Lumache

Release 0.1

Graziella

Oct 19, 2021

CONTENTS

The idea of introducing 5-style is to have a better understanding of how advanced BLD algsets will look like. The problem with BLD events is that once a person starts out with the beginner's blindfold method, there is a lot of excitement, and lots of things to improve on, then there is the awe of the 3-style method, and the execution speeds that can be achieved using it. In a few years after starting out with blindsolving, and optimizing oneself in every way possible in memorization, and speed optimal execution, floating algs and parity subsets, there is a big case of loss of motivation. Many BLDers have quit cubing after they try out 3-style for a few years. Ishaan, Marcell, Max and many more cubers do not see the next big step, and 3-style just feels like a big plateau. Also no one likes to improve in the most snail's pace way, by shaving off milliseconds by improving turn speed, or waiting for the cube gods to give us a lucky scramble. It is best if there exists one method that people can look upto after they try out and optimize with 3-style. 5-style is just a standalone method, most probably even after learning a big subset of 5-style edges, chances are that you need a 3-style alg to finish off the last 2 edge targets, so 5-style is an addition to 3-style to make solves more harmonious. It is one of the biggest algset that exists and even overshadows 1LLL (1 Look Last Layer in CFOP method). The main aim with trying out such a big algset is to understand cubing as a long term project and try to get more involved and efficient algsets in order to improve blindsolving. The ultimate aim is to improve a person's transfer learning capabilities by training in a rigorous method like 5style. By transfer learning I mean, getting useful mental improvements out of cubing. For example one byproduct of learning 5-style and letter quads might help improve linguistics and memory.

The contribution that I am making in this document is introducing 3 new concepts. The concept of letter quads for memorising cube targets thus reducing the information to be memorised, the concept of Yo notation to memorising 3-gen or 4-gen algs easily and of course the concept of 5-style edges and some useful stats and analysis about them.

Disclaimer: Please delve into this method only if you love speedcubing, and only if 3BLD/MBLD is your main event, otherwise this is not worth investing your time into.

Check out the *Usage* section for further information, including how to *Installation* the project.

Note: This project is under active development.

CHAPTER

ONE

CONTENTS

1.1 Usage

1.1.1 Installation

To use Lumache, first install it using pip:

```
(.venv) $ pip install lumache
```

1.1.2 Creating recipes

To retrieve a list of random ingredients, you can use the `lumache.get_random_ingredients()` function:

The `kind` parameter should be either `"meat"`, `"fish"`, or `"veggies"`. Otherwise, `lumache.get_random_ingredients()` will raise an exception.

For example:

```
>>> import lumache
>>> lumache.get_random_ingredients()
['shells', 'gorgonzola', 'parsley']
```

1.2 API

—

0.1 Background about blindsolving

5-Style is a method that tries to solve a 3x3 cube in around 50-70 moves which is almost the CFOP solutions that are not super advanced. Commutators are the go-to way of doing a 3x3 blindfolded. Not all 5-cycles can be described as a trivial commutator though, that's why I have the Yo notation to make alg memorisation easier. I will be describing details of Yo notation that are useful later in this document. This method is to be used in 3BLD to reduce the time of execution which stands at ~11 seconds now using the 3-style method with fancy parity and floating algs subsets. The main applicability of this method is for the MBLD since there is no restriction on the upper limit on the number of cubes as newer methods continue building complexity to improve the MBLD score. I am not sure if it will give a major boost to the 3BLD event, as exec is a lot TPS dependent and on muscle memory. But yes, letter quads, a side product of 5-style will be definitely useful to improve 3BLD memo and shave off more milliseconds than what noddon and throwing the cube cover does.

Origin of the word style is kinda not clear, but after the BH algs were developed a decade ago, they were slowly worked upon and made into a better algset called 3-style. Noah Arthurs in his tutorials introduced the word style to mean that the specific algset has more fingertrickable algs, and are more focussed on 3gen and regripless execution.

Unblackboxing this method to make it easily understandable is one crucial thing that this mega-manual tries to address. Has this idea been previously thought of? Most probably not. Cubing is quite new and there are still a lot of ideas that have not been tested out. Only time will tell which ideas are worth the effort and to be brought into the mainstream. Before we dive into the sea of letter quads, Yo notation and UF5 theory, let me give a backstory on how I have chosen this path of complexity.

