



From covering congruences to the superflip: a distributed computing experiment

Morley Davidson, Joseph Miller
(Kent State University)
and Bruce Norskog (Boston)

Common methodology

- Start with **general-purpose** software
- Formulate some **conjectures & plan** ahead
- Obtain or rewrite as **specific-purpose** code
- (**Distribute** computations if necessary)

Former D.C. experiment

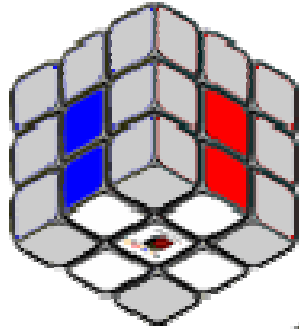
Partition polynomials: distribution of zeros, Mahler measure contribution on sub-arcs, exceptional zeros, factorization & discriminants, multiplicative properties of coefficients of their irreducible parts

Genesis of current experiment

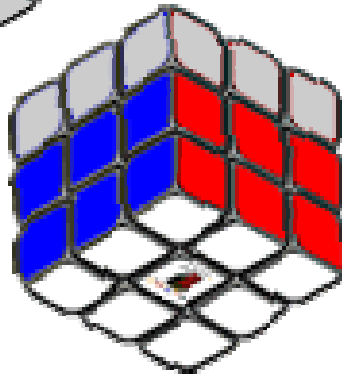
- Found tables online for **cubing systems** such as Fridrich, Petrus, and Roux. Got interested in move-count averages...
- ...wanted to learn an **advanced system** such as the ZB endgame (ZBF2L appl. to Fridrich & ZBLL appl. to Fridrich and Petrus)...
- ...wanted to create a **classification** scheme for LL permutation equiv classes based on covering subgroups.

Fridrich System

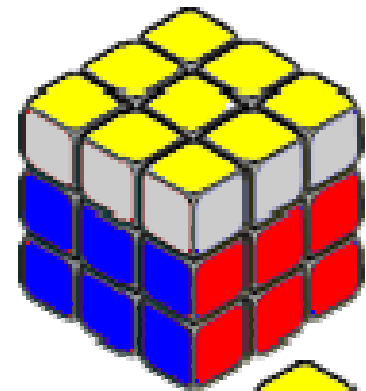
Step 1: “Cross”



**Steps 2,3,4,5 (“F2L”):
First Two Layers**

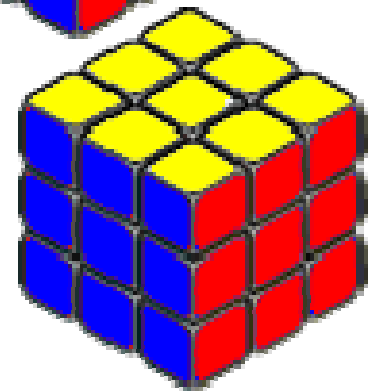


Step 6 (“OLL”): Orient Last Layer



Step 7 (“PLL”): Permute LL

(ZBF2L = F2L4&OELL)



Petrus System

Step 1: Form a $2 \times 2 \times 2$



Step 2: Extend to a $2 \times 2 \times 3$



Step 3: Orient Edges e.g.



to



Step 4: Complete a $2 \times 3 \times 3$



Step 5: Position Corners



Step 6: Orient Corners



Step 7: Position Edges



(ZBLL = Petrus Steps 5&6&7)

Fridrich LL approach

OLL: 216 equiv. classes, reduces to 57 mod U-turns and whole-cube rotations (mod Mrot+AUF), and to 40 mod LL (dihedral) symmetries (M+AUF).

PLL: 288 perms, reduces to 21 mod Mrot+AUF, 14 mod M+AUF, and 13 cases mod M+inv+AUF.

vs. LL in one step: 1212 cases mod M+inv+AUF.

Many ways to save moves

(even with color-fixed partitions, same number of steps, and single alg representatives):

Pochmann: **mis**placing a pair of F2L C/E pairs in **diagonally-opposite slots** gives shorter average PLL (while re-swapping slots).

