

非同型なパターンロックの列挙

An Enumeration of Distinct Lock Patterns from User Inputs

中川 幸一

KOUCHI NAKAGAWA

埼玉大学大学院 理工学研究科

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING, SAITAMA UNIVERSITY *

吉村 英竜

HIDETATSU YOSHIMURA †

埼玉県立大宮高等学校

SAITAMA PREFECTURAL OMIYA SENIOR HIGH SCHOOL ‡

Abstract

パターンロックとはスマートフォンなどのデバイスロックで用いられる画面ロックの一種である。非同型なパターンロックの総数を数え上げる先行研究 [1], [2] は存在しているが、デバイスが認識する各情報（出力）に対応する操作数（重複度）は算出されていない。本研究の目的は全ての出力を列挙し、重複度と総数を決定することである。

Abstract

Pattern locks are used for the lock screens in devices like smartphones. Although there is previous research [1], [2], which counted the total number of lock patterns, no one has computed the number of operations (Multiplicity) which corresponds to each information (Output) recognized by the device. The purpose of this study is to determine Multiplicity and total number by enumerating all Outputs.

1 はじめに

パターンロックとはスマートフォンなどのデバイスロックで用いられる画面ロックの一種である。このパターンロックを数学的に捉えその総数を数え上げた先行研究 [1] [2] は存在するが、それぞれの操作（以下、単に入力）で端末が受け取る操作情報（出力）に対し何通りの入力に対応しているか（重複度）を明らかにしたものはない。（重複の例に関しては図1を表記については4.1を参照。）そこで本研究では、高重複度の画面ロックを毎回違う入力でロック解除すれば操作を見られても推測されにくいと考え、全ての出力を列挙し、重複度と総数を決定することを目標とした。結果、パターンロックの総数は先行研究と同様の140,704通りの出力結果が得られ、重複度としては最大が217であることが分かった。

*k-nakagawa@h6.dion.ne.jp

†本研究は埼玉大学ハイグレード理数高校生育成プログラム（科学技術振興機構（JST）グローバルサイエンスキャンパス事業）の助成を受けたものです。

‡Y.Hide.Peppermint@outlook.jp

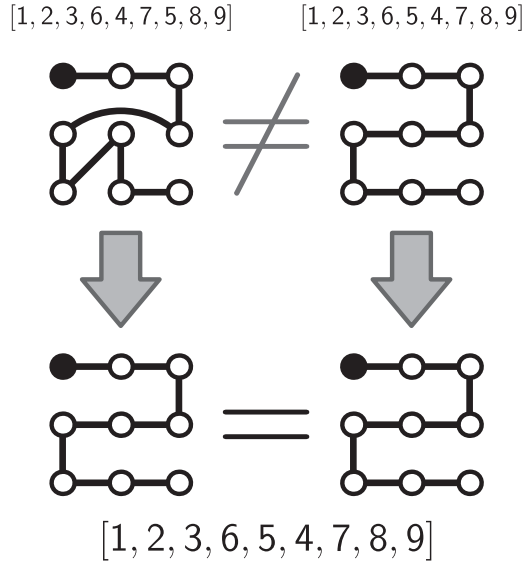


図 1: 重複の例

2 先行研究

Android のパターンロックにおける組み合わせの数は Otaku によって 9 点すべてを使用する場合 140,704 通りあることが計算されている [1]. また, [3] のプログラムコードによると, パターンロックにおける組み合わせを数え上げてはいたが, 重複しているものは順次取り除いているため, 重複具合までは調べ尽くされていなかった.

パターンロックの形状の考察としては, 対称性として 4 次二面体群 D_4 になっているため, 8 の倍数通りが現れていることは触れられていたが, それ以上のことは触れられていなかった.

3 パターンロックの制約

パターンロックは以下の 3 つの制約を持つ.

制約 1 : 3×3 の計 9 点から 4 点以上を一筆書きで結ぶ.

制約 2 : 選択していない点を通過することはできない. (ジャンプ禁止, 図 2)

制約 3 : 一度選択した点を通過することはできるが, その点をもう一度選択することはできない. (複数選択禁止, 図 3)

本研究では, より複雑な入力について考察をするため, 制約 1 の「4 点以上」という部分を「9 点すべて」に変えて考察する.

