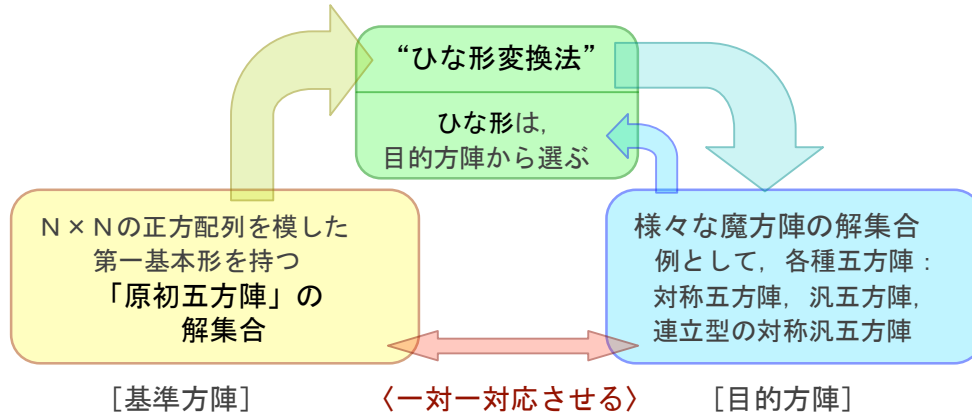


第4部 : 『魔方陣の最新研究』 摂田 寛二  
 第1章 : 原初方陣とひな形変換法の基礎的研究  
 第3節 : 原初五方陣の基礎的研究

新しい考え方である原初方陣とひな形変換法について、引続き各論的に研究する。



今回は、五次の魔方陣の中から（１）補数対称型，（２）汎対角線型，および（３）連立型の対称汎五方陣を選び、それに対応する「原初五方陣」と「ひな形」を仮定して、相互関係を研究してみたい。幸いにこれらの３つは、解の総数が確定しているのだから、それと同じ総数の「原初五方陣」を構成すれば良い。目的とするものが明瞭なのは、気分的に楽だ。

（３）の連立型は、（１）と（２）の性質を併せ持つものを言う。つまり補数対称型であつてかつ同時に汎対角線型でもあるもの。解の総数が 16 個あることが知られている。

そこで考えたのだが、懸案のひな形を（３）の連立型の中から選んではどうか。そうすれば、同時に（１）や（２）のひな形にもなり得るのではないか。

ここは、ひな形を共通に設定して、専ら原初五方陣の違いを研究しよう。

四次の場合は、原初方陣が共通で、ひな形が違った。五次では、その反対になるのかどうかを、実験して確かめてみようと思う。

### # 1. 「原初五方陣」の特徴

下図のように  $5 \times 5$  の正方二次元配列に、1 ~ 25 の自然数を全数個重複なくただ順に並べたものを原初五方陣の「第一基本形」と言おう。基本図としておなじみのものだ。

4	5	1	2	3	4	5	1	2	3	$1+2+3+4+5=15$	...	(1)
							6	7	8	$6+7+8+9+10=40$	...	(2)
9	10	6	7	8	9	10	6	7	8	$11+12+13+14+15=65$	...	(3)
							16	17	18	$16+17+18+19+20=90$	...	(4)
14	15	11	12	13	14	15	11	12	13	$21+22+23+24+25=115$	...	(5)
							1	6	11	$1+6+11+16+21=55$	...	(6)
19	20	16	17	18	19	20	16	17	18	$2+7+12+17+22=60$	...	(7)
							3	8	13	$3+8+13+18+23=65$	...	(8)
24	25	21	22	23	24	25	21	22	23	$4+9+14+19+24=70$	...	(9)
							5	10	15	$5+10+15+20+25=75$	...	(10)

0. これを方陣として見た場合、行列の定和が成り立たないから、魔方陣とは言えない。

1. しかし、ここでは「26の補数」が点対称に配置されている。

$$1+25=2+24=3+23=4+22=5+21=6+20=7+19=8+18=9+17=10+16=11+15=12+14(=13+13)=26$$

また、汎対角線 10 本の定和も成り立つ。

下図のように方陣空間を予め拡張しておくこと、汎対角線が見やすくなる。

$$\begin{aligned}
&1 + 7 + 13 + 19 + 25 = 65 \\
&1 + 10 + 14 + 18 + 22 = 65 \\
&2 + 8 + 14 + 20 + 21 = 65 \\
&2 + 6 + 15 + 19 + 23 = 65 \\
&3 + 9 + 15 + 16 + 22 = 65 \\
&3 + 7 + 11 + 20 + 24 = 65 \\
&4 + 10 + 11 + 17 + 23 = 65 \\
&4 + 8 + 12 + 16 + 25 = 65 \\
&5 + 6 + 12 + 18 + 24 = 65 \\
&5 + 9 + 13 + 17 + 21 = 65
\end{aligned}$$

また「クロスサム等和」  
が成り立つ。

$$\begin{aligned}
&1 + 7 = 2 + 6, \\
&2 + 8 = 3 + 7, \\
&3 + 9 = 4 + 8, \\
&4 + 10 = 5 + 9, \\
&6 + 12 = 7 + 11, \quad 7 + 13 = 8 + 12, \quad 8 + 14 = 9 + 13, \quad 9 + 15 = 10 + 14, \\
&11 + 17 = 12 + 16, \quad \dots\dots\dots
\end{aligned}$$

次のような「クロスサム等和」も成り立つ。

$$\begin{aligned}
&1 + 8 = 3 + 6, \quad 2 + 9 = 4 + 7, \quad 3 + 10 = 5 + 8, \\
&\text{拡張空間まで広げれば, } 4 + 6 = 1 + 9, \quad 5 + 7 = 2 + 10, \\
&6 + 13 = 8 + 11, \quad 7 + 14 = 9 + 12, \quad 8 + 15 = 10 + 13, \quad 9 + 11 = 6 + 14, \\
&10 + 12 = 7 + 15, \quad 11 + 18 = 13 + 16, \quad \dots\dots\dots
\end{aligned}$$

縦に見て、 $1 + 12 = 2 + 11, 6 + 17 = 7 + 16, 11 + 22 = 12 + 21, \dots\dots\dots$   
も成り立つ。

「外項・内項の等和」も成り立つ。

$$1 + 5 = 2 + 4 = 3 \times 2, \quad 6 + 10 = 7 + 9 = 8 \times 2, \quad 11 + 15 = 12 + 14 = 13 \times 2, \quad \dots\dots\dots$$

また  $1 + 21 = 6 + 16 = 11 \times 2, 2 + 22 = 7 + 17 = 12 \times 2, \dots\dots\dots$

このようにいろいろなことが成り立つ原初五方陣だが、肝心の行列の定和が成り立っていない。だから、原初五方陣を「魔方陣」と言うことはできないが、それでも既に連立型の「対称かつ汎五方陣」なのである。何の苦もなく奇跡の条件を実現してしまっている。

2. 良く知られている五次魔方陣のいろいろなタイプと比較して見よう。

性質\種類	原初五方陣	対称五方陣	汎五方陣	対称汎五方陣
補数対称性	○	○	×	○
汎対角線の定和	○ (65)	×	○ (65)	○ (65)
行列の定和	×	○(65)	○(65)	○(65)
クロスサムの等和	○	×	×	×
解の総数	いろいろ	48544	3600	16

3. では、1のような性質を保持したまま数字の並べ替えができるものであろうか。原初方陣の内部で並び替えの可能性を探ろう。付加する条件をどのようにしたら良いだろうか。

### # 2. 原初五方陣とひな形変換法によって補数対称型五方陣を作る

原初五方陣から対称五方陣に作るには、どのようにすれば良いだろうか。次のような手順を踏んで進むことにしよう。

- (1) 最良のひな形を選定する。
- (2) ひな形変換法を設計する。
- (3) 原初五方陣構成の基本条件をその「ひな形」から教わる。
- (4) 原初五方陣 48544 個を構成する。
- (5) その各個にひな形変換法を実行して、対称五方陣 48544 個を作る。

## 2-1. 共通の「ひな形」と「ひな形変換法」

上の (1) ~ (3) については、次図のようにする。

/原初五方陣					/ひな形													
24	25	21	22	23	24	25	21	22	23	3	25	17	14	6	3	25	17	14
4	5	1   2   3   4   5					1	2	12	9	1   23   20   12   9					1	23	
9	10	6   7   8   9   10					6	7	21	18	15   7   4   21   18					15	7	
14	15	11   12   13   14   15					11	12	10	2	24   16   13   10   2					24	16	
19	20	16   17   18   19   20					16	17	19	11	8   5   22   19   11					8	5	
24	25	21   22   23   24   25					21	22	3	25	17   14   6   3   25					17	14	
4	5	1	2	3	4	5	1	2	12	9	1	23	20	12	9	1	23	

### \* 原初五方陣構成のための基本条件式: C=65 \*

$$\begin{array}{ll}
 n1+n23+n20+n12+n9=C & \dots (1) \\
 n15+n7+n4+n21+n18=C & \dots (2) \\
 n24+n16+n13+n10+n2=C & \dots (3) \\
 n8+n5+n22+n19+n11=C & \dots (4) \\
 n17+n14+n6+n3+n25=C & \dots (5)
 \end{array}
 \quad
 \begin{array}{ll}
 | & n1+n15+n24+n8+n17=C & \dots (6) \\
 | & n23+n7+n16+n5+n14=C & \dots (7) \\
 | & n20+n4+n13+n22+n6=C & \dots (8) \\
 | & n12+n21+n10+n19+n3=C & \dots (9) \\
 | & n9+n18+n2+n11+n25=C & \dots (10)
 \end{array}$$

### \* 補数ペアの対称配置: CC=26 \*

$$\begin{aligned}
 n1+n25=n23+n3=n20+n6=n12+n14=n9+n17=n15+n11=n7+n19 \\
 =n4+n22=n21+n5=n18+n8=n24+n2=n16+n10=n13+n13=CC & \dots (cp)
 \end{aligned}$$

/\*\*/

/\* ひな形変換法 \*/

```

void damt(void){
    d[1]=c[1];   d[2]=c[23];   d[3]=c[20];   d[4]=c[12];   d[5]=c[9];
    d[6]=c[15];   d[7]=c[7];   d[8]=c[4];   d[9]=c[21];   d[10]=c[18];
    d[11]=c[24];  d[12]=c[16];  d[13]=c[13];  d[14]=c[10];  d[15]=c[2];
    d[16]=c[8];   d[17]=c[5];   d[18]=c[22];  d[19]=c[19];  d[20]=c[11];
    d[21]=c[17];  d[22]=c[14];  d[23]=c[6];   d[24]=c[3];   d[25]=c[25];
}

```

/\*\*/

ひな形は、連立型の対称汎五方陣の中から選んだ。そしてひな形変換法を上述のように設計する。原初五方陣構成の基本定義式も、ひな形から教わる。ひな形は、目的方陣そのものであるから、必ず補数対称型でなくては困る。それを原初五方陣にも反映させなければならない。そういうわけで上述のような基本定義式が必要になるのである。

基本定義式には当然、補数の定和と対称配置の条件式も加える。そして、これが肝心なことだが、それだけで十分である。他には何も要らない。「クロスサム等和」の条件式も、実は無くて全然かまわない。「外項・内項の等和式」も必要としない。

2-2. あとは、普通の魔方陣を構成するのと全く同じやり方で、原初五方陣を構成する。標準解リスト出力用の不等式を全然用いなければ、388352 個の原始解を得る。

リスト出力用不等式は、ひな形に敬意を表して、 $n1 < n25$  and  $n1 < n9 < n17$  とする。変換後の対称五方陣を損なわずに 48544 個の標準解を得られる。下のリストを見てほしい。

最初は、比較のために用いる基準のリスト。n1<n25 and n1<n5<n25 下で標準解を得る。  
次のリストで、原初五方陣と変換・再構成した対称五方陣とを対照させて出力した。

[補数対称型五方陣の標準解のリスト (部分) : チェック・サムつき: |Rw|:行和, |C|:列和]

1/ Rw C	7793/ Rw C	14601/ Rw C	21805/ Rw C
1 21 20 19 4 65 65	2 20 22 16 5 65 65	3 22 20 15 5 65 65	4 19 21 15 6 65 65
15 9 8 10 23 65 65	14 11 8 9 23 65 65	14 9 8 10 24 65 65	14 8 3 16 24 65 65
24 12 13 14 2 65 65	25 7 13 19 1 65 65	25 7 13 19 1 65 65	25 17 13 9 1 65 65
3 16 18 17 11 65 65	3 17 18 15 12 65 65	2 16 18 17 12 65 65	2 10 23 18 12 65 65
22 7 6 5 25 65 65	21 10 4 6 24 65 65	21 11 6 4 23 65 65	20 11 5 7 22 65 65
27597/ Rw C	33193/ Rw C	37313/ Rw C	41081/ Rw C
5 23 10 20 7 65 65	6 22 18 10 9 65 65	7 22 16 11 9 65 65	8 23 5 19 10 65 65
14 8 4 15 24 65 65	15 5 14 7 24 65 65	14 3 18 6 24 65 65	14 4 17 6 24 65 65
25 17 13 9 1 65 65	25 3 13 23 1 65 65	25 5 13 21 1 65 65	25 11 13 15 1 65 65
2 11 22 18 12 65 65	2 19 12 21 11 65 65	2 20 8 23 12 65 65	2 20 9 22 12 65 65
19 6 16 3 21 65 65	17 16 8 4 20 65 65	17 15 10 4 19 65 65	16 7 21 3 18 65 65
44129/ Rw C	46533/ Rw C	48065/ Rw C	
9 23 6 16 11 65 65	10 22 2 20 11 65 65	11 22 18 2 12 65 65	
14 5 4 18 24 65 65	12 7 18 5 23 65 65	10 9 6 19 21 65 65	
25 19 13 7 1 65 65	25 9 13 17 1 65 65	25 3 13 23 1 65 65	
2 8 22 21 12 65 65	3 21 8 19 14 65 65	5 7 20 17 16 65 65	
15 10 20 3 17 65 65	15 6 24 4 16 65 65	14 24 8 4 15 65 65	[Count = 48544]

[原初五方陣と再構成した補数対称型五方陣の対照リスト (部分) : 原初五方陣/P, 目的方陣/R]

1/P	251/R	7/P	293/R	12/P	505/R
1 2 3 4 5	1 23 20 12 9	1 2 3 5 4	1 23 19 12 10	1 2 3 6 5	1 23 18 12 11
6 7 8 9 10	15 7 4 21 18	7 6 9 10 8	15 6 5 22 17	8 19 16 11 22	9 19 6 21 10
11 12 13 14 15	24 16 13 10 2	11 12 13 14 15	24 18 13 8 2	17 12 13 14 9	24 4 13 22 2
16 17 18 19 20	8 5 22 19 11	18 16 17 20 19	9 4 21 20 11	4 15 10 7 18	16 5 20 7 17
21 22 23 24 25	17 14 6 3 25	22 21 23 24 25	16 14 7 3 25	21 20 23 24 25	15 14 8 3 25
23/P	339/R	27/P	495/R	30/P	283/R
1 2 3 8 5	1 23 14 20 7	1 2 3 9 5	1 23 11 22 8	1 2 3 10 7	1 23 14 21 6
12 9 10 7 4	11 9 8 21 16	15 19 16 8 12	6 19 9 21 10	12 8 9 6 4	11 8 10 19 17
15 20 13 6 11	24 22 13 4 2	20 22 13 4 6	24 14 13 12 2	15 21 13 5 11	24 22 13 4 2
22 19 16 17 14	10 5 18 17 15	14 18 10 7 11	16 5 17 7 20	22 20 17 18 14	9 7 16 18 15
21 18 23 24 25	19 6 12 3 25	21 17 23 24 25	18 4 15 3 25	19 16 23 24 25	20 5 12 3 25
33/P	567/R	35/P	357/R	42/P	600/R
1 2 3 11 5	1 23 9 22 10	1 2 3 12 5	1 23 20 17 4	1 2 3 14 5	1 23 8 22 11
17 19 18 10 12	6 19 11 21 8	6 10 11 4 8	7 10 12 21 15	18 17 19 11 10	6 17 14 21 7
20 22 13 4 6	24 14 13 12 2	19 17 13 9 7	24 18 13 8 2	20 22 13 4 6	24 16 13 10 2
14 16 8 7 9	18 5 15 7 20	18 22 15 16 20	11 5 14 16 19	16 15 7 9 8	19 5 12 9 20
21 15 23 24 25	16 4 17 3 25	21 14 23 24 25	22 9 6 3 25	21 12 23 24 25	15 4 18 3 25
49/P	590/R	51/P	380/R	54/P	589/R
1 2 3 15 6	1 23 9 22 10	1 2 3 16 8	1 23 14 21 6	1 2 3 17 4	1 23 11 20 10
17 18 19 10 12	5 18 15 20 7	12 7 11 6 4	9 7 16 18 15	15 14 19 10 8	5 14 17 22 7
21 22 13 4 5	24 14 13 12 2	17 21 13 5 9	24 22 13 4 2	21 20 13 6 5	24 18 13 8 2
14 16 7 8 9	19 6 11 8 21	22 20 15 19 14	11 8 10 19 17	18 16 7 12 11	19 4 9 12 21
20 11 23 24 25	16 4 17 3 25	18 10 23 24 25	20 5 12 3 25	22 9 23 24 25	16 6 15 3 25
57/P	581/R	61/P	587/R	72/P	448/R
1 2 3 18 5	1 23 12 20 9	1 2 3 20 4	1 23 14 17 10	1 2 3 21 6	1 23 19 12 10
14 15 19 9 10	4 15 18 21 7	12 11 19 10 8	5 11 20 22 7	7 4 15 10 8	9 4 21 20 11
22 20 13 6 4	24 16 13 10 2	21 17 13 9 5	24 18 13 8 2	17 12 13 14 9	24 18 13 8 2
16 17 7 11 12	19 5 8 11 22	18 16 7 15 14	19 4 6 15 21	18 16 11 22 19	15 6 5 22 17
21 8 23 24 25	17 6 14 3 25	22 6 23 24 25	16 9 12 3 25	20 5 23 24 25	16 14 7 3 25
77/P	599/R	83/P	265/R	92/P	351/R
1 2 3 22 5	1 23 14 16 11	1 2 4 3 6	1 22 21 10 11	1 2 4 5 3	1 22 19 15 8
12 9 19 11 8	6 9 22 21 7	5 7 8 11 12	17 7 3 20 18	7 9 10 8 6	12 9 5 23 16
20 16 13 10 6	24 18 13 8 2	9 10 13 16 17	24 14 13 12 2	14 15 13 11 12	24 20 13 6 2
18 15 7 17 14	19 5 4 17 20	14 15 18 19 21	8 6 23 19 9	20 18 16 17 19	10 3 21 17 14
21 4 23 24 25	15 10 12 3 25	20 23 22 24 25	15 16 5 4 25	23 21 22 24 25	18 11 7 4 25

