

GAPを用いたRubik's Cube 解法表示ソフトについて

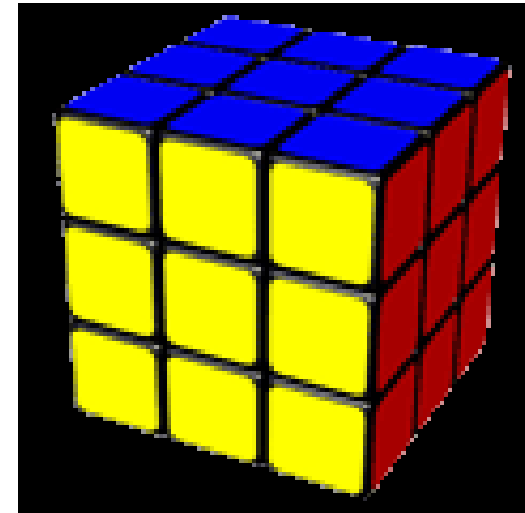
田崎拓馬 (オービック)
藤本光史 (福岡教育大学)

8/Nov/2007
RIMS, Kyoto

Computer Algebra - Design of Algorithms, Implementations
and Applications

Rubik's Cube について

- **Ernő Rubik**（ハンガリーの建築学者）によって考案された立方体パズル
- 1980年代に世界的なブーム
- $3 \times 3 \times 3$ のタイプが有名
 - ▣ $2 \times 2 \times 2$ Pocket Cube
 - ▣ $4 \times 4 \times 4$ Rubik's Revenge
 - ▣ $5 \times 5 \times 5$ Professor's Cube
- 速さを競うスピードキュービング世界大会が毎年開催
 - ▣ 2007年優勝者は日本人（中島悠・釧路高専2年）
 - ▣ 優勝タイム 12秒46（世界記録は9秒77）



Rubik's Cubeと数式処理

- Daniel Kunkle & Gene Cooperman, *Towards the Optimal Bound for Solutions to Rubik's Cube*, ISSAC2007
 - $3 \times 3 \times 3$ のルービックキューブを解くために必要な手数の上限は26手以下であることを計算により示した。
 - 1手の定義は face-turn metric

Rubik's Cubeと群

- 緑・黄・青・白・赤・橙の各面を時計と逆周りに90度回転させる操作をそれぞれ

g, y, b, w, r, o

とおく。

- ルービックキューブのすべての操作は、この **g, y, b, w, r, o** の積で表現できる。
- 言い換えれば、ルービックキューブのあらゆる模様(状態)は、 **g, y, b, w, r, o** の積で表現できる。

Rubik's Cubeと群

g, y, b, w, r, oは以下の置換で表現できる。

$$g := (17, 19, 24, 22)(18, 21, 23, 20)(6, 25, 43, 16) \\ (7, 28, 42, 13)(8, 30, 41, 11);$$

$$y := (33, 35, 40, 38)(34, 37, 39, 36)(3, 9, 46, 32) \\ (2, 12, 47, 29)(1, 14, 48, 27);$$

$$b := (25, 27, 32, 30)(26, 29, 31, 28)(3, 38, 43, 19) \\ (5, 36, 45, 21)(8, 33, 48, 24);$$

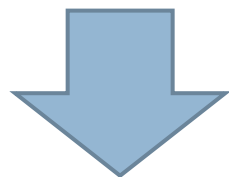
$$w := (9, 11, 16, 14)(10, 13, 15, 12)(1, 17, 41, 40) \\ (4, 20, 44, 37)(6, 22, 46, 35);$$

$$r := (1, 3, 8, 6)(2, 5, 7, 4)(9, 33, 25, 17) \\ (10, 34, 26, 18)(11, 35, 27, 19);$$

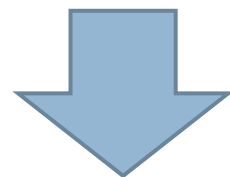
$$o := (41, 43, 48, 46)(42, 45, 47, 44)(14, 22, 30, 38) \\ (15, 23, 31, 39)(16, 24, 32, 40);$$