1. Context

As a kid, I have always been fascinated by the game of chess, and I played it for quite a while during high school. The thing that I used to find that separated the very high-class GM from a normal amateur chess player, was the amazing preparation the GM used to put to get his technique and repertoire correct. As an amateur player, I had a faint level of positional chess sense, and I was riding high on attacking chess and tactics. But this thought of how important chess preparation has always stayed in my mind.

I am always perplexed by the Rubik's Cube, and my perception keeps changing as I grow older. At first, it used to be a great feat just to solve one side, upon which I stayed satisfied for years, then one day I learned how to solve it completely by looking up an online tutorial.

Doing the cube blindfolded was the next challenge which I took up, which took me another four years to get the hang of, and I did it by leapfrogging from the Old Pochmann method to using M2/R2 in my solves. As the years went by, I slowly and steadily replaced each inefficient M2/R2 algorithm with a newer and faster 3-style algorithm. Another breakthrough in the blind scene came with the US BLDers smashing the blindfold times, by making really fast 3-style algorithms. This was the turning point, as TPS and thinkahead and better algs became the primary thing to invest time into.

It was again in the lockdown induced in the year 2020 due to the COVID pandemic that I understood the parallels between chess and cubing and realised that cubing does not have the intense opening preparation that normal classical chess have. Even speedchess has a lot of preparation at the titled players level. Having a lot of cases evaluated beforehand is a very new thing to cubing, as it was new to the chess world in the 1950s when Botvinnik introduced intense chess preparation and methodology that is widely adopted today by all professional chess players. Cubing is still new and the viability of having such preparation is shrugged off as a waste of time, as cubing records are still optimized on fingertricks, lookahead/pattern recognition, TPS and lots of practice.

1.1 Motivation

I attended my first major competition at the Asian Championships in Beijing in 2016. It was an amazing experience, and the major takeaway I had from the tournament was the impact I got from three cubers, who seemed to be on another level: Shivam Bansal, Kaijun Lin, and Gianfranco Huanqui. Kaijun Lin had already inspired me to take up the Roux method as my main solving method, and he had shown how BLD times are brought to such low and consistent times with practice and focus.

Gianfranco Huanqui is a revolutionary BLDer, who has made new kinds of finger tricks and made many new algs which are novel and fluently executable. On the final day of the Asians, Gianfranco Huanqui did over 300 3BLD solves in one day at the venue. I had lost count of the sub-20s, sub-18s he got and it was spectacular to watch him practice. In every solve, he looked at a point where he thought he could have improved, and continued self-learning in this way.

I also remember Shivam Bansal saying a mind-blowing fact after the prize distribution that, our mind is so powerful that we technically should be able to store petabytes worth of information in it which is even more than a supercomputer or a cluster of computer harnesses. By having such brain power the limits of MBLD is unreachable , he said. After the competition, I headed back to Chennai in India, feeling more driven to create something new. The next month (Nov 2016), I finally thought of taking the plunge into making a new method that I had always thought of but never did. I had decided to list out and memorize all the 5 cycle algorithms for 3x3, for both corners and edges, and also get some 4 cycle comms which come handy in finishing off edges in most of the cases and new parity algs. I wanted to make a memory element for each letter quad which could be retrieved doubly faster than 2 letter pairs, and I wanted a 12ish move count finger trick-able 5-cycle algorithm that could solve the case in the fastest time and with very less finger movement.

If I could have mained 3BLD as my main event, I would have never delved into complexities like 5-style. It is only logical to improve your TPS and fingertricks and get more into the flow state when you are maining 3BLD. It was mainly memorizing the dedges on the big cubes on the 4x4 and 5x5, which was becoming a bit difficult, as there are 23 targets, and it was super tough to get the flow with dedge memo and make it stick. I had the idea that if I adopt a system like letter quads, it will help solve the problem of dedge memo not being fluid. My motivation for trying out letter quads in the first place was wings on big cubes. Sometimes there are trash targets, and letter pairs don't stick. If I had been doing only 3BLD, I would have always done letter pairs, and would have found letter quads super unnecessary.