101/P	572/R	107/P	308/R	114/P	601/R
1 2 4 6 3	1 22 14 17 11	1 2 4 7 6	1 22 23 8 11	1 2 4 8 3	1 22 14 17 11
12 21 18 11 16	7 21 6 23 8	3 5 9 11 12	16 5 7 20 17	12 21 19 11 16	6 21 8 23 7
19 17 13 9 7	24 10 13 16 2	10 8 13 18 16	24 14 13 12 2	20 17 13 9 6	24 10 13 16 2
10 15 8 5 14	18 3 20 5 19	14 15 17 21 23	9 6 19 21 10	10 15 7 5 14	19 3 18 5 20
23 20 22 24 25	15 9 12 4 25	20 19 22 24 25	15 18 3 4 25	23 18 22 24 25	15 9 12 4 25
118/P	391/R	124/P	428/R	125/P	274/R
1 2 4 9 3	1 22 21 14 7	1 2 4 10 5	1 22 12 23 7	1 2 4 11 6	1 22 23 14 5
5 8 11 7 6	10 8 9 23 15	14 17 15 7 8	6 17 10 21 11	3 7 9 5 8	10 7 11 20 17
16 14 13 12 10	24 20 13 6 2	20 23 13 3 6	24 18 13 8 2	16 14 13 12 10	24 18 13 8 2
20 19 15 18 21	11 3 17 18 16	18 19 11 9 12	15 5 16 9 20	18 21 17 19 23	9 6 15 19 16
23 17 22 24 25	19 12 5 4 25	21 16 22 24 25	19 3 14 4 25	20 15 22 24 25	21 12 3 4 25
133/P	602/R	136/P	603/R	139/P	484/R
1 2 4 12 3	1 22 10 21 11	1 2 4 14 9	1 22 8 23 11	1 2 4 15 3	1 22 17 18 7
16 17 19 11 8	6 17 12 23 7	18 21 19 11 16	6 21 14 17 7	9 12 16 7 6	5 12 15 23 10
20 21 13 5 6	24 18 13 8 2	20 23 13 3 6	24 10 13 16 2	21 18 13 8 5	24 20 13 6 2
18 15 7 9 10	19 3 14 9 20	10 15 7 5 8	19 9 12 5 20	20 19 10 14 17	16 3 11 14 21
23 14 22 24 25	15 5 16 4 25	17 12 22 24 25	15 3 18 4 25	23 11 22 24 25	19 8 9 4 25
147/P	188/R	148/P	461/R	154/P	459/R
1 2 4 16 17	1 22 12 23 7	1 2 4 17 6	1 22 19 18 5	1 2 4 18 7	1 22 20 17 5
14 5 6 7 8	15 5 16 9 20	7 15 16 5 12	3 15 17 20 10	6 15 16 5 14	3 15 18 19 10
11 23 13 3 15	24 18 13 8 2	23 18 13 8 3	24 14 13 12 2	23 17 13 9 3	24 12 13 14 2
18 19 20 21 12	6 17 10 21 11	14 21 10 11 19	16 6 9 11 23	12 21 10 11 20	16 7 8 11 23
9 10 22 24 25	19 3 14 4 25	20 9 22 24 25	21 8 7 4 25	19 8 22 24 25	21 9 6 4 25
158/P	507/R	165/P	462/R	171/P	508/R
1 2 4 19 5	1 22 23 8 11	1 2 4 20 9	1 22 18 19 5	1 2 4 21 7	1 22 23 8 11
3 6 16 11 12	9 6 19 21 10	8 15 16 5 14	3 15 20 17 10	3 6 16 11 14	9 6 21 19 10
17 8 13 18 9	24 14 13 12 2	23 19 13 7 3	24 12 13 14 2	17 8 13 18 9	24 12 13 14 2
14 15 10 20 23	16 5 7 20 17	12 21 10 11 18	16 9 6 11 23	12 15 10 20 23	16 7 5 20 17
21 7 22 24 25	15 18 3 4 25	17 6 22 24 25	21 7 8 4 25	19 5 22 24 25	15 18 3 4 25
180/P	547/R	189/P	296/R	192/P	510/R
1 2 4 23 7	1 22 21 10 11	1 2 5 3 4	1 21 19 14 10	1 2 5 4 3	1 21 20 12 11
5 6 17 11 12	8 6 23 19 9	7 8 9 10 6	15 8 3 22 17	6 19 16 11 18	9 19 4 23 10
18 10 13 16 8	24 14 13 12 2	11 14 13 12 15	24 20 13 6 2	17 12 13 14 9	24 8 13 18 2
14 15 9 20 21	17 7 3 20 18	20 16 17 18 19	9 4 23 18 11	8 15 10 7 20	16 3 22 7 17
19 3 22 24 25	15 16 5 4 25	22 23 21 24 25	16 12 7 5 25	23 22 21 24 25	15 14 6 5 25
249/P	513/R	331/P	409/R	387/P	278/R
1 2 6 3 4	1 20 19 14 11	1 2 7 3 5	1 19 15 22 8	1 2 8 3 11	1 18 19 22 5
7 21 16 11 18	9 21 3 22 10	11 17 12 8 6	10 17 3 21 14	7 20 9 5 14	10 20 3 15 17
17 14 13 12 9	24 8 13 18 2	16 22 13 4 10	24 20 13 6 2	16 22 13 4 10	24 12 13 14 2
8 15 10 5 19	16 4 23 5 17	20 18 14 9 15	12 5 23 9 16	12 21 17 6 19	9 11 23 6 16
22 23 20 24 25	15 12 7 6 25	21 23 19 24 25	18 4 11 7 25	15 23 18 24 25	21 4 7 8 25
449/P	431/R	497/P	95/R	553/P	354/R
1 2 9 5 4	1 17 23 16 8	1 2 10 3 14	1 16 21 20 7	1 2 11 5 17	1 15 19 22 8
3 20 15 8 12	7 20 5 22 11	5 11 4 7 8	17 11 3 12 22	7 23 10 8 20	12 23 5 9 16
19 16 13 10 7	24 14 13 12 2	9 20 13 6 17	24 18 13 8 2	14 22 13 4 12	24 6 13 20 2
14 18 11 6 23	15 4 21 6 19	18 19 22 15 21	4 14 23 15 9	6 18 16 3 19	10 17 21 3 14
22 21 17 24 25	18 10 3 9 25	12 23 16 24 25	19 6 5 10 25	9 21 15 24 25	18 4 7 11 25
575/P	517/R	635/P	519/R	657/P	200/R
1 2 12 3 5	1 14 19 20 11	1 2 14 4 7	1 12 20 21 11	1 2 15 3 19	1 11 21 22 10
7 22 16 11 8	9 22 3 21 10	6 23 16 11 8	9 23 4 19 10	5 17 6 10 12	18 17 3 7 20
17 20 13 6 9	24 18 13 8 2	17 21 13 5 9	24 18 13 8 2	8 22 13 4 18	24 14 13 12 2
18 15 10 4 19	16 5 23 4 17	18 15 10 3 20	16 7 22 3 17	14 16 20 9 21	6 19 23 9 8
21 23 14 24 25	15 6 7 12 25	19 22 12 24 25	15 5 6 14 25	7 23 11 24 25	16 4 5 15 25
665/P	270/R	673/P	321/R	677/P	809/R
1 2 16 3 19	1 10 21 22 11	1 2 18 5 19	1 8 23 22 11	1 3 2 4 5	1 24 20 8 12
5 20 8 11 14	17 20 3 7 18	3 20 9 11 12	16 20 5 7 17	6 7 10 12 15	17 7 4 21 16
9 22 13 4 17	24 12 13 14 2	10 22 13 4 16	24 14 13 12 2	9 8 13 18 17	23 11 13 15 3
12 15 18 6 21	8 19 23 6 9	14 15 17 6 23	9 19 21 6 10	11 14 16 19 20	10 5 22 19 9
7 23 10 24 25	15 4 5 16 25	7 21 8 24 25	15 4 3 18 25	21 22 24 23 25	14 18 6 2 25
701/P	792/R	739/P	787/R	783/P	762/R
1 3 4 2 5	1 22 20 10 12	1 3 5 2 4	1 21 20 15 8	1 3 6 2 9	1 20 14 22 8
6 7 9 12 11	18 7 2 21 17	6 10 9 8 7	14 10 2 22 17	12 11 7 8 5	16 11 2 17 19
8 10 13 16 18	23 15 13 11 3	12 15 13 11 14	23 19 13 7 3	10 22 13 4 16	23 21 13 5 3
15 14 17 19 20	9 5 24 19 8	19 18 17 16 20	9 4 24 16 12	21 18 19 15 14	7 9 24 15 10
21 24 22 23 25	14 16 6 4 25	22 24 21 23 25	18 11 6 5 25	17 24 20 23 25	18 4 12 6 25

795/P	835/R	823/P	720/R	843/P	836/R
1 3 7 2 10	1 19 20 17 8	1 3 8 2 11	1 18 22 14 10	1 3 9 2 4	1 17 20 19 8
6 22 12 8 21	11 22 2 16 14	4 7 5 10 9	20 7 2 15 21	6 16 12 8 5	11 16 2 22 14
15 17 13 9 11	23 5 13 21 3	6 14 13 12 20	23 17 13 9 3	15 19 13 7 11	23 21 13 5 3
5 18 14 4 20	12 10 24 4 15	17 16 21 19 22	5 11 24 19 6	21 18 14 10 20	12 4 24 10 15
16 24 19 23 25	18 9 6 7 25	15 24 18 23 25	16 12 4 8 25	22 24 17 23 25	18 7 6 9 25
867/P	696/R	889/P	865/R	911/P	739/R
1 3 10 2 15	1 16 20 21 7	1 3 11 2 6	1 15 18 22 9	1 3 12 2 17	1 14 22 21 7
6 12 4 7 9	18 12 2 11 22	8 21 14 9 7	10 21 2 20 12	4 18 6 7 15	16 18 2 9 20
8 21 13 5 18	23 17 13 9 3	16 22 13 4 10	23 19 13 7 3	10 21 13 5 16	23 11 13 15 3
17 19 22 14 20	4 15 24 14 8	19 17 12 5 18	14 6 24 5 16	11 19 20 8 22	6 17 24 8 10
11 24 16 23 25	19 5 6 10 25	20 24 15 23 25	17 4 8 11 25	9 24 14 23 25	19 5 4 12 25
919/P	710/R	921/P	866/R	933/P	793/R
1 3 14 8 21	1 12 24 19 9	1 3 15 4 8	1 11 24 20 9	1 3 16 2 19	1 10 20 22 12
2 10 4 9 11	20 10 8 5 22	2 21 14 9 7	10 21 4 18 12	6 21 9 12 15	18 21 2 7 17
6 19 13 7 20	23 15 13 11 3	16 20 13 6 10	23 19 13 7 3	8 22 13 4 18	23 11 13 15 3
15 17 22 16 24	4 21 18 16 6	19 17 12 5 24	14 8 22 5 16	11 14 17 5 20	9 19 24 5 8
5 18 12 23 25	17 7 2 14 25	18 22 11 23 25	17 6 2 15 25	7 24 10 23 25	14 4 6 16 25
945/P	756/R	949/P	814/R	953/P	816/R
1 3 17 8 22	1 9 24 21 10	1 3 18 4 19	1 8 24 20 12	1 3 19 4 18	1 7 24 21 12
2 14 6 10 11	19 14 8 4 20	2 21 10 12 15	17 21 4 7 16	2 20 10 12 11	17 20 4 8 16
7 21 13 5 19	23 15 13 11 3	9 20 13 6 17	23 11 13 15 3	9 21 13 5 17	23 15 13 11 3
15 16 20 12 24	6 22 18 12 7	11 14 16 5 24	10 19 22 5 9	15 14 16 6 24	10 18 22 6 9
4 18 9 23 25	16 5 2 17 25	7 22 8 23 25	14 6 2 18 25	8 22 7 23 25	14 5 2 19 25
957/P	818/R	961/P	1157/R	1087/P	1158/R
1 3 20 8 21	1 6 24 22 12	1 4 2 3 5	1 24 19 10 11	1 4 3 2 5	1 23 20 10 11
2 19 10 12 11	17 19 8 5 16	7 6 9 11 12	18 6 3 21 17	6 7 9 11 12	18 7 2 21 17
9 22 13 4 17	23 15 13 11 3	8 10 13 16 18	22 14 13 12 4	8 10 13 16 18	22 14 13 12 4
15 14 16 7 24	10 21 18 7 9	14 15 17 20 19	9 5 23 20 8	14 15 17 19 20	9 5 24 19 8
5 18 6 23 25	14 4 2 20 25	21 23 24 22 25	15 16 7 2 25	21 24 23 22 25	15 16 6 3 25
1501/P	1613/R	1773/P	1967/R	2281/P	2556/R
1 5 2 4 3	1 24 20 12 8	1 6 2 3 4	1 24 17 12 11	1 7 2 3 8	1 24 11 20 9
6 7 10 8 9	15 7 4 23 16	9 5 10 11 8	19 5 3 22 16	15 22 16 9 21	12 22 3 18 10
11 12 13 14 15	21 17 13 9 5	7 12 13 14 19	20 18 13 8 6	14 20 13 6 12	19 5 13 21 7
17 18 16 19 20	10 3 22 19 11	18 15 16 21 17	10 4 23 21 7	5 17 10 4 11	16 8 23 4 14
23 22 24 21 25	18 14 6 2 25	22 23 24 20 25	15 14 9 2 25	18 23 24 19 25	17 6 15 2 25
2701/P	2831/R	3017/P	3185/R	3253/P	3423/R
1 8 2 3 4	1 24 21 9 10	1 9 2 4 3	1 24 12 20 8	1 10 2 3 4	1 24 19 12 9
5 6 11 10 12	19 6 3 22 15	14 21 19 8 15	10 21 4 23 7	7 5 11 9 8	20 5 3 22 15
7 9 13 17 19	18 14 13 12 8	16 20 13 6 10	17 11 13 15 9	6 12 13 14 20	16 18 13 8 10
14 16 15 20 21	11 4 23 20 7	11 18 7 5 12	19 3 22 5 16	18 17 15 21 19	11 4 23 21 6
22 23 24 18 25	16 17 5 2 25	23 22 24 17 25	18 6 14 2 25	22 23 24 16 25	17 14 7 2 25
3705/P	3711/R	4045/P	4154/R	4441/P	4485/R
1 11 2 3 14	1 24 19 16 5	1 12 2 3 7	1 24 15 17 8	1 14 2 3 10	1 24 15 20 5
7 8 6 5 17	22 8 3 12 20	11 5 10 8 6	22 5 3 19 16	11 7 9 5 8	22 7 3 16 17
4 16 13 10 22	15 9 13 17 11	4 17 13 9 22	14 20 13 6 12	4 20 13 6 22	12 18 13 8 14
9 21 20 18 19	6 14 23 18 4	20 18 16 21 15	10 7 23 21 4	18 21 17 19 15	9 10 23 19 4
12 23 24 15 25	21 10 7 2 25	19 23 24 14 25	18 9 11 2 25	16 23 24 12 25	21 6 11 2 25
4809/P	5026/R	5129/P	5297/R	5593/P	5693/R
1 15 2 3 4	1 24 19 12 9	1 16 2 3 4	1 24 11 21 8	1 17 2 3 4	1 24 15 20 5
7 18 20 9 21	16 18 3 22 6	15 12 17 8 6	19 12 3 22 9	11 12 16 5 7	18 12 3 22 10
10 12 13 14 16	11 5 13 21 15	7 21 13 5 19	10 20 13 6 16	8 20 13 6 18	9 19 13 7 17
5 17 6 8 19	20 4 23 8 10	20 18 9 14 11	17 4 23 14 7	19 21 10 14 15	16 4 23 14 8
22 23 24 11 25	17 14 7 2 25	22 23 24 10 25	18 5 15 2 25	22 23 24 9 25	21 6 11 2 25
5961/P	6111/R	6313/P	6358/R	6589/P	6705/R
1 18 2 3 4	1 24 15 20 5	1 19 2 3 8	1 24 15 20 5	1 20 2 3 4	1 24 15 17 8
11 17 19 5 12	16 17 3 22 7	11 10 14 5 9	22 10 3 18 12	11 12 19 8 10	21 12 3 22 7
10 20 13 6 16	8 14 13 12 18	4 20 13 6 22	7 17 13 9 19	5 17 13 9 21	6 16 13 10 20
14 21 7 9 15	19 4 23 9 10	17 21 12 16 15	14 8 23 16 4	16 18 7 14 15	19 4 23 14 5
22 23 24 8 25	21 6 11 2 25	18 23 24 7 25	21 6 11 2 25	22 23 24 6 25	18 9 11 2 25
6865/P	6891/R	7117/P	7217/R	7413/P	7443/R
1 21 2 4 9	1 24 18 19 3	1 22 2 5 6	1 24 17 15 8	1 23 2 6 11	1 24 16 19 5
8 14 16 3 15	20 14 4 17 10	9 10 19 8 12	23 10 5 20 7	10 14 18 5 17	22 14 6 15 8
6 19 13 7 20	5 11 13 15 21	3 15 13 11 23	4 14 13 12 22	4 19 13 7 22	3 9 13 17 23
11 23 10 12 18	16 9 22 12 6	14 18 7 16 17	19 6 21 16 3	9 21 8 12 16	18 11 20 12 4
17 22 24 5 25	23 7 8 2 25	20 21 24 4 25	18 11 9 2 25	15 20 24 3 25	21 7 10 2 25









とも確かめられた。不思議といえば、実に不思議だ。

なお原初五方陣の方は、出力順序をソートして入れ替え、清書しておいた。高速構成法を使うと、出力結果が必ずしもスマートではなくなってしまう恨みがある。やはり、第一基本型が最初に来た方が見やすい。なお、n/Rは、基準リストを参照して得た解番号だ。

この際、原初五方陣はそのままに、他のひな形とひな形変換法を用いて試してみたが、目的方陣の全数を再構成することはできなかった。原始解どうして試みても駄目だった。

他のひな形を用いて新たにひな形変換法を定義し、原初五方陣も新たに構成してみた。その場合には、目的方陣の全数を再構成できた。「最良のひな形」は、唯一無二ではない。

ただし、途中の原初五方陣は、前とは全然違うものになる。つまり、原初方陣が個別のひな形に依存していて、独立して存在しているわけではない。汎四方陣の場合とは異なる。

### # 3. 原初五方陣とひな形変換法によって汎対角線型五方陣を作る

同じ様にして、汎魔方陣を作れるだろうか。次の課題を実験しよう。

ひな形を選び、ひな形変換法を設計する。そしてひな形から教わった基本定義式を用いて原初五方陣 3600 個を構成する。その各個にひな形変換法を施し汎五方陣 3600 個を作る。

対称五方陣の場合と同じ「ひな形」を用いる。共通のひな形とひな形変換法を下のように定義する。前と違うのは、原初五方陣構成のための定義式。基本条件式に加えて、新たに汎対角線定和の定義式が必要になる。補数定和と対称配置の定義式が、今度は必要無い。

/原初五方陣

/ひな形

24 25 21 22 23 24 25 21 22	3 25 17 14 6 3 25 17 14
4 5   1   2   3   4   5   1 2 12 9	1 23 20 12 9   1 23
9 10   6   7   8   9   10   6 7 21 18	15 7   4   21   18   15 7
14 15   11   12   13   14   15   11 12 10 2	24 16 13 10   2 24 16
19 20   16   17   18   19   20   16 17 19 11	8   5   22   19   11   8 5
24 25   21   22   23   24   25   21 22 3 25	17   14   6   3   25   17 14
4 5 1 2 3 4 5 1 2 12 9	1 23 20 12 9 1 23

#### \* 原初五方陣構成のための基本条件式: C=65 \*

n1+n23+n20+n12+n9=C ... (1)	n1+n15+n24+n8+n17=C ... (6)
n15+n7+n4+n21+n18=C ... (2)	n23+n7+n16+n5+n14=C ... (7)
n24+n16+n13+n10+n2=C ... (3)	n20+n4+n13+n22+n6=C ... (8)
n8+n5+n22+n19+n11=C ... (4)	n12+n21+n10+n19+n3=C ... (9)
n17+n14+n6+n3+n25=C ... (5)	n9+n18+n2+n11+n25=C ... (10)

#### \* 汎対角線定和の定義式: C=65 \*

n1+n7+n13+n19+n25=C ... (11)	n1+n18+n10+n22+n14=C ... (12)
n23+n4+n10+n11+n17=C ... (13)	n23+n15+n2+n19+n6=C ... (14)
n20+n21+n2+n8+n14=C ... (15)	n20+n7+n24+n11+n3=C ... (16)
n12+n18+n24+n5+n6=C ... (17)	n12+n4+n16+n8+n25=C ... (18)
n9+n15+n16+n22+n3=C ... (19)	n9+n21+n13+n5+n17=C ... (20)

/\*\*/

/\* ひな形変換法 \*/

```
void damt(void){
    d[1]=c[1];   d[2]=c[23];   d[3]=c[20];   d[4]=c[12];   d[5]=c[9];
    d[6]=c[15];   d[7]=c[7];    d[8]=c[4];    d[9]=c[21];   d[10]=c[18];
    d[11]=c[24];  d[12]=c[16];  d[13]=c[13];  d[14]=c[10];  d[15]=c[2];
```

```

d[16]=c[8]; d[17]=c[5]; d[18]=c[22]; d[19]=c[19]; d[20]=c[11];
d[21]=c[17]; d[22]=c[14]; d[23]=c[6]; d[24]=c[3]; d[25]=c[25];
}
/**/

```

これで汎五方陣に対応する新たな原初五方陣が構成できる。標準解リスト出力用の不等式を用いなければ、28800個の原初五方陣が出来る。

不等式を用いるなら、ひな形に敬意を表して  $n1 < n25$  and  $n1 < n9 < n17$  としよう。そうすれば、原初五方陣も、再構成した汎五方陣も、3600個の標準解を出力するはずである。

「逆関数を解く」スタイルの原初方陣構成法は、次のようにプログラムする。

```

/* 原初五方陣を構成するためのプログラムの要点: Dictated by Kanji Setsuda */
/* Define Level 1: */
/* Set N1 */
void stp01(){
  short a;
  for(a=1;a<13;a++){
    if(uflg[a]==0){
      nm[1]=a; uflg[a]=1;
      stp02();
      uflg[a]=0;}
  }
}
/* Set N25(>n1) */
void stp02(){
  short a;
  for(a=25;a>nm[1];a--){
    if(uflg[a]==0){
      nm[25]=a; uflg[a]=1;
      stp03();
      uflg[a]=0;}
  }
}
/* Set N7 */
void stp03(){
  short a;
  for(a=1;a<26;a++){
    if(uflg[a]==0){
      nm[7]=a; uflg[a]=1;
      stp04();
      uflg[a]=0;}
  }
}
/* Set N19 */
void stp04(){
  short a;
  for(a=25;a>0;a--){
    if(uflg[a]==0){
      nm[19]=a; uflg[a]=1;
      stp05();
      uflg[a]=0;}
  }
}
/* Set n13=LSM-n1-n7-n19-n25 */
void stp05(){
  short a;
  a=LSM-nm[1]-nm[7]-nm[19]-nm[25];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[13]=a; uflg[a]=1;
      stp06();
      uflg[a]=0;}
  }
}
/* Define Level 2: */
/* Set N9(>n1) */
void stp06(){
  short a;
  for(a=nm[1]+1;a<26;a++){
    if(uflg[a]==0){
      nm[9]=a; uflg[a]=1;
      stp07();
      uflg[a]=0;}
  }
}
/* Set n17=LSM-n1-n9-n13-n25 & n9<n17 */
void stp07(){
  short a;
  a=LSM-nm[1]-nm[9]-nm[13]-nm[25];
  if((nm[9]<a)&&(a<26)){
    if(uflg[a]==0){
      nm[17]=a; uflg[a]=1;
      stp08();
      uflg[a]=0;}
  }
}
/* Set N21 */
void stp08(){
  short a;
  for(a=25;a>0;a--){
    if(uflg[a]==0){
      nm[21]=a; uflg[a]=1;
      stp09();
      uflg[a]=0;}
  }
}
/* Set n5=LSM-n9-n21-n13-n17 */
void stp09(){
  short a;
  a=LSM-nm[9]-nm[21]-nm[13]-nm[17];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[5]=a; uflg[a]=1;
      stp10();
      uflg[a]=0;}
  }
}

