

# 第3部：『魔方陣の新研究』

## 第3章：五方陣の最新研究

### 第1節：対称五方陣の最新研究：摂田 寛二

O. 「対称方陣」とは、普通の条件に加えて、方陣の幾何学的中心を中心点として対称の位置にある任意の2数の和が一定の値になる方陣のことを言う。

#### 1. 基本ポジションと変数名

n1	n2	n3	n4	n5
n6	n7	n8	n9	n10
n11	n12	n13	n14	n15
n16	n17	n18	n19	n20
n21	n22	n23	n24	n25

#### 2. 連立方程式群

$$\begin{aligned} n_1 + n_2 + n_3 + n_4 + n_5 &= K \quad \dots (1) \\ n_6 + n_7 + n_8 + n_9 + n_{10} &= K \quad \dots (2) \\ n_{11} + n_{12} + n_{13} + n_{14} + n_{15} &= K \quad \dots (3) \\ n_{16} + n_{17} + n_{18} + n_{19} + n_{20} &= K \quad \dots (4) \\ n_{21} + n_{22} + n_{23} + n_{24} + n_{25} &= K \quad \dots (5) \\ n_1 + n_6 + n_{11} + n_{16} + n_{21} &= K \quad \dots (6) \\ n_2 + n_7 + n_{12} + n_{17} + n_{22} &= K \quad \dots (7) \\ n_3 + n_8 + n_{13} + n_{18} + n_{23} &= K \quad \dots (8) \\ n_4 + n_9 + n_{14} + n_{19} + n_{24} &= K \quad \dots (9) \\ n_5 + n_{10} + n_{15} + n_{20} + n_{25} &= K \quad \dots (10) \end{aligned}$$

主対角線上の5数の和も定和Kに等しい。

$$n_1 + n_7 + n_{13} + n_{19} + n_{25} = K \quad \dots (11)$$

$$n_5 + n_9 + n_{13} + n_{17} + n_{21} = K \quad \dots (12)$$

重複しない自然数1～25の全数個を用いて上の連立方程式を満たさなければならぬ。

先ず、五方陣一般に共通の定数Kの値を最初に決めておこう。

(1)～(5)式から、 $n_1 + n_2 + \dots + n_{24} + n_{25} = 5*K$

左辺は、自然数1～25の総和に他ならないから、

$$\begin{aligned} 25*(25+1)/2 &= 5*K \\ \therefore K &= 65 \end{aligned} \quad \dots (13)$$

さらに「対称型」の五方陣では、次の補数定和式を付け加える。

$$\begin{aligned} n_1 + n_{25} &= n_2 + n_{24} = n_3 + n_{23} = n_4 + n_{22} = n_5 + n_{21} = n_6 + n_{20} = n_7 + n_{19} \\ &= n_8 + n_{18} = n_9 + n_{17} = n_{10} + n_{16} = n_{11} + n_{15} = n_{12} + n_{14} = C \quad \dots (14) \end{aligned}$$

$n_{13}$ は、中心の位置があるので、 $n_{13} + n_{13} = C$ と考えることにする。

定数Cの値を決めておこう。

対角線の式(11)に注目する。

$$n_1 + n_7 + n_{13} + n_{19} + n_{25} = 65 \quad \dots (11)$$

$$(n_1 + n_{25}) + n_{13} + (n_7 + n_{19}) = 65$$

$$(14)式から \quad C + C/2 + C = 65$$

$$\text{両辺を2倍して} \quad 5C = 2 \times 65$$

$$\text{ゆえに} \quad C = 130 / 5 = 26 \quad \dots (15)$$

$$\text{一方, } 2 \times n_{13} = C \text{ から, } n_{13} = 13 \quad \dots (16)$$

(14)式の12組の数は、すべて「26の補数」を成すことがわかった。

対称五方陣の中央には定数13が来る。これは、全ての対称配置の中心点に $n_{13}$ があることの表現である( $n_{13} + n_{13} = 26$ )。こうした魔方陣のことを「補数対称型五方陣」と言う。

対称の位置にある補数の配置を見るには、下図のような「補数表現」が便利だ。

もともと $n_{25}$ の位置には、 $n_1$ にある数値と足して26になる数値が入っている。つまり、 $n_1 + n_{25} = 26$ なのだが、それをあえて $n_1 + \underline{n_1} = 26$ と表記している。

$n_{24}$ には $n_2$ の補数が入っているのを、 $\underline{n_2}$ と表記している。

$n_{23}$ には $n_3$ の補数が入っているのを、 $\underline{n_3}$ と表記している。

.....