Distributed computation (DM, spring 09):
no such improvement for any other PLL
“wrong slots” variant, and never for ZBLL.

Step-Greedy Petrus

Step 1: Form a $2 \times 2 \times 2$

(8 choices; pick the cheapest)



Step 2: Extend to a $2 \times 2 \times 3$

(3 choices)



Step 3: Orient Edges

Step 4: Complete a $2 \times 3 \times 3$

(2 choices)



Move-count metrics

- **HTM**: each $\frac{1}{4}$ - or $\frac{1}{2}$ -turn is one move; a.k.a. HTM.
- **QTM**: each $\frac{1}{4}$ -turn is one move.
- **STM**: HTM but each “slice” turn only counts as one move.

Example: R^2DU' has HTM 3,
QTM 4,
STM 2.

Proved & Simulated Upper Bounds for Fridrich* and Petrus HTM means

	Colorfixed w/o cancels (proved)	Colorfixed w/ cancels to depth 1 (proved)	Step-Greedy w/o cancels (10000 simulations)	Step-Greedy w/ cancels to depth 1 (10000 simulations)
Fridrich*	51.63	50.58	48.9	47.7
Fridrich* w/ optimal LL	43.90	42.91	41.2	40.5
Petrus	49.48	48.41	47.4	46.4
Petrus w/ optimal ZBLL	42.91	42.08	40.7	39.9

Zborowski-Bruchem endgame

	HTM mean	# algs (perms)	# mod AUF+M (w/ AUF costs)	# mod AUF+M +inv (w/ AUF costs)	Minimal # for optimal step
ZBF2L	< 8.09	1200	158	≥ 125 (varies)	$\approx 1.4\times$ (varies)
ZBLL	< 12.64	7776	270	177 (exact)	466 or 292
Total	< 20.73	Too many!	428	≥ 302	$\approx 1.4\times$ (to stay < 20.73)

Further optimization strategies & distributed computations

- **Depth-first search** with Fridrich* has mean around 37 moves. Get around 33 moves using Petrus steps 1 and 2, then additional square, then ZB endgame (PBBZB).
- Try alternate algs (e.g., **All Optimals**)
- Back-tracking/insertions/**premoves**

CUBE CODING

- **CubeTools** written in **GAP** to assemble and analyze distributed output from Jelinek's **JACube** and Reid's **CubeExplorer**.
- **Auxiliary programs** (Visual Basic) for distributed computations: JACubeGUI, emailer, attachment downloader, results assimilator
- **C++ version of DFSAO**, currently using PBBZB cube partition with premoves.

DFSAO averages (simulated)

Number of premoves	Solution length (FTM)
0	25.46 (100K sims)
1	23.3 (10K sims), max 26
2	21.8 (1K sims), max 24
3	20.7 (1K sims), max 23

DFSAO superflip results

Number of premoves	Solution length
0	34
1-2	27
3-4	24
5-6	22
7	22 so far.
8	21 so far.

Mortal's Algorithm Conjecture: All scrambles can be solved in “low-twenties” via DFSAO with the PBBZB partition and at most 3 premoves.

22-move “human” superflip solution

- 2x2x2: D2 L' F' B' D2 L
- +square (to 2x2x3): L' F2 U' R2 F2 U L
- +square (to F2L minus slot): R' D2 B R2 B'
- ZBF2L: B R2 B' D2 B' D2 B R2
- ZBLL: D2
- Correction moves: U2 B U D F2

21-move “human” (?) superflip solution

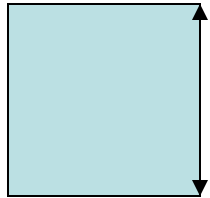
- 2x2x2: L2 D2 F2 R F' B' L'
- +square (to 2x2x3): L R2 D L' R D2 F2
- +square (to F2L minus slot): F' R2 F (R)
- ZBF2L: skip
- ZBLL: skip
- Correction moves: (R') U2 B R2 U' D R' F2