```

```

/* Define Level 3: */
/* Set N24 */
void stp10(){
  short a;
  for(a=25;a>0;a--){
    if(uflg[a]==0){
      nm[24]=a; uflg[a]=1;
      stp11();
      uflg[a]=0;}}
}
/* Set n20=LSM-n1-n7-n24-n13 */
void stp11(){
  short a;
  a=LSM-nm[1]-nm[7]-nm[24]-nm[13];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[20]=a; uflg[a]=1;
      stp12();
      uflg[a]=0;}}}
}
/* Set n2=LSM-n20-n9-n21-n13 */
void stp12(){
  short a;
  a=LSM-nm[20]-nm[9]-nm[21]-nm[13];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[2]=a; uflg[a]=1;
      stp13();
      uflg[a]=0;}}}
}
/* Set n6=LSM-n20-n24-n13-n2 */
void stp13(){
  short a;
  a=LSM-nm[20]-nm[24]-nm[13]-nm[2];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[6]=a; uflg[a]=1;
      stp14();
      uflg[a]=0;}}}
}
/* Define Level 4: */
/* Set N23 */
void stp14(){
  short a;
  for(a=25;a>0;a--){
    if(uflg[a]==0){
      nm[23]=a; uflg[a]=1;
      stp15();
      uflg[a]=0;}}
}
/* Set n12=LSM-n1-n23-n20-n9 */
void stp15(){
  short a;
  a=LSM-nm[1]-nm[23]-nm[20]-nm[9];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[12]=a; uflg[a]=1;

```

```

      stp16();
      uflg[a]=0;}}}
}
/* Set n15=LSM-n23-n2-n19-n6 */
void stp16(){
  short a;
  a=LSM-nm[23]-nm[2]-nm[19]-nm[6];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[15]=a; uflg[a]=1;
      stp17();
      uflg[a]=0;}}}
}
/* Set n8=LSM-n1-n15-n24-n17 */
void stp17(){
  short a;
  a=LSM-nm[1]-nm[15]-nm[24]-nm[17];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[8]=a; uflg[a]=1;
      stp18();
      uflg[a]=0;}}}
}
/* Set n18=LSM-n12-n24-n5-n6 */
void stp18(){
  short a;
  a=LSM-nm[12]-nm[24]-nm[5]-nm[6];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[18]=a; uflg[a]=1;
      stp19();
      uflg[a]=0;}}}
}
/* Set n11=LSM-n9-n18-n2-n25 */
void stp19(){
  short a;
  a=LSM-nm[9]-nm[18]-nm[2]-nm[25];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[11]=a; uflg[a]=1;
      stp20();
      uflg[a]=0;}}}
}
/* Set n4=LSM-n15-n7-n21-n18 */
void stp20(){
  short a;
  a=LSM-nm[15]-nm[7]-nm[21]-nm[18];
  if((0<a)&&(a<26)){
    if(uflg[a]==0){
      nm[4]=a; uflg[a]=1;
      stp21();
      uflg[a]=0;}}}
}
/* Define Level 5: */
/* Set n16=LSM-n12-n4-n8-n25 */
void stp21(){
  short a;
  a=LSM-nm[12]-nm[4]-nm[8]-nm[25];
  if((0<a)&&(a<26)){

```

```

    if(uflg[a]==0){
        nm[16]=a; uflg[a]=1;
        stp22();
        uflg[a]=0;}}
}
/* Set n10=LSM-n23-n4-n11-n17 */
void stp22(){
    short a,b;
    a=LSM-nm[23]-nm[4]-nm[11]-nm[17];
    if((0<a)&&(a<26)){
        b=LSM-nm[24]-nm[16]-nm[13]-nm[2];
        if(a==b){
            if(uflg[a]==0){
                nm[10]=a; uflg[a]=1;
                stp23();
                uflg[a]=0;}}}}
}
/* Set n22=LSM-n20-n4-n13-n6 */
void stp23(){
    short a,b;
    a=LSM-nm[20]-nm[4]-nm[13]-nm[6];
    if((0<a)&&(a<26)){
        b=LSM-nm[8]-nm[5]-nm[19]-nm[11];
        if(a==b){
            if(uflg[a]==0){
                nm[22]=a; uflg[a]=1;
                stp24();
                uflg[a]=0;}}}}
}
/* Set n14=LSM-n20-n21-n2-n8 */
void stp24(){
    short a,b,c;
    a=LSM-nm[20]-nm[21]-nm[2]-nm[8];
    if((0<a)&&(a<26)){
        b=LSM-nm[1]-nm[18]-nm[10]-nm[22];
        c=LSM-nm[23]-nm[7]-nm[16]-nm[5];
        if((a==b)&&(a==c)){
            if(uflg[a]==0){
                nm[14]=a; uflg[a]=1;
                stp25();
                uflg[a]=0;}}}}
}
/* Set n3=LSM-n9-n15-n16-n22 */
void stp25(){
    short a,b,c,d;
    a=LSM-nm[9]-nm[15]-nm[16]-nm[22];
    if((0<a)&&(a<26)){
        b=LSM-nm[20]-nm[7]-nm[24]-nm[11];
        c=LSM-nm[12]-nm[21]-nm[10]-nm[19];
        d=LSM-nm[17]-nm[14]-nm[6]-nm[25];
        if((a==b)&&(a==c)&&(a==d)){
            if(uflg[a]==0){
                nm[3]=a; uflg[a]=1;
                protorecord();
                uflg[a]=0;}}}}
}
/**/
/* Record the Answers */
void protorecord(){
    short n;
    cnt1++;
    for(n=1;n<26;n++){pn[cnt1][n]=nm[n];}
}
/**/

```

次のリストは、その実験結果の報告である。

最初に基準のリスト。n1<n25 and n1<n5<n25 で標準解を得ている。比較のために使用したい。次のリストで、原初五方陣と変換・再構成した汎五方陣とを対照させて出力した。

〔汎対角線五方陣の標準解のリスト（部分）：

チェック・サムつき：|Rw 行和, |Column 列和, |P1\P2Sums 汎対角線和]

1/ Rw P1\					37/ Rw P1\					73/ Rw P1\					109/ Rw P1\								
1	23	20	12	9	65 65\	1	23	19	12	10	65 65\	1	24	18	12	10	65 65\	1	24	17	13	10	65 65\
15	7	4	21	18	65 65\	14	7	5	21	18	65 65\	13	7	5	21	19	65 65\	12	8	5	21	19	65 65\
24	16	13	10	2	65 65\	25	16	13	9	2	65 65\	25	16	14	8	2	65 65\	25	16	14	7	3	65 65\
8	5	22	19	11	65 65\	8	4	22	20	11	65 65\	9	3	22	20	11	65 65\	9	2	23	20	11	65 65\
17	14	6	3	25	65 65\	17	15	6	3	24	65 65\	17	15	6	4	23	65 65\	18	15	6	4	22	65 65\
65 65 65 65 65 Column					65 65 65 65 65 Column					65 65 65 65 65 Column					65 65 65 65 65 Column								
/65/65/65/65/65 P2Sums					/65/65/65/65/65 P2Sums					/65/65/65/65/65 P2Sums					/65/65/65/65/65 P2Sums								
145/ Rw P1\					181/ Rw P1\					217/ Rw P1\					253/ Rw P1\								
1	18	25	12	9	65 65\	1	18	24	12	10	65 65\	1	19	23	12	10	65 65\	1	19	22	13	10	65 65\
15	7	4	16	23	65 65\	14	7	5	16	23	65 65\	13	7	5	16	24	65 65\	12	8	5	16	24	65 65\
19	21	13	10	2	65 65\	20	21	13	9	2	65 65\	20	21	14	8	2	65 65\	20	21	14	7	3	65 65\
8	5	17	24	11	65 65\	8	4	17	25	11	65 65\	9	3	17	25	11	65 65\	9	2	18	25	11	65 65\
22	14	6	3	20	65 65\	22	15	6	3	19	65 65\	22	15	6	4	18	65 65\	23	15	6	4	17	65 65\
65 65 65 65 65 Column					65 65 65 65 65 Column					65 65 65 65 65 Column					65 65 65 65 65 Column								
/65/65/65/65/65 P2Sums					/65/65/65/65/65 P2Sums					/65/65/65/65/65 P2Sums					/65/65/65/65/65 P2Sums								

289/ Rw P1\ 1 13 25 17 9 65 65\ 20 7 4 11 23 65 65\ 14 21 18 10 2 65 65\ 8 5 12 24 16 65 65\ 22 19 6 3 15 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					325/ Rw P1\ 1 13 24 17 10 65 65\ 19 7 5 11 23 65 65\ 15 21 18 9 2 65 65\ 8 4 12 25 16 65 65\ 22 20 6 3 14 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					361/ Rw P1\ 1 14 23 17 10 65 65\ 18 7 5 11 24 65 65\ 15 21 19 8 2 65 65\ 9 3 12 25 16 65 65\ 22 20 6 4 13 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					397/ Rw P1\ 1 14 22 18 10 65 65\ 17 8 5 11 24 65 65\ 15 21 19 7 3 65 65\ 9 2 13 25 16 65 65\ 23 20 6 4 12 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				
433/ Rw P1\ 1 8 25 17 14 65 65\ 20 12 4 6 23 65 65\ 9 21 18 15 2 65 65\ 13 5 7 24 16 65 65\ 22 19 11 3 10 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					469/ Rw P1\ 1 8 24 17 15 65 65\ 19 12 5 6 23 65 65\ 10 21 18 14 2 65 65\ 13 4 7 25 16 65 65\ 22 20 11 3 9 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					505/ Rw P1\ 1 9 23 17 15 65 65\ 18 12 5 6 24 65 65\ 10 21 19 13 2 65 65\ 14 3 7 25 16 65 65\ 22 20 11 4 8 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					541/ Rw P1\ 1 9 22 18 15 65 65\ 17 13 5 6 24 65 65\ 10 21 19 12 3 65 65\ 14 2 8 25 16 65 65\ 23 20 11 4 7 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				
577/ Rw P1\ 2 23 20 11 9 65 65\ 15 6 4 22 18 65 65\ 24 17 13 10 1 65 65\ 8 5 21 19 12 65 65\ 16 14 7 3 25 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					613/ Rw P1\ 2 23 19 11 10 65 65\ 14 6 5 22 18 65 65\ 25 17 13 9 1 65 65\ 8 4 21 20 12 65 65\ 16 15 7 3 24 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					649/ Rw P1\ 2 24 18 11 10 65 65\ 13 6 5 22 19 65 65\ 25 17 14 8 1 65 65\ 9 3 21 20 12 65 65\ 16 15 7 4 23 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					685/ Rw P1\ 2 24 16 13 10 65 65\ 11 8 5 22 19 65 65\ 25 17 14 6 3 65 65\ 9 1 23 20 12 65 65\ 18 15 7 4 21 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				
721/ Rw P1\ 2 18 25 11 9 65 65\ 15 6 4 17 23 65 65\ 19 22 13 10 1 65 65\ 8 5 16 24 12 65 65\ 21 14 7 3 20 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					757/ Rw P1\ 2 18 24 11 10 65 65\ 14 6 5 17 23 65 65\ 20 22 13 9 1 65 65\ 8 4 16 25 12 65 65\ 21 15 7 3 19 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					793/ Rw P1\ 2 19 23 11 10 65 65\ 13 6 5 17 24 65 65\ 20 22 14 8 1 65 65\ 9 3 16 25 12 65 65\ 21 15 7 4 18 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					829/ Rw P1\ 2 19 21 13 10 65 65\ 11 8 5 17 24 65 65\ 20 22 14 6 3 65 65\ 9 1 18 25 12 65 65\ 23 15 7 4 16 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				
865/ Rw P1\ 2 13 25 16 9 65 65\ 20 6 4 12 23 65 65\ 14 22 18 10 1 65 65\ 8 5 11 24 17 65 65\ 21 19 7 3 15 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					901/ Rw P1\ 2 13 24 16 10 65 65\ 19 6 5 12 23 65 65\ 15 22 18 9 1 65 65\ 8 4 11 25 17 65 65\ 21 20 7 3 14 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					937/ Rw P1\ 2 14 23 16 10 65 65\ 18 6 5 12 24 65 65\ 15 22 19 8 1 65 65\ 9 3 11 25 17 65 65\ 21 20 7 4 13 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					973/ Rw P1\ 2 14 21 18 10 65 65\ 16 8 5 12 24 65 65\ 15 22 19 6 3 65 65\ 9 1 13 25 17 65 65\ 23 20 7 4 11 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				
1009/ Rw P1\ 2 8 25 16 14 65 65\ 20 11 4 7 23 65 65\ 9 22 18 15 1 65 65\ 13 5 6 24 17 65 65\ 21 19 12 3 10 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					1045/ Rw P1\ 2 8 24 16 15 65 65\ 19 11 5 7 23 65 65\ 10 22 18 14 1 65 65\ 13 4 6 25 17 65 65\ 21 20 12 3 9 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					1081/ Rw P1\ 2 9 23 16 15 65 65\ 18 11 5 7 24 65 65\ 10 22 19 13 1 65 65\ 14 3 6 25 17 65 65\ 21 20 12 4 8 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					1117/ Rw P1\ 2 9 21 18 15 65 65\ 16 13 5 7 24 65 65\ 10 22 19 11 3 65 65\ 14 1 8 25 17 65 65\ 23 20 12 4 6 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				
1153/ Rw P1\ 3 22 20 11 9 65 65\ 15 6 4 23 17 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					1189/ Rw P1\ 3 22 19 11 10 65 65\ 14 6 5 23 17 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					1225/ Rw P1\ 3 24 17 11 10 65 65\ 12 6 5 23 19 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums					1261/ Rw P1\ 3 24 16 12 10 65 65\ 11 7 5 23 19 65 65\  65 65 65 65 65 Column /65 65 65 65 65 P2Sums				

24 18 12 10 1  65 65\ 7 5 21 19 13  65 65\ 16 14 8 2 25  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	25 18 12 9 1  65 65\ 7 4 21 20 13  65 65\ 16 15 8 2 24  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	25 18 14 7 1  65 65\ 9 2 21 20 13  65 65\ 16 15 8 4 22  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	25 18 14 6 2  65 65\ 9 1 22 20 13  65 65\ 17 15 8 4 21  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums
1297/ Rw P1\ 3 17 25 11 9  65 65\ 15 6 4 18 22  65 65\ 19 23 12 10 1  65 65\ 7 5 16 24 13  65 65\ 21 14 8 2 20  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1333/ Rw P1\ 3 17 24 11 10  65 65\ 14 6 5 18 22  65 65\ 20 23 12 9 1  65 65\ 7 4 16 25 13  65 65\ 21 15 8 2 19  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1369/ Rw P1\ 3 19 22 11 10  65 65\ 12 6 5 18 24  65 65\ 20 23 14 7 1  65 65\ 9 2 16 25 13  65 65\ 21 15 8 4 17  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1405/ Rw P1\ 3 19 21 12 10  65 65\ 11 7 5 18 24  65 65\ 20 23 14 6 2  65 65\ 9 1 17 25 13  65 65\ 22 15 8 4 16  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums
1441/ Rw P1\ 3 12 25 16 9  65 65\ 20 6 4 13 22  65 65\ 14 23 17 10 1  65 65\ 7 5 11 24 18  65 65\ 21 19 8 2 15  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1477/ Rw P1\ 3 12 24 16 10  65 65\ 19 6 5 13 22  65 65\ 15 23 17 9 1  65 65\ 7 4 11 25 18  65 65\ 21 20 8 2 14  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1513/ Rw P1\ 3 14 22 16 10  65 65\ 17 6 5 13 24  65 65\ 15 23 19 7 1  65 65\ 9 2 11 25 18  65 65\ 21 20 8 4 12  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1549/ Rw P1\ 3 14 21 17 10  65 65\ 16 7 5 13 24  65 65\ 15 23 19 6 2  65 65\ 9 1 12 25 18  65 65\ 22 20 8 4 11  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums
1585/ Rw P1\ 3 7 25 16 14  65 65\ 20 11 4 8 22  65 65\ 9 23 17 15 1  65 65\ 12 5 6 24 18  65 65\ 21 19 13 2 10  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1621/ Rw P1\ 3 7 24 16 15  65 65\ 19 11 5 8 22  65 65\ 10 23 17 14 1  65 65\ 12 4 6 25 18  65 65\ 21 20 13 2 9  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1657/ Rw P1\ 3 9 22 16 15  65 65\ 17 11 5 8 24  65 65\ 10 23 19 12 1  65 65\ 14 2 6 25 18  65 65\ 21 20 13 4 7  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1693/ Rw P1\ 3 9 21 17 15  65 65\ 16 12 5 8 24  65 65\ 10 23 19 11 2  65 65\ 14 1 7 25 18  65 65\ 22 20 13 4 6  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums
1729/ Rw P1\ 4 22 20 11 8  65 65\ 15 6 3 24 17  65 65\ 23 19 12 10 1  65 65\ 7 5 21 18 14  65 65\ 16 13 9 2 25  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1765/ Rw P1\ 4 22 18 11 10  65 65\ 13 6 5 24 17  65 65\ 25 19 12 8 1  65 65\ 7 3 21 20 14  65 65\ 16 15 9 2 23  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1801/ Rw P1\ 4 23 17 11 10  65 65\ 12 6 5 24 18  65 65\ 25 19 13 7 1  65 65\ 8 2 21 20 14  65 65\ 16 15 9 3 22  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1837/ Rw P1\ 4 23 16 12 10  65 65\ 11 7 5 24 18  65 65\ 25 19 13 6 2  65 65\ 8 1 22 20 14  65 65\ 17 15 9 3 21  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums
1873/ Rw P1\ 4 17 25 11 8  65 65\ 15 6 3 19 22  65 65\ 18 24 12 10 1  65 65\ 7 5 16 23 14  65 65\ 21 13 9 2 20  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1909/ Rw P1\ 4 17 23 11 10  65 65\ 13 6 5 19 22  65 65\ 20 24 12 8 1  65 65\ 7 3 16 25 14  65 65\ 21 15 9 2 18  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1945/ Rw P1\ 4 18 22 11 10  65 65\ 12 6 5 19 23  65 65\ 20 24 13 7 1  65 65\ 8 2 16 25 14  65 65\ 21 15 9 3 17  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	1981/ Rw P1\ 4 18 21 12 10  65 65\ 11 7 5 19 23  65 65\ 20 24 13 6 2  65 65\ 8 1 17 25 14  65 65\ 22 15 9 3 16  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums
2017/ Rw P1\ 4 12 25 16 8  65 65\ 20 6 3 14 22  65 65\ 13 24 17 10 1  65 65\ 7 5 11 23 19  65 65\ 21 18 9 2 15  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	2053/ Rw P1\ 4 12 23 16 10  65 65\ 18 6 5 14 22  65 65\ 15 24 17 8 1  65 65\ 7 3 11 25 19  65 65\ 21 20 9 2 13  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	2089/ Rw P1\ 4 13 22 16 10  65 65\ 17 6 5 14 23  65 65\ 15 24 18 7 1  65 65\ 8 2 11 25 19  65 65\ 21 20 9 3 12  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	2125/ Rw P1\ 4 13 21 17 10  65 65\ 16 7 5 14 23  65 65\ 15 24 18 6 2  65 65\ 8 1 12 25 19  65 65\ 22 20 9 3 11  65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums



65 65 65 65 65 Column /65/65/65/65/65 P2Sums	65 65 65 65 65 Column /65/65/65/65/65 P2Sums	65 65 65 65 65 Column /65/65/65/65/65 P2Sums	65 65 65 65 65 Column /65/65/65/65/65 P2Sums
2161/ Rw P1\ 4 7 25 16 13 65 65\ 20 11 3 9 22 65 65\ 8 24 17 15 1 65 65\ 12 5 6 23 19 65 65\ 21 18 14 2 10 65 65\ 2197/ Rw P1\ 4 7 23 16 15 65 65\ 18 11 5 9 22 65 65\ 10 24 17 13 1 65 65\ 12 3 6 25 19 65 65\ 21 20 14 2 8 65 65\ 2233/ Rw P1\ 4 8 22 16 15 65 65\ 17 11 5 9 23 65 65\ 10 24 18 12 1 65 65\ 13 2 6 25 19 65 65\ 21 20 14 3 7 65 65\ 2269/ Rw P1\ 4 8 21 17 15 65 65\ 16 12 5 9 23 65 65\ 10 24 18 11 2 65 65\ 13 1 7 25 19 65 65\ 22 20 14 3 6 65 65\ 2305/ Rw P1\ 5 22 19 11 8 65 65\ 14 6 3 25 17 65 65\ 23 20 12 9 1 65 65\ 7 4 21 18 15 65 65\ 16 13 10 2 24 65 65\ 2881/ Rw P1\ 6 23 20 2 14 65 65\ 5 12 9 21 18 65 65\ 24 16 3 15 7 65 65\ 13 10 22 19 1 65 65\ 17 4 11 8 25 65 65\ 3025/ Rw P1\ 7 23 20 1 14 65 65\ 5 11 9 22 18 65 65\ 24 17 3 15 6 65 65\ 13 10 21 19 2 65 65\ 16 4 12 8 25 65 65\ 3169/ Rw P1\ 8 22 20 1 14 65 65\ 5 11 9 23 17 65 65\ 24 18 2 15 6 65 65\ 12 10 21 19 3 65 65\ 16 4 13 7 25 65 65\ 3313/ Rw P1\ 9 22 20 1 13 65 65\ 5 11 8 24 17 65 65\ 23 19 2 15 6 65 65\ 12 10 21 18 4 65 65\ 16 3 14 7 25 65 65\ 3457/ Rw P1\ 10 22 19 1 13 65 65\ 4 11 8 25 17 65 65\ 23 20 2 14 6 65 65\ 12 9 21 18 5 65 65\ 16 3 15 7 24 65 65\ .....			
65 65 65 65 65 Column /65/65/65/65/65 P2Sums	65 65 65 65 65 Column /65/65/65/65/65 P2Sums	65 65 65 65 65 Column /65/65/65/65/65 P2Sums	65 65 65 65 65 Column /65/65/65/65/65 P2Sums
[Count = 3600]			

