MathsCubing I

Knowledge is the heritage of humanity, therefore each human can use it freely but he also has the duty to protect, to share, to improve it.
Morphocode CODE

Copyright

Title: MathsCubing (I)
Author: Morphocode CODE
Website: https://fan2cube.fr
Version: 21.8-24.1.8

ISBN : 9-798837-778070

Preface

A lot of people playing Rubik's Cube but don't know the mathematical properties of the Cube: The Dark Side of the Cube.
Then this book will enlighten you this wonderful part of the Cube, it also teaches you the correct terminolory and notation of the Rubik's Cube

1. MATHEMATIC FOR RUBIK'S CUBE

1.1. A GROUP

Let G be a set, with a law denoted '.' we say that (G, .) is a group if the law '.' verifies the following 4 properties:

1. a,b in G ⇒ a.b ∈G (internal law)
2. there exists an element e such that a.e = e.a = a (e = neutral element)
3. for each a, there exists a^{-1} such that $a.a^{-1} = a^{-1}.a = e$ (a^{-1} = inverse of a)
4. (a.b).c = a.(b.c) (associativity)

Moreover if we have: a.b = b.a (commute) We say that G is a commutative or abelian group.
Note: sometimes we write: ab = a.b and and 1 = e

<u>Warning</u> !! '1', 'a^{-1}' these are simple notations, symbols, characters, nothing to do with the integer 1, and the inverse: $3^{-1} = \frac{1}{3}$ e.g.

examples:

The group of 3 elements $\mathbb{Z}_3$ = {0, 1, 2} (the modulo 3 group)

$(\mathbb{Z}_3, +)$

+	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

Note : 2 = -1

The group of 2 elements $\mathbb{Z}_2$ = {0, 1} (the modulo 2 group)

($\mathbb{Z}_3$, +)

+	0	1
0	0	1
1	1	0

Note : 1 = -1

$(\mathbb{Z}, +)$
$(\mathbb{Q}^*, x)$
$(\mathbb{R}^*, x)$
$(\mathbb{Z}_p, +)$;p=prime

1.2. SUBGROUP

Let H be a subset of a group G, $H \subset G$ we say that H is a subgroup if:

1. $e \in H$
2. $x \in H \Rightarrow x^{-1} \in H$
3. $x,y \in H \Rightarrow xy \in H$

or

1. $H \neq \emptyset$
2. $\forall x,y \in H \Rightarrow xy^{-1} \in H$

A subgroup is said to be normal (or distinguished) if

$\forall h \in H, \forall g \in G \Rightarrow ghg^{-1} \in H$

Normal subgroups are very sought because they allow G to be 'divided', that is to say K = G/H, K is a group. When H is not normal we cannot divide G by H, however we can form what we call class of H.

1.3. THE CLASS OF H

Definition: Let H be a subgroup (not necessarily normal) of G, we call the left's class Hg (H is on the left) of G, the set:

$Hg = \{x \in G \mid x = hg, \text{ where } h \in H\}$, $g \in G$ given, fixed

It is therefore the elements $x \in G$ of the form $x = hg$ where $h \in H$, and $g \in G$ given, fixed.
The set of left-class is noted:

$H\backslash G = \{Ha, Hb, Hc, ... \}$, $a,b,c, ... \in G$ given, fixed

Class on the right (H on the right):
$gH = \{x \in G \mid x = gh, \text{ where } h \in H\}$, $g \in G$ given, fixed

$G\backslash H = \{aH, bH, cH, ... \}$, $a,b,c, ... \in G$ given, fixed

* H is normal $\Leftrightarrow$ $H\backslash G = G\backslash H$ class on the left = class on the right.

Three important properties:

1. $|H| = |Hg|$, a class has the same number of elements as H.
2. $|H\backslash G| = |G|/|H|$,
3. The class form a partition of G.

Same thing for the class on the right

1.4. PERMUTATION

We are given n objects $X = \{a,b,c,d, ...\}$ and n holes, the objects are in the holes. A permutation is a movement of these objects in these holes (we do not move the objects outside the holes !) .

And a permutation p will be noted :

p = [p(a),p(b),p(c),p(d), ...]

ex:

X = {1,2,3,4,5,6,7,8}

p = [1,5,8,6,3,7,2,4]
that means: p(1) = 1, p(2) = 5, p(3) = 8, p(4) = 6, p(5) = 3, p(6) = 7, p(7) = 2, p(8) = 4

q = [2,5,7,4,1,8,6,3]
it means: q(1) = 2, q(2) = 5, q(3) = 7, q(4) = 4, q(5) = 1, q(6) = 8, q(7) = 6, q(8) = 3
We denote S_n, or S_x the set of permutations with n objects.

1.5. THE K-CYCLES

k objects moving in cycles is called a k-cycle and we note it:

p = (a,b,c) or a→b→c
this means: a goes to b, b goes to c, and c goes to a,
in other words: p(a) = b, p(b) = c, p(c) = a. It is a 3-cycle, its length = 3 (number of letters)

p = (a,b) = a→b = a↔b
this means: a goes to b, b goes to a,
in other words: p(a) = b, p(b) = a. It is a 2-cycle (permute a,b , swap a,b , transpose a,b....) length = 2. We also say a transposition (a,b)

p = (a) = a→a = id = identity, a 1-cycle, length = 1

<u>Warning</u> !! (a,b,c) is different from [a,b,c]

e.g.
X={1,2,3,4,5}

$p = (4,2,1,3,5) \Rightarrow p(4) = 2, p(2) = 1, p(1) = 3, p(3) = 5, p(5) = 4$
$q = [4,2,1,3,5] \Rightarrow q(1) = 4, q(2) = 2, q(3) = 1, q(4) = 3, q(5) = 5$

$p \neq q$
and writing "[, ...]" requires writing all the elements of X, while "(, ...)" does not.

$X=\{1,2,3,4,5,6,7,8\}$
$p = (7,2,4,1,3)$
$p = [3,4,7,1,5,6,2,8]$

$r = (1,3,5,2,4)$
$r = [3,4,5,1,2,6,7,8]$
$p=[3,4,5,1,2,6,8,7]$, $q=(3,4,5,1,2,6,8,7)$
$p\neq q$ because $p(2) = 4$ and $q(2) = 6$

1.6. THE LAW '.' ON S_N

On S_n we define a law '.'
p.q = q o p (o = round = composition of functions)
So '.' it is the law of composition but from left to right.

<u>Properties of law '.'</u> :

1. p, q permutations ⇒ p.q permutation
2. identity : id = (a) = (b) = ...
3. for any permutation p, there exists an inverse permutation p^{-1}: $p.p^{-1} = p^{-1}.p = id$
4. (p.q).r = p.(q.r) associativity

NOTE: sometimes we write p.q = pq (we remove the '.' we're lazy!!) .

Let p be a permutation, we note p(x) = x.p read "p apply to x", we do from left to right, ex :

$p=(2,1,3)$, $q=(3,2,1)$

1.(2,1,3)(3,2,1) = 3.(3,2,1) = 2
2.(2,1,3)(3,2,1) = 1.(3,2,1) = 3
3.(2,1,3)(3,2,1) = 2.(3,2,1) = 1
then
(2,1,3)(3,2,1) = (1,2,3)

p = (1,3,4)(5,8,7,6) , q = (3,4,5,1,2,6,8,7)

1.(pq) = (1.p)q = 3.q = 4
2.(pq) = (2.p)q = 2.q = 6
3.(pq) = (3.p)q = 4.q = 5
4.(pq) = (4.p)q = 1.q = 2
5.(pq) = (5.p)q = 8.q = 7
6.(pq) = (6.p)q = 5.q = 1
7.(pq) = (7.p)q = 6.q = 8
8.(pq) = (8.p)q = 7.q = 3

(1,3,4)(5,8,7,6)(3,4,5,1,2,6,8,7) = (1,4,2,6)(3,5,7,8)

we do p then q, we apply x to the left

1.7. DISJOINT CYCLES

Def : We say that 2 cycles are disjoint if they have no elements in common. Ex
(a,b,c)(d,e)(f,g) disjoint
(a,b,c)(c,a,d) non-disjoint

Property :

if p and q, 2 disjoint cycles then: pq = qp they commute

Theorem 1:

Any permutation p can be decomposed into products of disjoint cycles and the decomposition is unique.
p = $(a,p(a),p^2(a), \ldots)$ $(b,p(b),p^2(b),\ldots)$ (....)

ex
X={1,2,3,4,5,6,7,8}
p = [3,2,4,1,8,5,6,7] is decomposed into the product of the cycles
p = [3,2,4,1,8,5,6,7] = (1,3,4)(5,8,7,6)

q = [4,3,5,1,2,6,8,7] = (1,4)(2,3,5)(7,8)

Theorem 2 :

Any permutation can be decomposed into products of transpositions (2-cycle) and the decomposition is not unique, but the parity of the number of transpositions is the same.

(a,b,c) = (a,b)(a,c)
(a,b,c,d) = (a,b)(a,c)(a,d)

Proof :
¤ a.(a,b)(a,c) = b.(a,c) = b
b.(a,b)(a,c) = a.(a,c) = c
c.(a,b)(a,c) = c.(a,c) = a

(a,b)(a,c) = (a,b,c)

¤ a.(a,b)(a,c)(a,d) = b.(a,c)(a,d) = b.(a,d) = b
b.(a,b)(a,c)(a,d) = a.(a,c)(a,d) = c.(a,d) = c
c.(a,b)(a,c)(a,d) = c.(a,c)(a,d) = a.(a,d) = d
d.(a,b)(a,c)(a,d) = d.(a,c)(a,d) = d.(a,d) = a

(a,b)(a,c)(a,d) = (a,b,c,d)

1.8. SIGNATURE

Even, odd permutation

We know that any permutation p can be decomposed (not unique) into a product of transpositions (2-cycle). Let t be

the number of transpositions in the decomposition, we say that p is even if t is even, odd if not.
For practical reasons in the calculations we note even = 1 and odd = -1

The signature of a permutation

By definition the signature of a permutation p is:

$sig(p) = (-1)^t$, where t is the number of transpositions in the decomposition.

We define the signature of a k-cycle by:
If (k-1) is even, the signature of this k-cycle is even, odd otherwise, that is :
$sig(k\text{-cycle}) = (-1)^{k-1}$

e.g.
sig(4-cycle) = 4-1 = 3 = odd.
sig(3-cycle) = 3-1 = 2 = even.
sig(2-cycle) = 2-1 = 1 = odd.
sig(1-cycle) = sig(a) = sig(id) = 1-1 = 0 = even.
so on ....

Properties :

1. sig(pq) = sig(p) . sig(q)
2. $sig(k\text{-cycle}) = (-1)^{k-1}$
3. $sig(p^{-1}) = [sig(p)]^{-1}$
4. sig(id) = 1

ex:
$p = (a,b)(a,c)(d,e) \Rightarrow t=3 \Rightarrow sig(p) = (-1)^3 \Rightarrow sig(p) = -1$, odd permutation (we have 3 pairs exchanges)
$q = (a,b,c) = sig(q) = (-1)^{3-1} = (-1)^2 = 1$;even

The set of even permutations will be noted: A_n, the number of permutations will be noted $|S_n|$ it is the cardinality of S_n or the number of elements of S_n. We have
$|S_n| = n!$ And
$|A_n| = n!/2$

Theorem 3 :

The 3-cycle (a,b,c) generates A_n

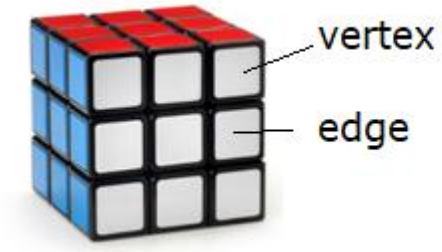
Theorem 4 :

The 3-cycle family { (a,b,x) with $x \neq a,b$ } generates A_n

2. THE RUBIK'S CUBE

2.1. PRESENTATION

He's the Rubik's Cube !



The Rubik's Cube

It's a rotating 3D puzzle with the stickers covering it.
The Rubik's Cube was invented by Erno Rubik (a Hungarian) around 1974 and then became famous from the 1980s.
When you turn the sides, the 6 colors scrambled ! the aim of the game is to restore the Cube to the solved state, i.e. one color per side.

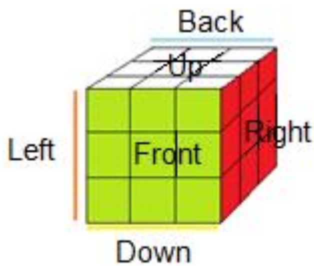
Rubik's Cube scrambled

2.2. FIXE THE CUBE, ORIENTE THE CUBE

Hold a Rubik's Cube (standard) in front of you or better still place it on a table with a side in front of you, so the Cube has 6 sides (or faces) named as follows (in order) :
(U)p > (D)own > (F)ront > (B)ack > (L)eft > (R)igth.
Abbreviated :
U > D > F > B > L > R

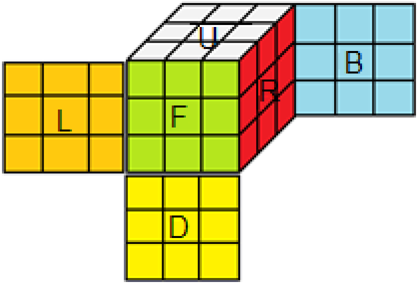
And for us the standard colors will be (in order) :
(w)hite > (y)ellow > (g)reen > (b)ue > (o)range > (r)ed.
Abbreviating :
w > y > g > b > o > r

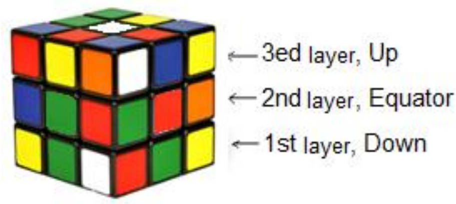
And these colors will be associated with the sides in the following ways :
(U)p)=(w)hite, (D)own=(y)ellow, (F)ront=(g)reen,
(B)ack=(b)ue, (L)eft=(o)range, (R)ight=(r)ed



Name of sides with standard colors

We say, we have oriented or fixed the Cube, this is the standard orientation of the Rubik's Cube.



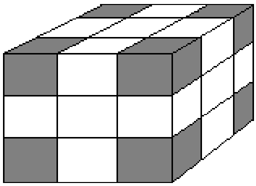


2.3. PIECES

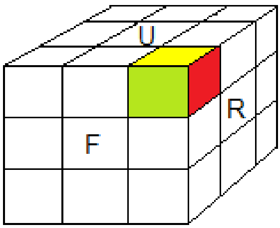
The first thing to do is to recognize the pieces. If you observe well, you will see that the Rubik's Cube has 3 kinds of pieces.

1. The pieces bearing 3 colors : the vertices (one vertex, tow vertices), they are the corners of the Cube. There are 8 vertices.

A vertex (a piece)



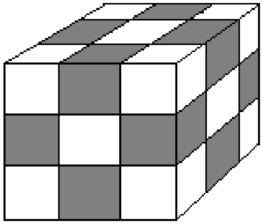
The 8 vertices



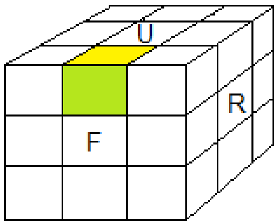
A vertex with its 3 colors

2. The pieces bearing 2 colors : The edges, they are located between two vertices. There are 12 edges.

An edge (a piece)

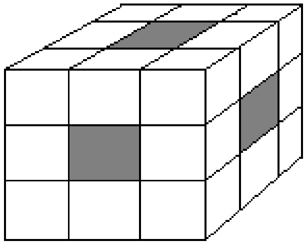


The 12 edges

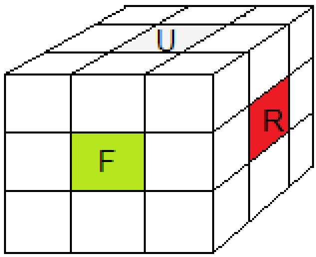


An edge with its 2 colors

3. The pieces bearing a single color : the centers, they are in the center of the side, and they determine the color of the sides : blue center ⇒ blue side, red center ⇒ red side and so on ... There are 6 centers.