1.2 Epiphany

I was attending Shaastra Open 2014, my second ever WCA competition. I was 18 years old at that time and had just finished a 4/8 MBLD attempt which felt quite satisfying. The competition went well, and I came second in 3BLD with a time of 2: 06, behind Kabyanil Talukdar who got a 1:20. After the prize distribution ceremony, Arunachaleswarar, an overzealous skewber, who did blindfolded Skewb solves by insane tracking, saw me doing M U M' U' on a 3x3. I showed him that these 4 moves are so efficient that they cycle 5 pieces without affecting the rest of the pieces. He added to me saying that, you should make a whole system out of this idea. I shrugged it off saying it's just too hard as there are many cases, running into over a million unique cases. The same day, earlier I talked to the MBLD winner Vikram Mada who did 6/6 using only single letter memorization (not even letter pairs) and discussed with him conveying how I wish to go beyond letter pairs and go to letter quads. I quickly calculated the number and said a quarter million cases. He said that this just looks impossible, saying that he was already having a tough time transitioning to 480 letter pairs and there I was talking about an algset that runs into a hundred thousand cases. I went back to college and borrowed a big Scrabble book from the library to just see how many 4 letter words exist out there, there were a lot but not enough to cover all the cases of letter quads which can emerge on a 3x3.

1.3 Roux-inspiration

I have been using the Roux method since the year 2016. The step in a Roux that fascinates even a normal cube solver is the LSE part or the last 6 edges. Most of the time we try and solve the LSE, we focus on getting the arrow edge orientation shape which will make all the edges "good", by performing M/M', U/U', M/M' moves.One night when I was going LSE only solves, I realized that the speedsolving approach to the LSE is quite simple and does a great job, and even with EOLR, UL/UR prediction and pinkie pie algsets, we totally avoid the concept of commutators in the solving process. Using 5-cycles on some of the LSE cases might be a thing to dwell upon and can be a possibility in some cases. The cuber will have to option-select between doing LSE the normal way or to recognise if a 5-cycle exists that needs to be solved and solve it.

1.4 Go Game Complexity

When I encountered the news of AlphaGo in 2016, I really wanted to learn the game of Go. Sadly, there were no tutorials and enough documentation on it other than AGA Go association, and to learn advanced techniques, I had to start learning the East Asian languages. Japan was always a powerhouse in Go since ancient times, and recently Korea has been competitive in the Go world. China had always produced good Go players, and they were the most shocked by the AlphaGo program. So, I had to learn Chinese, Korean, and Mandarin to really understand the content and new knowledge the Go world keeps creating. So, TLDR Go is so complex that I felt that such complexity will eventually come to other full information games like Twisty puzzles and in that the 3x3 puzzle.

1.5 Tabla Riyaaz

Indian rhythmic rich history has always played with the beat of 4, placing 4 beats anywhere has resulted in such rich rhythmic compositions.The major learning of the letter quads happened when I came back to Gujarat for my postgrad. I started learning the Tabla instrument, which I had learned from the age of 13-15, and now I had more maturity in understanding it. The phrases that are played are very important and the way they fit into the entire structure and are time bound within the time cycle (16 beats or 12 beats etc is beautiful. My main aim with the kuad language for letter quads is to be able to get fluency with 4 letter phrases in order to memorise LQs better and use them effectively as a memo system, and as a way to represent a unique UF5 case (5-style case from the UF buffer). All I can say is that I do not fear complexity, as the only way to make complex stuff more familiar is to grind and ponder over the patterns, digest them, and make connections among all these phrases and patterns.

2. Why try this method? Why not keep things simple and use good hardware and spam TPS?

I feel similar to a prepared Chess player, or a prepared Go player before I do a 3BLD solve. Rather than a nervy person spamming the Y-perm, and locking up and getting frustrated about the lockups, you will feel composed and at ease during the solve. 3BLD solve would be similar to counting up to the number 5 in mental effort, and if you get comfortable in it there will be, on an average, just five letter quads in a particular scramble. You will come off the beginner tag that every CFOP user or M2/OP user gets when he/she stops learning algs, after they learn OLL, PLL, M2 and just focus on finger tricks, and not newer algs. Till now I have gotten many easy scrambles officially. I once got a good 10/4 scramble in a competition in 2015. I reached a bottleneck in my improvement after that, which I could only improve on by drilling 3-style algorithms and getting all the algorithms in the algset sub-1 seconds. In hindsight, I do not want to reach another bottleneck, so I thought of developing this method.

The average movecount of UF5 is 10.4 STM, which is about 6-6.5 moves less than 2 3-style edge algs (The average move count of speed optimal edge 3-cycles from the UF buffer is 8.4, accounting for cancellations and simultaneous moves being 1 move). The improvement of 6-6.5 moves, or rather 3-3.25 moves per comm is very marginal, and overall I feel this method helps reduce the 3-style solution by about 20 moves. And if a top BLDer has 10 TPS overall and zero pauses, it is about 2 seconds of improvement. Also if we assume the same TPS can be achieved for 5-cycles, then there is an improvement by about 35-40% in execution time for edges.