[原初五方陣と再構成した汎対角線型五方陣の対照リスト (部分) :

解番号 : 原初五方陣/P, 目的方陣/R]

1/P	1/R	25/P	2/R	49/P	38/R
1 2 3 4 5	1 23 20 12 9	1 2 4 3 5	1 24 20 12 8	1 2 5 3 4	1 25 19 12 8
6 7 8 9 10	15 7 4 21 18	6 7 9 8 10	15 7 3 21 19	6 7 10 8 9	14 7 3 21 20
11 12 13 14 15	24 16 13 10 2	11 12 14 13 15	23 16 14 10 2	11 12 15 13 14	23 16 15 9 2
16 17 18 19 20	8 5 22 19 11	16 17 19 18 20	9 5 22 18 11	16 17 20 18 19	10 4 22 18 11
21 22 23 24 25	17 14 6 3 25	21 22 24 23 25	17 13 6 4 25	21 22 25 23 24	17 13 6 5 24
73/P	5/R	97/P	6/R	121/P	42/R
1 3 2 4 5	1 22 20 13 9	1 3 4 2 5	1 24 20 13 7	1 3 5 2 4	1 25 19 13 7
6 8 7 9 10	15 8 4 21 17	6 8 9 7 10	15 8 2 21 19	6 8 10 7 9	14 8 2 21 20
11 13 12 14 15	24 16 12 10 3	11 13 14 12 15	22 16 14 10 3	11 13 15 12 14	22 16 15 9 3
16 18 17 19 20	7 5 23 19 11	16 18 19 17 20	9 5 23 17 11	16 18 20 17 19	10 4 23 17 11
21 23 22 24 25	18 14 6 2 25	21 23 24 22 25	18 12 6 4 25	21 23 25 22 24	18 12 6 5 24
145/P	9/R	169/P	10/R	193/P	78/R
1 4 2 3 5	1 22 20 14 8	1 4 3 2 5	1 23 20 14 7	1 4 5 2 3	1 25 18 14 7
6 9 7 8 10	15 9 3 21 17	6 9 8 7 10	15 9 2 21 18	6 9 10 7 8	13 9 2 21 20
11 14 12 13 15	23 16 12 10 4	11 14 13 12 15	22 16 13 10 4	11 14 15 12 13	22 16 15 8 4
16 19 17 18 20	7 5 24 18 11	16 19 18 17 20	8 5 24 17 11	16 19 20 17 18	10 3 24 17 11
21 24 22 23 25	19 13 6 2 25	21 24 23 22 25	19 12 6 3 25	21 24 25 22 23	19 12 6 5 23
217/P	45/R	241/P	46/R	265/P	82/R
1 5 2 3 4	1 22 19 15 8	1 5 3 2 4	1 23 19 15 7	1 5 4 2 3	1 24 18 15 7
6 10 7 8 9	14 10 3 21 17	6 10 8 7 9	14 10 2 21 18	6 10 9 7 8	13 10 2 21 19
11 15 12 13 14	23 16 12 9 5	11 15 13 12 14	22 16 13 9 5	11 15 14 12 13	22 16 14 8 5
16 20 17 18 19	7 4 25 18 11	16 20 18 17 19	8 4 25 17 11	16 20 19 17 18	9 3 25 17 11
21 25 22 23 24	20 13 6 2 24	21 25 23 22 24	20 12 6 3 24	21 25 24 22 23	20 12 6 4 23



289/P	15/R	313/P	159/R	337/P	25/R
1 11 16 6 21	1 20 24 13 7	1 11 21 6 16	1 25 19 13 7	1 16 6 11 21	1 10 24 18 12
2 12 17 7 22	23 12 6 5 19	2 12 22 7 17	18 12 6 5 24	2 17 7 12 22	23 17 11 5 9
3 13 18 8 23	10 4 18 22 11	3 13 23 8 18	10 4 23 17 11	3 18 8 13 23	15 4 8 22 16
4 14 19 9 24	17 21 15 9 3	4 14 24 9 19	22 16 15 9 3	4 19 9 14 24	7 21 20 14 3
5 15 20 10 25	14 8 2 16 25	5 15 25 10 20	14 8 2 21 20	5 20 10 15 25	19 13 2 6 25
361/P	27/R	385/P	303/R	433/P	169/R
1 16 11 6 21	1 15 24 18 7	1 16 21 6 11	1 25 14 18 7	1 21 6 11 16	1 10 19 23 12
2 17 12 7 22	23 17 6 5 14	2 17 22 7 12	13 17 6 5 24	2 22 7 12 17	18 22 11 5 9
3 18 13 8 23	10 4 13 22 16	3 18 23 8 13	10 4 23 12 16	3 23 8 13 18	15 4 8 17 21
4 19 14 9 24	12 21 20 9 3	4 19 24 9 14	22 11 20 9 3	4 24 9 14 19	7 16 25 14 3
5 20 15 10 25	19 8 2 11 25	5 20 25 10 15	19 8 2 21 15	5 25 10 15 20	24 13 2 6 20
481/P	171/R	529/P	315/R	577/P	577/R
1 21 11 6 16	1 15 19 23 7	1 21 16 6 11	1 20 14 23 7	2 1 3 4 5	2 23 20 11 9
2 22 12 7 17	18 22 6 5 14	2 22 17 7 12	13 22 6 5 19	7 6 8 9 10	15 6 4 22 18
3 23 13 8 18	10 4 13 17 21	3 23 18 8 13	10 4 18 12 21	12 11 13 14 15	24 17 13 10 1
4 24 14 9 19	12 16 25 9 3	4 24 19 9 14	17 11 25 9 3	17 16 18 19 20	8 5 21 19 12
5 25 15 10 20	24 8 2 11 20	5 25 20 10 15	24 8 2 16 15	22 21 23 24 25	16 14 7 3 25
601/P	578/R	625/P	614/R	649/P	581/R
2 1 4 3 5	2 24 20 11 8	2 1 5 3 4	2 25 19 11 8	2 3 1 4 5	2 21 20 13 9
7 6 9 8 10	15 6 3 22 19	7 6 10 8 9	14 6 3 22 20	7 8 6 9 10	15 8 4 22 16
12 11 14 13 15	23 17 14 10 1	12 11 15 13 14	23 17 15 9 1	12 13 11 14 15	24 17 11 10 3
17 16 19 18 20	9 5 21 18 12	17 16 20 18 19	10 4 21 18 12	17 18 16 19 20	6 5 23 19 12
22 21 24 23 25	16 13 7 4 25	22 21 25 23 24	16 13 7 5 24	22 23 21 24 25	18 14 7 1 25
721/P	585/R	793/P	621/R	865/P	591/R
2 4 1 3 5	2 21 20 14 8	2 5 1 3 4	2 21 19 15 8	2 12 17 7 22	2 20 24 13 6
7 9 6 8 10	15 9 3 22 16	7 10 6 8 9	14 10 3 22 16	1 11 16 6 21	23 11 7 5 19
12 14 11 13 15	23 17 11 10 4	12 15 11 13 14	23 17 11 9 5	3 13 18 8 23	10 4 18 21 12
17 19 16 18 20	6 5 24 18 12	17 20 16 18 19	6 4 25 18 12	4 14 19 9 24	16 22 15 9 3
22 24 21 23 25	19 13 7 1 25	22 25 21 23 24	20 13 7 1 24	5 15 20 10 25	14 8 1 17 25
913/P	601/R	1009/P	745/R	1153/P	1153/R
2 17 7 12 22	2 10 24 18 11	2 22 7 12 17	2 10 19 23 11	3 1 2 4 5	3 22 20 11 9
1 16 6 11 21	23 16 12 5 9	1 21 6 11 16	18 21 12 5 9	8 6 7 9 10	15 6 4 23 17
3 18 8 13 23	15 4 8 21 17	3 23 8 13 18	15 4 8 16 22	13 11 12 14 15	24 18 12 10 1
4 19 9 14 24	6 22 20 14 3	4 24 9 14 19	6 17 25 14 3	18 16 17 19 20	7 5 21 19 13
5 20 10 15 25	19 13 1 7 25	5 25 10 15 20	24 13 1 7 20	23 21 22 24 25	16 14 8 2 25
1225/P	1157/R	1297/P	1161/R	1369/P	1197/R
3 2 1 4 5	3 21 20 12 9	3 4 1 2 5	3 21 20 14 7	3 5 1 2 4	3 21 19 15 7
8 7 6 9 10	15 7 4 23 16	8 9 6 7 10	15 9 2 23 16	8 10 6 7 9	14 10 2 23 16
13 12 11 14 15	24 18 11 10 2	13 14 11 12 15	22 18 11 10 4	13 15 11 12 14	22 18 11 9 5
18 17 16 19 20	6 5 22 19 13	18 19 16 17 20	6 5 24 17 13	18 20 16 17 19	6 4 25 17 13
23 22 21 24 25	17 14 8 1 25	23 24 21 22 25	19 12 8 1 25	23 25 21 22 24	20 12 8 1 24
1441/P	1167/R	1489/P	1177/R	1585/P	1321/R
3 13 18 8 23	3 20 24 12 6	3 18 8 13 23	3 10 24 17 11	3 23 8 13 18	3 10 19 22 11
1 11 16 6 21	22 11 8 5 19	1 16 6 11 21	22 16 13 5 9	1 21 6 11 16	17 21 13 5 9
2 12 17 7 22	10 4 17 21 13	2 17 7 12 22	15 4 7 21 18	2 22 7 12 17	15 4 7 16 23
4 14 19 9 24	16 23 15 9 2	4 19 9 14 24	6 23 20 14 2	4 24 9 14 19	6 18 25 14 2
5 15 20 10 25	14 7 1 18 25	5 20 10 15 25	19 12 1 8 25	5 25 10 15 20	24 12 1 8 20
1729/P	1729/R	1801/P	1733/R	1873/P	1737/R
4 1 2 3 5	4 22 20 11 8	4 2 1 3 5	4 21 20 12 8	4 3 1 2 5	4 21 20 13 7
9 6 7 8 10	15 6 3 24 17	9 7 6 8 10	15 7 3 24 16	9 8 6 7 10	15 8 2 24 16
14 11 12 13 15	23 19 12 10 1	14 12 11 13 15	23 19 11 10 2	14 13 11 12 15	22 19 11 10 3
19 16 17 18 20	7 5 21 18 14	19 17 16 18 20	6 5 22 18 14	19 18 16 17 20	6 5 23 17 14
24 21 22 23 25	16 13 9 2 25	24 22 21 23 25	17 13 9 1 25	24 23 21 22 25	18 12 9 1 25
1945/P	1773/R	2017/P	1743/R	2065/P	1753/R
4 5 1 2 3	4 21 18 15 7	4 14 19 9 24	4 20 23 12 6	4 19 9 14 24	4 10 23 17 11
9 10 6 7 8	13 10 2 24 16	1 11 16 6 21	22 11 9 5 18	1 16 6 11 21	22 16 14 5 8
14 15 11 12 13	22 19 11 8 5	2 12 17 7 22	10 3 17 21 14	2 17 7 12 22	15 3 7 21 19
19 20 16 17 18	6 3 25 17 14	3 13 18 8 23	16 24 15 8 2	3 18 8 13 23	6 24 20 13 2
24 25 21 22 23	20 12 9 1 23	5 15 20 10 25	13 7 1 19 25	5 20 10 15 25	18 12 1 9 25

2161/P	1897/R	2305/P	2305/R	2377/P	2309/R
4 24 9 14 19	4 10 18 22 11	5 1 2 3 4	5 22 19 11 8	5 2 1 3 4	5 21 19 12 8
1 21 6 11 16	17 21 14 5 8	10 6 7 8 9	14 6 3 25 17	10 7 6 8 9	14 7 3 25 16
2 22 7 12 17	15 3 7 16 24	15 11 12 13 14	23 20 12 9 1	15 12 11 13 14	23 20 11 9 2
3 23 8 13 18	6 19 25 13 2	20 16 17 18 19	7 4 21 18 15	20 17 16 18 19	6 4 22 18 15
5 25 10 15 20	23 12 1 9 20	25 21 22 23 24	16 13 10 2 24	25 22 21 23 24	17 13 10 1 24
2449/P	2313/R	2521/P	2349/R	2593/P	2319/R
5 3 1 2 4	5 21 19 13 7	5 4 1 2 3	5 21 18 14 7	5 15 20 10 25	5 19 23 12 6
10 8 6 7 9	14 8 2 25 16	10 9 6 7 8	13 9 2 25 16	1 11 16 6 21	22 11 10 4 18
15 13 11 12 14	22 20 11 9 3	15 14 11 12 13	22 20 11 8 4	2 12 17 7 22	9 3 17 21 15
20 18 16 17 19	6 4 23 17 15	20 19 16 17 18	6 3 24 17 15	3 13 18 8 23	16 25 14 8 2
25 23 21 22 24	18 12 10 1 24	25 24 21 22 23	19 12 10 1 23	4 14 19 9 24	13 7 1 20 24
2641/P	2329/R	2737/P	2473/R	2881/P	2881/R
5 20 10 15 25	5 9 23 17 11	5 25 10 15 20	5 9 18 22 11	6 7 8 9 10	6 23 20 2 14
1 16 6 11 21	22 16 15 4 8	1 21 6 11 16	17 21 15 4 8	11 12 13 14 15	5 12 9 21 18
2 17 7 12 22	14 3 7 21 20	2 22 7 12 17	14 3 7 16 25	1 2 3 4 5	24 16 3 15 7
3 18 8 13 23	6 25 19 13 2	3 23 8 13 18	6 20 24 13 2	16 17 18 19 20	13 10 22 19 1
4 19 9 14 24	18 12 1 10 24	4 24 9 14 19	23 12 1 10 19	21 22 23 24 25	17 4 11 8 25
2899/P	2883/R	2917/P	2885/R	2935/P	2897/R
6 8 7 9 10	6 22 20 3 14	6 9 7 8 10	6 22 20 4 13	6 10 7 8 9	6 22 19 5 13
11 13 12 14 15	5 13 9 21 17	11 14 12 13 15	5 14 8 21 17	11 15 12 13 14	4 15 8 21 17
1 3 2 4 5	24 16 2 15 8	1 4 2 3 5	23 16 2 15 9	1 5 2 3 4	23 16 2 14 10
16 18 17 19 20	12 10 23 19 1	16 19 17 18 20	12 10 24 18 1	16 20 17 18 19	12 9 25 18 1
21 23 22 24 25	18 4 11 7 25	21 24 22 23 25	19 3 11 7 25	21 25 22 23 24	20 3 11 7 24
2953/P	2887/R	2977/P	2935/R	3025/P	3025/R
6 16 1 11 21	6 5 24 18 12	6 21 1 11 16	6 5 19 23 12	7 6 8 9 10	7 23 20 1 14
7 17 2 12 22	23 17 11 10 4	7 22 2 12 17	18 22 11 10 4	12 11 13 14 15	5 11 9 22 18
8 18 3 13 23	15 9 3 22 16	8 23 3 13 18	15 9 3 17 21	2 1 3 4 5	24 17 3 15 6
9 19 4 14 24	2 21 20 14 8	9 24 4 14 19	2 16 25 14 8	17 16 18 19 20	13 10 21 19 2
10 20 5 15 25	19 13 7 1 25	10 25 5 15 20	24 13 7 1 20	22 21 23 24 25	16 4 12 8 25
3043/P	3027/R	3061/P	3029/R	3079/P	3041/R
7 8 6 9 10	7 21 20 3 14	7 9 6 8 10	7 21 20 4 13	7 10 6 8 9	7 21 19 5 13
12 13 11 14 15	5 13 9 22 16	12 14 11 13 15	5 14 8 22 16	12 15 11 13 14	4 15 8 22 16
2 3 1 4 5	24 17 1 15 8	2 4 1 3 5	23 17 1 15 9	2 5 1 3 4	23 17 1 14 10
17 18 16 19 20	11 10 23 19 2	17 19 16 18 20	11 10 24 18 2	17 20 16 18 19	11 9 25 18 2
22 23 21 24 25	18 4 12 6 25	22 24 21 23 25	19 3 12 6 25	22 25 21 23 24	20 3 12 6 24
3097/P	3031/R	3121/P	3079/R	3169/P	3169/R
7 17 2 12 22	7 5 24 18 11	7 22 2 12 17	7 5 19 23 11	8 6 7 9 10	8 22 20 1 14
6 16 1 11 21	23 16 12 10 4	6 21 1 11 16	18 21 12 10 4	13 11 12 14 15	5 11 9 23 17
8 18 3 13 23	15 9 3 21 17	8 23 3 13 18	15 9 3 16 22	3 1 2 4 5	24 18 2 15 6
9 19 4 14 24	1 22 20 14 8	9 24 4 14 19	1 17 25 14 8	18 16 17 19 20	12 10 21 19 3
10 20 5 15 25	19 13 6 2 25	10 25 5 15 20	24 13 6 2 20	23 21 22 24 25	16 4 13 7 25
3187/P	3171/R	3205/P	3173/R	3223/P	3185/R
8 7 6 9 10	8 21 20 2 14	8 9 6 7 10	8 21 20 4 12	8 10 6 7 9	8 21 19 5 12
13 12 11 14 15	5 12 9 23 16	13 14 11 12 15	5 14 7 23 16	13 15 11 12 14	4 15 7 23 16
3 2 1 4 5	24 18 1 15 7	3 4 1 2 5	22 18 1 15 9	3 5 1 2 4	22 18 1 14 10
18 17 16 19 20	11 10 22 19 3	18 19 16 17 20	11 10 24 17 3	18 20 16 17 19	11 9 25 17 3
23 22 21 24 25	17 4 13 6 25	23 24 21 22 25	19 2 13 6 25	23 25 21 22 24	20 2 13 6 24
3241/P	3175/R	3265/P	3223/R	3313/P	3313/R
8 18 3 13 23	8 5 24 17 11	8 23 3 13 18	8 5 19 22 11	9 6 7 8 10	9 22 20 1 13
6 16 1 11 21	22 16 13 10 4	6 21 1 11 16	17 21 13 10 4	14 11 12 13 15	5 11 8 24 17
7 17 2 12 22	15 9 2 21 18	7 22 2 12 17	15 9 2 16 23	4 1 2 3 5	23 19 2 15 6
9 19 4 14 24	1 23 20 14 7	9 24 4 14 19	1 18 25 14 7	19 16 17 18 20	12 10 21 18 4
10 20 5 15 25	19 12 6 3 25	10 25 5 15 20	24 12 6 3 20	24 21 22 23 25	16 3 14 7 25
3331/P	3315/R	3349/P	3317/R	3367/P	3329/R
9 7 6 8 10	9 21 20 2 13	9 8 6 7 10	9 21 20 3 12	9 10 6 7 8	9 21 18 5 12
14 12 11 13 15	5 12 8 24 16	14 13 11 12 15	5 13 7 24 16	14 15 11 12 13	3 15 7 24 16
4 2 1 3 5	23 19 1 15 7	4 3 1 2 5	22 19 1 15 8	4 5 1 2 3	22 19 1 13 10
19 17 16 18 20	11 10 22 18 4	19 18 16 17 20	11 10 23 17 4	19 20 16 17 18	11 8 25 17 4
24 22 21 23 25	17 3 14 6 25	24 23 21 22 25	18 2 14 6 25	24 25 21 22 23	20 2 14 6 23



く、原初五方陣の解集合との間に「一対一対応の関係」があることをも明示している。

なお、追加試験として、ひな形を変えて新たにひな形変換法を定義し、そのつど原初方陣を作り直して調べたところ、どのひな形も、目的の汎五方陣の解 3600 個を再構成することができた。「最良のひな形」が唯一無二でないことがわかる。

また、原初方陣はそのままに、ひな形のみを変えて調べたところ、再構成できない解がけっこう残った。「標準解出力のための不等式」が、合わない解を選び出すためだ。

そこで、原初方陣も目的方陣も、不等式無しの「原始解」のレベルで比較してみたところ、何と目的方陣の全数を再構成することができた。この場合ひな形は別個でも、原初方陣は同じであるから、原初方陣の解集合は一つあれば十分だということになる。

ひな形を変えても、原初方陣解の出力順が変わるだけで、内容自体は変わらないのだ。

つまり、汎五方陣では、原初五方陣が個別のひな形に必ずしも依存せず、独立してただ一種類存在していることを示している。いったいこれは何を意味するのだろうか？

#### # 4. 原初五方陣とひな形変換法によって連立型の対称汎五方陣を作る

〔連立型対称汎五方陣の標準解のリスト (全) :

チェック・サムつき: |Rw 行和, |Column 列和, |P1\|P2Sums 汎対角線和]