$n_{14}$ には $n_{12}$ の補数が入っているのを、 $\underline{n_{12}}$ と表記している。

補数対称型五方陣の「補数表現形」:  
[原形] [補数表現]

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

1	2	3	4	5
6	7	8	9	10
11	12	13	12	11
10	9	8	7	6
5	4	3	2	1

$$\begin{aligned}
 n_1 + n_1 &= 26 \\
 n_2 + n_2 &= 26 \\
 n_3 + n_3 &= 26 \\
 n_4 + n_4 &= 26 \\
 n_5 + n_5 &= 26 \\
 n_6 + n_6 &= 26 \\
 \dots\dots \\
 n_{10} + n_{10} &= 26 \\
 n_{11} + n_{11} &= 26 \\
 n_{12} + n_{12} &= 26 \\
 n_{13} + n_{13} &= 26
 \end{aligned}$$

補数表現形を用いると、例えば主対角線の定和が、計算なしで「直視」できる。

$$\begin{aligned}
 n_1 + n_7 + n_{13} + n_{19} + n_{25} &= n_1 + n_7 + n_{13} + n_7 + n_1 \\
 &= (n_1 + n_1) + (n_7 + n_7) + n_{13} = 26 + 26 + 13 = 65 \\
 n_5 + n_9 + n_{13} + n_{17} + n_{21} &= n_5 + n_9 + n_{13} + n_9 + n_5 \\
 &= (n_5 + n_5) + (n_9 + n_9) + n_{13} = 26 + 26 + 13 = 65
 \end{aligned}$$

### 3. ここで行または列の交換移動による「異形」の出現を考える。

先ず最初に考えなければならないのは、鏡映反転と90度ずつの回転によって出来る異形である。下図を見てほしい。

0/を、原形とする。1/は、0/を90度時計回りに回転させた形。2/は、1/をさらに90度時計回りに回転させた形。3/は、2/をさらに90度時計回りに回転させた形だ。

4/は、主対角線  $1+7+13+19+25$  を軸として 0/を対称反転させた形。5/は、4/を90度時計回りに回転させた形。6/は、5/をさらに90度時計回りに回転させた形。そして 7/は、6/をさらに90度時計回りに回転させた形である。

#### \* 鏡映反転と90度ずつの回転の概念図 \*

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

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

我々の慣習では、このような8個の異形を全て「同一解」と見て、別個には数えない。

このタイプの変形は、平面方陣ならば次数・タイプに関係なく、いつでも起きるからである。だから、標準解のリストには、専用の不等式を用いてふるいを掛け、代表解1個のみを出力するようにする。例えば、五方陣では、 $n_1 < n_5 < n_{21}; n_1 < n_{25}$ ; を用いる。

ここで、対称五方陣の解が、いったい幾つぐらいあるのかをコンピュータに計数させてみよう。次のようなプログラムを用意して実行した。

```
//** Self-Complementary Magic Squares of Order 5: **
//** Compact List of 48544 Standard Solutions **
//** File: 'SCMS55S.c' Dictated by Kanji Setsuda **
```

```

/** on Feb. 5, 2005; Sep.30, 2006 */
/** Revised on Jan.25, 2015 */
/** with Mac OSX 10.10.1 & Xcode 6.1.1 */
//
/** Basic Form for SCMS55 */
// .-.-.-.-.-. .-.-.-.-.-.
// | 1| 2| 3| 4| 5| | 1|23| 5|24|12|
// |---+---+---+--| |---+---+---+--|
// | 6| 7| 8| 9|10| | 8|15|19|17| 6|
// |---+---+---+--| |---+---+---+--|
// |11|12|13|14|15| |22|16|13|10| 4|
// |---+---+---+--| |---+---+---+--|
// |16|17|18|19|20| |20| 9| 7|11|18|
// |---+---+---+--| |---+---+---+--|
// |21|22|23|24|25| |14| 2|21| 3|25|
// '---'---'---'
/** Basic Conditions for SCMS55: C=65; */
// n1+n2+n3+n4+n5=C ...rw1; | n1+n6+n11+n16+n21=C ...cl1;
// n6+n7+n8+n9+n10=C ...rw2; | n2+n7+n12+n17+n22=C ...cl2;
// n11+n12+n13+n14+n15=C ...rw3; | n3+n8+n13+n18+n23=C ...cl3;
// n16+n17+n18+n19+n20=C ...rw4; | n4+n9+n14+n19+n24=C ...cl4;
// n21+n22+n23+n24+n25=C ...rw5; | n5+n10+n15+n20+n25=C ...cl5;
/** Self-complementary Conditions for SCMS55: SC=26; */
// n1+n25=n2+n24=n3+n23=n4+n22=n5+n21=n6+n20=
// n7+n19=n8+n18=n9+n17=n10+n16=n11+n15=n12+n14=n13+n13=SC; ...scc
//
#include <stdio.h>
//
long cnt;
long cntr[5];
short cnt11, cnt13;
short cnt2, cnt3;
short LSM, CC;
short nm[26], uflg[26], area[26];
short nt[5][26];
//
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 stp21(void), stp22(void), stp23(void), stp24(void);
void stp25(void), stp26(void), stp27(void), stp28(void);
void stp29(void), stp30(void), stp31(void), stp32(void);
//
void ansprint(void);
void prans(short x);
//
/* Main Program */
int main(){
short n;
printf("\n");
printf(" ** Self-Complementary Magic Squares of Order 5: **\n");
printf(" ** Compact List 1 of 48544 Standard Solutions starting with n1=1 **\n");
for(n=0;n<26;n++){nm[n]=0; uflg[n]=0;}
LSM=65; CC=26; cnt=0; cnt11=0; cnt13=0; cnt3=0;
nm[13]=13; uflg[13]=1;
stp01();
uflg[13]=0; nm[13]=0;
if(cnt3>0){prans(cnt3);}
printf(" [Solution Counts(n1=1/n3=1/All) = %d/%d/%ld] OK!\n",cnt11,cnt13,cnt);
printf("\n");
printf(" ** Compact List 2 of 48544 Standard Solutions starting with n3=1 **\n");
for(n=0;n<26;n++){nm[n]=0; uflg[n]=0;}
LSM=65; CC=26; cnt=0; cnt11=0; cnt13=0; cnt3=0;
nm[13]=13; uflg[13]=1;
}