3. How to make learning 5-cycles less daunting?

Learn how to grind algorithms in an efficient manner to help you get a ton of algorithms into your head and muscle memory. The best way to tackle this humongous algset method is to consider only one alg at a time, learn that one alg in a day, and move on to another alg. The daily rhythm is the best strategy to get everything solidly sorted out in your mind. I also wish to make a video series focussing on subcategories of the algset and how to make a huge memory map of algs in the head. Learn the Yo Notation.

There are about 126,720 edge algs, but you need not learn each and every alg. Not all algs will be 40% faster than 2 3-style UF comms. I will be publishing a subset of the method on my Youtube '5AlgSolve', so that the subset which gives maximum advantage with less effort is first focussed on.

For memorising 126,720 letter quads, I like to call them as the citizens of the micronation 'UF5'. Each of the letter quad has an identity and all the personalised things that a human being has. Similar to how some math superstars make personal connections with random numbers, the way to adopt all these 126,720 citizens is to personalize them.

4. How hard is 5-Style compared to the well known hard algsets like ZBLL?

5-Style has ~12600 edge algorithms and ~60000 corner algorithms. For the corners, the [R U D] 3-style algorithms are already really fast, and the edge algorithms move count has been reduced by ~40%. 5-Style algset size is similar to (ZBLL) 2 which is just plain crazy.

5. 5-Style Solve Examples

My orientation is Yellow on top and Orange in front My lettering scheme is also a bit different, so please do check it out before walkthrough these solves. Before the reconstruction and after scrambling, please rotate to the YO orientation.

5.1 Effectiveness?

There is about 40% improvement in the move count, with little loss on the finger trick ability of the solve. Since you may not detect any insertion sequence in some 5-cycle alg quickly, it is best if you make triggers of batch 4 moves each and memorize each 4 move block using the Yo notation. Memorizing the algorithm via the Singmaster notation is quite cumbersome and difficult. There is World record potential using this method. On a good 10/8 solve (9 algs), and assuming a memo of 5 seconds, total times of ~11-12 seconds are possible with execution of just 5-6 seconds.

5.2 Is 5-style better than 3-style?

5-Style is assumed to be the extension to the efficient and finger-tricky 3x3 blindfolded method of 3-style. 3-style is a really fast method. The current WR is less than 20 seconds using 3-style. 3-style for corners is already quite optimized considering finger tricks and regrips. 5-Style does not immediately triumph over 3-style when it comes to only corners, as we combine two similar corner comms (if they are coming in succession), and get a very efficient and fast execution. Cube explorer is quite slow in finding good 5 cycles for corners, and one of the reasons is due to bad orientation of one of the corners in the 4 corners that need to be cycled, the overall algorithm count is long, hence inefficient. To make

5-Style work on corners, some addendum kind of work has to be done, where 3-style is extended out in some likely and unlikely cases, and categorized, and scaled to 5 cycles fully. Currently UFR corners are faster on the execution as compared to 5-style UFR edges.

6. How to implement this method in your own solves?

The number of unique edge cases in 5-cycles is a whopping 126,720 cases. That's a lot to be honest. These many cases will take a lifetime to learn if you try to compute your alg learning time.

If you look into a normal scramble, not everything is a 5-cycle, there are also a few 4-cycles which form due to going into cycle breaks or edge targets finishing up during a trace [Form: ABAC]or going into the parity setup [Form: ABCA].

For corners, using 5 cycles is still questionable, as there are only 8 corners in a 3x3, and 7 targets in a normal setup, so it is best to solve them using [R U D] 3-style algs from the most optimal buffer UFR.

The occurrence of any given letter quad in a scramble is extremely sparse. And there is no way to deal with this sparsity than to be prepared for every case beforehand.

The chance that any letter quad comes up again in a solve in edge memo is 3/126720 = 1/42240 = 0.002367%. The chance that any letter quad comes up again in a solve in corner memo is 2/68040 = 0.002939%.

This is just insane sparsity that a normal human being just cannot handle. You need a ton of patience developing each of the letter quad, which has a contribution of only 5/194760 = 0.002567% in the entire picture!!!

6.1 "How long before I master this method?"

