L-019

直交するラテン立方体と$(3,n)$しきい値法
Mutually orthogonal Latin cubes and $(3,n)$ threshold scheme

野澤 友希† 足立 智子†
Yuuki Nozawa Tomoko Adachi

1. はじめに

ラテン立方体は、ラテン方陣を3次元に拡張したものである。ラテン方陣や直交配列から、秘密分散法が構成できる。直交するラテン方陣の組($MOLS$)から、$(2,n)$しきい値法が構成でき、強さtの直交配列から(t,n)しきい値法が構成できる。文献[1]より、$GF(q)$上の3変数の一次多項式u本から、行列Mが得られれば、MOCを構成できる。MOCがわかれば、強さ3の直交配列が構成できる。しかし、$GF(5)$であっても、行列Mを探すのは時間がかかる。本研究では、$GF(3)$での行列Mの特徴から、$GF(5)$での行列Mを効率的に探す初期行列Cの特徴を調べる。

2. 用語の説明と先行研究

2.1 ラテン立方体と秘密分散法

有限体$GF(q)$の性質を用いて、互いに直交するラテン立方体が構成できる。二つのラテン立方体を重ね合わせたときに、シンボルの順序対がすべて異なる場合、直交するという。任意の二つが直交するような、位数qのd次元ラテン超立方体の複数個の組を、d-$MOC(q)$と記す。文献[1,2]より、$GF(q)$上のd変数の一次多項式u本から、行列Mが得られれば、d-$MOC(q)$を構成できる。[3]では、行列M(行数u)を探索した。d-$MOC(q)$から直交配列ができ、$(d,u\text{-}1)$しきい値法が構成できる。

定理 1 ([1]) uをd以上の整数とする。
$f_i(x_1,x_2,\dots,x_d)=a_{i,1}x_1+a_{i,2}x_2+\cdots+a_{i,d}x_d\ (1\le i\le u)$
を$GF(q)$上の一次多項式とする。このとき、$f_1,f_2,\dots,f_u$によって生成される超立方体は、行列$M=(a_{i,j})_{u\times d}$のすべてのd行が線型独立であるときかつその場合に限り、d-$MOC(q)$を形成する。

定理 2 ([4,5]) 直交配列$OA_1(t,n+1,q)$の存在と、シンボルq種類の(t,n)しきい値法の存在は、必要十分条件である。

2.2 先行研究の結果

[3]では、d=3の場合に、サイズu×3の行列Aから、任意の3行を抜き出した3×3行列Bの階数が3ならば、定理3の条件を満たすため、Aは求める行列Mとなる。初期行列Cから、条件を満たす行列Aを探すプログラムをPythonで作成した。

Step1 初期行列Cに適する行の候補を挙げる。
Step2 初期行列Cに新しく1行加えた行列Aを作成する。
Step3 Aから任意の3行を抜き出し3×3行列Bの階数を調べる。
Step4 階数が3なら、Aへ行の候補からさらに新しく1行加える。階数が3でないなら、Aから加えた1行を消し、行の候補から別の1行を加える。行の候補がなくなれば終了。

† 静岡理工科大学 Shizuoka Institute of Science and Technology

上述の手法[3]では、$GF(3)$の場合サイズ3×3の一つの初期行列C_1 = [[0,1,0],[2,1,0],[1,0,1]]に対してすべての組み合わせを探索して最大行数u=6の行列Aを求めることができた。しかし、$GF(5)$の場合は初期行列C_1に対して、計算時間がかかりすぎ、u=12の時点で探索を中断した。

3. 実験 1 に関する提案手法とその結果

3.1 提案手法

本研究では、d=3について調べる。先行研究[3]では、$GF(3)$の場合に、初期行列を2つ(零要素を含む)しか調べていなかった。そこで本実験では、$GF(3)$の場合に、非零要素の組み合わせでできるサイズ3×3の初期行列すべてのパターンについて行列Aを探索する。本実験の目的は、探索したサイズu×3の行列Aについて、各行の要素の組み合わせの特徴を見つけることである。行列の要素を非零に限定する手法は、文献[2]でも用いられている。

$GF(3)$で考えられるすべての大きさ3×3の初期行列${}_8C_3$=56通りのうち、[1]の条件を満たすものは44通りである。これらについて[3]を改良した手法で行列Mの探索を行う。