The 6 centers



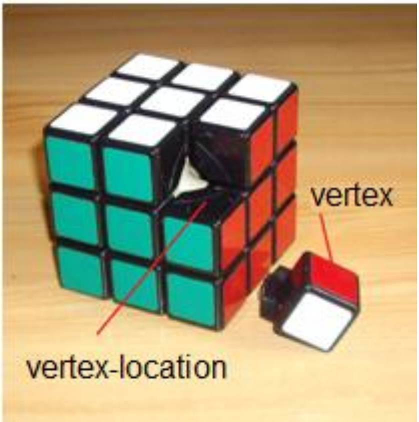
A center with its color

The vertices, the edges, the centers, these pieces remain in their camp, that is to say an edge always remains an edge it never becomes a center or a vertex, same thing for the centers and the vertices. All these pieces are distinct (there are 6 + 8 + 12 = 26) so they are unique. For the Rubik's Cube what is important are edges and vertices.

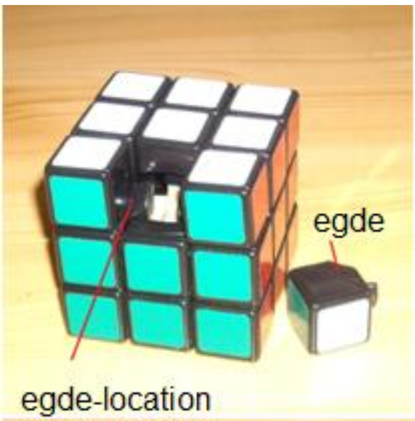
When scrambleing the Cube the vertices and the edges move, but no vertex takes the place of an edge and vice versa, the vertices remain in the clan of the vertices the edges remain in the clan of the edges.

2.4. LOCATIONS

We must make the distinction between a location (a position, a hole) and a piece (edge, vertex, center) : like the house and the people living. The location is the house , the piece is the people (the people live in his house).



Location and piece

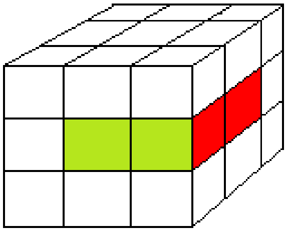


Location and piece

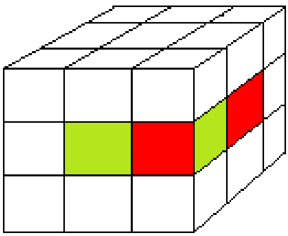
Each piece of the Rubik's Cube has a unique location, its location .

3. ORIENTATION THE PIECES

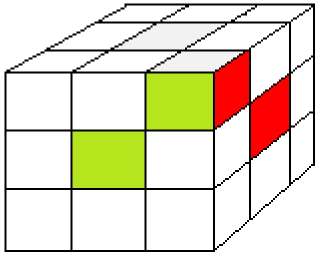
When we place a piece in its own location, it can be well placed or badly placed, we will say it is well oriented or badly oriented. Correct orientation means its colors match the center's color



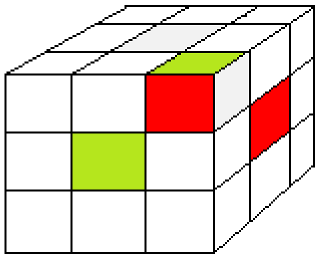
Edge good oriented



Edge bad oriented

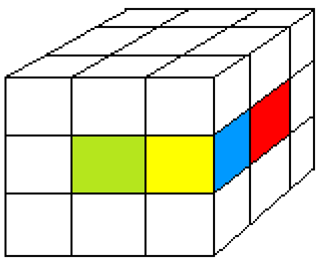


Vertex good oriented

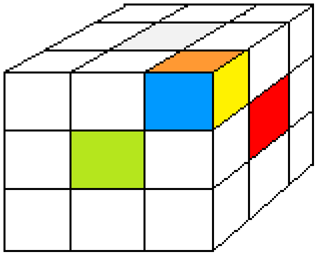


Vertex bad oriented

But then, how do you know if a piece is oriented correctly or incorrectly when it is in another location than its own location ?



Good or bad orientation ?

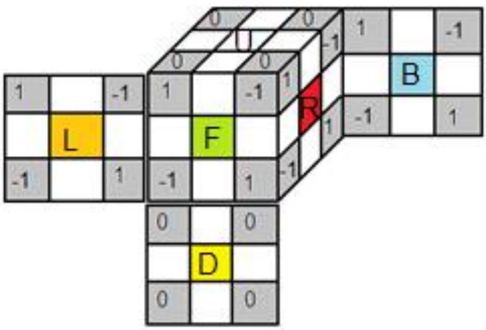


Good or bad orientation ?

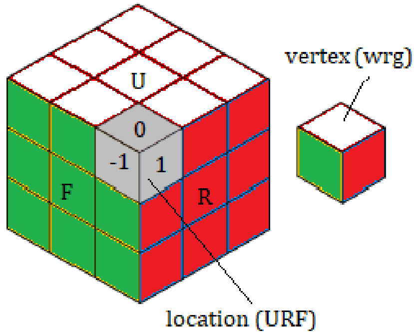
To answer this question we must go through a marking system

3.1. THE MARKING OF THE FACET-VERTEX

A vertex-location has facets and we decide to mark the facets as follows : 0 on the Up facets and the Down facets . Then, clockwise, we mark 1, -1. This is the usual marking.



vertices-locations with the facets marked in clockwise
0 = good orientation

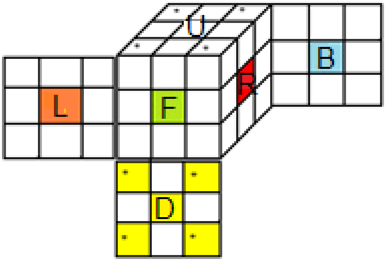


And by convention the order of the sides U and D is :
U > D (sides marked zero 0).

A vertex-location is a facets object, it has a name, the initial of the sides that compose it, and they are denoted in parentheses.
For the names of the vertices-locations we use the rule : "dominant side + clockwise"
Which gives us the 8 names of the vertices-locations :
(URF), (UFL), (ULB), (UBR)
(DFR), (DLF), (DBL), (DRB)

3.2. THE DOMINANT COLOR OF A VERTEX

For a vertex, what is its dominant color ? and why ? To know the dominant a vertex's color is very simple once the marking is given. In the solved state, the dominant color is the color which is on zero 0. Because in the solved state all the vertices are in good orientation.



* = the dominant color (* placed on 0)

And that gives the order of the colors :
(w)hite > (y)ellow (color marked zero 0)

A vertex (a piece) is an object with colors, it has a name, the initial of the colors which compose it, and they are denoted in parentheses .

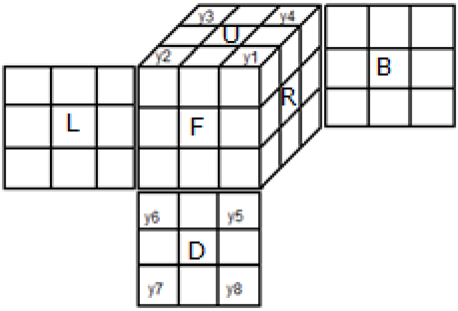
For the names of the vertices we use the rule : "dominant color + clockwise"

Which gives us the 8 names of the vertices :
(wrg), (wgo), (wob), (wbr)
(ygr), (yog), (ybo), (yrb)

3.3. VERTICES NUMBERING

We will number the vertices in y_i as shown in fig. below
y_1 = (wrg), y_2 = (wgo), y_3 = (wob), y_4 = (wbr),
y_5 = (ygr), y_6 = (yog), y_7 = (ybo), y_8 = (yrb).
$y = (y_1,y_2,y_3, \dots ,y_8)$

Note: We placed the y_i on the facet marked 0



The numbered vertices:y_i

Initially the vertices locations contain the y_i as follows:
(URF) = y_1, (UFL) = y_2, (ULB) = y_3, (UBR) = y_4
(DFR) = y_5, (DLF) = y_6, (DBL) = y_7, (DRB) = y_8

3.4. ORIENTATION OF THE VERTICES

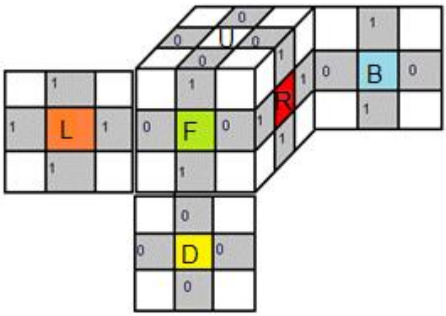
On the one hand, there are locations with 3 facets marked 0, 1, -1 and on the other hand the vertices having 3 colors, one of which is dominant. When a vertex is lodged in a location and its dominant color is on the facet marked -1, we say its orientation is -1, and if its dominant color is on 1, its orientation is 1, on 0 its orientation is 0, in this case we say that the vertex is well oriented, good orientation.

We denote, for example:
$(URF)^+$ that means the vertex in (URF) twisted in clockwise or in "+" or "1".
$(URF)^-$ that means the vertex in (URF) twisted in anti-clockwise or in "-" or "-1".
Let the vertex (wrg) is in (DFR) :

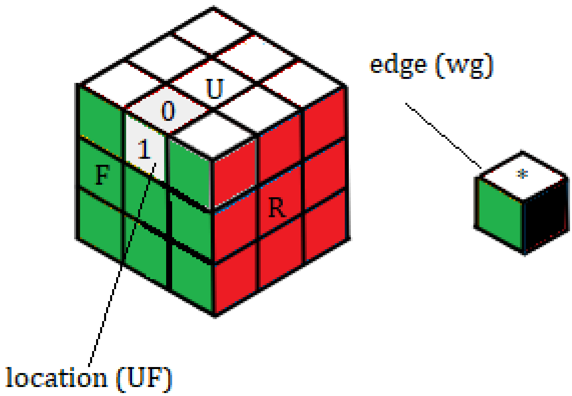
-if its dominant color w rotates 1/3 turn in clockwise so is orientation is 1
-if its dominant color w rotates 1/3 turn in anti-clockwise so is orientation is -1
-if its dominant color w on D so is orientation is 0, good orientation.

3.5. THE MARKING OF THE FACET-EDGE

We decide to mark the facet-egde as shown in fig. below : 0 on U and D, then 0 on F and B, this is the standard marking of edges in Rubik's Cube.



Edge-location with marked facets
0=good orientation

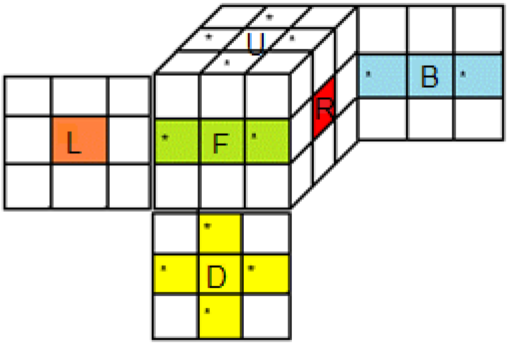


And this gives the order of the sides :
U > D > F > B > L > R (sides marked zero 0).
An edge-location is a facets object, it has a name, the initial of the sides that compose it, and they are denoted in parentheses.
For the names of the edges-locations we use the rule : "dominant side"

Here are the 12 names of the edges-locations :
(UF), (UL), (UB), (UR)
(FR), (FL), (BL), (BR)
(DF), (DL), (DB), (DR)

3.6. THE DOMINANT COLOR OF A EDGE

For an edge, what is its dominant color ? and why ? Once the marking is given, in the solved state, the dominant color is the color which is on zero 0. Because in the solved state all the edges are in the correct orientation.



* = the dominants colors (* placed on 0)

And this gives the order of the colors:
(w)hite > (y)ellow > (g)reen > (b)lue > (o)range > (r)ed ;(colors marked zero 0).

An edge is a colors object, it has a name, the initial of the colors that compose it, and they are noted in parentheses.

For the names of the edges we use the rule : "dominant color"

Here are the 12 names of the edges:

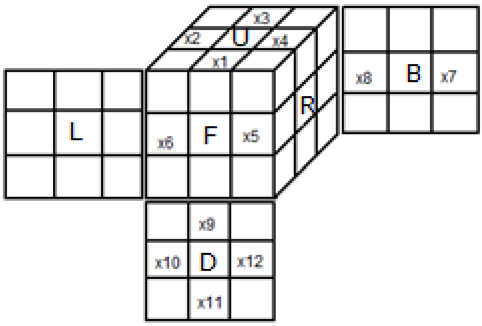
(wg), (wo), (wb), (wr)
(gr), (go), (bo), (br)
(yg), (yo), (yb), (yr)

3.7. EDGE NUMBERING

We will number the edges in x_i as shown in fig. below

x_1=(wg), x_2=(wo), x_3=(wb), x_4=(wr),
x_5=(gr), x_6=(go), x_7=(bo), x_8=(br),
x_9=(yg), x_{10}=(yo), x_{11}=(yb), x_{12}=(yr).
x = (x_1,x_2,x_3, ..., x_{12})

Note: we placed the x_i on the facet marked 0



Numbered edges: x_i

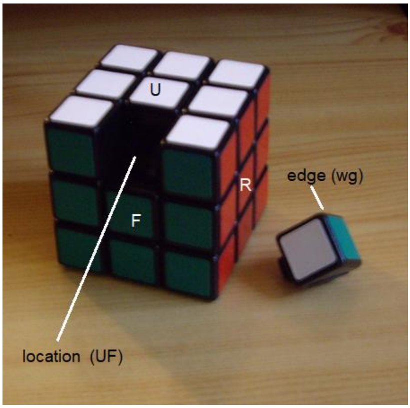
Initially the locations contain the x_i as follows:
(UF)=x_1, (UL)=x_2, (UB)=x_3, (UR)=x_4
(FR)=x_5, (FL)=x_6, (BL)=x_7, (BR)=x_8
(DF)=x_9, (DL)=x_{10}, (DB)=x_{11}, (DR)=x_{12}

3.8. ORIENTATION OF THE EDGES

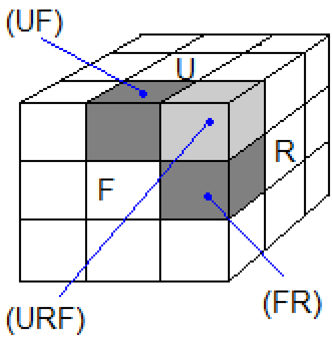
As for the vertices, we have on the one hand, locations with 2 facets marked 0, 1 and on the other hand edges having 2 colors, one of which is dominant. When an edge is lodged in a location and its dominant color is on 1, we say its orientation is 1, and if its dominant color is on 0, its orientation is 0 in this case we say this edge is well oriented.

Let the vertex (gr) is in (UF) :
-if its dominant color g on F so is orientation is 1
-if its dominant color g on U (dominant side) so is orientation is 0, good orientation.

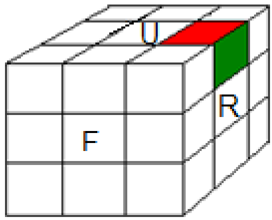
We denote, for example:
(UF)⁻ that means the edge in location (UF) has "-" or "1" .



