

Roux, an advanced approach to cubing

Dominic Zimmer

December 24, 2015

Contents

1	Prologue	3
1.1	Abstract	3
1.2	Perspectives	3
1.3	Terminology and notation	3
2	Roux - theory	4
2.1	History	4
2.2	Idea	4
3	Roux - practical	5
3.1	First block	5
3.2	Second block	5
3.3	Corners Last Layer	5
3.4	Last six edges	5
4	Appendix: Algorithms	6
4.1	2 Look CLL	6
4.2	CMLL	6
4.3	LSE	6

1 Prologue

1.1 Abstract

Below, I'm going to introduce, explain and discuss a method to solve the Rubiks Cube called *Roux*. I will assume the reader not to know how to solve the Rubiks Cube however the geometric understanding of the cube is required. Thus I will also mention some very basic information which might seem redundant to others.

1.2 Perspectives

Apart from Roux, there are plenty methods out there of which I will mention the most prevalent ones. First off, there is the *Beginner's Method*, also known as *Layer by Layer Method* which does what the name implies: it solves the cube layer by layer. This method seems pretty intuitive to most people as it starts off by solving one Face (which is easy to begin with). The most common method is closely related to the Beginner's Method: *Fridrich's Method* or more commonly known as *CFOP* (I will admit, that name might sound very stange at first, though it is justified). CFOP starts off by forming a cross on one side, extending that cross to the first two layers, orienting and permutating the last layer (leading to its unique name). You can see how closely related it is to the Beginner's Method: The key difference being that the Beginner's Method splits building the first two layers into two seperate steps whereas CFOP does this more efficiently. There are quite some more methods which deserve to be named here, for instance *Petrus' Method*, but I'm going to leave it at that.

1.3 Terminology and notation

2 Roux - theory

2.1 History

2.2 Idea

3 Roux - practical

3.1 First block

3.2 Second block

3.3 Corners Last Layer

3.4 Last six edges

4 Appendix: Algorithms

4.1 2 Look CLL

4.2 CMLL

4.3 LSE