[3]のアルゴリズムでは、行の候補を追加する操作において、まず[1,1,1]を試し、次に[1,1,2],[1,2,1],...,[2,2,2]といったように、辞書順に候補を確認していた。このとき、[1,1,1]の次の行に[2,2,2]が入るパターンと、[2,2,2]の次の行に[1,1,1]が入るパターンの双方を確認しており、重複が発生していた。そこで、辞書順が前の候補に戻らないようにアルゴリズムを改良した。さらに、初期行列に一行加えたときに求める条件を満たさない行の候補を、最初に省くように改良した。

3.2 結果および考察

本実験の結果、得られた行列Aの行数uはu=4,6であった。uの最大値は、[3]の結果と同じu=6であった。

得られた行列Aは、行数u=4のものが、図1の①から⑧(およびそのシンボル1と2を入れ変えたもの)であった。行数u=6のものが、図1の⑨であった。すなわち、$GF(3)$の場合に行列Aは図1の計9パターンに分類できた。

①	②	③	④	⑤	⑥	⑦	⑧	⑨
111	111	111	111	111	111	111	111	112
112	112	112	112	221	221	221	221	121
121	121	212	212	121	121	212	212	122
122	211	122	211	122	211	122	211	221
								212
								211

図1 実験1で得られた行列A

ここで、Aの最大行数u=4となった①から⑧の行列にはすべて[1,1,1]が含まれており、u=6となった⑨には、[1,1,1]が含まれていなかった。

この結果から、Aが[1,1,1]のような同一要素を並べた行を含まない方が、最大行数uは大きくなると考えられる。

この特徴を用いて、次節では$GF(5)$の実験2,3を行う。

4. 実験 2 に関する提案手法とその結果

4.1 提案手法

本実験では、計算回数を減らすために、[1,1,1]を含むサイズ3×3の初期行列C_2 = [[1,1,2],[1,2,1],[1,1,1]]を用いて、$GF(5)$で行列Mの探索を行う。

本実験の目的は、先行研究[3]では中断してしまった$GF(5)$での探索を最後まで行い、探索できたサイズu×3の行列Aについて、各行の要素の組み合わせの特徴を見つけることである。

4.2 結果および考察

本実験の結果、得られた行列Aのうち最も行数uが大きかったのはu=13であった。これは、[3]の結果(u=12で中断)よりもよくなった。得られたu=13の行列Aは、図2の①から⑭の14通りあった。

①	②	③	④	⑤	⑥	⑦
[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]
[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]
[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]
[1 3 3]	[1 3 3]	[1 3 3]	[1 3 3]	[1 3 4]	[1 3 4]	[1 3 4]
[1 3 4]	[1 3 4]	[1 3 4]	[1 4 3]	[1 4 3]	[1 4 3]	[1 4 3]
[1 4 3]	[1 4 3]	[2 3 4]	[2 1 4]	[1 4 4]	[1 4 4]	[2 1 4]
[2 1 4]	[2 1 4]	[2 4 1]	[2 4 3]	[2 1 4]	[2 1 4]	[2 4 1]
[2 4 1]	[2 4 1]	[3 4 1]	[3 1 4]	[2 4 1]	[2 4 1]	[3 2 2]
[3 2 4]	[3 4 2]	[3 4 2]	[3 2 4]	[3 2 4]	[3 4 2]	[3 2 4]
[4 1 3]	[4 1 3]	[4 1 2]	[4 1 3]	[4 1 3]	[4 1 3]	[4 1 3]
[4 2 3]	[4 2 3]	[4 1 3]	[4 2 1]	[4 2 3]	[4 2 3]	[4 2 3]
[4 3 1]	[4 3 1]	[4 2 3]	[4 3 1]	[4 3 1]	[4 3 1]	[4 3 1]
[4 3 2]	[4 3 2]	[4 3 1]	[4 3 2]	[4 3 2]	[4 3 2]	[4 3 2]