Rubik's Cubeと群

- **g, y, b, w, r, o** を生成元とする置換群（48次対称群の部分群）としてルービックキューブ群が定義できる。



有限群



数式処理で扱うことが可能

GAP

- Groups, Algorithms, Programming の頭文字
- 1986年 アーヘン工科大学で開発スタート
- 1997年アーヘン工科大学からセントアンドリュース大学に拠点が移って現在に至る

- 計算群論に特化した数式処理システム
- 位数1024を除いて2000までの有限群をデータライブラリとして持つ

ルービックキューブ群を定義する

- **g, y, b, w, r, o**の置換表現を入力し、次を実行するだけ

```
gap> cube := Group(g,y,b,w,r,o);
```

- 群の位数は次で求められる。

```
gap> Size(cube);
```

```
43252003274489856000
```

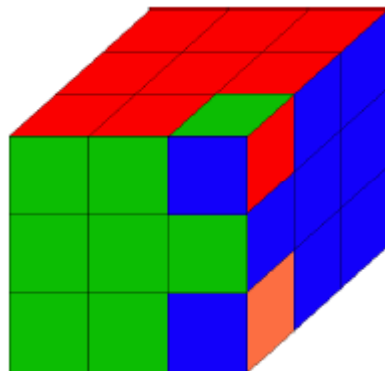
Rubik's Cubeに関する問題

- ルービックキューブの解法表示
 - バラバラにした状態から元に戻す手順を求める
- 与えられた模様は実現できるか

- 一連の操作を何回繰り返すと元に戻るか

与えられた模様は実現できるか

- ルービックキューブ群におけるメンバーシップ問題



```
gap> (8,19,25)(24,30,43) in cube;  
true
```

```
gap> (8,19,25)(24,43,30) in cube;  
false
```

何回繰り返すと元に戻るか

- ルービックキューブ群における元の位数問題
 - 例えば「黄→緑⁻¹→白⁻¹→青⁻¹」という一連の操作だと何回で元に戻るか？

```
gap> Order(y*g-1*w-1*b-1);
```

```
12
```

ルービックキューブの解法表示

Input: ルービックキューブの状態の置換表現

Output: 戻す操作 (**g, y, b, w, r, o** の積)

- 群の元を生成元の積で表す問題
 - ▣ GAPの組込関数GroupHomomorphismByImagesを用いたGetWordOfElementsという関数を定義して利用

```
gap> p:=GetWordOfElements(cube,["g","y","b","w","r","o"],  
(8,19,25)(24,30,43));
```

ルービックキューブの解法表示

$g^*r^*w^*g^*w^{-1}*g^{-1}*r^{-1}*g^{-1}*b^{-1}*g^{-1}*o^{-1}*g^*o^*b^*g^2*b$
 $*g^*b^{-1}*r^*g^{-1}*r^{-1}*g^*b^*g^{-1}*b^{-1}*g^*r^{-1}*b^*r^*b^{-1}*g$

gap> p⁻¹;