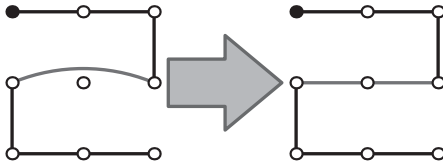


図 2: ジャンプ禁止

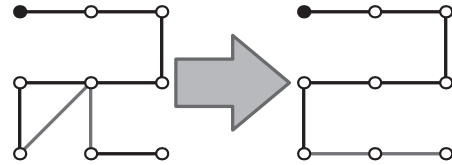


図 3: 複数選択禁止

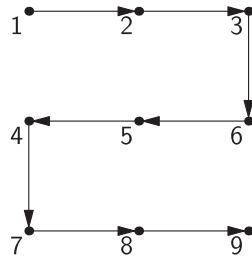


図 4: 数字と点の対応

4 提案手法

以下の手順によってパターンロックの形状と重複度を求める。

1. 入力の数字表現
2. 通過点の補充
3. 2回以上選択された点の削除（仮出力を得る）
4. 仮出力が入力と一致したかを調べる
 - (a) 一致: 仮出力を出力
 - (b) 不一致: 仮出力を入力と捉えて手順 2～4 を行う
5. 得られた異なる出力を数え、同じ出力に対応する入力を集計

4.1 入力の数字表現

3×3 の点に対して左上から右下に順に 1 から 9 までの番号を振ることにする。(電話の番号配置と同じ) 1～9 の数字の並び(順列)をもって順に対応する点を辿ることによって入力や出力の表現とする。このとき、左端の数字に対応する点から順に点を結んで一筆書きをすることとする。例えば、[1, 2, 3, 6, 5, 4, 7, 8, 9] という入力に対しては、図 4 のようなパターンロックの形状が対応する。

また、このとき入力に対する総数は $9!(= 362880)$ 通りとなる。

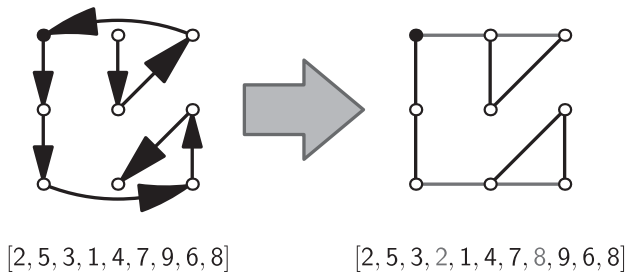


図 5: 通過点の補充の例

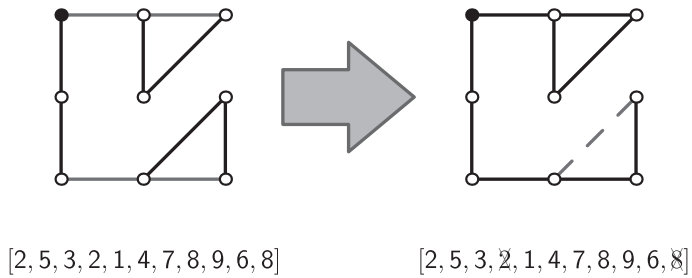


図 6: 点の削除の例

4.2 通過点の補充

パターンロックの制約 2 により以下の数字の並びの間にジャンプした数字を補う。

2 を補う順列 : (1, 3), (3, 1)

4 を補う順列 : (1, 7), (7, 1)

5 を補う順列 : (1, 9), (2, 8), (3, 7), (4, 6), (6, 4), (7, 3), (8, 2), (9, 1)

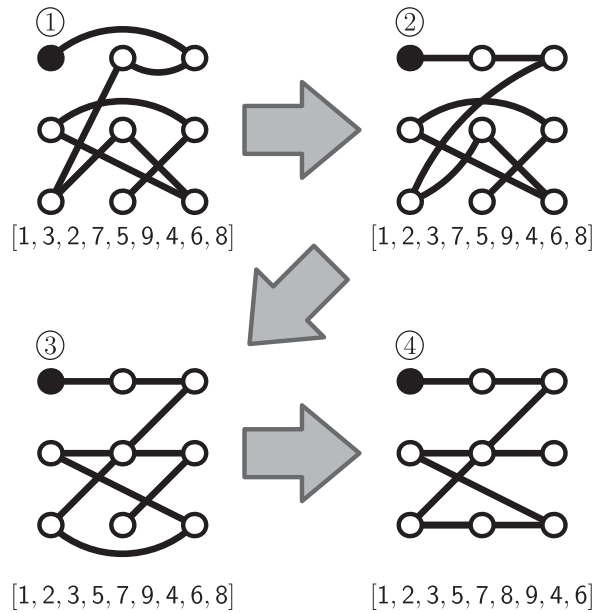
6 を補う順列 : (3, 9), (9, 3)

8 を補う順列 : (7, 9), (9, 8)

例えば, $[2, 5, 3, 1, 4, 7, 9, 6, 8]$ という入力に対しては, $[2, 5, 3, 2, 1, 4, 7, 8, 9, 6, 8]$ と変形する。(図 5 参照)

4.3 2 回以上選択された点の削除

手順 2 で補われた数字の列に対して, パターンロックの制約 3 により 2 回以上選択された点の削除を行う。このとき, 左端から順に数字を確認して, 2 回目以降に表れる数字をすべて削除する。例えば, $[2, 5, 3, 2, 1, 4, 7, 8, 9, 6, 8]$ という入力に対しては, $[2, 5, 3, 1, 4, 7, 8, 9, 6]$ と変形する(図 6 参照)