⑧	⑨	⑩	⑪	⑫	⑬	⑭
[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]	[1 1 2]
[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]	[1 2 1]
[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]	[1 1 1]
[1 3 4]	[1 3 4]	[1 3 4]	[1 3 4]	[1 3 4]	[1 3 4]	[1 4 3]
[1 4 3]	[1 4 3]	[1 4 3]	[1 4 3]	[1 4 3]	[1 4 4]	[1 4 4]
[2 1 4]	[2 1 4]	[2 1 4]	[2 1 4]	[2 1 4]	[2 3 4]	[2 1 4]
[2 4 1]	[2 4 1]	[2 4 1]	[2 4 1]	[2 4 1]	[2 4 1]	[2 4 3]
[3 2 2]	[3 2 4]	[3 2 4]	[3 4 2]	[3 4 2]	[3 4 1]	[3 1 4]
[3 4 2]	[3 4 4]	[4 1 1]	[3 4 4]	[4 1 1]	[3 4 2]	[3 2 4]
[4 1 3]	[4 1 3]	[4 1 3]	[4 1 3]	[4 1 3]	[4 1 2]	[4 1 3]
[4 2 3]	[4 2 3]	[4 2 3]	[4 2 3]	[4 2 3]	[4 1 3]	[4 2 1]
[4 3 1]	[4 3 1]	[4 3 1]	[4 3 1]	[4 3 1]	[4 2 3]	[4 3 1]
[4 3 2]	[4 3 2]	[4 3 2]	[4 3 2]	[4 3 2]	[4 3 1]	[4 3 2]

図2　実験2で得られた行列A

この14通りの行列Aについて、各行の組み合わせを調べ、次の特徴が判明した。

14通りすべてに含まれている行のパターンは[4,1,3],[4,3,1]であった。

また、片方が含まれるとき、もう片方が含まれないような対の関係にある行のパターンがあった。④⑭のみに含まれる行のパターンは[2,4,3],[3,1,4],[4,2,1]であり、それ以外の12個に含まれる行のパターンは、[1,3,4],[2,4,1],[4,2,3]であった。③⑬のみに含まれる行のパターンは[2,3,4],[3,4,1],[4,1,2]であり、それ以外の12個に含まれる行のパターンは、[1,4,3],[[2,1,4],[4,3,2]であった。

5. 実験 3 に関する提案手法とその結果

5.1 提案手法

本実験では、uを大きくするために、次のサイズ7×3の初期行列C_3を用いて、$GF(5)$で行列Mの探索を行う。

$$C_3 = [[1,1,2],[1,2,1],[4,1,3],\ [4,3,1],[1,3,4],[2,4,1],[4,2,3]]$$

このC_3の作り方を以下に示す。行列C_2から[1,1,1]を取り除き、14通りすべてに含まれている行のパターン[4,1,3],[4,3,1]を加える。さらに、12通りに含まれている行のパターン[1,3,4],[2,4,1],[4,2,3]を加える。

本実験の目的は、$GF(5)$の場合に、uが大きな行列Aを見つけることである。

5.2 結果および考察

本実験の結果、得られた行列Aのうち最も行数uが大きかったのはu=16であった。これは、[3]の結果(u=12で中断)や実験2の結果(u=13)よりもよくなった。

得られたu=13の行列Aは122通りあった。そのうちの一つが図3である。

[1 1 2]
[1 2 1]
[4 1 3]
[4 3 1]
[1 3 4]
[2 4 1]
[4 2 3]
[1 1 3]
[1 3 1]
[1 4 3]
[1 4 4]
[2 1 4]
[3 2 2]
[3 2 4]
[3 4 2]
[4 3 2]

図3 実験3で得られた行列A

6. おわりに

本研究ではd=3について行列M(行数u)を探索し、$GF(5)$の場合に[3]よりも行数uが大きいものが得られた。この結果と文献[1,2]より、3-$MOC(5)$が構成でき、強さ3で列数uの直交配列ができる。さらに、本研究の結果と文献[4,5]より(3,u-1)しきい値法が構成できる。

よって、大きなuの行列Mが探索できれば、参加者数n=u-1が多い$(3,n)$しきい値法を構成することができる。本研究の結果より、(3,15)しきい値法が構成できることがわかった。

参考文献

[1] J.T. Ethier, G.L. Mullen, “Strong forms of orthogonality for sets of hypercubes”, Discrete Math., vol.312, no.12, pp.2050–2061, 2012.
[2] X.-N. Lu, T. Adachi, “On Dimensionally Orthogonal Diagonal Hypercubes”, IEICE TRANS. FUNDAMENTALS, vol.E103-A, no.10, pp.1211-1217, 2020.
[3] 野澤友希, 足立智子, 行列を用いたラテン超方格と直交配列の探索, FIT2024 予稿集,no.1, 75-76.
[4] E. Dawson, E.S. Mahmoodian, A. Rahilly, “Orthogonal arrays and ordered threshold schemes”, Australian. J. Comb., 8(1993), pp.27-44.
[5] D. R. Stinson, “Combinatorial Designs and Cryptography, Revisited”, 50 years of Comb., Graph Theory, and Computing, Ed. F. Chung et al., CRC Press, 2020, pp.335-357.