1/ Rw P1\ 1 23 20 12 9 65 65\ 15 7 4 21 18 65 65\ 24 16 13 10 2 65 65\ 8 5 22 19 11 65 65\ 17 14 6 3 25 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		2/ Rw P1\ 1 23 20 14 7 65 65\ 15 9 2 21 18 65 65\ 22 16 13 10 4 65 65\ 8 5 24 17 11 65 65\ 19 12 6 3 25 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		3/ Rw P1\ 1 15 24 18 7 65 65\ 23 17 6 5 14 65 65\ 10 4 13 22 16 65 65\ 12 21 20 9 3 65 65\ 19 8 2 11 25 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		4/ Rw P1\ 1 15 22 18 9 65 65\ 23 19 6 5 12 65 65\ 10 2 13 24 16 65 65\ 14 21 20 7 3 65 65\ 17 8 4 11 25 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	
5/ Rw P1\ 2 23 19 11 10 65 65\ 14 6 5 22 18 65 65\ 25 17 13 9 1 65 65\ 8 4 21 20 12 65 65\ 16 15 7 3 24 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		6/ Rw P1\ 2 23 19 15 6 65 65\ 14 10 1 22 18 65 65\ 21 17 13 9 5 65 65\ 8 4 25 16 12 65 65\ 20 11 7 3 24 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		7/ Rw P1\ 2 14 25 18 6 65 65\ 23 16 7 4 15 65 65\ 9 5 13 21 17 65 65\ 11 22 19 10 3 65 65\ 20 8 1 12 24 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		8/ Rw P1\ 2 14 21 18 10 65 65\ 23 20 7 4 11 65 65\ 9 1 13 25 17 65 65\ 15 22 19 6 3 65 65\ 16 8 5 12 24 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	
9/ Rw P1\ 4 23 17 11 10 65 65\ 12 6 5 24 18 65 65\ 25 19 13 7 1 65 65\ 8 2 21 20 14 65 65\ 16 15 9 3 22 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		10/ Rw P1\ 4 23 17 15 6 65 65\ 12 10 1 24 18 65 65\ 21 19 13 7 5 65 65\ 8 2 25 16 14 65 65\ 20 11 9 3 22 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		11/ Rw P1\ 4 12 25 18 6 65 65\ 23 16 9 2 15 65 65\ 7 5 13 21 19 65 65\ 11 24 17 10 3 65 65\ 20 8 1 14 22 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		12/ Rw P1\ 4 12 21 18 10 65 65\ 23 20 9 2 11 65 65\ 7 1 13 25 19 65 65\ 15 24 17 6 3 65 65\ 16 8 5 14 22 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	
13/ Rw P1\ 5 23 16 12 9 65 65\ 11 7 4 25 18 65 65\ 24 20 13 6 2 65 65\ 8 1 22 19 15 65 65\ 17 14 10 3 21 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		14/ Rw P1\ 5 23 16 14 7 65 65\ 11 9 2 25 18 65 65\ 22 20 13 6 4 65 65\ 8 1 24 17 15 65 65\ 19 12 10 3 21 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		15/ Rw P1\ 5 11 24 18 7 65 65\ 23 17 10 1 14 65 65\ 6 4 13 22 20 65 65\ 12 25 16 9 3 65 65\ 19 8 2 15 21 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums		16/ Rw P1\ 5 11 22 18 9 65 65\ 23 19 10 1 12 65 65\ 6 2 13 24 20 65 65\ 14 25 16 7 3 65 65\ 17 8 4 15 21 65 65\  65 65 65 65 65 Column /65/65/65/65/65 P2Sums	

[Count = 16]

次の研究対象は、連立型の補数対称型かつ汎対角線型魔方陣である。  
 はたして原初五方陣とひな形変換法によって、これを再構成できるものだろうか。  
 先ず目的とする対称汎五方陣の解全 16 個を上のリストに掲げておこう。かつて鈴木睦教授が全数を発見・確定し、自ページに発表した例の「珠玉の」16 個である。  
 ちなみに私のこのリストは、 $n1 < n5$  and  $n1 < n5 < n21$  の条件下で整理してある。

今度は、この 16 個の中から  $n1 = 1$  の解 4 個を選んでひな形とし、並行試験する。原初方陣は変えずにひな形とひな形変換法のみを変えて果たしてうまく行くか、実験してみよう。  
 共通の原初方陣は、次の条件下で構成する。

/原初五方陣					/ひな形													
24	25	21	22	23	24	25	21	22	23	3	25	17	14	6	3	25	17	14
4	5	1	2	3	4	5	1	2	3	12	9	1	23	20	12	9	1	23
9	10	6	7	8	9	10	6	7	8	21	18	15	7	4	21	18	15	7
14	15	11	12	13	14	15	11	12	13	10	2	24	16	13	10	2	24	16
19	20	16	17	18	19	20	16	17	18	19	11	8	5	22	19	11	8	5
24	25	21	22	23	24	25	21	22	23	3	25	17	14	6	3	25	17	14
4	5	1	2	3	4	5	1	2	3	12	9	1	23	20	12	9	1	23

**\* 原初五方陣構成のための基本条件式: C=65 \***

$n1+n23+n20+n12+n9=C \quad \dots (1)$	$n1+n15+n24+n8+n17=C \quad \dots (6)$
$n15+n7+n4+n21+n18=C \quad \dots (2)$	$n23+n7+n16+n5+n14=C \quad \dots (7)$
$n24+n16+n13+n10+n2=C \quad \dots (3)$	$n20+n4+n13+n22+n6=C \quad \dots (8)$
$n8+n5+n22+n19+n11=C \quad \dots (4)$	$n12+n21+n10+n19+n3=C \quad \dots (9)$
$n17+n14+n6+n3+n25=C \quad \dots (5)$	$n9+n18+n2+n11+n25=C \quad \dots (10)$

**\* 補数ペアの対称配置: CC=26 \***

$$n1+n25=n23+n3=n20+n6=n12+n14=n9+n17=n15+n11=n7+n19 \\ =n4+n22=n21+n5=n18+n8=n24+n2=n16+n10=n13+n13=CC \quad \dots (cp)$$

**\* 汎対角線定和の条件式: C=65 \***

$n1+n7+n13+n19+n25=C \quad \dots (11)$	$n1+n18+n10+n22+n14=C \quad \dots (12)$
$n23+n4+n10+n11+n17=C \quad \dots (13)$	$n23+n15+n2+n19+n6=C \quad \dots (14)$
$n20+n21+n2+n8+n14=C \quad \dots (15)$	$n20+n7+n24+n11+n3=C \quad \dots (16)$
$n12+n18+n24+n5+n6=C \quad \dots (17)$	$n12+n4+n16+n8+n25=C \quad \dots (18)$
$n9+n15+n16+n22+n3=C \quad \dots (19)$	$n9+n21+n13+n5+n17=C \quad \dots (20)$

**\* リスト出力用の不等式:** ひな形に敬意を表して  $n1 < n25$  and  $n1 < n9 < n17$  としよう。

/\* 原初五方陣を構成するためのプログラムの要点: Dictated by Kanji Setsuda \*/

```

/* Define Level 1: */
/* Set n1 & n25 & n1<n25 */
void stp01(){
  short a,b;
  for(a=1;a<13;a++){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[1]=a; nm[25]=b;
      uflg[a]=1; uflg[b]=1;
      stp02();
      uflg[b]=0; uflg[a]=0;}
  }
}

/* Set n24 & n2 */
void stp02(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[24]=a; nm[2]=b;
      uflg[a]=1; uflg[b]=1;
      stp03();
      uflg[b]=0; uflg[a]=0;}
  }
}

/* Set n8 & n18 */

```

```

void stp03(){
  short a,b;
  for(a=1;a<26;a++){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[8]=a; nm[18]=b;
      uflg[a]=1; uflg[b]=1;
      stp04();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n17 & n9 & n1<n9<n17 */
void stp04(){
  short a,b;
  for(a=25;a>13;a--){b=PSM-a;
    if(b>nm[1]){
      if((uflg[a]==0)&&(uflg[b]==0)){
        nm[17]=a; nm[9]=b;
        uflg[a]=1; uflg[b]=1;
        stp05();
        uflg[b]=0; uflg[a]=0;}}
  }
}
/* Set n15=65-n1-n24-n8-n17 & n11 */
void stp05(){
  short a,b;
  a=LSM-nm[1]-nm[24]-nm[8]-nm[17];
  b=LSM-nm[25]-nm[2]-nm[18]-nm[9];
  → if((0<a)&&(a<26)&&(a+b==PSM)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[15]=a; nm[11]=b;
      uflg[a]=1; uflg[b]=1;
      stp06();
      uflg[b]=0; uflg[a]=0;}}
}
/**/
/* Level 2 */
/* Set n7=65-n20-n24-n11-n3 & n19 */
void stp09(){
  short a,b;
  a=LSM-nm[20]-nm[24]-nm[11]-nm[3];
  b=LSM-nm[6]-nm[2]-nm[15]-nm[23];
  if((0<a)&&(a<26)&&(a+b==PSM)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[7]=a; nm[19]=b;
      uflg[a]=1; uflg[b]=1;
      if(nm[1]+nm[7]+nm[13]+nm[19]+nm[25]==LSM){stp10();}
      uflg[b]=0; uflg[a]=0;}}
}
/* Set n21=65-n20-n2-n8-n14 & n */
void stp10(){
  short a,b;
  a=LSM-nm[20]-nm[2]-nm[8]-nm[14];
  b=LSM-nm[12]-nm[18]-nm[24]-nm[6];
  if((0<a)&&(a<26)&&(a+b==PSM)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[21]=a; nm[5]=b;
      uflg[a]=1; uflg[b]=1;
      if(nm[9]+nm[21]+nm[13]+nm[5]+nm[17]==LSM){stp11();}
      uflg[b]=0; uflg[a]=0;}}
}

```

```

→/* Set n23 & n3 */
void stp06(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[23]=a; nm[3]=b;
      uflg[a]=1; uflg[b]=1;
      stp07();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n20 & n6 */
void stp07(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[20]=a; nm[6]=b;
      uflg[a]=1; uflg[b]=1;
      stp08();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n12=65-n1-n23-n20-n9 & n14 */
void stp08(){
  short a,b;
  a=LSM-nm[1]-nm[23]-nm[20]-nm[9];
  b=LSM-nm[25]-nm[3]-nm[6]-nm[17];
  if((0<a)&&(a<26)&&(a+b==PSM)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[12]=a; nm[14]=b;
      uflg[a]=1; uflg[b]=1;
      stp09();
      uflg[b]=0; uflg[a]=0;}}
}

```

```

}
/* Set n4=65-n15-n7-n21-n18 & n22 */
void stp11(){
short a,b;
a=LSM-nm[15]-nm[7]-nm[21]-nm[18];
b=LSM-nm[11]-nm[19]-nm[5]-nm[8];
if((0<a)&&(a<26)&&(a+b==PSM)){
if((uflg[a]==0)&&(uflg[b]==0)){
nm[4]=a; nm[22]=b;
uflg[a]=1; uflg[b]=1;
if(nm[20]+nm[4]+nm[13]+nm[22]+nm[6]==LSM){stp12();}
uflg[b]=0; uflg[a]=0;}}
}
/* Set n16=65-n23-n7-n5-n14 & n10 */
void stp12(){
short a,b;
a=LSM-nm[23]-nm[7]-nm[5]-nm[14];
b=LSM-nm[3]-nm[19]-nm[21]-nm[12];
if((0<a)&&(a<26)&&(a+b==PSM)){
if((uflg[a]==0)&&(uflg[b]==0)){
nm[16]=a; nm[10]=b;
uflg[a]=1; uflg[b]=1;
if(nm[24]+nm[16]+nm[13]+nm[10]+nm[2]==LSM){stp13();}
uflg[b]=0; uflg[a]=0;}}
}
/**/
/* Checks Sums of Some 4 Entries */
void stp13(void){
short sm1, sm2, sm3, sm4;
sm1=nm[1]+nm[18]+nm[10]+nm[22]+nm[14];
sm2=nm[25]+nm[8]+nm[16]+nm[4]+nm[12];
sm3=nm[9]+nm[15]+nm[16]+nm[22]+nm[3];
sm4=nm[17]+nm[11]+nm[10]+nm[4]+nm[23];
if((sm1==LSM)&&(sm2==LSM)&&(sm3==LSM)&&(sm4==LSM)){protorecord();}
}
/**/
/* Record the Answers */
void protorecord(){
short n;
cnt1++;
for(n=1;n<26;n++){pn[cnt1][n]=nm[n];}
}
/**/

```

4個のひな形とひな形変換法は、次のように記述する。

```

1/M | void damt1(void) {
1 23 20 12 9 | d[1]=p[1]; d[2]=p[23]; d[3]=p[20]; d[4]=p[12]; d[5]=p[9];
15 7 4 21 18 | d[6]=p[15]; d[7]=p[7]; d[8]=p[4]; d[9]=p[21]; d[10]=p[18];
24 16 13 10 2 | d[11]=p[24]; d[12]=p[16]; d[13]=p[13]; d[14]=p[10]; d[15]=p[2];
8 5 22 19 11 | d[16]=p[8]; d[17]=p[5]; d[18]=p[22]; d[19]=p[19]; d[20]=p[11];
17 14 6 3 25 | d[21]=p[17]; d[22]=p[14]; d[23]=p[6]; d[24]=p[3]; d[25]=p[25];
/**/
2/M | void damt2(void) {
1 23 20 14 7 | d[1]=p[1]; d[2]=p[23]; d[3]=p[20]; d[4]=p[14]; d[5]=p[7];
15 9 2 21 18 | d[6]=p[15]; d[7]=p[9]; d[8]=p[2]; d[9]=p[21]; d[10]=p[18];
22 16 13 10 4 | d[11]=p[22]; d[12]=p[16]; d[13]=p[13]; d[14]=p[10]; d[15]=p[4];
8 5 24 17 11 | d[16]=p[8]; d[17]=p[5]; d[18]=p[24]; d[19]=p[17]; d[20]=p[11];
19 12 6 3 25 | d[21]=p[19]; d[22]=p[12]; d[23]=p[6]; d[24]=p[3]; d[25]=p[25];
/**/

```



```

3/M | void damt3(void) {
1 15 24 18 7 | d[ 1]=p[ 1]; d[ 2]=p[15]; d[ 3]=p[24]; d[ 4]=p[18]; d[ 5]=p[ 7];
23 17 6 5 14 | d[ 6]=p[23]; d[ 7]=p[17]; d[ 8]=p[ 6]; d[ 9]=p[ 5]; d[10]=p[14];
10 4 13 22 16 | d[11]=p[10]; d[12]=p[ 4]; d[13]=p[13]; d[14]=p[22]; d[15]=p[16];
12 21 20 9 3 | d[16]=p[12]; d[17]=p[21]; d[18]=p[20]; d[19]=p[ 9]; d[20]=p[ 3];
19 8 2 11 25 | d[21]=p[19]; d[22]=p[ 8]; d[23]=p[ 2]; d[24]=p[11]; d[25]=p[25];
/**/
4/M | void damt4(void) {
1 15 22 18 9 | d[ 1]=p[ 1]; d[ 2]=p[15]; d[ 3]=p[22]; d[ 4]=p[18]; d[ 5]=p[ 9];
23 19 6 5 12 | d[ 6]=p[23]; d[ 7]=p[19]; d[ 8]=p[ 6]; d[ 9]=p[ 5]; d[10]=p[12];
10 2 13 24 16 | d[11]=p[10]; d[12]=p[ 2]; d[13]=p[13]; d[14]=p[24]; d[15]=p[16];
14 21 20 7 3 | d[16]=p[14]; d[17]=p[21]; d[18]=p[20]; d[19]=p[ 7]; d[20]=p[ 3];
17 8 4 11 25 | d[21]=p[17]; d[22]=p[ 8]; d[23]=p[ 4]; d[24]=p[11]; d[25]=p[25];

```

次のリストが、その並行試験の実行結果だ。途中を省略せずに全数個を掲げる。  
 左端の縦1列に共通の原初五方陣。右側に順に、変えたひな形別に再構成結果をリスト出力した。第1番解の横一列には、ひな形そのものが並んでいる (/R1, /R2, /R3, /R4)。

[原初五方陣と再構成した連立型対称汎五方陣の対照リスト： 解番号： 原初/P, 再構成解/R]

1/P	1/R1	2/R2	3/R3	4/R4
1 2 3 4 5	1 23 20 12 9	1 23 20 14 7	1 15 24 18 7	1 15 22 18 9
6 7 8 9 10	15 7 4 21 18	15 9 2 21 18	23 17 6 5 14	23 19 6 5 12
11 12 13 14 15	24 16 13 10 2	22 16 13 10 4	10 4 13 22 16	10 2 13 24 16
16 17 18 19 20	8 5 22 19 11	8 5 24 17 11	12 21 20 9 3	14 21 20 7 3
21 22 23 24 25	17 14 6 3 25	19 12 6 3 25	19 8 2 11 25	17 8 4 11 25
2/P	2/R1	1/R2	4/R3	3/R4
1 4 3 2 5	1 23 20 14 7	1 23 20 12 9	1 15 22 18 9	1 15 24 18 7
6 9 8 7 10	15 9 2 21 18	15 7 4 21 18	23 19 6 5 12	23 17 6 5 14
11 14 13 12 15	22 16 13 10 4	24 16 13 10 2	10 2 13 24 16	10 4 13 22 16
16 19 18 17 20	8 5 24 17 11	8 5 22 19 11	14 21 20 7 3	12 21 20 9 3
21 24 23 22 25	19 12 6 3 25	17 14 6 3 25	17 8 4 11 25	19 8 2 11 25
3/P	3/R1	0/R2	0/R3	2/R4
1 16 11 6 21	1 15 24 18 7	1 15 24 8 17	1 23 10 14 17	1 23 20 14 7
2 17 12 7 22	23 17 6 5 14	23 7 16 5 14	15 19 2 21 8	15 9 2 21 18
3 18 13 8 23	10 4 13 22 16	20 4 13 22 6	22 6 13 20 4	22 16 13 10 4
4 19 14 9 24	12 21 20 9 3	12 21 10 19 3	18 5 24 7 11	8 5 24 17 11
5 20 15 10 25	19 8 2 11 25	9 18 2 11 25	9 12 16 3 25	19 12 6 3 25
4/P	4/R1	0/R2	0/R3	1/R4
1 16 11 6 21	1 15 22 18 9	1 15 22 8 19	1 23 10 12 19	1 23 20 12 9
4 19 14 9 24	23 19 6 5 12	23 9 16 5 12	15 17 4 21 8	15 7 4 21 18
3 18 13 8 23	10 2 13 24 16	20 2 13 24 6	24 6 13 20 2	24 16 13 10 2
2 17 12 7 22	14 21 20 7 3	14 21 10 17 3	18 5 22 9 11	8 5 22 19 11
5 20 15 10 25	17 8 4 11 25	7 18 4 11 25	7 14 16 3 25	17 14 6 3 25
5/P	5/R1	6/R2	7/R3	8/R4
2 1 3 5 4	2 23 19 11 10	2 23 19 15 6	2 14 25 18 6	2 14 21 18 10
7 6 8 10 9	14 6 5 22 18	14 10 1 22 18	23 16 7 4 15	23 20 7 4 11
12 11 13 15 14	25 17 13 9 1	21 17 13 9 5	9 5 13 21 17	9 1 13 25 17
17 16 18 20 19	8 4 21 20 12	8 4 25 16 12	11 22 19 10 3	15 22 19 6 3
22 21 23 25 24	16 15 7 3 24	20 11 7 3 24	20 8 1 12 24	16 8 5 12 24
6/P	6/R1	5/R2	8/R3	7/R4
2 5 3 1 4	2 23 19 15 6	2 23 19 11 10	2 14 21 18 10	2 14 25 18 6
7 10 8 6 9	14 10 1 22 18	14 6 5 22 18	23 20 7 4 11	23 16 7 4 15
12 15 13 11 14	21 17 13 9 5	25 17 13 9 1	9 1 13 25 17	9 5 13 21 17
17 20 18 16 19	8 4 25 16 12	8 4 21 20 12	15 22 19 6 3	11 22 19 10 3
22 25 23 21 24	20 11 7 3 24	16 15 7 3 24	16 8 5 12 24	20 8 1 12 24