Location (UF) and edge (wg)

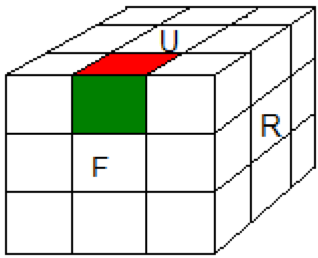


Locations

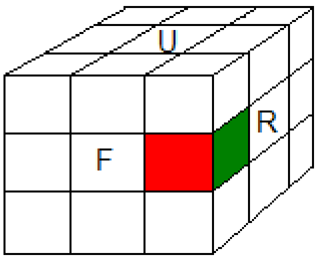


The edge (gr) in (UR)

We can put the edge (gr) in (UR) or (UF)

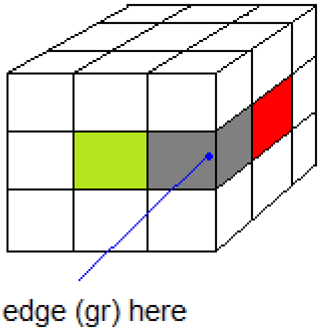


The edge (gr) in (UF)

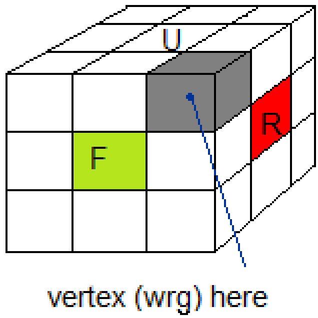


The edge (gr) in (FR)

Each piece of the Rubik's Cube has a unique location, its location, the location determined by the centers

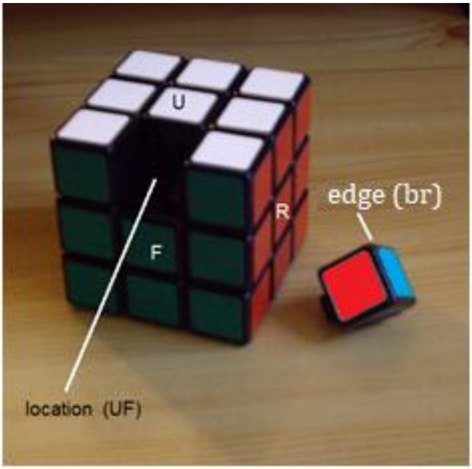


The edge's location (gr)

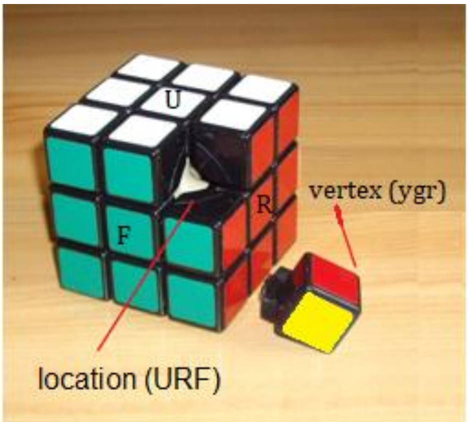


The vertex's location (wrg)

For convenience we say the edge (UF) instead of "the edge contained in location (UF)" , in the same way we simply say the vertex (URF) instead of "the vertex contained in location (URF)".



we say : edge (UF) instead of (br)



we say: vertex (URF) instead of (ygr)

Notice : For the Rubik's Cube we are only interested in the vertices and edges, but sometimes we are also interested in the centers (SuperRubik's Cube) in this case the location of the centers will be denoted:
(U), (D), (F), (B), (L), (R) .

and the centers
(w), (y), (g), (b), (o), (r) .

4. THE CONFIGURATIONS $(G^+,.)$

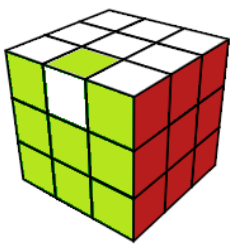
In Rubik's Cube the concept of configuration (pattern of stickers) is very important without him it's impossible to build the theory.

The configuration group G^+ of the Rubik's Cube is :
$G^+ = (S_{12} \times \mathbb{Z}_2^{12}) \times (S_8 \times \mathbb{Z}_3^8)$
with law '.' :

$(u,x,v,y)(u',x',v',y') = (uu',x+u(x'),vv',y+v(y'))$
$uu' = u' \circ u$, $vv' = v' \circ v$
$u(x) = (x_{u(1)} , x_{u(2)} , \ldots, x_{u(12)})$
$v(y) = (y_{v(1)} , y_{v(2)} , \ldots, y_{v(8)})$

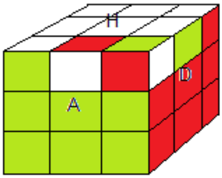
→In G^+ we can permute the vertices between them without touching the edges.
→In G^+ we can twist one vertex without touching the other pieces.
→In G^+ we can permute the edges between them without touching the vertices.
→In G^+ we can flip one edge without touching the other pieces.

Here are some visualizations of the configurations



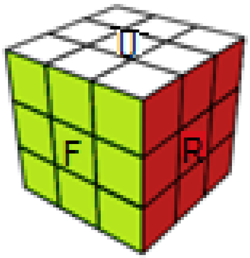
a configuration

another configuration



a configuration

another configuration



Solved configuration e, one color per side

NOTICE: The centers must not be moved !!
center (U)p=(w)hite, center (F)ront=(g)reen, center (R)ight=(r)ed ...

Another way of seeing G^+: We disassemble the Cube and reassemble it randomly, we hold the Cube in such a way that:
(U)p = (w)hite, (D)own = (y)ellow, (F)ront = (g)reen, (B)ack = (b)lue, (L)eft = (o)range, (R)right = (r)red

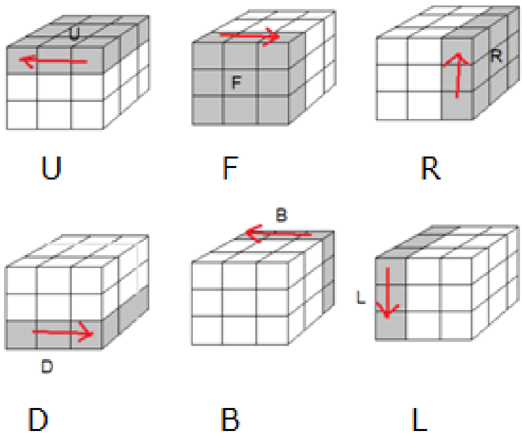
The pattern obtained is a configuration, i.e. an element of G^+, there are many unsolvable configurations, we have one in 12 chance of having a solvable configuration.

5. THE ROTATIONS

The Rubik's Cube has several types of rotations.

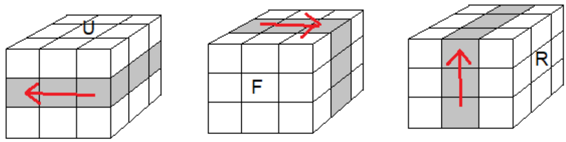
<u>The basic rotations</u> {U, D, F, B, L, R}

F = We stand in front of the Front side and turn the Front side 90° (clockwise) .
F' = F^{-1} = We stand in front of the Front side and turn the Front side -90° (anti-clockwise) .
F² = FF = We stand in front of the Front side and, we turn the Front side 180°



<u>Slice rotations</u> {u, r, f}

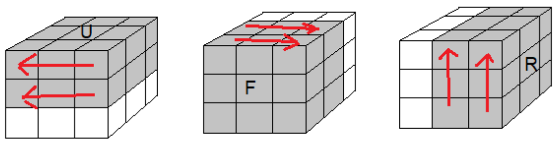
f = We stand in front of the Front side and we turn the slice-front 90° (clockwise) .
f'= We stand in front of the Front side and we turn the slice-front -90° (anti-clockwise) .
f² = ff = We stand in front of the Front side and we turn the slice-front 180°.



u f r

<u>Block rotations</u> {U*, R*, F*}

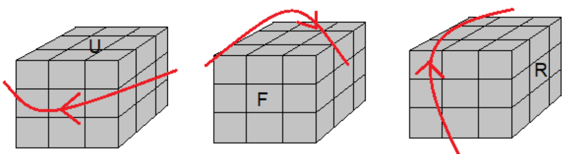
F* = We stand in front of the Front side and we turn the block (Ff) 90° (clockwise) .
F*'= We stand in front of the Front side and we turn the block (Ff) -90° (anti-clockwise) .
F*² = F*F* = We stand in front of the Front side and we turn the block (Ff) 180°.



U* F* R*

<u>Cube rotations</u> {^tU, ^tR, ^tF}

^tF = We stand in front of the Front side and turn the whole Cube 90° (clockwise).
$^tF'$ = $(^tF)'$ = We stand in front of the Front side and turn the whole Cube -90° (anti-clockwise) .
$^tF^2$ = $^tF^tF$ = We stand in front of the Front side and we turn the whole Cube 180°.

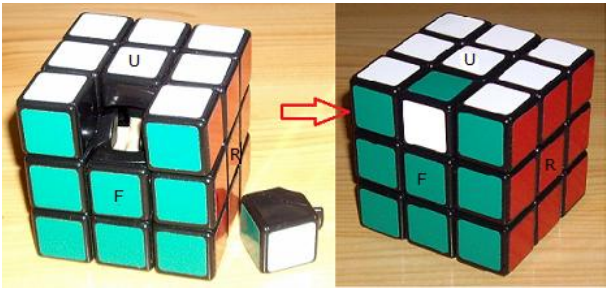


^tU ^tF ^tR

We will define three extended rotations following :

Rotation Γ: flip the edge (UF)[1].

1. Remove the edge (UF)
2. flip 180°
3. Then we put it back

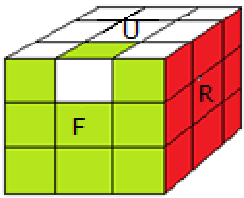


By definition this maneuver is called extended rotation Γ which results the edge in (UF) flipped

$(UF)^{-} = \Gamma$

And associated configuration

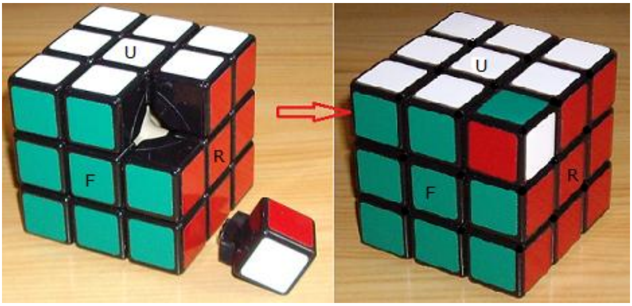
$\Gamma \rightarrow (id,x,id,0)$, $x = (1,0,0,0,0,0,0,0,0,0,0,0)$; this flips (UF), and does not touch the other pieces.



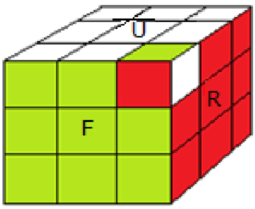
$\Gamma = (UF)^{-}$

[1] by abuse of language we say the edge (UF) instead of the edge in location (UF)

Rotation Ψ: twist the vertex (URF)[2].
1. We remove the vertex (URF)
2. twist 120° clockwise
3. Then we put it back



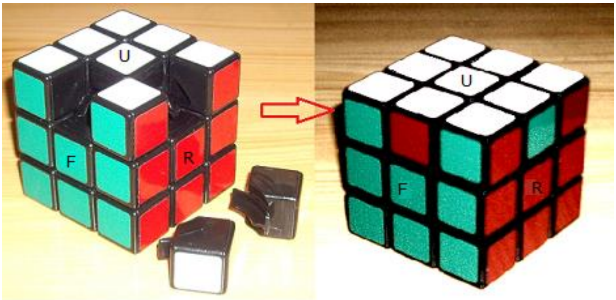
This maneuver, by definition, is called extended rotation Ψ which results the vertex in (URF) twisted 120° clockwise
$(URF)^+ = \Psi$
And associated configuration
$\psi \rightarrow (id,0,id,y)$, $y = (1,0,0,0,0,0,0,0)$; this twists the vertex (URF) 120°, and does not touch the other pieces.



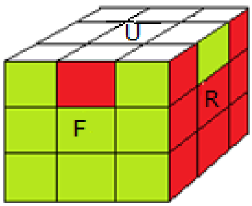
$\psi = (URF)^+$

Rotation Ω: swap two edges.
1. We remove the edges (UF), (UR)
2. swap (UF)↔(UR) = (UF,UR)
3. Then we put them back

[2] by abuse of language we say the vertex (URF) instead of the vertex in location (URF)



This maneuver, by definition, is called extended rotation Ω which results the edges in (UF), (UR) swapped
(UF)↔(UR) = (UF,UR) = Ω
And associated configuration
Ω → (u,0,id,0) , u = (1,4) ; this permute (x_1,x_4), and does not touch the other pieces .



Ω = (UF,UR)

6. FORMULAS (M,.)

Another very important concept to understand : the concept of formula (= movement, = scrambling, = maneuver),
A formula is a finite sequence of basic rotations {U, D, F, B, L, R} and their inverse with the rule :

* We agree to avoid doing FF', F'F, DD', D'D, ...
e.g.:

$FUD'B^2RL'$; ok
$LDUU'R^2DB$; forbidden: UU'
URU'R' ; ok

A basic rotation is therefore a formula e.g. rotation F.

$R^2LD'rUR'BF$, $RF\,{}^tUL'R^2$ are not formulas strictly speaking because they contain r and ${}^tU$ which are not a basic rotations, but in practice we also call them formulas.

We denote :
UU' = U'U = DD' = D'D = ... = I
By definition it is the identical formula (neutral formula, empty formula), I = do no thing.

The set of formulas formed from the basic rotations {U, D, F, B, L, R} (and their inverse) will be denoted:

M = <U, D, F, B, L, R>

we say that M is generated by the basic rotations .
We define on M a law '.' concatenation , and we have :

⇒Let V, T be two formulas, it is clear that VT is still a formula

⇒there is a neutral formula I : VI = IV = V

⇒For any formula V there is an inverse formula V' for example :
$V = UB'D^2FL'R$
$V' = R'LF'D^2BU'$; read V backwards and prime↔non-prime and we have:
VV' = V'V = I

⇒Let V, T, S be three formulas, we do (VT) then S is the same as doing V then (TS) i.e. we have:
(VT)S = V(TS) ; associativity

So (M,.) is a group, the formulas group of the Rubik's Cube

Let M^+ = <U, D, F, B, L, R, Γ, ψ, Ω> we call the group of extended formulas of the Rubik's Cube

<u>Note</u> : Here M and M^+ is infinite (because we can have U', UUU, UUUUUU, UUUUUUUUU, etc ....)

<u>Commutator</u> : A commutator is a formula of the form : VTV'T' where V, T are formulas and we denote it
[VT] = VTV'T' (read VT hook)

<u>Conjugate</u> : A conjugate is a formula of the form : VTV' where V, T are formulas and we denote it
T^V = VTV' (read T power V)
It is also said that VTV' is a conjugate of T.
The notation T^V is more suitable because it preserves the property of the conjugation, in fact :

$(S^T)^S = S(T^V)S' = (SV)T(V'S') = (SV)T(SV)'$
$= T^{SV}$ (as a number $(3^5)^2 = 3^{2x5}$)

We may note $\frac{T}{V} = VYV'$ it is a good notation also, because it preserves also the property of the conjugation :

$$\frac{\left(\frac{T}{V}\right)}{S} = S\left(\frac{T}{V}\right)S' = S(VTV')S' = (SV)T(V'S') = (SV)T(SV)'$$
$= \frac{T}{SV}$ as a number $\frac{\left(\frac{3}{5}\right)}{2} = \frac{3}{2x5}$

In the notation T^V we gain 2 characters , $\frac{T}{V}$ we gain 1 character. Some people note [T:V] = VTV' then an abbreviation is longer than the original writing !!

7. STATES (G,.)

When we apply a formula V to a configuration, we get another configuration. We say M acts on G^+ and we denote this action '•' and here are the four axioms for the action :

$G^+ \times M \to G^+$
$(r,V) \to r\bullet V = s \in G^+$

A1) $\forall r$; $r\bullet I = r$;identity
A2) $\forall r,V,T$; $(r\bullet V)\bullet T = r\bullet(VT)$;associative
A3) $(r\in G^+, V\in M$; $r\bullet V = r) \Rightarrow V = I$;free (I is the only formula having fixed points)
A4) $\forall r,V,T$; $(r\bullet V)(r\bullet T) = r\bullet(VT)$;compatibility law in M and G

$r\bullet V = s$ We say V generate s (by r) (read : V applied to r giving s)
r=initial configuration, s=final configuration

→s comes from V (by r)
→V generate s (by r)

when r=e solved configuration
We simply say that s comes from V or that V generates s

$e\bullet V = s$

By definition the Rubik's Cube group G is defined by:

$G = \{s\in G^+ \mid s=e\bullet V , V\in M\}$
the elements of G are called states, e=solved state.

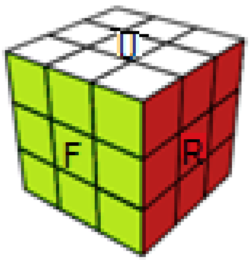
Let see if G is a subgroup of G^+

1) $e\bullet I = e$, $I\in M \Rightarrow e\in G$
2) $s\in G \Rightarrow s=e\bullet V$, $V\in M$
Let $s'=e\bullet V'$
$ss' = (e\bullet V)(e\bullet V') = e\bullet(VV') = e\bullet I = e$

s's = (e•V')(e•V) = e•(V'V) = e•I = e
it shows that s' is the inverse of s and s'∈G because V'∈M
3) s=e•V , t=e•T , V,T ∈M
st = (e•V)(e•T) = e•(VT) ⇒ st∈G because , VT∈M
So (G,.) is a group the group of the Rubik's Cube.