There is no estimate on how long it will take, it all depends on the effort you put in, and the focus you garner up as you take a deep dive into this method. The best way to implement this method right away is to always do some solves with it. And the one way of learning is by deliberate learning (getting very analytical after each solve , on what all things you did and how you improve on it).

7. My current motivation

Currently, I am doing Machine learning in huge sized image cube data. And generally, the number of classes that the model has to classify is ~10,000 classes.Once I wondered, if I am training models using GPU to distinguish between hundreds of thousands of classes, why would I not do the same to make memorization of a cube easier. In a cube we similarly create thousands of categories, each consisting of a unique 4 letters combination. The think-ahead becomes clearer in a solution because of this categorization.

Another source that spurred me to stick with 4 letter combinations is the Indian art of tabla percussion. In tabla, there is a rich verbal language to represent the rhythmic sounds that the surface of both the drums makes. In that, there are a lot of divisions and basic counting mathematics, that makes it possible to have 16 beat or 10 beat cycles. In tabla playing, there is a concept called the 'rela' which involves playing at very high speeds, with as much as 4-8 sounds in one count of the rhythmic cycle. This gave me the idea of having an impulse of 4 letters at once while memorizing too. For those who did not understand the 'tabla instrument' analogy I gave about, I also claim to compare the algorithm complexity of 5-style with the western classical music instrument of the piano in the Western World. A famous pianist tries to bring in a lot of abstract emotions in his/her playing. There are no 21+53 or 480 or 500 set pieces that they have and the number combination of the notes they produce goes into the hundreds of thousands.

8. How does the method work? (Types of cycles and swaps that emerge through this method)

We know that for a 3-cycle on a cube there are several types of commutators we form out of it where each one of them is derived (I will not be deriving it here as it is a bit more mathematical) They are classified as: Pure Commutator A9 Cycle Shifts Columns Per Specials Orthogonals

Reference to this section: https://www.speedsolving.com/wiki/index.php/Beyer-Hardwick_Method https://www.speedsolving.com/forum/threads/bh-tutorial.12268/

9. The big ADVANTAGE: 5-Style and Letter Quads extends the boundaries of the Multiple Blindfolded (MBLD) WCA event

I previously made a video on how to memorize algorithms that do not have triggers and how to memorize these algorithms using Memory techniques. The main motivation behind making 5-Style for me, is to make big MBLD attempts more seamless as this event is so information and memorization loaded, and also accuracy has to be spot on throughout the attempt. This method is particularly very useful for the MBLD event since there is no restriction on the upper limit of the number of cubes, the complexity of memorization and execution can be increased to improve in the MBLD event.

10. Future Scope

Get all the top BLDers to contribute to this mammoth alg database, and to make a lot of videos classifying these algorithms , and making new finger tricks, a few new types emerge.. You can use a bluetooth cube too to drill out and time all these algorithms. https://briefcubing.com/?enable-5-style or Tao-Yu's alg trainer

I also have created documentation for the Yo notation used to encode the non-intuitive 5-Style algorithms, and I have also created documentation for Letter Quads. There will have to be new techniques of remembering algorithms, without the involvement of cramming or muscle memory. There is a technique which I have developed which uses Yo notation to memorize the algorithm in batches of 4 moves. So, algorithm memorization will involve a lot of metacognition.

An example, remembering algorithms via triggers will work in the case (oiag) : [U : [M,F]] but not in the case (dula): F' U' F D' F' U R' D' R U R' D R U' F D, which has some 3 move insertion in its sequence but no set triggers or [A,B] inside it. Average move count of UF3 is 8.46, and for UF is 10.6, so for extra 2.14 turns we get 2 more solved pieces, if we work out on the paper but in reality UF5 will only be marginally better, as fingertricks are still not heavily worked upon like for 3-style. It's just 2 more moves to solve 2 more pieces, but it is not as simple as it sounds, as we need to be able to do 10+ TPS on UF5 algs as well.

There are about 180 people in the 5-style discord who help out a bit in verifying algs and making them more fingertrickable. To make the transition path easier, we need to compare whether the tradeoff of 2 shorter 3-style algs is better or one 5-Style is better for all the cases. Eg, the hypothesis that for the 3x3 corners, the margin of move count difference is less for 3-style and 5-style. So, using 5-style for corners is not preferable.

5-Style for BigBLD:

I have made algs for other piece types like x+w in 4BLD/5BLD wing algorithms (centers being preserved). The algs generated are not that great and it is tough to get used to them. The length of each 5-cycle algorithm for wings on big cubes will shoot up in move count as many slice moves cannot be used in tandem, they will not be center safe.