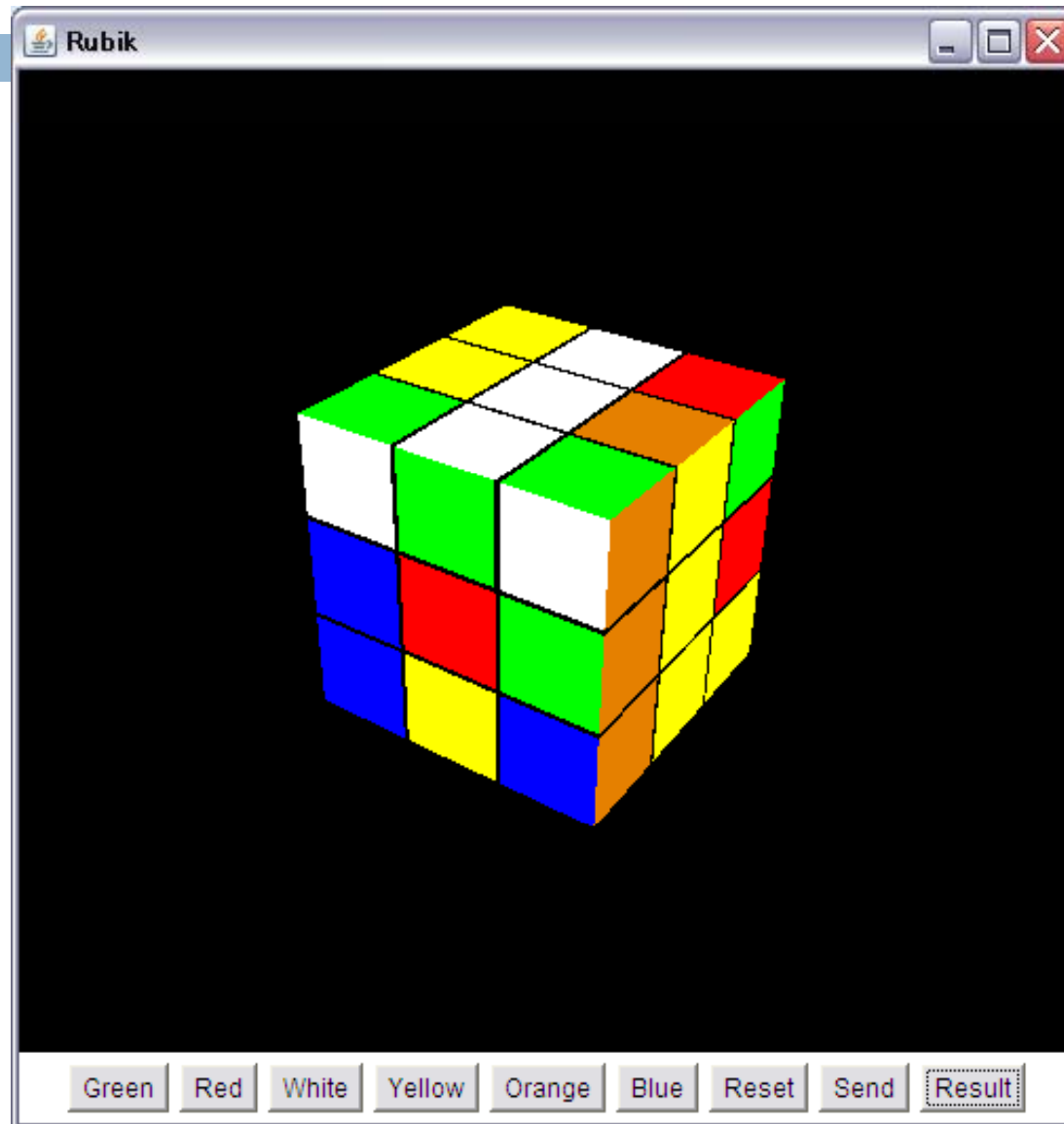
$g^{-1}*b^*r^{-1}*b^{-1}*r^*g^{-1}*b^*g^*b^{-1}*g^{-1}*r^*g^*r^{-1}*b^*g^{-1}*b^{-1}$
 $*g^{-2}*b^{-1}*o^{-1}*g^{-1}*o^*g^*b^*g^*r^*g^*w^*g^{-1}*w^{-1}*r^{-1}*g^{-1}$

この操作を右側から順に行うと元に戻せる！

解法表示ソフトについて

- マウスやペンでドラッグすることにより、回転・移動・拡大が可能
- 各面の色のボタンをクリックすることにより、その色の面を反時計回りに90度回転
- **Reset** ボタンを押すことで、最初の状態に戻す
- **Send** ボタンを押すことで、現在のルービックキューブの状態(置換表現)を **GAP** に渡し、計算を開始
- **Result** ボタンで **GAP** が計算した手順でルービックキューブの自動操作を実行

スクリーンショット



実装方法



- GUI 部
 - Java3Dを用いて作成
- 解法エンジン
 - GAPを利用
- GUIとGAPの通信
 - 原始的なファイルのやりとりで実現

デモ



活用法

- 代数学の群論の授業
 - 置換表現を求める
 - Lagrangeの定理の理解を深める

- ジュニアサイエンス
 - 子ども達に楽しんでもらう
 - パズルの解法に数学が利用できることを体験

今後の課題

- GAPとの通信にOpenMathやOpenXMを利用
- LBL法などの高速解法モードの追加
- マウスやペンによるキューブ操作の実現
- キューブの20個のピースをバラバラにした状態からマウスやペンを使って様々な模様を作成できるようにする。
→その模様が実現可能かどうかを判定させる