Efficient Semi-rectangular Scheme motivated by Subgroup Covers

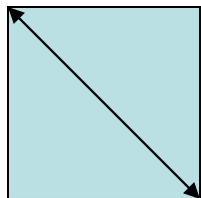
- “Factorial-radix” representation based on semi-direct product structure works well on all 7776 ZBLL or all 62208 LL permutations...but...!
- ...we only want to cover equivalence classes $\text{mod } M+\text{inv}+\text{AUF}$ (ZBLL has 177, LL 1212).
- Idea: start with PLL subgroup's classes and see how efficiently they extend via semidirect product structure to cover ZBLL's CO types.

PLL and ZBLL CP classes

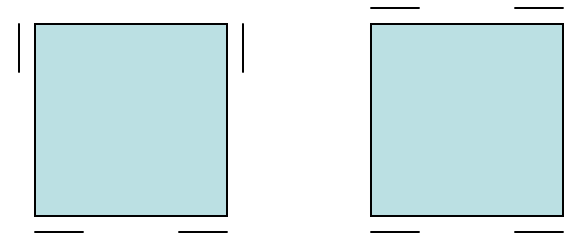
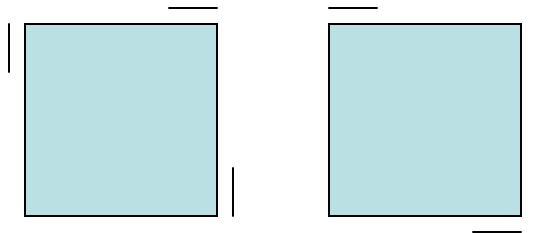
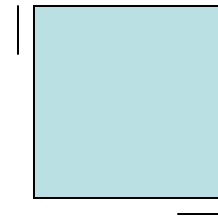
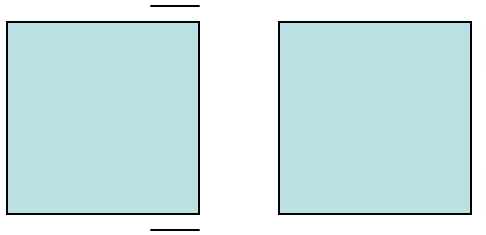
- **N** = null CP case (2-gen ZBLL's)
 - (e?)
- **R** = right-swap CP



- **D** = diagonal-swap CP



Non-PLL CO subclasses



& rotational variants (though not all)

ZBLL CO subclasses by CP class

- **N**=null CP: 6 CO subclasses
- **D**=diag swap: same 6 CO's
- **R**=right swap: 16 (for 271) or 11 CO's

EP types for R (CP) classes (0 redundant)

- Those with 180-degree + R/L reflection symmetry have same 7 EP types as PLL R's.
- All the rest have all possible 12 EP types (original 7 cases extended almost symmetrically).

PoNDeR-based covers for ZBL mod MAUF and MinvAUF

- Efficient with either 271-system or 177-system; has only 32 classes in 177-system, each having either 7 or 12 EP types. In 271-system, there are only 6 symmetrically-placed M-redundancies (placeholders).
- For 177-system, drop redundant inv-redundancies (nice surprise: they are fairly consistently placed)

Extension to a cover for LL mod MinvAUF

- PNDR 177-alg ZBLL extends naturally to 8x177-system for **LL-at-once** (1416 vs. 1212) based on the 8 non-reduced EO cases.
- To compensate, many algs could be replaced by **“Concatenation”** and **“Slice-conjugate”** algs:
- $(L R D^2 L' D' L D^2 R' D L')(L D' R D^2 L' D R' F' B' D^2 F B)$ cancels 12 moves!
- $R^m(R U R' U R U^2 R')R'^m$ is an optimal non-ZBLL LL-alg.

Distributed computing search for Concatenation LL Algs

- 103 max-10 algs generate $62+513=574$ of 1212
- 274 max-11 algs generate $157+656=813$ of 1212.