7/P	7/R1	0/R2	0/R3	6/R4
2 17 12 7 22	2 14 25 18 6	2 14 25 8 16	2 23 9 15 16	2 23 19 15 6
1 16 11 6 21	23 16 7 4 15	23 6 17 4 15	14 20 1 22 8	14 10 1 22 18
3 18 13 8 23	9 5 13 21 17	19 5 13 21 7	21 7 13 19 5	21 17 13 9 5
5 20 15 10 25	11 22 19 10 3	11 22 9 20 3	18 4 25 6 12	8 4 25 16 12
4 19 14 9 24	20 8 1 12 24	10 18 1 12 24	10 11 17 3 24	20 11 7 3 24
8/P	8/R1	0/R2	0/R3	5/R4
2 17 12 7 22	2 14 21 18 10	2 14 21 8 20	2 23 9 11 20	2 23 19 11 10
5 20 15 10 25	23 20 7 4 11	23 10 17 4 11	14 16 5 22 8	14 6 5 22 18
3 18 13 8 23	9 1 13 25 17	19 1 13 25 7	25 7 13 19 1	25 17 13 9 1
1 16 11 6 21	15 22 19 6 3	15 22 9 16 3	18 4 21 10 12	8 4 21 20 12
4 19 14 9 24	16 8 5 12 24	6 18 5 12 24	6 15 17 3 24	16 15 7 3 24
9/P	9/R1	10/R2	11/R3	12/R4
4 1 3 5 2	4 23 17 11 10	4 23 17 15 6	4 12 25 18 6	4 12 21 18 10
9 6 8 10 7	12 6 5 24 18	12 10 1 24 18	23 16 9 2 15	23 20 9 2 11
14 11 13 15 12	25 19 13 7 1	21 19 13 7 5	7 5 13 21 19	7 1 13 25 19
19 16 18 20 17	8 2 21 20 14	8 2 25 16 14	11 24 17 10 3	15 24 17 6 3
24 21 23 25 22	16 15 9 3 22	20 11 9 3 22	20 8 1 14 22	16 8 5 14 22
10/P	10/R1	9/R2	12/R3	11/R4
4 5 3 1 2	4 23 17 15 6	4 23 17 11 10	4 12 21 18 10	4 12 25 18 6
9 10 8 6 7	12 10 1 24 18	12 6 5 24 18	23 20 9 2 11	23 16 9 2 15
14 15 13 11 12	21 19 13 7 5	25 19 13 7 1	7 1 13 25 19	7 5 13 21 19
19 20 18 16 17	8 2 25 16 14	8 2 21 20 14	15 24 17 6 3	11 24 17 10 3
24 25 23 21 22	20 11 9 3 22	16 15 9 3 22	16 8 5 14 22	20 8 1 14 22
11/P	11/R1	0/R2	0/R3	10/R4
4 19 14 9 24	4 12 25 18 6	4 12 25 8 16	4 23 7 15 16	4 23 17 15 6
1 16 11 6 21	23 16 9 2 15	23 6 19 2 15	12 20 1 24 8	12 10 1 24 18
3 18 13 8 23	7 5 13 21 19	17 5 13 21 9	21 9 13 17 5	21 19 13 7 5
5 20 15 10 25	11 24 17 10 3	11 24 7 20 3	18 2 25 6 14	8 2 25 16 14
2 17 12 7 22	20 8 1 14 22	10 18 1 14 22	10 11 19 3 22	20 11 9 3 22
12/P	12/R1	0/R2	0/R3	9/R4
4 19 14 9 24	4 12 21 18 10	4 12 21 8 20	4 23 7 11 20	4 23 17 11 10
5 20 15 10 25	23 20 9 2 11	23 10 19 2 11	12 16 5 24 8	12 6 5 24 18
3 18 13 8 23	7 1 13 25 19	17 1 13 25 9	25 9 13 17 1	25 19 13 7 1
1 16 11 6 21	15 24 17 6 3	15 24 7 16 3	18 2 21 10 14	8 2 21 20 14
2 17 12 7 22	16 8 5 14 22	6 18 5 14 22	6 15 19 3 22	16 15 9 3 22
13/P	13/R1	14/R2	15/R3	16/R4
5 2 3 4 1	5 23 16 12 9	5 23 16 14 7	5 11 24 18 7	5 11 22 18 9
10 7 8 9 6	11 7 4 25 18	11 9 2 25 18	23 17 10 1 14	23 19 10 1 12
15 12 13 14 11	24 20 13 6 2	22 20 13 6 4	6 4 13 22 20	6 2 13 24 20
20 17 18 19 16	8 1 22 19 15	8 1 24 17 15	12 25 16 9 3	14 25 16 7 3
25 22 23 24 21	17 14 10 3 21	19 12 10 3 21	19 8 2 15 21	17 8 4 15 21
14/P	14/R1	13/R2	16/R3	15/R4
5 4 3 2 1	5 23 16 14 7	5 23 16 12 9	5 11 22 18 9	5 11 24 18 7
10 9 8 7 6	11 9 2 25 18	11 7 4 25 18	23 19 10 1 12	23 17 10 1 14
15 14 13 12 11	22 20 13 6 4	24 20 13 6 2	6 2 13 24 20	6 4 13 22 20
20 19 18 17 16	8 1 24 17 15	8 1 22 19 15	14 25 16 7 3	12 25 16 9 3
25 24 23 22 21	19 12 10 3 21	17 14 10 3 21	17 8 4 15 21	19 8 2 15 21
15/P	15/R1	0/R2	0/R3	14/R4
5 20 15 10 25	5 11 24 18 7	5 11 24 8 17	5 23 6 14 17	5 23 16 14 7
2 17 12 7 22	23 17 10 1 14	23 7 20 1 14	11 19 2 25 8	11 9 2 25 18
3 18 13 8 23	6 4 13 22 20	16 4 13 22 10	22 10 13 16 4	22 20 13 6 4
4 19 14 9 24	12 25 16 9 3	12 25 6 19 3	18 1 24 7 15	8 1 24 17 15
1 16 11 6 21	19 8 2 15 21	9 18 2 15 21	9 12 20 3 21	19 12 10 3 21

16/P	16/R1	0/R2	0/R3	13/R4
5 20 15 10 25	5 11 22 18 9	5 11 22 8 19	5 23 6 12 19	5 23 16 12 9
4 19 14 9 24	23 19 10 1 12	23 9 20 1 12	11 17 4 25 8	11 7 4 25 18
3 18 13 8 23	6 2 13 24 20	16 2 13 24 10	24 10 13 16 2	24 20 13 6 2
2 17 12 7 22	14 25 16 7 3	14 25 6 17 3	18 1 22 9 15	8 1 22 19 15
1 16 11 6 21	17 8 4 15 21	7 18 4 15 21	7 14 20 3 21	17 14 10 3 21

[解の総数 = 16/16x4]

[再構成した解のモニター結果]

エラー: 0/R1, 8/R2, 8/R3, 0/R4

[Calculated and Listed by Kanji Setsuda on Nov. 1, 2010 with MacOSX & Xcode3]

再構成した目的方陣の解番号は、元の基準解のリストのそれをわざわざ参照している。0/Rとあるのは、基準リストに無かった解を示す。どれもが  $n5 < n21$  に違反している。第1番解と第4番解をひな形にした場合に、目的方陣の総数をめでたく再構成することができた。第1番解は当然としても(原初五方陣そのものを第1番解に基づいて作っている!)、第4番解も合格するという点が興味深い。「最良のひな形」は、決して唯一無二ではない。第2番解、第3番解のひな形では半数の再構成に失敗しているが、これは最初に定義した「標準解出力用の不等式群」に合わなくなったためだ。変換法の実験ではよく起こることで、想定外の混乱に見舞われた。こういう場合には、不等式を全然使わない「原始解」の集合どうして変換を行うと良いことがわかっている。

下に、原始解集合の間でひな形変換法を実行した結果を示す。各々128個ある。ひな形も変換法も計8個ある。その並行試験を試みた。原初方陣だけは共通で、一切変えていない。

[原初五方陣と再構成した連立型対称汎五方陣の対照リスト-2: 解番号: 原初/P, 再構成/R]

1/P	1/R1	2/R2	3/R3	4/R4
1 2 3 4 5	1 23 20 12 9	1 23 10 12 19	1 23 20 14 7	1 23 10 14 17
6 7 8 9 10	15 7 4 21 18	15 17 4 21 8	15 9 2 21 18	15 19 2 21 8
11 12 13 14 15	24 16 13 10 2	24 6 13 20 2	22 16 13 10 4	22 6 13 20 4
16 17 18 19 20	8 5 22 19 11	18 5 22 9 11	8 5 24 17 11	18 5 24 7 11
21 22 23 24 25	17 14 6 3 25	7 14 16 3 25	19 12 6 3 25	9 12 16 3 25
2/P	2/R1	1/R2	4/R3	3/R4
1 2 3 4 5	1 23 10 12 19	1 23 20 12 9	1 23 10 14 17	1 23 20 14 7
16 17 18 19 20	15 17 4 21 8	15 7 4 21 18	15 19 2 21 8	15 9 2 21 18
11 12 13 14 15	24 6 13 20 2	24 16 13 10 2	22 6 13 20 4	22 16 13 10 4
6 7 8 9 10	18 5 22 9 11	8 5 22 19 11	18 5 24 7 11	8 5 24 17 11
21 22 23 24 25	7 14 16 3 25	17 14 6 3 25	9 12 16 3 25	19 12 6 3 25
3/P	3/R1	4/R2	1/R3	2/R4
1 4 3 2 5	1 23 20 14 7	1 23 10 14 17	1 23 20 12 9	1 23 10 12 19
6 9 8 7 10	15 9 2 21 18	15 19 2 21 8	15 7 4 21 18	15 17 4 21 8
11 14 13 12 15	22 16 13 10 4	22 6 13 20 4	24 16 13 10 2	24 6 13 20 2
16 19 18 17 20	8 5 24 17 11	18 5 24 7 11	8 5 22 19 11	18 5 22 9 11
21 24 23 22 25	19 12 6 3 25	9 12 16 3 25	17 14 6 3 25	7 14 16 3 25
4/P	4/R1	3/R2	2/R3	1/R4
1 4 3 2 5	1 23 10 14 17	1 23 20 14 7	1 23 10 12 19	1 23 20 12 9
16 19 18 17 20	15 19 2 21 8	15 9 2 21 18	15 17 4 21 8	15 7 4 21 18
11 14 13 12 15	22 6 13 20 4	22 16 13 10 4	24 6 13 20 2	24 16 13 10 2
6 9 8 7 10	18 5 24 7 11	8 5 24 17 11	18 5 22 9 11	8 5 22 19 11
21 24 23 22 25	9 12 16 3 25	19 12 6 3 25	7 14 16 3 25	17 14 6 3 25
5/P	5/R1	6/R2	7/R3	8/R4
1 6 11 16 21	1 15 24 8 17	1 15 22 8 19	1 15 24 18 7	1 15 22 18 9
2 7 12 17 22	23 7 16 5 14	23 9 16 5 12	23 17 6 5 14	23 19 6 5 12
3 8 13 18 23	20 4 13 22 6	20 2 13 24 6	10 4 13 22 16	10 2 13 24 16
4 9 14 19 24	12 21 10 19 3	14 21 10 17 3	12 21 20 9 3	14 21 20 7 3
5 10 15 20 25	9 18 2 11 25	7 18 4 11 25	19 8 2 11 25	17 8 4 11 25

<b>6/P</b>					<b>6/R1</b>					<b>5/R2</b>					<b>8/R3</b>					<b>7/R4</b>				
1	6	11	16	21	1	15	22	8	19	1	15	24	8	17	1	15	22	18	9	1	15	24	18	7
4	9	14	19	24	23	9	16	5	12	23	7	16	5	14	23	19	6	5	12	23	17	6	5	14
3	8	13	18	23	20	2	13	24	6	20	4	13	22	6	10	2	13	24	16	10	4	13	22	16
2	7	12	17	22	14	21	10	17	3	12	21	10	19	3	14	21	20	7	3	12	21	20	9	3
5	10	15	20	25	7	18	4	11	25	9	18	2	11	25	17	8	4	11	25	19	8	2	11	25
<b>7/P</b>					<b>7/R1</b>					<b>8/R2</b>					<b>5/R3</b>					<b>6/R4</b>				
1	16	11	6	21	1	15	24	18	7	1	15	22	18	9	1	15	24	8	17	1	15	22	8	19
2	17	12	7	22	23	17	6	5	14	23	19	6	5	12	23	7	16	5	14	23	9	16	5	12
3	18	13	8	23	10	4	13	22	16	10	2	13	24	16	20	4	13	22	6	20	2	13	24	6
4	19	14	9	24	12	21	20	9	3	14	21	20	7	3	12	21	10	19	3	14	21	10	17	3
5	20	15	10	25	19	8	2	11	25	17	8	4	11	25	9	18	2	11	25	7	18	4	11	25
<b>8/P</b>					<b>8/R1</b>					<b>7/R2</b>					<b>6/R3</b>					<b>5/R4</b>				
1	16	11	6	21	1	15	22	18	9	1	15	24	18	7	1	15	22	8	19	1	15	24	8	17
4	19	14	9	24	23	19	6	5	12	23	17	6	5	14	23	9	16	5	12	23	7	16	5	14
3	18	13	8	23	10	2	13	24	16	10	4	13	22	16	20	2	13	24	6	20	4	13	22	6
2	17	12	7	22	14	21	20	7	3	12	21	20	9	3	14	21	10	17	3	12	21	10	19	3
5	20	15	10	25	17	8	4	11	25	19	8	2	11	25	7	18	4	11	25	9	18	2	11	25
<b>9/P</b>					<b>9/R1</b>					<b>10/R2</b>					<b>11/R3</b>					<b>12/R4</b>				
2	1	3	5	4	2	23	19	11	10	2	23	9	11	20	2	23	19	15	6	2	23	9	15	16
7	6	8	10	9	14	6	5	22	18	14	16	5	22	8	14	10	1	22	18	14	20	1	22	8
12	11	13	15	14	25	17	13	9	1	25	7	13	19	1	21	17	13	9	5	21	7	13	19	5
17	16	18	20	19	8	4	21	20	12	18	4	21	10	12	8	4	25	16	12	18	4	25	6	12
22	21	23	25	24	16	15	7	3	24	6	15	17	3	24	20	11	7	3	24	10	11	17	3	24
<b>17/P</b>					<b>17/R1</b>					<b>18/R2</b>					<b>19/R3</b>					<b>20/R4</b>				
4	1	3	5	2	4	23	17	11	10	4	23	7	11	20	4	23	17	15	6	4	23	7	15	16
9	6	8	10	7	12	6	5	24	18	12	16	5	24	8	12	10	1	24	18	12	20	1	24	8
14	11	13	15	12	25	19	13	7	1	25	9	13	17	1	21	19	13	7	5	21	9	13	17	5
19	16	18	20	17	8	2	21	20	14	18	2	21	10	14	8	2	25	16	14	18	2	25	6	14
24	21	23	25	22	16	15	9	3	22	6	15	19	3	22	20	11	9	3	22	10	11	19	3	22
<b>25/P</b>					<b>25/R1</b>					<b>26/R2</b>					<b>27/R3</b>					<b>28/R4</b>				
5	2	3	4	1	5	23	16	12	9	5	23	6	12	19	5	23	16	14	7	5	23	6	14	17
10	7	8	9	6	11	7	4	25	18	11	17	4	25	8	11	9	2	25	18	11	19	2	25	8
15	12	13	14	11	24	20	13	6	2	24	10	13	16	2	22	20	13	6	4	22	10	13	16	4
20	17	18	19	16	8	1	22	19	15	18	1	22	9	15	8	1	24	17	15	18	1	24	7	15
25	22	23	24	21	17	14	10	3	21	7	14	20	3	21	19	12	10	3	21	9	12	20	3	21
<b>33/P</b>					<b>33/R1</b>					<b>34/R2</b>					<b>39/R3</b>					<b>40/R4</b>				
6	1	11	21	16	6	15	19	3	22	6	15	17	3	24	6	15	19	23	2	6	15	17	23	4
7	2	12	22	17	18	2	21	10	14	18	4	21	10	12	18	22	1	10	14	18	24	1	10	12
8	3	13	23	18	25	9	13	17	1	25	7	13	19	1	5	9	13	17	21	5	7	13	19	21
9	4	14	24	19	12	16	5	24	8	14	16	5	22	8	12	16	25	4	8	14	16	25	2	8
10	5	15	25	20	4	23	7	11	20	2	23	9	11	20	24	3	7	11	20	22	3	9	11	20
<b>41/P</b>					<b>41/R1</b>					<b>42/R2</b>					<b>47/R3</b>					<b>48/R4</b>				
7	2	12	22	17	7	14	20	3	21	7	14	16	3	25	7	14	20	23	1	7	14	16	23	5
6	1	11	21	16	18	1	22	9	15	18	5	22	9	11	18	21	2	9	15	18	25	2	9	11
8	3	13	23	18	24	10	13	16	2	24	6	13	20	2	4	10	13	16	22	4	6	13	20	22
10	5	15	25	20	11	17	4	25	8	15	17	4	21	8	11	17	24	5	8	15	17	24	1	8
9	4	14	24	19	5	23	6	12	19	1	23	10	12	19	25	3	6	12	19	21	3	10	12	19
<b>49/P</b>					<b>49/R1</b>					<b>50/R2</b>					<b>55/R3</b>					<b>56/R4</b>				
9	4	14	24	19	9	12	20	3	21	9	12	16	3	25	9	12	20	23	1	9	12	16	23	5
6	1	11	21	16	18	1	24	7	15	18	5	24	7	11	18	21	4	7	15	18	25	4	7	11
8	3	13	23	18	22	10	13	16	4	22	6	13	20	4	2	10	13	16	24	2	6	13	20	24
10	5	15	25	20	11	19	2	25	8	15	19	2	21	8	11	19	22	5	8	15	19	22	1	8
7	2	12	22	17	5	23	6	14	17	1	23	10	14	17	25	3	6	14	17	21	3	10	14	17
<b>57/P</b>					<b>57/R1</b>					<b>58/R2</b>					<b>63/R3</b>					<b>64/R4</b>				
10	5	15	25	20	10	11	19	3	22	10	11	17	3	24	10	11	19	23	2	10	11	17	23	4
7	2	12	22	17	18	2	25	6	14	18	4	25	6	12	18	22	5	6	14	18	24	5	6	12

8	3	13	23	18	21	9	13	17	5	21	7	13	19	5	1	9	13	17	25	1	7	13	19	25
9	4	14	24	19	12	20	1	24	8	14	20	1	22	8	12	20	21	4	8	14	20	21	2	8
6	1	11	21	16	4	23	7	15	16	2	23	9	15	16	24	3	7	15	16	22	3	9	15	16
			65/P				65/R1					66/R2					71/R3					72/R4		
16	1	11	21	6	16	15	9	3	22	16	15	7	3	24	16	15	9	23	2	16	15	7	23	4
17	2	12	22	7	8	2	21	20	14	8	4	21	20	12	8	22	1	20	14	8	24	1	20	12
18	3	13	23	8	25	19	13	7	1	25	17	13	9	1	5	19	13	7	21	5	17	13	9	21
19	4	14	24	9	12	6	5	24	18	14	6	5	22	18	12	6	25	4	18	14	6	25	2	18
20	5	15	25	10	4	23	17	11	10	2	23	19	11	10	24	3	17	11	10	22	3	19	11	10
			73/P				73/R1					74/R2					79/R3					80/R4		
17	2	12	22	7	17	14	10	3	21	17	14	6	3	25	17	14	10	23	1	17	14	6	23	5
16	1	11	21	6	8	1	22	19	15	8	5	22	19	11	8	21	2	19	15	8	25	2	19	11
18	3	13	23	8	24	20	13	6	2	24	16	13	10	2	4	20	13	6	22	4	16	13	10	22
20	5	15	25	10	11	7	4	25	18	15	7	4	21	18	11	7	24	5	18	15	7	24	1	18
19	4	14	24	9	5	23	16	12	9	1	23	20	12	9	25	3	16	12	9	21	3	20	12	9
			81/P				81/R1					82/R2					87/R3					88/R4		
19	4	14	24	9	19	12	10	3	21	19	12	6	3	25	19	12	10	23	1	19	12	6	23	5
16	1	11	21	6	8	1	24	17	15	8	5	24	17	11	8	21	4	17	15	8	25	4	17	11
18	3	13	23	8	22	20	13	6	4	22	16	13	10	4	2	20	13	6	24	2	16	13	10	24
20	5	15	25	10	11	9	2	25	18	15	9	2	21	18	11	9	22	5	18	15	9	22	1	18
17	2	12	22	7	5	23	16	14	7	1	23	20	14	7	25	3	16	14	7	21	3	20	14	7
			89/P				89/R1					90/R2					95/R3					96/R4		
20	5	15	25	10	20	11	9	3	22	20	11	7	3	24	20	11	9	23	2	20	11	7	23	4
17	2	12	22	7	8	2	25	16	14	8	4	25	16	12	8	22	5	16	14	8	24	5	16	12
18	3	13	23	8	21	19	13	7	5	21	17	13	9	5	1	19	13	7	25	1	17	13	9	25
19	4	14	24	9	12	10	1	24	18	14	10	1	22	18	12	10	21	4	18	14	10	21	2	18
16	1	11	21	6	4	23	17	15	6	2	23	19	15	6	24	3	17	15	6	22	3	19	15	6
			97/P				97/R1					98/R2					99/R3					100/R4		
21	6	11	16	1	21	15	4	8	17	21	15	2	8	19	21	15	4	18	7	21	15	2	18	9
22	7	12	17	2	3	7	16	25	14	3	9	16	25	12	3	17	6	25	14	3	19	6	25	12
23	8	13	18	3	20	24	13	2	6	20	22	13	4	6	10	24	13	2	16	10	22	13	4	16
24	9	14	19	4	12	1	10	19	23	14	1	10	17	23	12	1	20	9	23	14	1	20	7	23
25	10	15	20	5	9	18	22	11	5	7	18	24	11	5	19	8	22	11	5	17	8	24	11	5
			105/P				105/R1					106/R2					107/R3					108/R4		
22	7	12	17	2	22	14	5	8	16	22	14	1	8	20	22	14	5	18	6	22	14	1	18	10
21	6	11	16	1	3	6	17	24	15	3	10	17	24	11	3	16	7	24	15	3	20	7	24	11
23	8	13	18	3	19	25	13	1	7	19	21	13	5	7	9	25	13	1	17	9	21	13	5	17
25	10	15	20	5	11	2	9	20	23	15	2	9	16	23	11	2	19	10	23	15	2	19	6	23
24	9	14	19	4	10	18	21	12	4	6	18	25	12	4	20	8	21	12	4	16	8	25	12	4
			113/P				113/R1					114/R2					115/R3					116/R4		
24	9	14	19	4	24	12	5	8	16	24	12	1	8	20	24	12	5	18	6	24	12	1	18	10
21	6	11	16	1	3	6	19	22	15	3	10	19	22	11	3	16	9	22	15	3	20	9	22	11
23	8	13	18	3	17	25	13	1	9	17	21	13	5	9	7	25	13	1	19	7	21	13	5	19
25	10	15	20	5	11	4	7	20	23	15	4	7	16	23	11	4	17	10	23	15	4	17	6	23
22	7	12	17	2	10	18	21	14	2	6	18	25	14	2	20	8	21	14	2	16	8	25	14	2
			121/P				121/R1					122/R2					123/R3					124/R4		
25	10	15	20	5	25	11	4	8	17	25	11	2	8	19	25	11	4	18	7	25	11	2	18	9
22	7	12	17	2	3	7	20	21	14	3	9	20	21	12	3	17	10	21	14	3	19	10	21	12
23	8	13	18	3	16	24	13	2	10	16	22	13	4	10	6	24	13	2	20	6	22	13	4	20
24	9	14	19	4	12	5	6	19	23	14	5	6	17	23	12	5	16	9	23	14	5	16	7	23
21	6	11	16	1	9	18	22	15	1	7	18	24	15	1	19	8	22	15	1	17	8	24	15	1
			128/P				128/R1					127/R2					126/R3					125/R4		
25	24	23	22	21	25	3	6	14	17	25	3	16	14	7	25	3	6	12	19	25	3	16	12	9
20	19	18	17	16	11	19	22	5	8	11	9	22	5	18	11	17	24	5	8	11	7	24	5	18
15	14	13	12	11	2	10	13	16	24	2	20	13	6	24	4	10	13	16	22	4	20	13	6	22
10	9	8	7	6	18	21	4	7	15	8	21	4	17	15	18	21	2	9	15	8	21	2	19	15
5	4	3	2	1	9	12	20	23	1	19	12	10	23	1	7	14	20	23	1	17	14	10	23	1