①	1	3	2	7	5	9	4	6	8
	1	2	3	2	7	5	9	4	5 6 8
②	1	2	3	2	7	5	9	4	5 6 8
	1	2	3	5	7	5	9	4	5 6 8
	1	2	3	5	7	5	9	4	5 6 8
③	1	2	3	5	7		9	4	6 8
	1	2	3	5	7	8	9	4	5 6 8
	1	2	3	5	7	8	9	4	5 6 8
④	1	2	3	5	7	8	9	4	6

図 7: 繰り返しが最大の例

4.4 仮出力が入力と一致したかを調べる

手順 3 で得られた仮出力の順列が入力時の順列と一致しているかを確認する。一致した場合はこの仮出力を出力結果の一つとし、一致しなかった場合はこの仮出力を入力と見なして再度手順 2~4 を繰り返す。尚、一致しなかったということは初期の入力に対して最終的な出力結果の順列がまだ得られていないということであり、調べたところによると最大で 4 回まで繰り返すことが分かった。(図 7 参照)

5 結果

提案手法によって得られた結果は表1のようになった。種類の合計値は先行研究と同様の140704通りの出力を得た。また、本研究の課題である、高重複度のパターンロックは、重複度217というものが最大であり、図8のように8通り得られた。

6 応用

本提案手法を応用すると、逆順に辿った（開始位置が異なる）場合についての考察も出来る。得られた結果は表2のようになった。先ほど求めた140704通りの異なるパターンロックの形状のうち、逆順に辿っても同じ形状が得られるものは32256通りとなり、これは最終的に得られる形状として、本質的に異なるものは32256通りであることを示している。この逆順も考慮したときの高重複度のパターンロックは、重複度265というものが最大であり、8通り得られた。形状自体は図8と同じで、開始位置はどちらの端点からでもよいだけの違いであった。

参 考 文 献

- [1] Otaku, Cedric's weblog (<http://beust.com/weblog2/archives/000497.html>)Retrieved 2017-12-19.
- [2] 石黒, 福島, 清本, 三宅「モバイル端末のロック解除向けパターン認証の安全性評価」, IEICE technical report, Vol.112(No.128), pp.273-278, 2012-07-12.
- [3] にほ録, Android のアンロックパターンは 389,112 通り
(<http://blog.goo.ne.jp/nihongi/e/d0cfde12d440b92c181b6afd2b7dc4cb>)Retrieved 2017-12-19.

表 1: 計算結果 (開始位置の考慮あり)

重複度	種類	重複度	種類	重複度	種類	重複度	種類
217	8	48	352	28	184	12	240
192	16	47	8	26	16	11	336
143	8	45	8	24	472	10	232
105	24	41	8	23	8	9	264
95	8	40	64	21	136	8	4528
80	64	38	8	20	8	7	4416
69	16	36	8	19	32	6	4864
66	16	35	296	18	176	5	4304
64	8	34	8	16	64	4	3832
63	8	32	136	15	368	3	4344
55	8	30	8	14	192	2	4224
54	8	29	16	13	192	1	106160

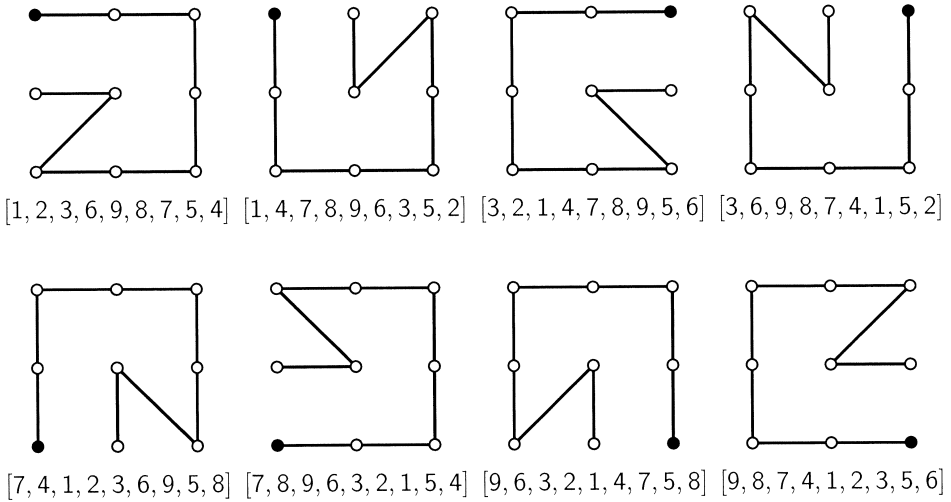


図 8: 重複度 217 のパターンロックの形状 (開始位置の考慮あり)

表 2: 計算結果 (開始位置の考慮なし)

重複度	種類	重複度	種類	重複度	種類
265	8	60	8	34	16
210	8	56	176	32	32
200	8	53	8	15	64
191	8	50	200	14	64
160	16	48	80	13	160
77	16	46	40	12	160
74	16	44	32	11	168
72	8	43	8	10	3400
71	8	42	16	9	72
63	8	40	48	8	856
62	16	38	80	2	10320