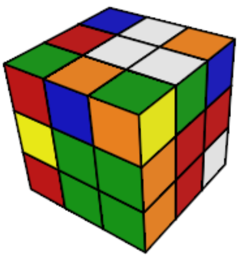
G is the set of configurations coming from M.
The elements of G are called the states

Here are some states

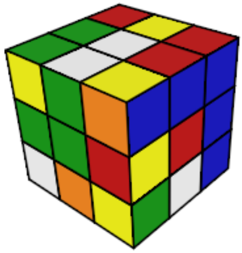


Solved state e, one color per side

Here are tow states



A state



Another state

NOTICE: The centers must not be moved !!
center (U)p=(w)hite, center (F)ront=(g)reen, center (R)ight=(r)ed ...

8. THE LAWS OF THE RUBIK'S CUBE

When playing Rubik's Cube you may have noticed this.

You :

-cannot flip one edge without touching the other edges.
-cannot twist one vertex without touching the other vertices.
-cannot swap two edges without touching the vertices.

Well, the Rubik's Cube has three laws known as the fundamental theorem of Cubology :
What are the necessary and sufficient conditions for a configuration to be a state ?

We prove the following theorem .

Fundamental theorem of Cubology :

$s=(u,x,v,y) \in G^+$ is an element of G if and only if :

$$(F) \sum_{i=1}^{12} x_i = 0 \pmod 2 \text{ , abbreviated } x = 0 \pmod 2$$

$$(T) \sum_{i=1}^{8} y_i = 0 \pmod 3 \text{ , abbreviated } y = 0 \pmod 3$$

(P) sig(u) = sig(v)

Then
$G = \{s \in G^+$ satisfying (F), (T), (P) $\}$
(G,.) is a subgroup of $(G^+,.)$ these elements of G^+ which verify (F), (T), (P).

Proof :
¤ Necessary condition :
We start with a formula V∈M, such that

e•V = s.
We have to show that s∈G.
We will reason by induction over the length of |V| = n

A) For n=1 ⟹ V=Z=basic rotation
We can see that all Z the basic rotations verify (F), T(), (P)
let see for example the basic rotation F :

e•F = (p,a,q,b)
p = (x_1,x_5,x_9,x_6) = (1,5,9,6) ; we use 1,5 ... rather than x_1,x_5, ...
a = (1,0,0,0,1,1,0,0,1,0,0,0)
q = (y_1,y_5,y_6,y_2) = (1,5,6,2)
b = (-1,1,0,0,1,-1,0,0)
then
a = 0 (mod 2)
b = 0 (mod 3)
sig(p) = sig(q)

and the rotation R :
e•R = (p,a,q,b)
p = (4,8,12,5)
a = (0,0,0,0,0,0,0,0,0,0,0,0)
q = (1,4,8,5)
b = (1,0,0,-1,-1,0,0,1)
then
a = 0 (mod 2)
b = 0 (mod 3)
sig(p) = sig(q)

We do the same thing with the other basic rotations.
So the properties (F), (T), (P) are verified for n=1

B) Supposing that these properties are true for n, let us show that it still remains true for n+1.
Let V be a formula of length n+1 and e•V=(u',x',v',y') the associated state. Now we go from n to n+1 by a basic rotation Z, hence
V = QZ ; |Q|=n
Or

e•Q=(u,x,v,y) , e•Z=(p,a,q,b)
e•V = e•(QZ) = (e•Q)(e•Z) ,
(u',x',v',y') = (u,x,v,y)(p,a,q,b)

* (u',x') = (u,x)(p,a) = (up, x+u(a))
x' = x + u(a)
a = 0 (mod 2) ; according to A)
u(a)=0 (mod 2)
x = 0 (mod 2) ; IH
x' = 0 (mod 2)

* (v',y') = (v,y)(q,b) = (vq, y+v(b))
y' = y + v(b)
b = 0 (mod 3) ; according to A)
v(b)=0 (mod 3)
y = 0 (mod 3) ; IH
y' = 0 (mod 3)

* u' = up , v' = vq

sig(u') = sig(up) = sig(u) sig(p)
= sig(u) sig(q) ; according to A)
= sig(v) sig(q) ; IR
= sig(v')

Therefore the properties (F), (T), (P) are true for all n
i.e. for all V∈M ⇒ e•V = s∈G

¤ <u>Sufficient condition</u> :
Let s∈G so we must to find a V∈M so that
e•V = s.

Let go ...

It is enough to solve the Cube it is possible because s verifies (F), (T), (P) we will therefore have a big big formula N such that
s•N = e

So let V=N' we have
e•V = s this means that s comes from V∈M.

<u>(F) Law of flip</u> : When we flip the edges we always flip two edges.
<u>(T) Law of twist</u> : When we twist the vertices we always twist :
- two vertices in the opposite directions
- three vertices in the same directions.

<u>(P) Law of parity</u> : When we permute a pair of edges we are obliged to also permute a pair of vertices and vice versa.

If we have a SuperRubik's Cube (oriented centers) we have a 4th law

<u>(C) Law of centers</u> : When we twist the centers without touching the other pieces, the sum of the degrees of the six centers is a multiple of 180.

So in the solved state we can have:

- the centers 180° ,
-or couples of centers at (90°, -90°)
-or couples of centers at (90°, 90°)
-or couples of centers at (-90°, -90°)
....

8.1. THE NUMBER OF STATES

To calculate the number of states it is enough to examine the laws, the constraints $\mathcal{N}$

(F) x=0 (mod 2) ⇒ 2 choices
(T) y=0 (mod 3) ⇒ 3 choices
(P) sig(u)=sig(v) ⇒ 2.2/2 choices
finally we have
$\mathcal{N}$ = 2.3.2 = 12
So
$|G| = |G^+| / \mathcal{N} = |G^+| / 12$

$|G| = 12!\ 2^{12} \times 8!\ 3^{8} / 2.3.2 = 12!.2^{10} \times 8!.3^{7}$
$|G| = 43\ 252\ 003\ 274\ 489\ 856\ 000$.

9. FORMULAS AND STATES

The axiom A3 shows that two formulas giving the same configuration will be considered identical
Indeed :

$s\bullet V = s\bullet T$
$(s\bullet V)\bullet T' = (s\bullet T)\bullet T'$
$s\bullet(VT') = s\bullet(TT')$
$s\bullet(VT') = s\bullet I = s$
$s\bullet(VT') = s$

Only I have fixed points (A3) so
$VT' = I \Rightarrow V = T$

especially
$e\bullet V = e\bullet T \Rightarrow V = T$

for example :

$F' = F^3$
$F^4 = R^4$
$(U^2R^2)^3 (D^2R^2)^3 = (U^2L^2)^3 (D^2L^2)^3$
$D = (RL'F^2B^2RL')\ U\ (RL'F^2B^2RL')$
$F^2B^2L^2R^2 = R^2L^2B^2F^2$
$F'D^2FU^2F'D^2FU^2 = U^2RDR'U^2RD'R'$
$FR'F'R = U'RUR'F'UFU'$

Let

$f: M \rightarrow G^+$
$f(V) = e\bullet V$; e=solved state
Note f is injective (axiom A3)

We define the following equivalence relation
$V,T \in M$, $V\sim T$ ssi $V,T \in Ker(f))$
and
$M/\sim = M/Ker(f) = M \cong Im(f) \subset G^+$ fini

So this implies that M is finite , we do the same thing for M^+ so M^+ is finished too .

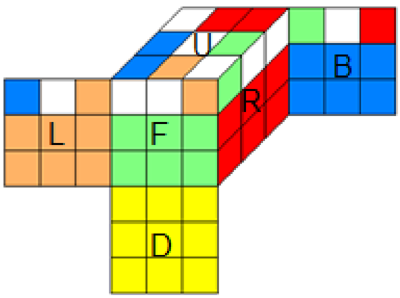
9.1. FORMULA'S LENGTH

The length of a formula V is the number of rotations it contains and we denote it |V| , e.g. :

|I| = 0 ; there is no rotation in I
$|F| = 1$, $|F^2| = 2$,
$S = FR^3F'R'U^2B'^2$, $V = UBDR^3LF'$
|S| = 10, |V| = 8

Among the formulas generate s, there are formulas of minimum length, for example the formulas C, N below give the same state s but N is minimal (* = minimal).

$C = R^2F^2RU^2L'U^2RU^2BUL'B'L'R'F'L'U'R$ (23)
$N = L'RB'LU^2F'L'B'FRB'RF'LFR$ (17*)



State s

To find a minimal formula we use the program cube514qtm.exe (Cube Explorer by Herbert Kociemba): We give a state, it finds a formula or a minimal formula.
http://kociemba.org/cube.htm

9.2. FORMULA'S ORDER

Let V be a formula, we say that V is the order d, if d is the smallest integer such that :

$V^d = I$

That is, we apply V, d times we return to the initial state e.g. :

$F^4 = I$; F is order 4

$[RU]^6 = I$; [RU] is order 6

One shows that the maximum order of a formula is 1260, and there are only 73 orders.

$(UR')^{63} = I \Rightarrow (U)^-(R)^+$; this proof the centers rotate
$(UR)^{105} = I \Rightarrow (U)^+(R)^+$
$(UR'UF'D^2)^{1260} = I$; the maximal order

9.3. MOVE

When we apply a formula V to the state r we find another state s.

r•V = s ; r=initial state, s=final state

The states s obtained by moving the pieces.
We denote:

¤ (URF)→(UBR)→(UFL) or (URF,UBR,UFL)
That means:
-The vertex in (URF) goes to the location (UBR),
-The vertex in (UBR) goes to the location (UFL),
-The vertex in (UFL) goes to the location (URF),
We say we have a 3-cycle-vertices

¤ (URF)↔(UFL) or (URF,UFL)
That means :
-The vertex in (URF) goes to the location (UFL),
-The vertex in (UFL) goes to the location (URF),
We say we have a 2-cycle-vertices

By abuse of language we say
"The vertex (URF) goes to vertex (UBR)" instead of "The vertex in location (URF) goes to location (UBR)"

It's the same for edges:
We denote:

¤ (UF)→(UR)→(UL) or (UF,UR,UL)
That means :
-The edge in (UF) goes to the location (UR),
-The edge in (UR) goes to the location (UL),
-The edge in (UL) goes to the location (UF),
We say we have a 3-cycle-edges

¤ (UF)↔(UL) or (UF,UL)
That means :
-The edge in (UF) goes to the location (UL),
-The edge in (UL) goes to the location (UF),
We say we have a 2-cycle-edges

A formula therefore moves pieces, we write :
V = (URF)→(UBR)→(UFL) or
V = (URF,UBR,UFL)

To say that V moves the vertices (URF),(UBR),(UFL)
examples:

[RU] L'[UR]L = (ULB)→(UFL)→(UBR)

[RU] = (FR,UR,UB) (DFR,URF) (UBR,ULB)

F[RU]F' = (UB,UF,UR) (URF,UFL) (UBR,ULB)
....

10. FAMOUS FORMULAS

$M = \langle U,D,F,B,L,R \rangle$

¤ We can have 5 generators
$M = \langle U,F,B,L,R \rangle$ where $D = (RL'F^2B^2RL')\ U\ (RL'F^2B^2RL')$ (Roger Penrose)

¤ We can have 2 generators
$M = \langle T,K \rangle$ where $T = UBLUL'U'B'$ and $K = R^2FLD'R'$ (Frank Barnes)

¤ $RU^2D'BD'$, this formula has the maximum order 1260
$(RU^2D'BD')^{1260} = I$

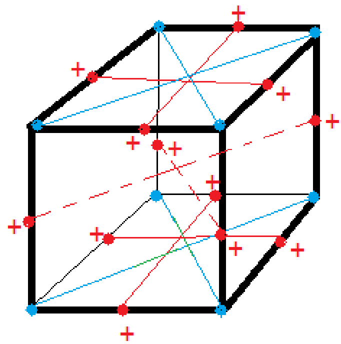
¤SuperFlip Φ (12 edges fliped) :
$\Phi = (UF)^-(UL)^-(UB)^-(UR)^-$
$(FR)^-(FL)^-(BL)^-(BR)^-$
$(DF)^-(DL)^-(DB)^-(DR)^-$

* SuperFlip =
$\Phi = R'U^2BL'\ .FU'BDF\ .UD'LD^2\ .F'RB'DF'\ .U'B'UD'$ (24*)
(Michael Reid ,1995, by computer)

$|\Phi| = 24$ (minimal formula, Jerry Bryan, 1995)

¤ SuperFlip4Spot Π :
$\Pi = (UF^-,UB^-)(UL^-,UR^-)(FR^-,BL^-)(FL^-,BR^-)$
$(DF^-,DB^-)(DL^-,DR^-)$
$(URF,ULB)(UFL,UBR)(DFR,DBL)(DLF,DBR)$

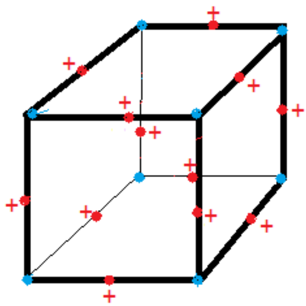
Michael Reid (1998) has found this formula by computer
$\Pi = U^2D^2L\ F^2\ .U'DR^2\ BU'D'R.\ LF^2R\ UD'\ R'LUF'B'$ (26*)
and has proven to be one of the shortest formulas $|\Pi| = 26$ this formula is called the SuperFlip4Spot, it is one of the states furthest from the Cube.



SuperFlip4Spot

* 6 pairs of edges are exchanged
* All edges flipped
* 4 pairs of vertices are exchanged

Why we call it SuperFlip4Spot because we have:
$\Pi = \Phi\ \Omega = \Omega\ \Phi$
4Spot = Ω = $DB^2F^2DU'L^2R^2U'$ (12*)
We don't know many states of length 26, apart from this SuperFlip4Spot !

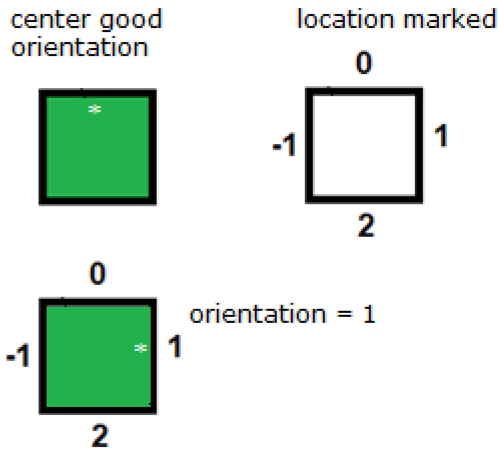


SuperFlip

* 12 edges flipped

10.1. CENTER ORIENTATION

If the centers are oriented (with an image for example, like the Rubik Hello Kitty, or Rubik Pochmann...), we have the Super Rubik's Cube and has a 4th law, called the law of centers (C). Indeed :
We orient the centers as follows:



* = 0, good orientation

As the edges, and the vertices
We arbitrarily decide one side of the center is dominant '*' (star), and mark the location of the center as 0 (dominant side '*'), then clockwise 1, 2, -1
When the dominant side '*' points to 0, 1, 2, -1 we say that the orientation of the center is 0 (well oriented) 1,2,-1.
Let c (1=90°) be the sum of the orientations of the 6 centers : $c = c_1 + c_2 + c_3 + c_4 + c_5 + c_6$.

We see that the law of centers (C) is:
(C): $(-1)^c = sig(u)$

The orientation number c of the centers is even if sig(u) is even, odd if sig(u) is odd.
If we twist the centers without touching the other pieces the signature of the edges is even (sig(edges) = 1) therefore
$(-1)^c = 1$
$c = 2k$
i.e. the number (1=90°) of orientations of the centers is even.

Passing through the degree
$90c = 180k$
90c = the number of degrees of the 6 centers,

So when we twist the centers without touching the other pieces the number of degrees of the centers is a multiple of 180.

This explains why in the solved state, we have:
¤ 180° centers.
¤ Or pairs of centers at (90°,-90°)
¤ Or pairs of centers at (90°,90°)
¤ Or pairs of centers at (-90°,-90°)
so on ...