[解の総数 = 128/128x4]

[再構成した解のモニター結果]

エラー: 0/R1, 0/R2, 0/R3, 0/R4

[原初五方陣と再構成した連立型対称汎五方陣の対照リスト-3: 解番号: 原初/P, 再構成/R]

1/P 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	5/R1 1 15 24 8 17 23 7 16 5 14 20 4 13 22 6 12 21 10 19 3 9 18 2 11 25	6/R2 1 15 22 8 19 23 9 16 5 12 20 2 13 24 6 14 21 10 17 3 7 18 4 11 25	7/R3 1 15 24 18 7 23 17 6 5 14 10 4 13 22 16 12 21 20 9 3 19 8 2 11 25	8/R4 1 15 22 18 9 23 19 6 5 12 10 2 13 24 16 14 21 20 7 3 17 8 4 11 25
2/P 1 2 3 4 5 16 17 18 19 20 11 12 13 14 15 6 7 8 9 10 21 22 23 24 25	7/R1 1 15 24 18 7 23 17 6 5 14 10 4 13 22 16 12 21 20 9 3 19 8 2 11 25	8/R2 1 15 22 18 9 23 19 6 5 12 10 2 13 24 16 14 21 20 7 3 17 8 4 11 25	5/R3 1 15 24 8 17 23 7 16 5 14 20 4 13 22 6 12 21 10 19 3 9 18 2 11 25	6/R4 1 15 22 8 19 23 9 16 5 12 20 2 13 24 6 14 21 10 17 3 7 18 4 11 25
3/P 1 4 3 2 5 6 9 8 7 10 11 14 13 12 15 16 19 18 17 20 21 24 23 22 25	6/R1 1 15 22 8 19 23 9 16 5 12 20 2 13 24 6 14 21 10 17 3 7 18 4 11 25	5/R2 1 15 24 8 17 23 7 16 5 14 20 4 13 22 6 12 21 10 19 3 9 18 2 11 25	8/R3 1 15 22 18 9 23 19 6 5 12 10 2 13 24 16 14 21 20 7 3 17 8 4 11 25	7/R4 1 15 24 18 7 23 17 6 5 14 10 4 13 22 16 12 21 20 9 3 19 8 2 11 25
4/P 1 4 3 2 5 16 19 18 17 20 11 14 13 12 15 6 9 8 7 10 21 24 23 22 25	8/R1 1 15 22 18 9 23 19 6 5 12 10 2 13 24 16 14 21 20 7 3 17 8 4 11 25	7/R2 1 15 24 18 7 23 17 6 5 14 10 4 13 22 16 12 21 20 9 3 19 8 2 11 25	6/R3 1 15 22 8 19 23 9 16 5 12 20 2 13 24 6 14 21 10 17 3 7 18 4 11 25	5/R4 1 15 24 8 17 23 7 16 5 14 20 4 13 22 6 12 21 10 19 3 9 18 2 11 25
5/P 1 6 11 16 21 2 7 12 17 22 3 8 13 18 23 4 9 14 19 24 5 10 15 20 25	1/R1 1 23 20 12 9 15 7 4 21 18 24 16 13 10 2 8 5 22 19 11 17 14 6 3 25	2/R2 1 23 10 12 19 15 17 4 21 8 24 6 13 20 2 18 5 22 9 11 7 14 16 3 25	3/R3 1 23 20 14 7 15 9 2 21 18 22 16 13 10 4 8 5 24 17 11 19 12 6 3 25	4/R4 1 23 10 14 17 15 19 2 21 8 22 6 13 20 4 18 5 24 7 11 9 12 16 3 25
6/P 1 6 11 16 21 4 9 14 19 24 3 8 13 18 23 2 7 12 17 22 5 10 15 20 25	3/R1 1 23 20 14 7 15 9 2 21 18 22 16 13 10 4 8 5 24 17 11 19 12 6 3 25	4/R2 1 23 10 14 17 15 19 2 21 8 22 6 13 20 4 18 5 24 7 11 9 12 16 3 25	1/R3 1 23 20 12 9 15 7 4 21 18 24 16 13 10 2 8 5 22 19 11 17 14 6 3 25	2/R4 1 23 10 12 19 15 17 4 21 8 24 6 13 20 2 18 5 22 9 11 7 14 16 3 25
7/P 1 16 11 6 21 2 17 12 7 22 3 18 13 8 23 4 19 14 9 24 5 20 15 10 25	2/R1 1 23 10 12 19 15 17 4 21 8 24 6 13 20 2 18 5 22 9 11 7 14 16 3 25	1/R2 1 23 20 12 9 15 7 4 21 18 24 16 13 10 2 8 5 22 19 11 17 14 6 3 25	4/R3 1 23 10 14 17 15 19 2 21 8 22 6 13 20 4 18 5 24 7 11 9 12 16 3 25	3/R4 1 23 20 14 7 15 9 2 21 18 22 16 13 10 4 8 5 24 17 11 19 12 6 3 25
8/P 1 16 11 6 21 4 19 14 9 24 3 18 13 8 23 2 17 12 7 22 5 20 15 10 25	4/R1 1 23 10 14 17 15 19 2 21 8 22 6 13 20 4 18 5 24 7 11 9 12 16 3 25	3/R2 1 23 20 14 7 15 9 2 21 18 22 16 13 10 4 8 5 24 17 11 19 12 6 3 25	2/R3 1 23 10 12 19 15 17 4 21 8 24 6 13 20 2 18 5 22 9 11 7 14 16 3 25	1/R4 1 23 20 12 9 15 7 4 21 18 24 16 13 10 2 8 5 22 19 11 17 14 6 3 25
9/P 2 1 3 5 4 7 6 8 10 9 12 11 13 15 14	13/R1 2 14 25 8 16 23 6 17 4 15 19 5 13 21 7	14/R2 2 14 21 8 20 23 10 17 4 11 19 1 13 25 7	15/R3 2 14 25 18 6 23 16 7 4 15 9 5 13 21 17	16/R4 2 14 21 18 10 23 20 7 4 11 9 1 13 25 17



17 16 18 20 19	11 22 9 20 3	15 22 9 16 3	11 22 19 10 3	15 22 19 6 3
22 21 23 25 24	10 18 1 12 24	6 18 5 12 24	20 8 1 12 24	16 8 5 12 24
17/P	21/R1	22/R2	23/R3	24/R4
4 1 3 5 2	4 12 25 8 16	4 12 21 8 20	4 12 25 18 6	4 12 21 18 10
9 6 8 10 7	23 6 19 2 15	23 10 19 2 11	23 16 9 2 15	23 20 9 2 11
14 11 13 15 12	17 5 13 21 9	17 1 13 25 9	7 5 13 21 19	7 1 13 25 19
19 16 18 20 17	11 24 7 20 3	15 24 7 16 3	11 24 17 10 3	15 24 17 6 3
24 21 23 25 22	10 18 1 14 22	6 18 5 14 22	20 8 1 14 22	16 8 5 14 22
25/P	29/R1	30/R2	31/R3	32/R4
5 2 3 4 1	5 11 24 8 17	5 11 22 8 19	5 11 24 18 7	5 11 22 18 9
10 7 8 9 6	23 7 20 1 14	23 9 20 1 12	23 17 10 1 14	23 19 10 1 12
15 12 13 14 11	16 4 13 22 10	16 2 13 24 10	6 4 13 22 20	6 2 13 24 20
20 17 18 19 16	12 25 6 19 3	14 25 6 17 3	12 25 16 9 3	14 25 16 7 3
25 22 23 24 21	9 18 2 15 21	7 18 4 15 21	19 8 2 15 21	17 8 4 15 21
33/P	35/R1	36/R2	37/R3	38/R4
6 1 11 21 16	6 18 25 12 4	6 18 5 12 24	6 18 25 14 2	6 18 5 14 22
7 2 12 22 17	15 2 9 16 23	15 22 9 16 3	15 4 7 16 23	15 24 7 16 3
8 3 13 23 18	19 21 13 5 7	19 1 13 25 7	17 21 13 5 9	17 1 13 25 9
9 4 14 24 19	3 10 17 24 11	23 10 17 4 11	3 10 19 22 11	23 10 19 2 11
10 5 15 25 20	22 14 1 8 20	2 14 21 8 20	24 12 1 8 20	4 12 21 8 20
41/P	43/R1	44/R2	45/R3	46/R4
7 2 12 22 17	7 18 24 11 5	7 18 4 11 25	7 18 24 15 1	7 18 4 15 21
6 1 11 21 16	14 1 10 17 23	14 21 10 17 3	14 5 6 17 23	14 25 6 17 3
8 3 13 23 18	20 22 13 4 6	20 2 13 24 6	16 22 13 4 10	16 2 13 24 10
10 5 15 25 20	3 9 16 25 12	23 9 16 5 12	3 9 20 21 12	23 9 20 1 12
9 4 14 24 19	21 15 2 8 19	1 15 22 8 19	25 11 2 8 19	5 11 22 8 19
49/P	51/R1	52/R2	53/R3	54/R4
9 4 14 24 19	9 18 22 11 5	9 18 2 11 25	9 18 22 15 1	9 18 2 15 21
6 1 11 21 16	12 1 10 19 23	12 21 10 19 3	12 5 6 19 23	12 25 6 19 3
8 3 13 23 18	20 24 13 2 6	20 4 13 22 6	16 24 13 2 10	16 4 13 22 10
10 5 15 25 20	3 7 16 25 14	23 7 16 5 14	3 7 20 21 14	23 7 20 1 14
7 2 12 22 17	21 15 4 8 17	1 15 24 8 17	25 11 4 8 17	5 11 24 8 17
57/P	59/R1	60/R2	61/R3	62/R4
10 5 15 25 20	10 18 21 12 4	10 18 1 12 24	10 18 21 14 2	10 18 1 14 22
7 2 12 22 17	11 2 9 20 23	11 22 9 20 3	11 4 7 20 23	11 24 7 20 3
8 3 13 23 18	19 25 13 1 7	19 5 13 21 7	17 25 13 1 9	17 5 13 21 9
9 4 14 24 19	3 6 17 24 15	23 6 17 4 15	3 6 19 22 15	23 6 19 2 15
6 1 11 21 16	22 14 5 8 16	2 14 25 8 16	24 12 5 8 16	4 12 25 8 16
...				
121/P	125/R1	126/R2	127/R3	128/R4
25 10 15 20 5	25 3 16 12 9	25 3 6 12 19	25 3 16 14 7	25 3 6 14 17
22 7 12 17 2	11 7 24 5 18	11 17 24 5 8	11 9 22 5 18	11 19 22 5 8
23 8 13 18 3	4 20 13 6 22	4 10 13 16 22	2 20 13 6 24	2 10 13 16 24
24 9 14 19 4	8 21 2 19 15	18 21 2 9 15	8 21 4 17 15	18 21 4 7 15
21 6 11 16 1	17 14 10 23 1	7 14 20 23 1	19 12 10 23 1	9 12 20 23 1
128/P	124/R1	123/R2	122/R3	121/R4
25 24 23 22 21	25 11 2 18 9	25 11 4 18 7	25 11 2 8 19	25 11 4 8 17
20 19 18 17 16	3 19 10 21 12	3 17 10 21 14	3 9 20 21 12	3 7 20 21 14
15 14 13 12 11	6 22 13 4 20	6 24 13 2 20	16 22 13 4 10	16 24 13 2 10
10 9 8 7 6	14 5 16 7 23	12 5 16 9 23	14 5 6 17 23	12 5 6 19 23
5 4 3 2 1	17 8 24 15 1	19 8 22 15 1	7 18 24 15 1	9 18 22 15 1

[解の総数 = 128/128x4]

[再構成した解のモニター結果]

エラー: 0/R1, 0/R2, 0/R3, 0/R4

[Calculated and Listed by Kanji Setsuda on Nov. 1, 2010 with MacOSX & Xcode3]

n1=1の解8個の全部が、「最良のひな形」として再構成に成功した。実に公平だ。



これらの実験結果は、いったい全体何を意味しているのだろうか？

原初方陣の「独立性・普遍性」が、いかにも印象的だ。実験中に原初方陣は全然変えていないから、原初方陣が個別のひな形に依存せず、ひな形から独立していると言える。

実際、どのひな形に準拠して原初五方陣を構成しても、解の出力順序こそ変われ、内容的には同一のものができる。原初五方陣の解集合は、同じものしか作れないのだ。

つまり、対称汎五方陣にとって真に「原初的な」もの、根源的なものが客観的・実体的にただ一種類存在しているわけで、それを根拠にひな形変換法が様々な個性的現象を描き出す。かかる原初方陣は、だから、目的方陣の「基本図」として、目的方陣に基本的な「構造」を提供しているのだと思われる。そして、その構造は、全ての解に共通なのだ。

だが、補数対称型の五方陣の場合には、そういう「構図」はついに見つからなかった。見つかったのは、わずかに汎対角線型と連立型の場合のみ。つまり、二者に共通である汎対角線型方陣にそういう特徴(=「運命」)があるのだと推理される。

これはまた、目的方陣の「対称性の低さ／高さ」を物語っていると思われる。言うならば、汎対角線型の魔方陣は、対称性が真に高いのだ。

四次に続いて五次の場合でも、実に興味深い結論が得られたと言わざるを得ない。

### #5. ひな形構成法の一提案

もう一つの特長は、3タイプに共通のひな形だ。こういうひな形は、日頃から是非暗記をしておきたいもの。というのは、一つ覚えておくと、それによってひな形変換法や原初方陣を作り出し、最後には目的方陣全数個を方法的に構成する「基本材料」になり得るから。

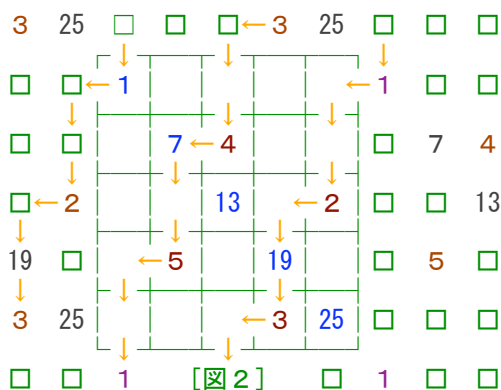
暗記の手助けになるように、次のような「手作りの構成法」を提案したい。紙と鉛筆があれば、間違いなく構成できるひな形の作り方である。

基本図の主対角線1本をコピーして始まる。対角線上の各数を起点にして原初方陣の横方向の各行を書き写す。右横に動くのではなく、専ら左下がりに「桂馬とび」する。



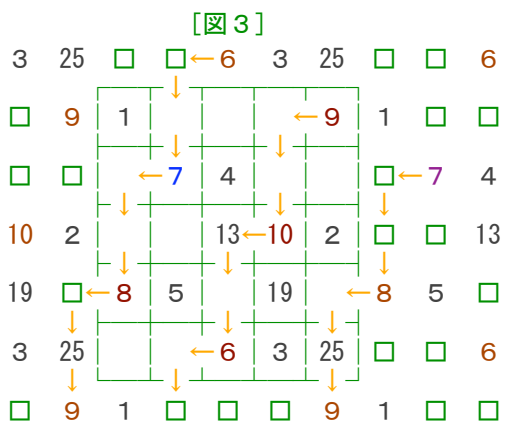
[図1]

右上図2のような空白方陣を用意し、図1の主対角線を1本コピーする。

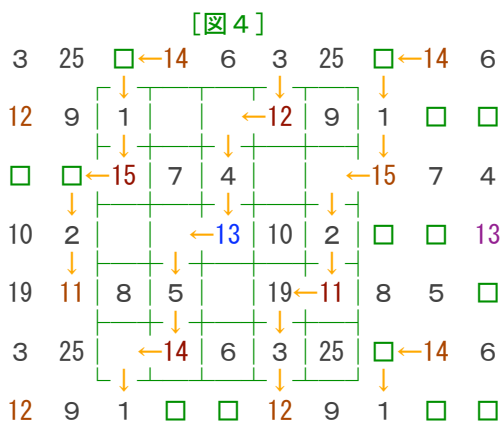


[図2]

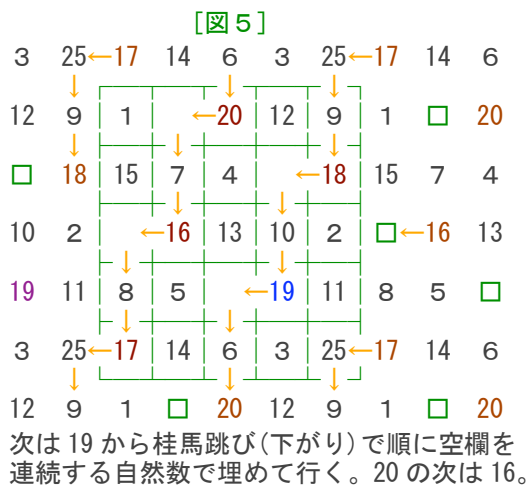
上図のように1から桂馬跳び(下がり)で順に連続する自然数2~5を埋めて行く。



次は7から桂馬跳び(下がり)で順に空欄を連続する自然数で埋めて行く。10の次は6。



次は13から桂馬跳び(下がり)で順に空欄を連続する自然数で埋めて行く。15の次は11。



### #6. 結論：付：原初方陣とひな形変換法の研究史

いったい「原初方陣」とは何なのか。何度もこの根本的な問いに突き当たる。

三方陣、四方陣、五方陣で各論的に原初方陣と主要な魔方陣との対応関係を考察して来たが、様々な性質がわかって来たと同時に、いよいよ謎も深まった、という感じがする。

何故にこうも一対一対応するのか。不思議だ。相手は、何しろ全てなのだから。

小生が勝手に強引に無理矢理にこじつけたからだ、などと言わないでいただきたい。確かに手順は複雑過ぎるし、論証は無いしで、大胆な結論を恣意的に急ぎ過ぎているという自覚が無いわけではない。

しかし、小生が説明したかったのは、過程ではなくて、現象の裏側或いは内側にある「連関」だ。その一部を諸氏に何とか見える形にしてアピールしたかったのである。

バラバラに個性的に存在するかに見えた各種の魔方陣の中に、いわば共通の「宿命」みたいなものがあって、それをこの原初方陣が示してくれているのではないのか。宿命は、形にも現れるが、解の総数にも現れる。

そして、原初方陣の研究を深めることで、各魔方陣に通用するいわば「マスターキー」のようなものを得られるのではないのか。そんな期待を持つにいった。

もしこの方法に何らかの普遍性があるとすれば、それは多分、方陣空間そのものの性質あるいは「正方配列」の性質そのものに根拠があるのである。

どんな方陣でも、必ず配列に入る。そして配列の制約を受ける。いや、配列の恩恵に浴する。そこから得たもののおかげで、奇跡的な一致があちこちで起きるのではあるまいか。

もしかしたら、魔方陣の様々な問題を、解の総数の計算や計数をも含めて、原初方陣の側に移して解決できるのではあるまいか。そんな想いにも駆られる。

この特質にいち早く気付いたのは、海外の数学者たちであった。英国のオルレンショウ&ブラーエ両氏が四、八、十二……次の正方連結方陣で解析を試みて、一書を著わしている。その紹介記事が、Scientific American誌1999年11月号に載った。そして、その概要を小生に教えてくださったのが、わが鈴木睦教授であった。2001年2月のことである。

ただならないものを感じた私たちは、この方法が他の次数、他のタイプの魔方陣についても適用できるのではないかと考えた。小生は、早速各論的に調べ始めた。

最初のうちは、全てを英国流に追試していたが、四次や八次の場合は確かにうまく行っても、五次・七次の原初方陣(英国流では“Reversible Squares”)の構成法に戸惑うことが多くなった。目的方陣のタイプが違くと、作り方まで根本的に変えなければならない。その変え方が必ずしも、一律ではない。「クロスサム等和」や「外項・内項の等和」を条件に用いるのが、いつでも正しいとは限らない。そうとわかった時にはどうしたらよいのか?