Also, I have been working on making new comms for x-centers which are 5-cycles, the progress has been documented here: Link

5-Style system for FMC (5C insertions): If the 5-cycle is completely made and introduced, then it is very useful to use in FMC events. FMC solvers generally do efficient 2x2x3 block building, get a skeleton and do L5C without trying to look for some lucky insertion to reduce L5C. If we already know the L5C algset, we focus on block-building in the FMC attempt and do a 5 Corner insertion somewhere in between the solution using a 5-cycle algorithm (~move length 10-16). But at the end of the day, the method of DR or domino reduction or using fancy block building with insertions is much easier to learn and perform for the FMC event.

11. How to scale this method and make it complete and well verified?

The letter quads sometimes feel like feature engineering in old Machine learning terms, with a lot of toiling into making the data labeled and complete. The best way to memorize a 3x3 scramble is to not use 2-letter or 4-letter, but do pattern abstract comprehension on the 3x3 (piecewise or sticker wise pattern making). Many new kinds of finger tricks coming out of 5-Style algs need to be analyzed. Because many of the move-sequences are different from the well known CFOP triggers and new ways of finger tricking will be found out.

12. Plan to Learning 5-Style method in 5 Years from scratch 5-Style in 5 years! You can plan out 5-style and try to reach the goal of using it in your solves, in about 5 years (kind of, give or take a few years) . The plan may not be perfect, and I hope I get more feedback, so that I can improve on the plan and make it more realistic for new blindsolvers.

The total hours upto the year 2020, that I have spent on making this method work is, 50 hours on fingertrick video making, 50 hours on documentation, 10 hours on kb trainer, 100 hours on software, 700 hours on making algs, 500 hours on stuff that I replaced later on, 1000 hours on letter quads making. I am not counting the time when I do background thinking of UF5, so overall about 100 days of human time for me. For a person who will be deriving from my resources and learning from scratch the hours will be much less, it will be more towards reinforcing the algs, understanding them intuitively and drilling. I cannot give an estimate on that.

13. Byproducts of 5style which can help for speedcubing in general

Letter Quads Letter Quads is a memorisation system, similar to letter pairs in blindsolving, we have a group of letters [A … X] that we club in groups of four and try to make a coherent image out of it. The advantage letter quads have over letter pairs, is that we have to use just 1 image every 4 letters, which makes memorisation much faster, and of less information.

Yo Notation Yo notation is a way to encode face turn, slice turns and wide turns of a 3x3 uniquely as a letter. The main objective of creating such an encoding system is that it makes learning algorithms much easier. In most of the speedsolving methods, the algorithm can be broken down into triggers like R U R' U', R' F R F' etc, and most of the cases in blindsolving can be broken down into commutator notation [A: [B, C]], which makes it easier to remember as the algorithm makes sense and there is a pattern that we get out of it. For 5-style edges, a lot of the algs can be decomposed into commutators and nested commutators, but it cannot be expressed in a few triggers as there are just too many types of algs. So, there was a need to create a system which makes memorising algorithms easier, as each face turn can be a unique letter U=a, U'=b, U2=c and so on, and the entire algorithm string can be broken down into groups of 4 to be memorised as images. So for example, R U2 R' D, becomes jckd in Yo notation, which is just the letter quad for justin kid image.

14. Feedback about this method

Thank you for reading through this document, I hope it was not too complicated to understand or ridiculous to imagine. It would help me a lot if you take some of your time and fill up the feedback form about your thoughts about this method. Feedback

15. Creator of this algset

Hi, I am Abhijeet.

I am a Master's student in Machine Learning and planning to do a Ph.D. in Theoretical Astrophysics sometime later. I have been speedcubing for over 7 years and I know how to solve a Rubik's Cube since 2008. I procrastinated a lot before finally deciding to commit to this method. Since I am already a few years into it, I will not be looking back. If this method fails to improve over 3-style, the side-fruit that I will cherish over is the extensive use of letter quads in my memo. Contact me at mail ID: 5stylerepertoire@gmail.com

SS Forum dedicated to the discussion of 5 cycles: https://www.speedsolving.com/forum/threads/5cyles.61725/ https://www.speedsolving.com/threads/5-style-method-pros-and-cons.73119 https://www.speedsolving.com/threads/5-style-example-solve-game.79974/ https://www.speedsolving.com/threads/daily-letter-quads.80224/