In a formula we count the basic rotations +1, it's inverse -1 exp:

F[RU]F'U $\Rightarrow$ 1+1+1-1-1-1+1 $\Rightarrow$ $(U)^+ = 90°$
$(RU)^{105} = I \Rightarrow (R)^+ = 90°, (U)^+ = 90°$
$(RU')^{63} = I \Rightarrow (R)^- = -90°, (U)^+ = 90°$

This proof the centers turn.

What is the number of states of the Super Rubik's Cube ?

Each center has 4 orientations as we have 6 centers hence we have $4^6/2 = 2048$ (divided by 2 because of the law $(-1)^c = sig(u)$) additional configurations which must be multiplied with the number of states of the Rubik's Cube.

$|G_s| = 2048\ |G|$

Noticed :
the formula :
$|G_e^+| = |G| = |M| / |M_e|$
$G_e^+ = \{ \mu \in G^+ \mid \mu = e \bullet V, V \in M\} = G$ = the orbit of e
$M_e = \{ V \in M \mid e \bullet V = e\}$ = stabilizer of e.
As no formula leaves a state fixed except I we have:
$M_e = \{I\}$
hence $|G| = |M|$.

11. THE CONJUGATION

While handling your Rubik's Cube, you may be unknowingly using the technique of conjugation.

11.1. THE CONJUGATION TECHNIQUE

The conjugation technique (CT) said if you can do something at one point then you can do it at any point! We'll take an ex to understand. Here is a formula

$T = FU^2F^2.D'[U'L']D.F^2U'F'U'$

which flips two edges (UF) and (UR) and leaves intact the other pieces of the Cube (we will say that the rest of the Cube is invariant) so according to the CT we can flip all the edges! for example if we want to flip (FR) and (UB) how to do?
Well it's very simple, just bring (FR) to (UF) and (UB) to (UR) then apply the formula then put the edges (FR) and (HD) back in their initial location, that's all !!
take a closer look:

1. F'B'R' : Bring (FR) to (UF), (UB) to (UR).
2. Apply formula: T
3. RBF : Put (FR) and (UB) back in their original location.

In other words to flip (FR) and (UB) we do:
$(FR)^{-}(UB)^{-}$ = (F'B'R') .(FU²F².D'[U'L']D.F²U'F'U') .(RBF)
Thus we can flip all the edges.

T = FU²F².D'[U'L']D.F²U'F'U'

$(FR)^{-}(UB)^{-}$ = (F'B'R')T(RBF)

The CT is always written in the format: XYX' as you noticed on the example above (F'B'R').T.(F'B'R')'. This is why the writing XYX' is called conjugation. for example: FDF' is a conjugation
Recall the inverse of a formula X' :
X = FDU'LR'B²L'U
X' = U'LB²'RL'UD'F' (read backwards and prime↔non-prime)

A K-formula is a formula that modifies a single piece of the side K thus leaving the other pieces of side K intact, it can of course modify the other sides.
E.g :

→ C = R'DRFDF' is an U-formula because it modifies a single piece in (URF) of the Up side.

→ Z = [RU] is a L-formula because it modifies a single piece in (ULB) of the Left side, but it is also a D-formula because it modifies a single piece in (DFR) of the Down side.

→ V = $FUD'L^2U^2D^2RU$ is an U-formula because it modifies a single piece in (UF) of the Up side.

11.2. CLEAN FORMULAS

Reminder: A clean (or independent) formula is a formula that does a targeted job and leaves the rest of the Cube unchanged.
The formula T is clean. Clean formulas are highly sought after because they allow the Conjugation Technique to be applied without any precautions to be taken.
But how do we find clean formulas? how do we make the clean formulas ??. Hey! well, we make them from the K-formula !!

Suppose we have a L-formula which modifies a Left-vertex and leaves other parts of the Left side invariant (it can modify the other sides of course) then we can construct a clean formula. Let's see on an example

Here is a L-formula Z = [RU] which modifies the vertex in (ULB) and leaves the side L invariant, we will build a clean formula from Z
Just take S = [ZL] = ZLZ'L'

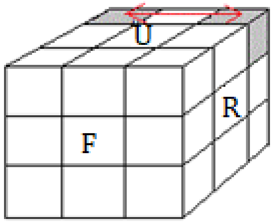
Explanation

1. We apply Z: the vertex in (ULB) is exchanged, but the rest of the Cube is perturbed

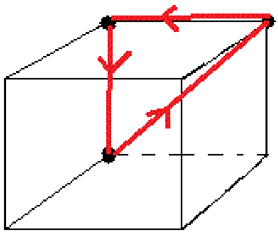
2. We bring the vertex in (DBL) -guinea pig vertex- to (ULB): L

3. We apply Z': which exchanges the vertex in (DBL) and repairs the rest of the Cube at the same time.

4. We put the vertex in (DBL) back in its initial location: L'
The formula S = ZLZ'L' moves 3 vertices (ULB)→(DBL)→(UBR) and leaves the rest of the Cube invariant.



Z = [RU]



S = [ZL] = ZLZ'L'
[RU].L[UR]L' = (ULB)→(DBL)→(UBR)

In the same idea

Let $Z^2 = [RU]^2$ which twists the vertex (ULB) and leaves the side L invariant, we will build a proper formula from Z^2
As in the previous example we take $S = [Z^2L] = Z^2LZ^2{}'L'$

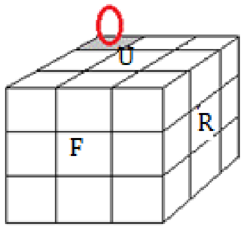
Explanation

1. We apply Z^2: the vertex (ULB) is twisted, but the rest of the Cube is disturbed

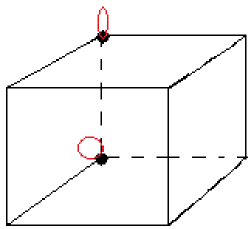
2. We bring the vertex (DBL) -guinea pig vertex- to (ULB): L

3. We apply Z^2': which twists (DBL) inversely and repairs the rest of the Cube at the same time.

4. We put the vertex (DBL) back in its initial location: L'
The formula S = $Z^2 L Z^2$'L' thus twists 2 vertices (ULB), (DBL) and leaves the rest of the Cube invariant.



$Z^2 = [RU]^2$



S = [Z^2L] = $Z^2 L Z^2$'L'

Similarly we can build a proper formula from V, it suffices to take S = [VU] = VUV'U' which flips two edges (UF) and (UR).

The way to construct a proper formula as above is called Principle of Commutation. The Principle of Commutation is always written in the format: XYX'Y' abbreviating [XY], this is why the writing XYX'Y' is called commutation

In Rubik's Cube we use 2 principles:
The Conjugation: XYX' = Y^X and
Commutator: XYX'Y' = [XY]
XYX' is called a conjugate of Y
commutator X,Y: [XY] = XYX'Y' .

12. THE MINIMAL ALGORITHM

The question... I have been asking myself the following question for a long time:
"What is the minimum number of formulas must be memorized to be able to restore the Cube?" .

It is clear that the algorithm using the least formula is not made for SpeedCubing, in SpeedCubing to go faster we memorize the maximum possible formulas (about 119 formulas !!) but here the question is not to save time but rather space, memory ... for example we store these formulas in the memory of a ROBOT and it is he who restores the Cube with adequate algorithms .

But first let's see what an algorithm is:

The algorithm
A resolution algorithm with input $\mathcal{A}$, is a finite sequences of actions:

at each stable of the resolution:
- You can hold the Cube however you want.
- We can use conjugation
- The formulas V must be in $\mathcal{A}$
- We can use the operations: V', V^n, ${}^tX$, ${}^tX^2$, ... (X=basic rotation)

An action: place, orient, slide, permute, flip, twist, ...

example: $\mathcal{A}$ ={V,C}
A1: Place the centers: (${}^tR$ ${}^tU$) V (${}^tR$ ${}^tU$) V' (${}^tU'$ ${}^tR'$) V (${}^tU'$ ${}^tR'$)
A2: Place the edges: C .L'VL
A3: Rotate vertices: V^4 C'
.....

In other words, at each step of the solution, you can use the conjugation, the inverse, the power, and you can hold the Cube as you want.

Let's ask ourselves the following question: How many formulas does an algorithm use?
So it's reasonable to say that the number of formulas the algorithm uses is the number of formulas in the input to run the algorithm.

Some part of the resolution does not need formulas, it is intuitive and we should note this part "0" it means "no need for formulas, it is intuitive"

The Cube is said to be restored by $\mathcal{A}$, or solved by $\mathcal{A}$.

GOAL: Find $\mathcal{A}$ with the least possible formulas.

The groupe of the Rubik's Cube G has 4 components (4 parts) dividing into two clans: the clan of edges (S_{12} x $\mathbb{Z}_2^{12}$) and the clan of vertices (S_8 x $\mathbb{Z}_3^8$), each clan has 2 parts: permutations represented by S , orientations represented by $\mathbb{Z}$.
This suggests that there are 4 resolution steps:

1. Place the edges
2. Orient the edges
3. Place vertices
4. Orient vertices

12.1. THE 4 FORMULAS

The idea is to have 4 formulas for the 4 steps of the resolution. With the Cube Explorer program (cube514qtm.exe) we find the following 4 formulas, which allow you to restore the Cube. The set $\mathcal{A}$ therefore contains 4 formulas

Alg0(4):
1. (UF)↔(UR) = RU'RU FD'FDF² RFR'F' R'
2. $(UF)^-(UL)^-$ = FRBLUL'UB'R'F'L'U'LU'
3. (UFL)→(UBR)→(ULB) = RUB'D'BUB'DBU²R'
4. $(UFL)^-(ULB)^+$ = L'UR'U'B'R'BLB'RBURU'

When we arrive at step (3), the edges are well arranged and therefore well placed (sig(u)=1), therefore sig(v)=1, we can thus place all the vertices with the 3-cycle (with using conjugation).

We therefore have an algorithm with 4 formulas.
These formulas are minimal but have no structure!! we do not understand what the formula does!!!
So we can restore the Cube with 4 formulas.
Reminder: the word "restore" is in the sense that we had defined: "there is an algorithm such that..."

<u>First reduction</u>

We can reduce $\mathcal{A}$ to 3 elements, using W
W = L'UR'U² LU'L'U² LRU' = (UFL,URF)(UF,UR)
We place the edges by W, then we use this same W to place the vertices, which is possible because once the edges are well placed, we will have an even number of pairs of vertices to place (parity law) so we do not disturb the edges which are already well placed.

Alg1(3):
1. (UF)↔(UR) = W
2. (UFL)↔(URF) = W
3. $(UF)^-(UL)^-$ = FRBLUL'UB'R'F'L'U'LU'
4. $(UFL)^-(ULB)^+$ = L'UR'U'B'R'BLB'RBURU'

Algorithm with 3 formulas.

<u>Second reduction</u>

We see that the last two lines (3), (4) twisting the pieces, so we wonder if we can replace them with a single formula? the answer is affirmative, indeed if we flippe 2 times an edge we return to its initial state, on the other hand it is necessary to twist 3 times a vertex to return to its initial state, therefore if we find a formula which twist 2 edges and 2 vertex we will have won, here is one:

S = LB'R'BU'FUB'F'RBR'UL'U'R = $(UF)^{+}(UL)^{+}(UFL)^{-}(UBR)^{+}$

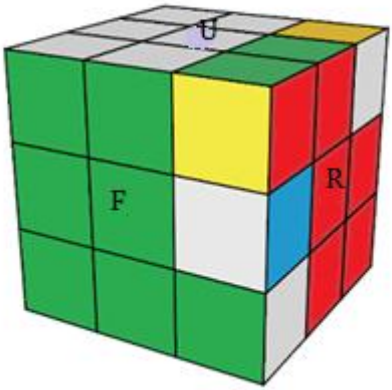
Alg2(2) :
1. (UF)↔(UR) = W
2. (UFL)↔(URF) = W
3. $(UF)^{-}(UL)^{-}$ = S
4. $(UFL)^{+}(UBR)^{-}$ = S^2

We thus reduce $\mathcal{A}$ to 2 elements, we have an algorithm with 2 formulas !! It's really not bad but we wonder if we can do better? i.e. to have a one-element algorithm !!?? and then if we can make the formulas more comprehensive? more structured?? The answer is yes" .

The [RU] bracket

The problem with these formulas is that they have no structure, you can't see what they do and they are not easy to remember!! .
So we would like to have structured formulas to understand what is happening, what the formula does and easy to remember...

We examine the formula [RU], we want to build an algorithm around [RU] that is to say at each step of resolution it must appear the bracket [RU].
Let's take a good look at what this switch does. [RU] acts on the Cube as a kind of 'Z' and we denote it Z = [RU]



[RU]=(FR,UR,UB)(URF,DFR)(ULB,UBR)

[RU] acts on the Cube:

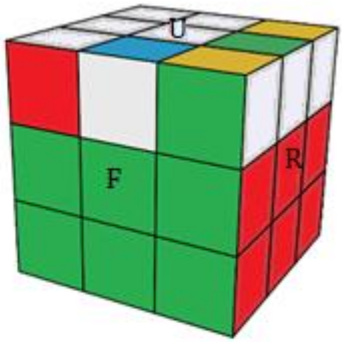
Edges: (FR)→(UR)→(UB) = (FR,UR,UB)

Vertices: (URF)↔(DFR).(ULB)↔(UBR) = (URF,DFR)(ULB,UBR)

[RU]= (FR,UR,UB)(URF,DFR)(ULB,UBR)

We will build an algorithm around the [RU]. The idea is to have 4 "structured" formulas which correspond to the 4 stages of the resolution:
And there you have it, if you look closely, you have everything you need!!!

From [RU] if you want everything to happen on the Up side, just do F[RU]F' , therefore this moves 3 edges of side Up



(UF)→(UR)→(UB) = F[RU]F'

To orient the edges it is enough to observe O=F[RU]F', O leaves only one good edge, we bring the 2nd good edge by H, because we only want to have 2 bad edges, that is to say we takes F[RU]F'.U = J .

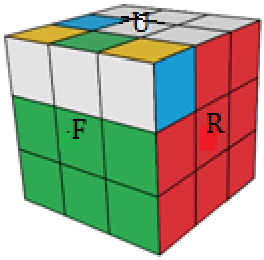
J swaps 2 edges so we can use J to place all the edges.
To reverse the edges we will examine J closely.
J permutes 2 edges indeed are x_1, x_2 , x_3 in the following way:

$$x_1, x_2, x_3 \xrightarrow{J} x_2^-, x_1, x_3^-$$

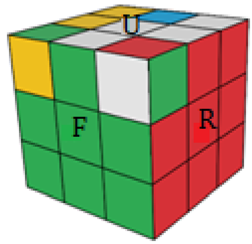
x_1, x_2 swaped and x_2^- flipped so just do J² and we will flip the edges:

$$x_1, x_2, x_3 \xrightarrow{J} x_2^-, x_1, x_3^- \xrightarrow{J} x_1^+, x_2^-, x_3^{--} = x_3$$

x_1 and x_2 are flipped
J² flips 2 edges.