```

```

stp21();
uflg[13]=0; nm[13]=0;
if(cnt3>0){prans(cnt3);}
printf(" [Solution Counts(n3=1/n1=1/All) = %d/%d/%ld]  OK!\n",cnt13,cnt11,cnt);
printf("\n");
printf("** Calculated and Listed by Kanji Setsuda on\n");
printf("    Jan.25, 2015 with MacOSX 10.10.1 & Xcode 6.1.1 **\n");
printf("\n");
return 0;
}
/* Begin The Search 1 *
/** Normalizing Inequality Conditions: n1<n5<n21 and n1<n25;
/* Define Level 1: *
/* Set N1 & n25 & N1<n25 *
void stp01(){
short a,b;
for(a=1;a<13;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){cnt2=0;
nm[1]=a; nm[25]=b;
uflg[a]=1; uflg[b]=1;
stp02();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set N3 & n23 *
void stp02(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){if(nm[1]==1){cnt2=0;}
nm[3]=a; nm[23]=b;
uflg[a]=1; uflg[b]=1;
stp03();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set N2 & n24 *
void stp03(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[2]=a; nm[24]=b;
uflg[a]=1; uflg[b]=1;
stp04();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set N4 & n22 *
void stp04(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[4]=a; nm[22]=b;
uflg[a]=1; uflg[b]=1;
stp05();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set n5=LSM-n1-n2-n3-n4 & n21 & n1<n5<n21 *
void stp05(){
short a,b;
a=LSM-nm[1]-nm[2]-nm[3]-nm[4];
if((nm[1]<a)&&(a<26)){
b=LSM-nm[25]-nm[24]-nm[23]-nm[22];
if((a<b)&&(a+b==CC)){
if((uflg[a]==0)&&(uflg[b]==0)) {

```

```

    nm[5]=a; nm[21]=b;
    uflg[a]=1; uflg[b]=1;
    stp06();
    uflg[b]=0; uflg[a]=0;}}
}

/* Define Level 2: *
/* Set N6 & n20 *
void stp06(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
    nm[6]=a; nm[20]=b;
    uflg[a]=1; uflg[b]=1;
    stp07();
    uflg[b]=0; uflg[a]=0;}}
}

/* Set N16 & n10 *
void stp07(){
short a,b;
for(a=25;a>0;a--){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
    nm[16]=a; nm[10]=b;
    uflg[a]=1; uflg[b]=1;
    stp08();
    uflg[b]=0; uflg[a]=0;}}
}

/* Set n11=LSM-n1-n6-n16-n21 & n15 *
void stp08(){
short a,b;
a=LSM-nm[1]-nm[6]-nm[16]-nm[21];
if((0<a)&&(a<26)){
    b=LSM-nm[25]-nm[20]-nm[10]-nm[5];
    if(a+b==CC){
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[11]=a; nm[15]=b;
            uflg[a]=1; uflg[b]=1;
            stp09();
            uflg[b]=0; uflg[a]=0;}}}
}

/* Define Level 2: *
/* Set N7 & n19 *
void stp09(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
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 N9 & n17 *
void stp10(){
short a,b;
for(a=25;a>0;a--){b=CC-a;
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){stp11();}
    uflg[b]=0; uflg[a]=0;}}
}

/* Set n8=LSM-n6-n7-n9-n10 & n18 *