随分と長い間、試行錯誤した。悪戦苦闘の末に、結局正攻法に戻って、根本的な解決法を思い付いた。いわゆる「逆関数を解く」という方法で原初方陣を構成する。

原初方陣から目的方陣を構成するひな形変換法を「関数」と考えれば、目的方陣から原初方陣を構成する方法は「逆関数」である。それを具体的に解けばよい。この時に目的方陣の全数はもはや必要ではなく、ひな形1個があれば十分である。ひな形から原初方陣の構成条件を教わるだけで済むからだ。後はプログラムを組んで、コンピュータに計算させる。

この方針ならば、全次数の各タイプを一律に同じ方法で扱うことができる。

この試みは結局すべて成功したが、最初の内は不満だらけだった。

プログラムを組んでそのようにコンピュータに実行させても、結果が中々に出て来ない。正解が出るには出るが、実行速度がきわめて遅い。効率に欠ける点が、不満だった。

しかし、しばらくして原初方陣の作り方に高速のプログラミング方法が見つかった。要するに、概念通りに忠実に求めれば良かったのである。変数名を対応するものに書き換えるだけで、手順を同じにした。これで、けっこう高速になった。原関数より速くなったということはないが、原初方陣構成法だけが特別遅いということは無くなった。原関数と同じ程度のスピードで作れるようになったのである。

ただし、高速性を獲得した代わりに、解の出力順序が必ずしもスマートではなくなった。第一基本形がいつでも最初に現れるとは限らなくなった。「副作用」があるのである。

原初方陣では、なるべくならば、第一基本形をいつも最初に出力するようにしたい。

それは、この方法でも不可能ではないのだが、そうするためには、定義する変数の順序をかなり変えてやる必要がある。そうすると、殆どの場合実行速度がどんどん落ちてしまう。外見と性能が必ずしも一致しないのである。

外見を取るか高速性能を取るか、悩みのタネはつきない。だが、小生は高速性能を取る。外見は後でいくらでも整えることができるからだ。ソート・プログラムを使用して。

七次以上の高次方陣になると、別の悩みが増える。何をやるにも時間を食うのである。一つのタイプの解の総数を求めようにも、なかなか答を得られない。目的方陣の解の総数を知らないと、原初方陣を幾つ作ったら良いのかわからない。その点でもつまづく。

かくして「オイラー方陣」の出番が来た。全貌を速やかに知るためには完全オイラー型の汎魔方陣を作らざるを得ない。そのかわりに「ノン・オイラー型」は潔く見限る。

しかし、オイラー型の目的方陣に対応する原初方陣を「オイラー法」でどうやって作ったら良いのか？ それ自身がオイラー型でないものをはたして作れるものか？

試行錯誤を重ね、何とか結果を出すことができた。最新稿をぜひお読みいただきたい。

その後あれこれ研究が進むにつれて、小生自身は原初方陣が実は「基本図の集合」なのではないかと思うようになった。

配列自身と同じ第一基本形が、全ての方陣の「基本図」であることに異論はあるまい。

その他の原初方陣は、基本図の並び替えの問題を概念的に表す「変換図」の集合なのではあるまいか。

そうであればこそ、原初方陣の解が目的方陣の解と同じ数だけあり、しかも1つのひな形で全部の目的方陣を作り出す力があるということなのではないだろうか。

今の小生は、原初方陣が「基本図の集合」と論理的に同値であると、殆ど確信している。

と同時に、何故1個のひな形変換法で全部を作れるのかを、ますます不思議に感じる。

というのは、その意味が、ひな形をたった1個の「大基本解」と仮定することと同じだからだ。特に五次や七次の奇数次方陣では、連立型の中からひな形を選べば、3つのタイプのひな形を共通の1個に仮定することができる。そうすると、ひな形変換法も1個だけですむ。

このことが意味するのは、まさに「そのひな形が、3タイプ全ての五方陣の『大基本解』である」ということになる！

(On March 3, 2001; March 10, 2003; September 30, 2006 and November 7, 2010;  
MacOSX and Xcode 3.1.2 による計数, 作表, 執筆: 撰田 寛二(Kanji Setsuda))

E-Mail Address: jag12001@nifty.com

## 附録：プログラムの実際例

```
/** 原初五方陣と連立型対称汎五方陣 **/  
/** 'ProtoT5SmLP.c' Dictated by Kanji Setsuda **/  
/** on Jun.14, '05; Sep.27, '06; Oct. 4, '10 **/  
/** Working with MacOSX & Xcode3.1.2 **/  
**/  
/* Using Library */  
#include <stdio.h>  
**/  
/* Global Variables */  
short int cnt, cnt1;  
short LSM, PSM, cnt3;  
short nm[26], uflg[26];  
short dnm[26];  
short ln[129];  
short d[6][26];  
short st[129][26], pn[129][26];  
**/  
/* Sub-Routines 1: */  
void deep01(void), deep02(void), deep03(void), deep04(void);  
void deep05(void), deep06(void), deep07(void), deep08(void);  
void deep09(void), deep10(void), deep11(void), deep12(void);  
void deep13(void), deep14(void);  
void ansrecord(void);  
void prans(short x);  
**/  
/* Sub-Routines 2: */  
void stp01(void), stp02(void), stp03(void), stp04(void);  
void stp05(void), stp06(void), stp07(void), stp08(void);  
void stp09(void), stp10(void), stp11(void), stp12(void);  
void stp13(void), stp14(void);  
void protorecord(void);  
void sortp(void);  
void exc(short x);  
void printall(void), pr3ans(short x);  
void damt(void);  
short findoldn(void);  
void prmon(void);  
**/  
/* Main Program */  
int main(){  
  short m,n;  
  printf("\n ** 原初五方陣と連立型対称汎五方陣 **\n");  
  for(n=0;n<26;n++){nm[n]=0; uflg[n]=0;}  
  LSM=65; PSM=26; cnt=0; cnt3=0;  
  nm[13]=13; uflg[13]=1;  
  deep01();  
  if(cnt3>0){prans(cnt3);}  
  printf(" [Count = %d]\n",cnt);  
  printf("\n **原初五方陣と再構成した連立型対称汎五方陣：原初/P 再構成/R **\n");  
  for(n=0;n<26;n++){nm[n]=0; uflg[n]=0;}  
  for(m=0;m<cnt;m++){ln[m]=0;}  
  LSM=65; PSM=26; cnt1=0;  
  nm[13]=13; uflg[13]=1;  
  stp01();  
  sortp();  
  printall();  
  if(cnt3>0){pr3ans(cnt3);}  
  printf(" [Count = %d/%d]\n",cnt1,cnt);  
  printf("\n [モニター・リスト：基準解と再構成した連立型対称汎五方陣との対比]\n");  
  prmon();  
  printf(" OK! [リスト作成：Kanji Setsuda on Oct. 4, 2010 with MacOSX & Xcode3]\n");  
  return 0;
```

```

}
/**/
/* Level 1: */
/* Set n1 & n25 */
void deep01(){
  short a,b;
  for(a=1;a<26;a++){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[1]=a; nm[25]=b;
      uflg[a]=1; uflg[b]=1;
      deep02();
      uflg[b]=0; uflg[a]=0;}
    }
}
/* Set n11 & n15 */
void deep02(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[11]=a; nm[15]=b;
      uflg[a]=1; uflg[b]=1;
      deep03();
      uflg[b]=0; uflg[a]=0;}
    }
}
/* Set n16 & n10 */
void deep03(){
  short a,b;
  for(a=1;a<26;a++){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[16]=a; nm[10]=b;
      uflg[a]=1; uflg[b]=1;
      deep04();
      uflg[b]=0; uflg[a]=0;}
    }
}
/* Set n21 & n5 */
void deep04(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[21]=a; nm[5]=b;
      uflg[a]=1; uflg[b]=1;
      deep05();
      uflg[b]=0; uflg[a]=0;}
    }
}
/* Set n6=65-n1-n11-n16-n21 & n20 */
void deep05(){
  short a,b;
  a=LSM-nm[1]-nm[11]-nm[16]-nm[21];
  b=LSM-nm[25]-nm[15]-nm[10]-nm[5];
  if((0<a)&&(a<26)&&(a+b==PSM)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[6]=a; nm[20]=b;
      uflg[a]=1; uflg[b]=1;
      deep06();
      uflg[b]=0; uflg[a]=0;}}
}
/* Set n2 & n24 */
void deep06(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){

```



```

        nm[2]=a; nm[24]=b;
        uflg[a]=1; uflg[b]=1;
        deep07();
        uflg[b]=0; uflg[a]=0;}
    }
}
/* Set n3 & n23 */
void deep07(){
    short a,b;
    for(a=25;a>0;a--){b=PSM-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[3]=a; nm[23]=b;
            uflg[a]=1; uflg[b]=1;
            deep08();
            uflg[b]=0; uflg[a]=0;}
        }
}
/* Set n4=65-n1-n2-n3-n5 & n22 */
void deep08(){
    short a,b;
    a=LSM-nm[1]-nm[2]-nm[3]-nm[5];
    b=LSM-nm[25]-nm[24]-nm[23]-nm[21];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[4]=a; nm[22]=b;
            uflg[a]=1; uflg[b]=1;
            deep09();
            uflg[b]=0; uflg[a]=0;}}
}
/* Level 2: */
/* Set n7=65-n3-n11-n20-n24 & n19 */
void deep09(){
    short a,b;
    a=LSM-nm[3]-nm[11]-nm[20]-nm[24];
    b=LSM-nm[23]-nm[15]-nm[6]-nm[2];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[7]=a; nm[19]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[1]+nm[7]+nm[13]+nm[19]+nm[25]==LSM){deep10();}
            uflg[b]=0; uflg[a]=0;}}
}
/* Set n9=65-n3-n15-n16-n22 & n17 */
void deep10(){
    short a,b;
    a=LSM-nm[3]-nm[15]-nm[16]-nm[22];
    b=LSM-nm[23]-nm[11]-nm[10]-nm[4];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[9]=a; nm[17]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[5]+nm[9]+nm[13]+nm[17]+nm[21]==LSM){deep11();}
            uflg[b]=0; uflg[a]=0;}}
}
/* Set n8=65-n6-n7-n9-n10 & n18 */
void deep11(){
    short a,b;
    a=LSM-nm[6]-nm[7]-nm[9]-nm[10];
    b=LSM-nm[20]-nm[19]-nm[17]-nm[16];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[8]=a; nm[18]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[3]+nm[8]+nm[13]+nm[18]+nm[23]==LSM){deep12();}

```

```

        uflg[b]=0; uflg[a]=0;}}
}
/* Set n12=65-n2-n7-n17-n22 & n14 */
void deep12(){
short a,b,c,d;
a=LSM-nm[2]-nm[7]-nm[17]-nm[22];
b=LSM-nm[24]-nm[19]-nm[9]-nm[4];
c=LSM-nm[4]-nm[8]-nm[16]-nm[25];
d=LSM-nm[1]-nm[10]-nm[18]-nm[22];
if((0<a)&&(a<26)){
if((a==c)&&(b==d)&&(a+b==PSM)){
if((uflg[a]==0)&&(uflg[b]==0)){
nm[12]=a; nm[14]=b;
uflg[a]=1; uflg[b]=1;
if(nm[11]+nm[12]+nm[13]+nm[14]+nm[15]==LSM){deep13();}
uflg[b]=0; uflg[a]=0;}}}
}
/* Checks Sums of Some 4 Entries */
void deep13(void){
short sm1, sm2;
sm1=nm[5]+nm[6]+nm[12]+nm[18]+nm[24];
sm2=nm[2]+nm[8]+nm[14]+nm[20]+nm[21];
if((sm1==LSM)&&(sm2==LSM)){ansrecord();}
}
/**/
/* Record the Answers */
void ansrecord(){
short n;
cnt++;
for(n=1;n<26;n++){st[cnt][n]=nm[n];}
d[cnt3][0]=cnt;
for(n=1;n<26;n++){d[cnt3][n]=nm[n];}
cnt3++; if(cnt3==5){prans(cnt3); cnt3=0;}
}
/**/
/* Print the Answers */
void prans(short x){
short l,l5,m,n;
for(n=0;n<x;n++){
printf("%15d/",d[n][0]);
if(n+1<x){printf(" ");}
}
printf("\n");
for(l=0;l<5;l++){l5=l*5;
for(m=0;m<x;m++){
printf(" %3d%3d%3d%3d%3d",
d[m][l5+1],d[m][l5+2],d[m][l5+3],d[m][l5+4],d[m][l5+5],d[m][l+26],d[m][l+31]);
if(m+1<x){printf(" ");}
}
printf("\n");
}
}
/**/
/* Make the Prototype Squares and Apply 'DAMT' */
/* Define Level 1: */
/* Set n1 & n25 */
void stp01(){
short a,b;
for(a=1;a<26;a++){b=PSM-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[1]=a; nm[25]=b;
uflg[a]=1; uflg[b]=1;
stp02();
uflg[b]=0; uflg[a]=0;}
}
}

```



```

}
}
/* Set n24 & n2 */
void stp02(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[24]=a; nm[2]=b;
      uflg[a]=1; uflg[b]=1;
      stp03();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n8 & n18 */
void stp03(){
  short a,b;
  for(a=1;a<26;a++){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[8]=a; nm[18]=b;
      uflg[a]=1; uflg[b]=1;
      stp04();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n17 & n9 */
void stp04(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[17]=a; nm[9]=b;
      uflg[a]=1; uflg[b]=1;
      stp05();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n15=65-n1-n24-n8-n17 & n11 */
void stp05(){
  short a,b;
  a=LSM-nm[1]-nm[24]-nm[8]-nm[17];
  b=LSM-nm[25]-nm[2]-nm[18]-nm[9];
  if((0<a)&&(a<26)&&(a+b==PSM)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[15]=a; nm[11]=b;
      uflg[a]=1; uflg[b]=1;
      stp06();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n23 & n3 */
void stp06(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[23]=a; nm[3]=b;
      uflg[a]=1; uflg[b]=1;
      stp07();
      uflg[b]=0; uflg[a]=0;}
  }
}
/* Set n20 & n6 */
void stp07(){
  short a,b;
  for(a=25;a>0;a--){b=PSM-a;
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[20]=a; nm[6]=b;

```

```

        uflg[a]=1; uflg[b]=1;
        stp08();
        uflg[b]=0; uflg[a]=0;}
    }
}
/* Set n12=65-n1-n23-n20-n9 & n14 */
void stp08(){
    short a,b;
    a=LSM-nm[1]-nm[23]-nm[20]-nm[9];
    b=LSM-nm[25]-nm[3]-nm[6]-nm[17];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[12]=a; nm[14]=b;
            uflg[a]=1; uflg[b]=1;
            stp09();
            uflg[b]=0; uflg[a]=0;}}
}
/* Level 2 */
/* Set n7=65-n20-n24-n11-n3 & n19 */
void stp09(){
    short a,b;
    a=LSM-nm[20]-nm[24]-nm[11]-nm[3];
    b=LSM-nm[6]-nm[2]-nm[15]-nm[23];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[7]=a; nm[19]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[1]+nm[7]+nm[13]+nm[19]+nm[25]==LSM){stp10();}
            uflg[b]=0; uflg[a]=0;}}
}
/* Set n21=65-n20-n2-n8-n14 & n */
void stp10(){
    short a,b;
    a=LSM-nm[20]-nm[2]-nm[8]-nm[14];
    b=LSM-nm[12]-nm[18]-nm[24]-nm[6];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[21]=a; nm[5]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[9]+nm[21]+nm[13]+nm[5]+nm[17]==LSM){stp11();}
            uflg[b]=0; uflg[a]=0;}}
}
/* Set n4=65-n15-n7-n21-n18 & n22 */
void stp11(){
    short a,b;
    a=LSM-nm[15]-nm[7]-nm[21]-nm[18];
    b=LSM-nm[11]-nm[19]-nm[5]-nm[8];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[4]=a; nm[22]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[20]+nm[4]+nm[13]+nm[22]+nm[6]==LSM){stp12();}
            uflg[b]=0; uflg[a]=0;}}
}
/* Set n16=65-n23-n7-n5-n14 & n10 */
void stp12(){
    short a,b;
    a=LSM-nm[23]-nm[7]-nm[5]-nm[14];
    b=LSM-nm[3]-nm[19]-nm[21]-nm[12];
    if((0<a)&&(a<26)&&(a+b==PSM)){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[16]=a; nm[10]=b;
            uflg[a]=1; uflg[b]=1;
            if(nm[24]+nm[16]+nm[13]+nm[10]+nm[2]==LSM){stp13();}

```

```

        uflg[b]=0; uflg[a]=0;}}
}
/**/
/* Checks Sums of Some 4 Entries */
void stp13(void){
    short sm1, sm2, sm3, sm4;
    sm1=nm[1]+nm[18]+nm[10]+nm[22]+nm[14];
    sm2=nm[25]+nm[8]+nm[16]+nm[4]+nm[12];
    sm3=nm[9]+nm[15]+nm[16]+nm[22]+nm[3];
    sm4=nm[17]+nm[11]+nm[10]+nm[4]+nm[23];
    if((sm1==LSM)&&(sm2==LSM)&&(sm3==LSM)&&(sm4==LSM)){protorecord();}
}
/**/
/* Record the Answers */
void protorecord(){
    short n;
    cnt1++;
    for(n=1;n<26;n++){pn[cnt1][n]=nm[n];}
}
/**/
/* Print the Prototypes and the Objects */
void printall(){
    short m,n;
    for(m=0;m<6;m++){for(n=0;n<26;n++){d[m][n]=0;}}
    cnt3=0;
    for(m=1;m<=cnt1;m++){
        d[cnt3*2][0]=m;
        for(n=1;n<26;n++){nm[n]=pn[m][n]; d[cnt3*2][n]=nm[n];}
        damt();
        d[cnt3*2+1][0]=findoldn();
        for(n=1;n<26;n++){d[cnt3*2+1][n]=dnm[n];}
        cnt3++; if(cnt3==3){pr3ans(cnt3); cnt3=0;}
    }
}
/**/
/* Print the Answers */
void pr3ans(short x){
    short l,l5,m,m2,n;
    for(n=0;n<x;n++){m=d[n*2+1][0];
        if(m<10){printf("%14d/P%3d/R ",d[n*2][0],m);}
        else if(m<100){printf("%14d/P%4d/R ",d[n*2][0],m);}
        else{printf("%14d/P%5d/R",d[n*2][0],m);}
        if(n+1<x){printf(" ");}}
    printf("\n");
    for(l=0;l<5;l++){l5=l*5;
        for(m=0;m<x;m++){m2=m*2;
            printf(" ");
            for(n=1;n<6;n++){
                printf("%3d",d[m2][l5+n]);}
            printf(" ");
            for(n=1;n<6;n++){
                printf("%3d",d[m2+1][l5+n]);}
            if(m+1<x){printf(" ");}
        }
        printf("\n");
    }
}
/**/
/* Find its Old Number */
short findoldn(){
    short int fn,cn,m,n;
    fn=0;
    for(m=1;m<=cnt;m++){cn=0;
        for(n=1;n<26;n++){if(dnm[n]==st[m][n]){cn++;}else{break;}}}
}

```

```

    if(cn==25){fn=m; break;}
}
if(cn<25){fn=0;}
ln[fn]++;
return fn;
}
/**/
/* Print the Monitor List of Correspondence */
void prmon(){
short m;
printf(" ??: %d",ln[0]);
for(m=1;m<=cnt1;m++){
    if(m%40==1){printf("\n%5d:",m);}
    printf("%2d",ln[m]);
}
printf("\n");
}
/**/
/* Do-it-After-the-Model Transformation */
void damt(void){
    dnm[1]=nm[1]; dnm[2]=nm[23]; dnm[3]=nm[20]; dnm[4]=nm[12]; dnm[5]=nm[9];
    dnm[6]=nm[15]; dnm[7]=nm[7]; dnm[8]=nm[4]; dnm[9]=nm[21]; dnm[10]=nm[18];
    dnm[11]=nm[24]; dnm[12]=nm[16]; dnm[13]=nm[13]; dnm[14]=nm[10]; dnm[15]=nm[2];
    dnm[16]=nm[8]; dnm[17]=nm[5]; dnm[18]=nm[22]; dnm[19]=nm[19]; dnm[20]=nm[11];
    dnm[21]=nm[17]; dnm[22]=nm[14]; dnm[23]=nm[6]; dnm[24]=nm[3]; dnm[25]=nm[25];
}
/**/
/* Sort the Prototypes */
void sortp(){
long d1,d2;
short m,mx,f;
mx=cnt1-1;
do{f=0;
    for(m=1,m<=mx;m++){
        d1=((pn[m][1]*25+pn[m][2])*25+pn[m][3])*25+pn[m][4]*25+pn[m][5];
        d2=((pn[m+1][1]*25+pn[m+1][2])*25+pn[m+1][3])*25+pn[m+1][4]*25+pn[m+1][5];
        if(d1>d2){exc(m); f=1;}
        else if((d1==d2)&&(pn[m][6]>pn[m+1][6])){exc(m); f=1;}
        else if((d1==d2)&&(pn[m][6]==pn[m+1][6])&&(pn[m][7]>pn[m+1][7])){exc(m); f=1;}
        else if((d1==d2)&&(pn[m][6]==pn[m+1][6])&&(pn[m][7]==pn[m+1][7])){
            if(pn[m][11]>pn[m+1][11]){exc(m); f=1;}
            else if((pn[m][11]==pn[m+1][11])&&(pn[m][16]>pn[m+1][16])){exc(m); f=1;}
        }
    }
    mx--;
}while(f>0);
}
/**/
void exc(short x){
short n,d;
for(n=0;n<26;n++){
    d=pn[x][n]; pn[x][n]=pn[x+1][n]; pn[x+1][n]=d;
}
}
/**/

```