J = F[RU]F'.U

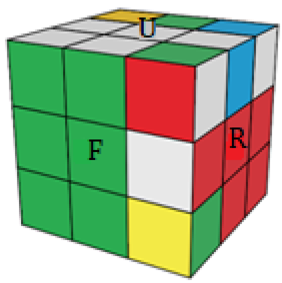


$(UL)^{-}(UB)^{-} = (F[RU]F'.U)^2$

Now let's see for the vertices:
[RU] modifies a single piece of side L, it exchanges the vertex (ULB) of side L, with (UBR) another vertex of the Cube, we can easily make a 3-cycle-vertices like this:

[[RU],L] = [RU].L'[UR]L = (ULB)→(UFL)→(UBR)
Q = [RU].L'[UR]L

To orient the vertices, let's observe $[RU]^2$:

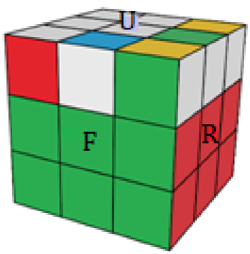


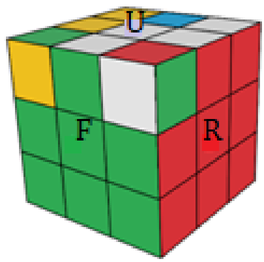
$[RU]^2$

$[RU]^2$ modifies a single piece of side L, it twists the vertex (ULB) of side L, just place the vertex (UFL) in (ULB) and apply the inverse of $([RU]^2)' = [UR]^2$ to rotate 2 vertices:
$[[RU]^2,L] = [RU]^2 .L'[UR]^2L = (ULB)^{-}(UFL)^{-}$
$T= [RU]^2 .L'[UR]^2L$

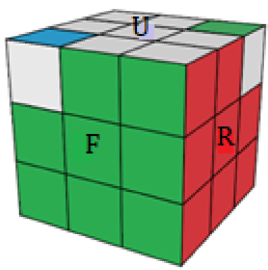
So we have $\mathcal{A}$ = {J, Q, T}
And the associated Alg3(3) algorithm:
1. (UL)↔(UB) = J
2. $(UL)^{-}(UB)^{-} = J^{2}$
3. (ULB)→(UFL)→(UBR) = Q
4. $(ULB)^{+}(UFL)^{-}$ = T

step (3) is possible because the edges are well placed so the state of the vertices is even, so we can place them all by a 3-cycle.
Here the formulas are much clearer and more structured so easy to memorize and we see what a formula does , moreover they are built around [RU].

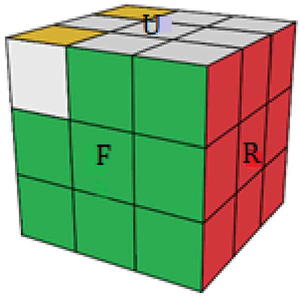




(UB)→(UF)→(UR) = F[RU]F' $(UL)^{-}(UB)^{-}$ = $(F[RU]F'.U)^{2}$



(ULB)→(UFL)→(UBR) = [RU] .L'[UR]L



(UFL)$^+$(ULB)$^-$ = [RU]² .L'[UR]²L

Note: From O, J, Q, T we can adapt to have a quite reasonable resolution algorithm, this is what I did:

1. Place the edges:
   * Bottom: (UF)→(DF) = F²
   * Equator: (UB)→(FR) = [RU]
   * High: (UB)→(UF)→(UR) = F[RU]F' = O

→ If there are 2 adjacent edges: hold the cube (UL), (UF) adjacent and apply HO.
→ If we have 2 opposite edges: hold the cube (HP), (HA) opposite and apply O ⇒ we return to the adjacent case.

2. Flip edges: (UL)$^-$(UB)$^-$ = (F[RU]F'.U)²
3. Placing the vertices:
   (ULB)→(UFL)→(UBR) = [RU].L'[UR]L
4. Twist Vertices:
   (UFL)$^+$(ULB)$^-$ = [RU]² .L'[UR]²L

Four Solving Equations

1. (UL)↔(UB) = J
2. (UL)$^-$(UB)$^-$ = J²
3. (ULB)→(UFL)→(UBR) = Q
4. (ULB)$^-$(UFL)$^-$ = T

We will also say the 4 equations of restoration.

Here, we go up the Cube with an algorithm built around the [RU] switch, each step we see [RU] appear.

a) The algorithm is simple to understand: We place the edges then orient them. We place vertices and then orient them.

b) 3 formulas only.

c) The formulas are structured so easy to understand and memorize.

The minimal algorithm J

We found a "structured" algorithm with 3 formulas, recently I saw that we can go even further...

The formula J = F[RU]F'U is really interesting, we effect:

1. J swaps 2 edges and 2 vertices
2. J^2 (or J^6) flips 2 edges
3. J^4 twists 3 vertices

As J also permutes 2 vertices, it can therefore be used to place the vertices (after having placed the edges), and J^4 twists 3 vertices which makes it possible to orient all the vertices. and here it is wonderful, we can do everything with J !!...

* We start by placing all the edges by J

* After having placed all the edges, we then place the vertices thanks to J too! we place (ULB)↔(URF) (with the conjugation) taking care not to touch the edges (UL) and (UB) i.e. not to use the rotations U, L and B, that is possible because the vertices are now in an even state (sig(u)=1=sig(v)).

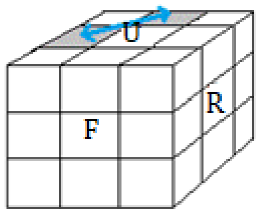
* We flip the edges by J^2 (or J^6),

* We twist the vertices by J^4

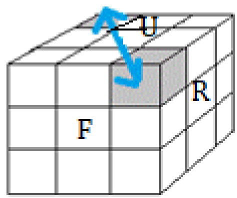
Hallelujah!! a single formula to restore the Cube!!! ...
$\mathcal{A}$ = {J = F[RU]F'.U}

And the associated minimal algorithm:

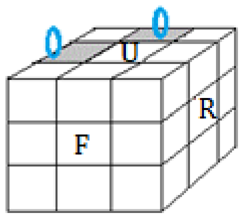
- (UL)↔(UB) = J
- (ULB)↔(URF) = J
- $(UL)^{-}(UB)^{-} = J^2$ (ou J^6)
- $(ULB)^{+}(UFL)^{+}(URF)^{+} = J^4$



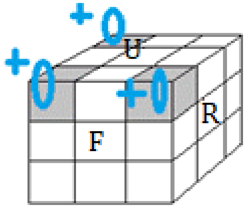
(UL)↔(UB) = F[RU]F'.U



(ULB)↔(URF) = F[RU]F'.U



$(UL)^{-}(UB)^{-} = (F[RU]F'.U)^2$



$(ULB)^{+}(UFL)^{+}(URF)^{+} = (F[RU]F'.U)^{4}$

OK, it is truly extraordinary that the Cube can be restored with only a formula built around a [RU] !! It's amazing isn't it?
This algorithm in practice is not really usable but in theory it can be very useful, e.g. one can program a Robot to solve the Cube, apart from the algorithm one has to store the formulas in memory, so if the memory costs dear one can save money by using the minimal algorithm with only one formula to store.
Note: J^4 and J^6 are proper and therefore independent (the order of execution does not intervene)
We have 4 equations to restore the Cube, these equations use only one formula J

- (UL,UB) = J
- (ULB,URF) = J
- $(UL)^{-}(UB)^{-} = J^2$
- $(ULB)^{+}(UFL)^{+}(URF)^{+} = J^4$

It is really striking that there is an analogy with Maxwell's 4 equations in electromagnetics

- $\mathrm{div}(B) = 0$
- $\mathrm{rot}(E) = -\frac{\partial B}{\partial t}$
- $\mathrm{div}(E) = \frac{\rho}{\varepsilon}$
- $\mathrm{rot}(B) = \mu j + \varepsilon\mu \frac{\partial E}{\partial t}$

NOTE :

We could say that J and Z=[RU] are the two constants of the structure M, and they are related by
$J = Z^F U$
Finally, the minimal algorithm is by definition:

1. Place the edges : Q
2. Place the vertices : Q
3. Flip the edges : Q^2
4. Twist the vertices : Q^4

where Q is a formula, the formula Q in the minimal algorithm is named first formula

Application: Let's follow the following scenario:

A team of explorers is on Mars, following an explosion the entire communication system is off, only the SMS channel remains to communicate with Earth.

The oxygen system is damaged and there are only 45 minutes left to breathe. Fortunately the crew discovered an oxygen system installed by extra-terrestrials but to trigger it you have to solve a Rubik's Cube!!!

No member of the team knows how to solve it, except the Robot R18, although the solving program is intact but the Robot has lost a memory part of its data, more precisely the entry of a Rubik's Cube formula...

-Hello! Hello ! Earth....Help....

OK, we have to send the Robot a formula? but what formula? and then the communication is expensive a character sent costs 100.000$!!!
So how to save the crew at minimum cost?

Solution: We send the Robot by SMS the formula F[RU]F'U !!!
8 characters...

or $Z^F U$, only 3 characters if the Robot is equipped with the latest version of the solving algorithm!!

Important note: We often hear or in the media: "The Rubik's Cube can be restored in a maximum of 20 moves". it's not very clear !! ...

→ First of all, it is not the word "move" that should be used because a move is a finite sequence of rotations!

→ It should be said 20 rotations, but should it still be specified that R^2 counts 1 rotation, e.g. $UR^2B'L^2F$ is worth 5 rotations.

More precisely in August 2014 Tomas Rokicki and Morley Davidson prove the following theorem:

$\forall$s state $\exists$V formula, with $|V| \leq 26$ such that $s \bullet V = e$

The problem is that we don't find V easily for any s !!.
So it's a different problem with the minimal algorithm. In the problem the minimal algorithm we seek the minimum number of formulas to memorize to get by. In the SpeedCubing, to go faster we memorize around 60 to 120 formulas!!!

The number 26 is called the number of God (for the Rubik's Cube) because the Cube can be restored from any state up to 26 rotations. It took 33 years to find that number!!

One wonders which are the states (the furthest) which require 26 rotations to get out of it? Michael Reid found this formula on the computer

$\Pi = U^2D^2L\ F^2\ .U'DR^2\ BU'D'R.\ LF^2R\ UD'\ R'LUF'B'$

and proved that it is the shortest formula $|\Pi| = 26$ this state is named the SuperFlip4Spot, so it is one of the furthest states from the Rubik's Cube.

Why it's called SuperFlip4Spot because it's the product of SuperFlip and 4Spot.
SuperFlip4Spot = SuperFlip. 4Spot = 4Spot . SuperFlip
No other states of length 26 are known, apart from this SuperFlip4Spot!!

We already remember in July 2010, Tomas Rokicki, Herbert Kociemba, Morley Davidson, and John Dethridge demonstrated that the diameter of the Rubik's Cube is 20f ($|F|^2=1$) (30 years of hardship), that means:

$\forall$s state $\exists$V formula, with $|V| \leq 20f$ such that $s \bullet V = e$

Here too we are looking for the furthest formulas V with $|V| = 20f$

* The SuperFlip (12 flipped edges) is one of the furthest states in f-rotation, indeed in 1992 Dik .T. Winter found a SuperFlip formula
Φ = FBU²RF².R²B²U'DF.U²R'L'UB².DR²UB²U

of length $|\Phi| = 20f$ it was necessary to wait until 1995 for Michael Reid to demonstrate that it is the shortest formula in f-rotation. In summary, the SuperFlip is one of the furthest states in f-rotation

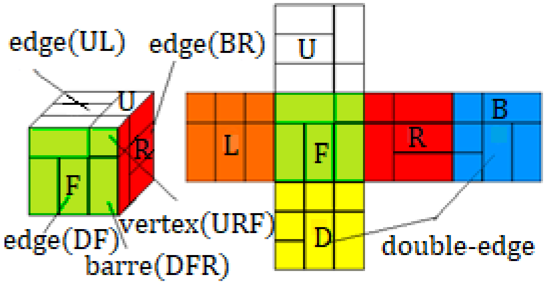
Michael Reid came up with this formula:
Φ = R'U²BL'.FU'BDF.UD'LD².F'RB'DF'.U'B'UD'

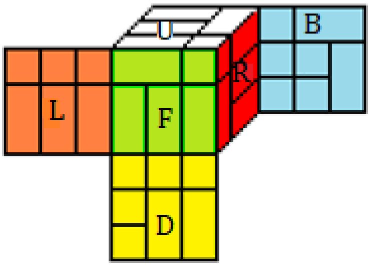
in 1995 by computer for the SuperFlip and it was Jerry Bryan who demonstrated (1995) that it is the shortest formula $|\Phi| = 24$.

* The SuperFlip plays an important role, it is the only one (apart from the identity e) in the center of G (the Rubik's Cube group) for a given formula, it is always interesting to have one of the shortest writing.
Other examples of the minimal algorithm

Here we can give 2 examples of the minimal algorithm.

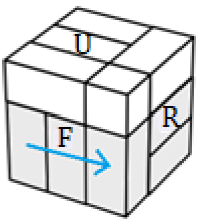
The Cube Bandage
The Bandage Cube is a Rubik's Cube bandage invented by Meffert around 1981.



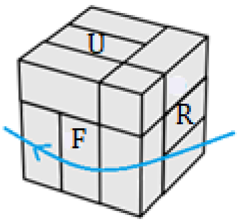


Bandage Cube

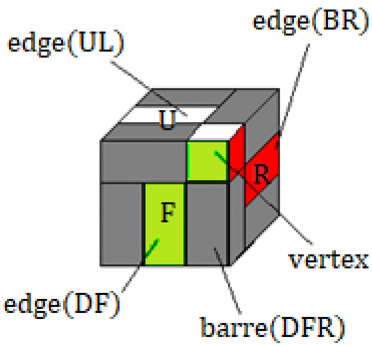
Les rotations :



D*



^{t}U



Position START

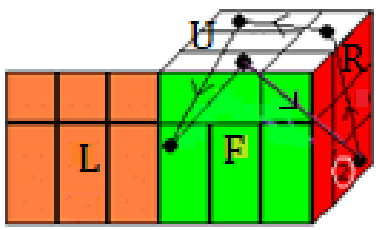
It is amazing that the Bandage Cube has a minimal 2-formula algorithm.

We start by placing ourselves in the START position.
START position:

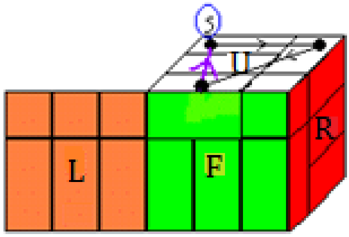
1. The double edge ┘ must be in their location (DB).
2. The small vertex must be in position (URF) (even badly oriented, but it will be well oriented automatically)
3. The other edges should be neatly arranged (DL), (BR), (DF), (UL).
4. That we can do the rotations U,R,F

The minimal algorithm $\mathcal{A}$ = {Q, T}

We have two formulas:

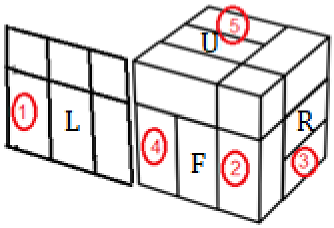


Q = FRUF .R'F²U'



T = F'U .LF'L'F².RU'R'

The bars are arranged in the following order: (DBL), (DFR), (DRB), (DLF), (ULB)



Arrange in this order

1) Setting the bar (DBL):
Find the bar (BL) (recognizable by its 3 colors). This bar must be brought to position (ULB) using Q or (^{t}R^{t}U) Q (^{t}U'^{t}R') (when this bar is in (DFR))
then once in (ULB) we place it with: (RD*R') Q (RD*'R')
Explanation: The formula Q covers the cube except 2 zones: (DFR) and (DBL), but thanks to the conjugation we manage to place (DBL)

2) Set the bar (DFR):
Find the bar (DFR) (recognizable by its 3 colors). This bar must be brought into position (DLF) using Q
then once in (DLF) we place it with: (^{t}U'^{t}R') Q (^{t}R^{t}U)
Note: (^{t}U'^{t}R') means "hold the Cube correctly" before applying Q .

3) Place the bar (DRB):
Find the bar (DRB) (recognizable by its 3 colors), nothing to complicate, we place it with: Q
Note: No problem moving the bar (DRB) to its place because the bar is in the area covered by Q

4) Place the bar (DLF):
Find the bar (DLF) (recognizable by its 3 colors). This bar must be brought into position (ULB) using: T
then once in (ULB) we place it with:
(^{t}R^{t}U) Q (^{t}R^{t}U) Q' (^{t}U'^{t}R') Q (^{t}U'^{t}R')
Note: This complicated formula swaps (DLF,ULB)(UFL,UBR)

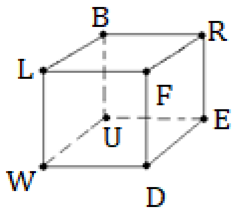
5) Set the bar (ULB):
Find the bar (ULB) (recognizable by its 3 colors). We place it with T (we apply T several times if necessary).
Noticed:
- The bar (ULB) is necessarily at the Up and T (generates a 3-cycle vertices) covers the entire Up so no problem moving it to (ULB).
- Sometimes we use the inverse formula to move faster, it depends where the bar is and where we want to move it.
For example if we want
to move (UFL) to (DLF) it is better to use Q' than Q.

The Skewb

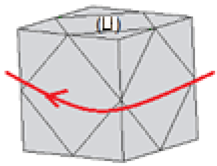
The Skewb is a fairly well-known twist, it too has a minimal algorithm with one formula!

Skewb

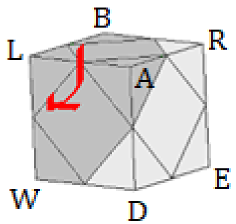
Moves :



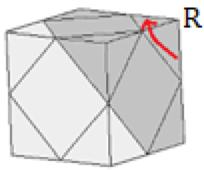
Rotations



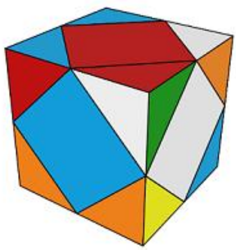
^{t}U



L =120°



R=120°

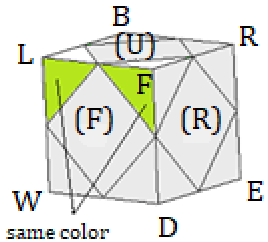


Position START

We chose Up=white, Front=green, Right=red, ...
We place ourselves in the START position
START position (isolate vertices)
The 4 white vertices are at the Up.

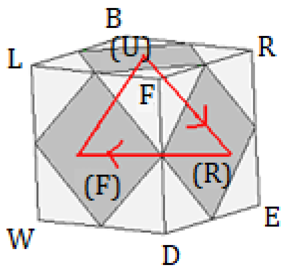
The minimum algorithm $\mathcal{A}$ = {[RL']}

A- Place the Up vertices
Correctly place the Up vertices with [RL']: F↔B = [RL']



Note: Once the Top vertices are well placed, the Low vertices are automatically well placed.

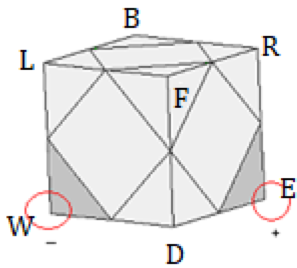
B- Placing centers
The centers (marked by the vertices) are moved by the following formula:
Circular permutation of 3 centers:
(U)→(R)→(F) = $[RL']^2$



(U)→(R)→(F) = $[RL']^2$

We first try to place the center High (white) then to have 2 centers adjacent to (U). The cube is held as needed to apply the formula.

C- Twist vertices
For this we use the following formula:
Twist 2 vertices:
$(W)^{-}(E)^{+}$ = $[RL']^{3}$ ${}^{t}U^{2}$ $[RL']^{3}$



$(W)^{-}(E)^{+}$ = $[RL']^{3}$ ${}^{t}U^{2}$ $[RL']^{3}$

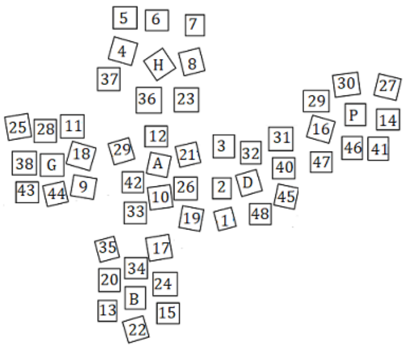
Here too, the cube is held as needed to apply the formula. The Skewb has a minimal one-formula algorithm like the Rubik's Cube!

13. THE PERMUTATIONS OF THE STICKERS (Λ,.)

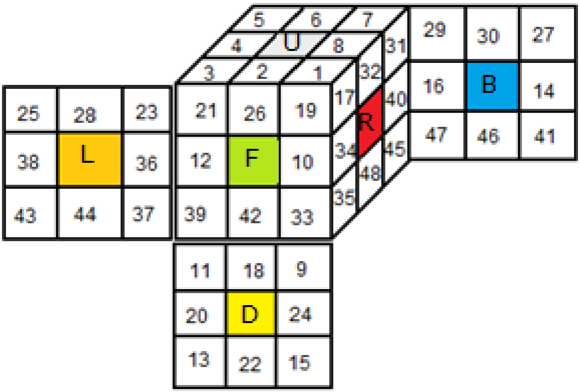
There is another important antagonism of the Rubik's Cube: Permutations (of stickers).
Let S_{48} the set of permutations of the 48 stickers with law '.'
pq = p.q = qop ; o=function composition.

Imagine on one hand we have stickers scattered on the table and on the other hand we have a monochrome Rubik's Cube fixed thus giving 6 rotations {U,D,F,B,L,R}. When we turn a face, we see the stickers move.

V = URL'DF



p_v



So there is a morphism of (M, .) to $(S_{48}, .)$

$M \to S_{48}$
$V \to p_V \in S_{48}$

Let
$\Lambda = < p_U , p_D, p_F, p_B, p_L, p_R > \subset S_{48}$
$\Lambda^+ = < p_U , p_D, p_F, p_B, p_L, p_R, p_\Gamma, p_\Psi, p_\Omega > \subset S_{48}$

$(\Lambda,.)$ the permutations groupe of the Rubik's Cube
$(\Lambda^+,.)$ the group of extended permutations of the Rubik's Cube .

We must have :
$|\Lambda| = |G| = |M|$
$|\Lambda^+| = |G^+| = |M^+|$

Standard permutations

2-cycle-edges, 3-cycle-vertices:

p_U = (2,4,6,8)(26,28,30,32)
(1,3,5,7)(17,21,25,29)(19,23,27,31);
p_D = (18,24,22,20)(42,48,46,44)
(9,15,13,11)(33,45,41,37)(35,47,43,39);
p_F = (2.34,18.36)(26.10,42.12)
(1.35,11.23)(9.37,3.17)(19.33,39.21);
p_B = (6,38,22,40)(30,14,46,16)
(7,25,13,45)(29,27,41,47)(31,5,43,15);
p_L = (4,12,20,14)(28,36,44,38)
(3,39,13,27)(21,11,41.5)(23,37,43,25);
p_R = (8,16,24,10)(32,40,48,34)
(1,29,15,33)(17,31,45,35)(19,7,47,9);

Extended permutations

p_Γ = (2.26);
p_Ψ = (1,17,19);
p_Ω = (2.8)(26.32);

Slice permutations

p_u = (10,36,14,40)(34,12,38,16);
p_r = (2,30,22,42)(26,6,46,18);
p_f = (4,32,24,44)(28,8,48,20);

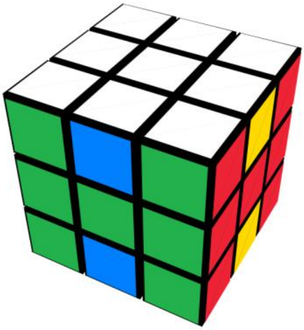
The GAP
https://www.gap-system.org/index.html
https://fan2cube.fr/gap_rubikcube.txt

In the cmd window we go to the GAP folder
C:\Users\name> cd\gap4r4\bin
C:\gap4r4\bin>gap < gap_rubikcube.txt

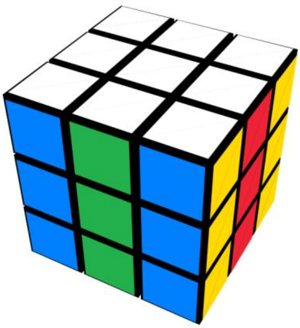
```
gap> (1,3,5,7)(2,4,6,8)(17,21,25,29)(19,23,27,31)(26,28,30,32)
gap> (9,15,13,11)(18,24,22,20)(33,45,41,37)(35,47,43,39)(42,48,46,44)
gap> (1,35,11,23)(2,34,18,36)(3,17,9,37)(10,42,12,26)(19,33,39,21)
gap> (5,43,15,31)(6,38,22,40)(7,25,13,45)(14,46,16,30)(27,41,47,29)
gap> (3,39,13,27)(4,12,20,14)(5,21,11,41)(23,37,43,25)(28,36,44,38)
gap> (1,29,15,33)(7,47,9,19)(8,16,24,10)(17,31,45,35)(32,40,48,34)
gap> (2,26)
gap> (1,17,19)
gap> (2,8)(26,32)
gap> (10,36,14,40)(12,38,16,34)
gap> (2,30,22,42)(6,46,18,26)
gap> (4,32,24,44)(8,48,20,28)
gap> <permutation group with 9 generators>
gap> <permutation group with 6 generators>
gap> gap>
gap> |Lamda+| = 519024039293878272000
gap> |Lamda| = 43252003274489856000
gap> N = 12
gap> |G+| = 519024039293878272000
gap> |G| = |G+|/N = 43252003274489856000
gap> Indice = |G+|/|G| = 12
gap> gap> >
C:\GAP4R4\bin>
```

14. PRETTY PATTERNS

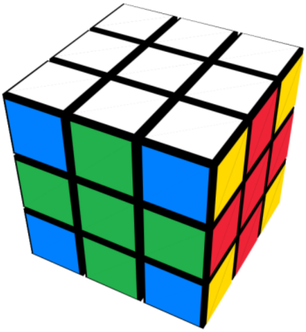
In this chapter we have a collection of pretty patterns



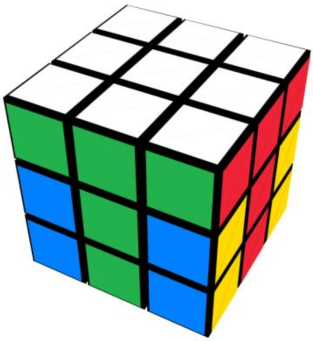
4U = $(U^2R^2L^2F^2\ B^2)^2$



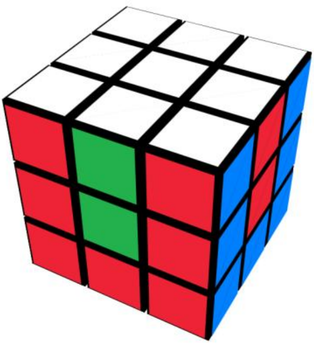
4I = $(F^2R^2B^2)^2$



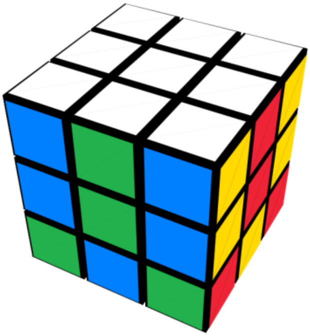
4plus = UF²R²F² U'DL² B²L²D'



4T = RLU²R'L'FBU²F'B'U²

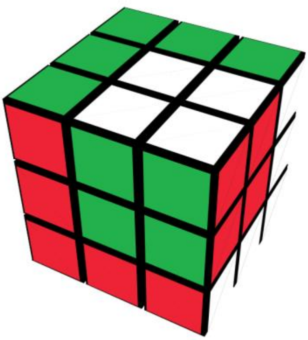


4U = FR'B'D²L' UD'B D²RFL'

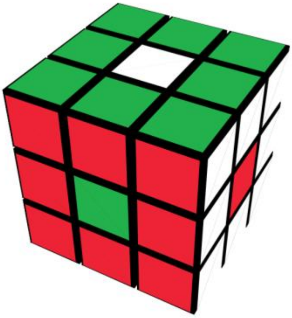


4Y = $(R^2F^2L^2)^2 D^2$

6flag = LUF^2RL' U^2B'UD B^2LFB' R'LF' RL'R

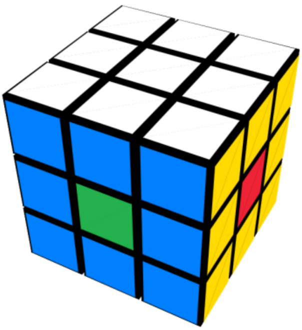


2Cube = FLFU'RUF^2L^2U'L'BD'B'L^2U



6Spots = UD'RL'FB'UD'

3Cube = U'L'U'F' R² B'RF UB²UB'L U'FURF'



4Spot = Ω = F²B²UD' R²L²UD'
 One of the factors of SuperFlip4Spot Π

SuperFlip = Φ = R'U²BL'FU'BDFUD'LD²F'RB'DF'U'B'UD'

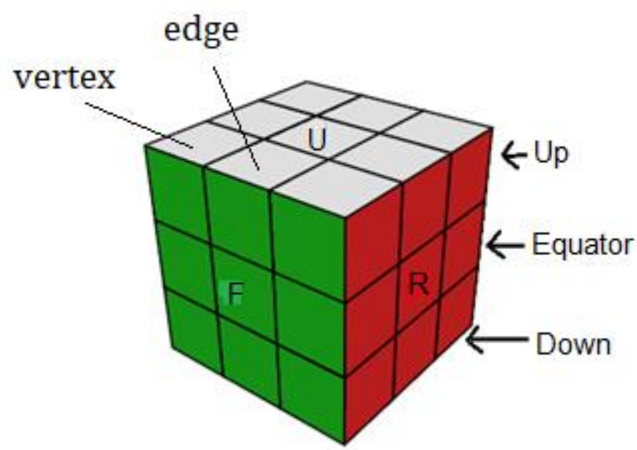
SuperFlip4Spot = Π = U²D²LF²U'DR²BU'D'RLF²R UD'
R'LUF'B'
Π = ΦΩ = ΩΦ

The SuperFlip4Spot is a SuperFar (length = 26) the only one we know !

15. SOLUTION OF RUBIK'S CUBE

A book on the Rubik's Cube without a solving algorithm is not a book on the Rubik's Cube! it is therefore reasonable to provide a resolution algorithm in this last chapter.

The algorithm [UR] Autor : Morphocode
Year : 2011



This method is divided into 5 steps, and each step uses 2 formulas so in all 10 formulas to restore the Cube. The strategy is very classic, layer by layer. First we finish the Down (1st layer) then the Equator (2nd layer) then the Up (3rd layer)

- Finish the Down : arrange the edges then the vertices .
- Finish the Equator .
- Finish the Up : arrange the edges then the vertices .

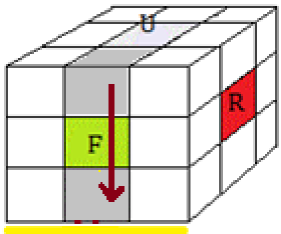
To fix the ideas we will take a standard Rubik's Cube with :

(U)p=(w)hite, (D)own=(y)ellow, (F)ront=(g)reen,
(B)ack=(b)lue, (L)eft=(o)range, (R)ight=(r)ed

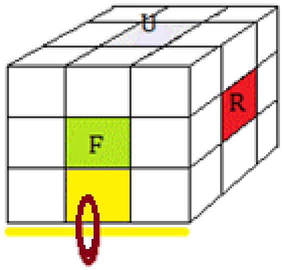
15.1. ARRANGE THE DOWN-EDGES

We are going to arrange (place and orient) the Down-edges, that is making the Down-Cross .

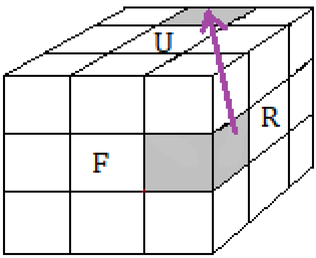
<u>Green center in front of you</u>: Find the edge (yg), then place it just above i.e. in (UF) and then do F² to come down the edge.
If (DF) has a bad orientation, we can flip it with the formula:
$(DF)^{-}$ = DRD'F



(UF)→(DF) = F²



$(DF)^{-}$ = DRD'F

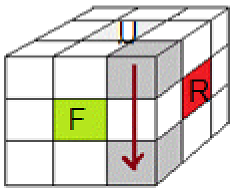


[UR]

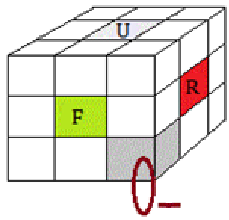
15.2. ARRANGE THE DOWN-VERTICES

We are going to arrange (place and orient) the Down-vertices, that is finish the Down side.
Find the vertex (ygr), then place it just above i.e. in (URF) (see fig) then come down it with the formula :
[UR] = URU'R'

If (DFR) has a bad orientation, we can twist (2 times if necessary) with the formula:
$(DFR)^{-}$ = $[UR]^2$



(URF)→(DFR)=[UR]=URU'R'



$(DFR)^{-}$ = $[UR]^2$

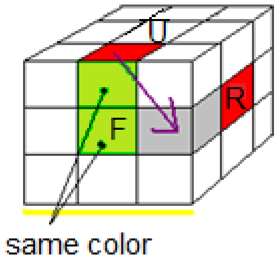
<u>Note</u>: If a vertex is in a bad location, we dislodge it by putting anything in it !! .
We do the same thing for the other Down-vertices (yellow vertices)

15.3. ARRANGE THE EQUATOR-EDGES

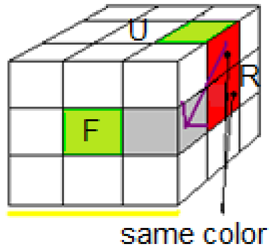
Find an equator-edge, that is an edge that has no color w(hite) = U(p) , it is necessary to place the edge as required by the formula before applying it:
Depending on the case, the corresponding formula is applied:

* (UF)=(gr) , F=green: (UF)→(FR) = [UR][U'F'] (we could say: [UR]=prepare and [U'F']=place)

* (UR)=(gr) , R=red: (UR)→(FR) = [U'F'][UR] (we could say: [U'F']=prepare and [UR]=place)



(UF)→(FR)=[UR][U'F']



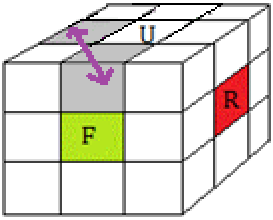
(UR)→(FR)=[U'F'][UR]

<u>Note</u>: If an edge is in a bad location, we dislodge it by putting anything in it !! .

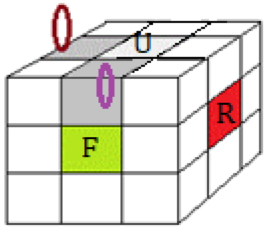
15.4. ARRANGE THE UP-EDGES

We place the Up-edges thanks to the formula: F[UR]F'U'
(UL)↔(UF) = F[UR]F'U'

<u>NOTE</u> : If we have 2 opposite edges to place, we simply apply the same formula F[UR]F'U' and we return to the case 2 adjacent edges to place. It is important do not use the conjugations because the formula is not clean (it will destroy the Down side !!).
The Up-egdes are now in its location and now we're going orient them.
Flip 2 adjacent edges : $(UL)^-(UF)^-$ = $(F[UR]F'U')^2$



(UL)↔(UF) = F[UR]F'U'



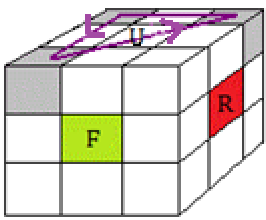
$(UL)^{-}(UF)^{-}$ = (F[UR]F'U')²

NOTICE: If we have tow opposite edges to flip, we simply apply the same formula (F[UR]F'U')² and we return to the case of the tow adjacent edges to flip. becareful do not use the conjugations because the formula is not clean (it will destroy the Down side !!)

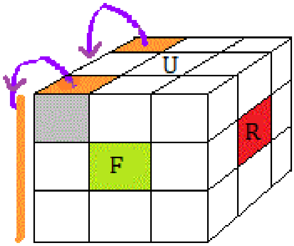
* When flipping the Up-edges, it is impossible to flip a single edge (without touching the rest of the Cube). In fact, the law of flips says that when you flip the edges, you always flip two edges.

15.5. ARRANGE THE UP-VERTICES

We will place the Up-vertices with the formula:
(UFL)→(ULB)→(UBR) = [UR] .L'[RU]L
We orient the vertices with the formula: get down color Up to the Left side
$(UFL)^{-}(ULB)^{+}$ = [UR]² .L' [RU]²L



(UFL)→(ULB)→(UBR) = [UR] .L'[RU]L



$(UFL)^{-}(ULB)^{+}$ = [UR]² .L'[RU]²L

Note : This formula is clean, so we can use conjugation without taking any precautions.

* When placing Up-vertices, it is impossible to swap two vertices (without touching the rest of the Cube). Indeed the law of parity says that when one permutes two vertices one is obliged to also permute two edges and vice versa.

* When twisting the Up-vertices, it is impossible to twist a single vertex +1 or -1 (without touching the rest of the Cube). Indeed the law of twists says that when we twist the vertices we twist:
- Either two vertices of opposite direction: (+1, -1),
- Or three vertices of the same direction: (+1, +1, +1), (-1, -1, -1).

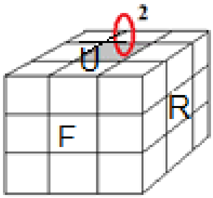
15.6. TWISTING THE CENTERS

Here are two additional formulas to twist the centers, if ever your centers are oriented like : Hello Kitty Cube, Fisher Cube ...

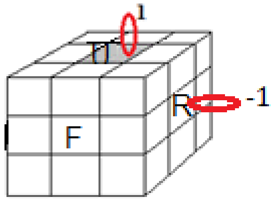
Twist the Up-center to 180°:
$(U)^{++}$ = (URLU²R'L')² ;(14*)

Twist the Up-center 90° the Right-center to -90°:
$(U)^{+}(R)^{-}$ = Uru'r' .U'rur'

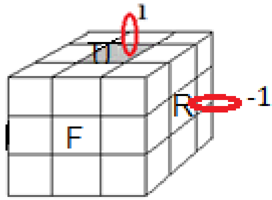
$(U)^{+}(R)^{-}$ = U'DBF'R'LURL'FB'D'UR' ; (14*)



$(U)^{++} = (URLU^2R'L')^2$



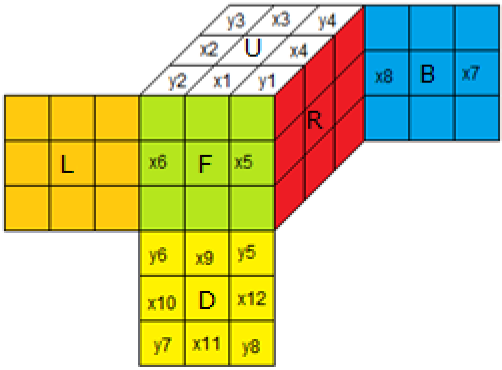
$(U)^{+}(R)^{-}$ = Uru'r' .U'rur'



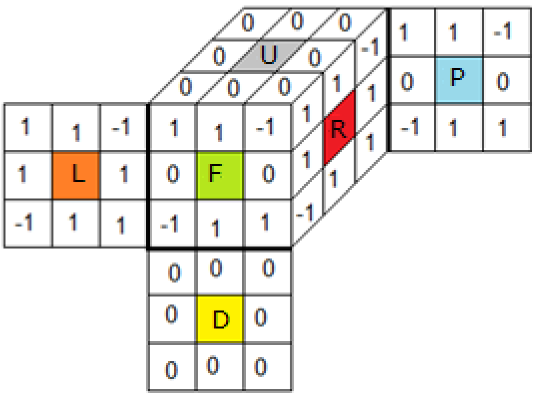
$(U)^{+}(R)^{-}$ = U'DBF'R'LURL'FB'D'UR'

And there you go, now the Rubik's Cube has no more secrets for you ......

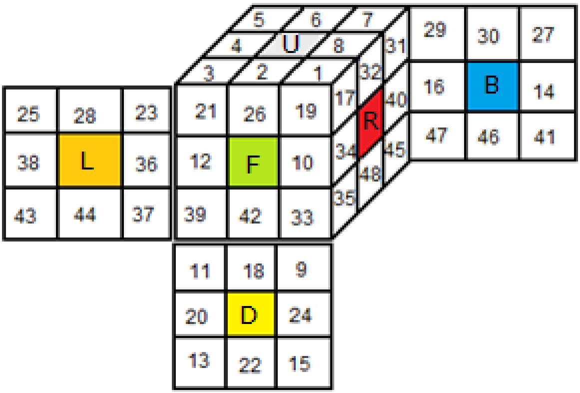
16. ADDITIONAL FORMULAS



Numbering diagram
$x = (x_1, x_2, \ldots, x_{12})$
$y = (y_1, y_2, \ldots, y_8)$



Orientation diagram



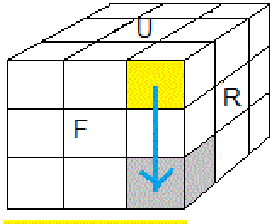
sticker diagram
$x_i = (2i, 2i+24)$
$y_i = (2i-1, 4i+13, 4i+15)$

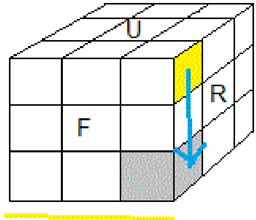
This is the Alibaba's cave !! here you will find shortcuts, formulas to go faster ....

I. <u>Shortcuts</u>
You've succeeded to restore the Cube with these 10 formulas, but maybe you want to go faster. There are shortcuts ! You just have to learn by heart and adapt to each situation.

<u>Get down the vertex (URF)→(DFR)</u>

The vertex (URF) can have only 3 situations:
1. if the Down color (yellow) is on the Front we go directly down by: [UR]
2. if the Down color (yellow) is on the Right, we use: [U'F ']
3. if the Down color (yellow) is on the Up, rotate it:

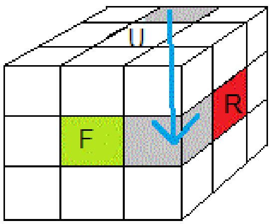


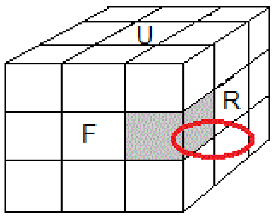


(URF)→(DFR)=[UR]

(URF)→(DFR)=[U'F']

Place : (UB)→(FR) = [R'F][UR]' = [R'F][RU] (discovered by me, commutator Y=[R'F] and Z=[UR])
Flip edge (FR): $(FR)^{-}$ = $(RU^2R'U)^2$ .F'U'F



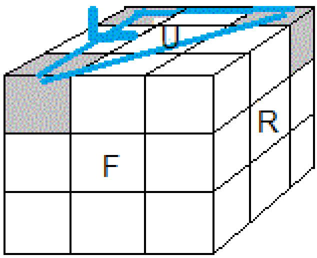


(UB)→(FR) = [R'F][RU]

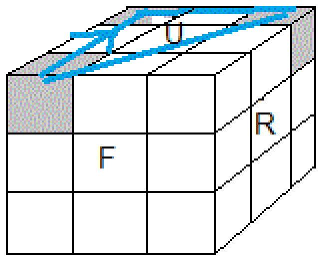
$(FR)^{-}$ = $(RU^2R'U)^2$ .F'U'F

II. <u>Inverse formulas</u>

Inverse 3-cycle vertices



(ULB)→(UFL)→(UBR)
=[UR].L'[RU]L



(ULB)←(UFL)←(UBR)
=L'[UR]L.[RU]

III. <u>Swap vertices and edges</u>

Glide: (UFL)↔(UBR).(UF)↔(UR) = F'UF'U'.R'DR'D'.R^2[F'R']F
Glide: (ULB)↔(UBR).(UF)↔(UR) = [R'U^2]R'FR.
UR'U'R'.F'R^{2}U'
Glide: (ULB)↔(UBR).(UF)↔(UR) = [BU]B'R .B^{2}U' .[B'U']B'R'
Glide: (ULB)↔(UBR).(UF)↔(UR) = UR^{2}U'R^{2}D.B^{2}L^{2}UL^{2}D'.B^2

IV. <u>Formulas for edges</u>

We do not touch the vertices of course

- (UL)→(UR)→(UB) = R^2.U'R'U'.(RU)^2.RU'R (glide without disturbing the vertices)
- (UF)→(UB)→(UR) = R^2.UFB'.R^2.BF' U.R^2 ; glide 3 edges
- (UF)$^-$(UR)$^-$ = FU^{2}F^2 .D' [U'L']D .F^{2}U' F' U' (clean formula)
- (UF)$^-$(UR)$^-$ = (FUD'L^{2}U^{2}D^{2}R) U (R'D^{2}U^{2}L^{2}DU'F') U' (commutator, clean formula)
- (UF)↔(UR).(DF)↔(DR) = D^{2}RL' DU' RUD' F^{2}LR' D'
- (UF)$^-$(UB)$^-$ = FU' RF' U.RL'.B' UR' BU'.LR'
- (UF)↔(UB).(FR)↔(BR) = (R^{2}U^2)^3

V. <u>Formulas for vertices</u>

- (URF)→(UBR)→(ULB) = $BL'BR^2\ .B'LBR^2\ .B^2$ (glide without disturbing the edges)
- (ULB)→(UFL)→(URF) = $FR'FL^2\ .F'RFL^2\ .F^2$ (glide without disturbing the edges)
- (URF)→(UBR)→(UFL) = ULU'.R'.UL' U'.R (not disturbing the edges)
- (ULB)→(UFL)→(UBR) = $LF'LB^2\ .L'FLB^2\ .L^2$;glide 3 vertices
- (URF)↔(UFL) = BU' F' UB' U' FU^2
- (URF)↔(UBR) = $(R^2U)^2(R^2U^2)^2\ .F^2U'\ F^2UF^2U'$ (disturbing the edges)
- $(URF)^+(UBR)^+(UBL)^+$ = $RUR'U\ .RU^2R'U^2$ (disturbing the edges)
- (ULB)→(URF)→(UBR) = $R^2\ .B^2RFR'.\ B^2RF'\ R$;glide 3 vertices
- $(UFL)^-(URF)^+$ = R'DRFDF'. U'.FD'F'R'D'R.U
- $(UFL)^-\ (ULB)^+$ = $RUR'U\ .RU^2R'U^2\ .R'U'RU'\ .R'U^2RU^2$

VI. <u>Four independent formulas</u>

- (UF)↔(UR) = $F'UF'U'.R'DR'D'.R^2[F'R']F$
- $(UF)^-(UR)^-$ = $FU^2F^2\ .D'\ [U'L']D.\ F^2U'F'U'$
- (URF)→(UBR)→(ULB) = $R^2\ .B^2RFR'.\ B^2RF'R$
- $(UFL)^-(URF)^+$ = $UFD'F^2U\ .L^2U'L^2.\ FU'F^2DF^2$

These formulas are independent, they allow to restore the Cube in the order:
Edges then vertices :
→Place the edges then orient (or inverse)
→Place the vertices then orient (or inverse)
These formulas are found by Cube Explorer, they have no structure !! we do not understand what the formula does !!!

VII. Formulas in <U,R>

When one mixes the Cube only by U, R and that the resolution must also use these two rotations, one must have the formulas only in U, R. (the Siamese)

Move

(UL)→(UR)→(UB) = $R^2U'R'U'.(RU)^2.RU'R$;glide

Twist

$(ULB)^+(UFL)^-$ = $RUR'U .RU^2R'U^2.R'U'RU'R'.U^2RU^2$
$(UBR)^+(URF)^-$ = $(RU')^3(R'U)^3$; disturbing the edges
$(URF)^+(UBR)^-$ = $(R'U[RU'])^3 .U'(R'U[RU'])^{-3}U$
$(UFL)^+(ULB)^+(UBR)^+$ = $([U'R]^2 .U'[RU']^2U)^2$
$(URF)^+(UBR)^+(ULB)^+$ = $RUR'U .RU^2R'U^2$

Twist the centers

$(U)^{++}=(RUR'U)^5$
$(U)^+(R)^-=R'K .(U'K)^2 .U'R$ where $K = (R'U')^2R(UR)^2$; not easy to find !
In the same way one can impose to scrable the Cube only by rotations U, R, F and that the resolution also uses only these rotations. We must therefore have the formulas comprising only U, R, F (not easy !)

SOMMAIRE

Biographie

* Cube Explorer (Herbert Kociemba) : We give a state , and the program finds a corresponding formula in f-metric (cube514htm.exe) or q-metric (cube514qtm.exe)

http://kociemba.org/cube.htm

*Here are the javascripts to calculate max order of a formula.
https://fan2cube.fr/javascript/ordre_maxi.html
https://fan2cube.fr/javascript/ordre_calcul.html

* GAP, is a program that calculates the order of a group of permutations, ...
https://www.gap-system.org/Releases/index.html

* Quizzes to test your knowledge
https://fan2cube.fr/certificat/mc1.html

* A cube simulator
http:/pMetro.su/pCubes.zip

* Rubik's resolution
https://fan2cube.fr/softs/rubiks-solver-master.zip

* Rubik animation
https://fan2cube.fr/softs/rubik_animation.zip

* Virtualcube
https://fan2cube.fr/softs/virtualcubejs2017.zip