```

```

void stp11(){
    short a,b;
    a=LSM-nm[6]-nm[7]-nm[9]-nm[10];
    if((0<a)&&(a<26)){
        b=LSM-nm[20]-nm[19]-nm[17]-nm[16];
        if(a+b==CC){
            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){stp12();}
                uflg[b]=0; uflg[a]=0;}}}
    }
    /* Set n12=LSM-n2-n7-n17-n22 & n14 */
    void stp12(){
        short a,b;
        a=LSM-nm[2]-nm[7]-nm[17]-nm[22];
        if((0<a)&&(a<26)){
            b=LSM-nm[24]-nm[19]-nm[9]-nm[4];
            if(a+b==CC){
                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){ansprint();}
                    uflg[b]=0; uflg[a]=0;}}}
    }
    //
    /* Begin The Search 2 */
    /** Normalizing Inequality Conditions: n3<n11<n15 and n3<n23;
    ** Define Level 1: */
    /* Set N3 & n23 & n3<n23 */
    void stp21(){
        short a,b;
        for(a=1;a<13;a++){b=CC-a;
            if((uflg[a]==0)&&(uflg[b]==0)){cnt2=0;
                nm[3]=a; nm[23]=b;
                uflg[a]=1; uflg[b]=1;
                stp22();
                uflg[b]=0; uflg[a]=0;}}
    }
    /* Set N1 & n25 */
    void stp22(){
        short a,b;
        for(a=1;a<26;a++){b=CC-a;
            if((uflg[a]==0)&&(uflg[b]==0)){if(nm[3]==1){cnt2=0;}
                nm[1]=a; nm[25]=b;
                uflg[a]=1; uflg[b]=1;
                stp23();
                uflg[b]=0; uflg[a]=0;}}
    }
    /* Set N2 & n24 */
    void stp23(){
        short a,b;
        for(a=1;a<26;a++){b=CC-a;
            if((uflg[a]==0)&&(uflg[b]==0)){
                nm[2]=a; nm[24]=b;
                uflg[a]=1; uflg[b]=1;
                stp24();
                uflg[b]=0; uflg[a]=0;}}
    }
    /* Set N4 & n22 */
    void stp24(){
        short a,b;

```

```

for(a=1;a<26;a++){b=CC-a;
  if((uflg[a]==0)&&(uflg[b]==0)){
    nm[4]=a; nm[22]=b;
    uflg[a]=1; uflg[b]=1;
    stp25();
    uflg[b]=0; uflg[a]=0;}
}
}

/* Set n5=LSM-n1-n2-n3-n4 & n21 *
void stp25(){
short a,b;
a=LSM-nm[1]-nm[2]-nm[3]-nm[4];
if(0<a)&&(a<26)){
  b=LSM-nm[25]-nm[24]-nm[23]-nm[22];
  if(a+b==CC){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[5]=a; nm[21]=b;
      uflg[a]=1; uflg[b]=1;
      stp26();
      uflg[b]=0; uflg[a]=0;}}}
}

/* Define Level 2: *
/* Set N6 & n20 *
void stp26(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
  if((uflg[a]==0)&&(uflg[b]==0)){
    nm[6]=a; nm[20]=b;
    uflg[a]=1; uflg[b]=1;
    stp27();
    uflg[b]=0; uflg[a]=0;}}
}

/* Set N16 & n10 *
void stp27(){
short a,b;
for(a=25;a>0;a--) {b=CC-a;
  if((uflg[a]==0)&&(uflg[b]==0)){
    nm[16]=a; nm[10]=b;
    uflg[a]=1; uflg[b]=1;
    stp28();
    uflg[b]=0; uflg[a]=0;}}
}

/* Set n11=LSM-n1-n6-n16-n21 & n15 & n3<n11< n15 *
void stp28(){
short a,b;
a=LSM-nm[1]-nm[6]-nm[16]-nm[21];
if((nm[3]<a)&&(a<26)){
  b=LSM-nm[25]-nm[20]-nm[10]-nm[5];
  if((a<b)&&(a+b==CC)){
    if((uflg[a]==0)&&(uflg[b]==0)){
      nm[11]=a; nm[15]=b;
      uflg[a]=1; uflg[b]=1;
      stp29();
      uflg[b]=0; uflg[a]=0;}}}
}

/* Define Level 2: *
/* Set N7 & n19 *
void stp29(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
  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){stp30();}
        uflg[b]=0; uflg[a]=0;
    }
}
/* Set N9 & n17 *
void stp30(){
short a,b;
for(a=25;a>0;a--) {b=CC-a;
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){stp31();}
    uflg[b]=0; uflg[a]=0;
}
}
/* Set n8=LSM-n6-n7-n9-n10 & n18 *
void stp31(){
short a,b;
a=LSM-nm[6]-nm[7]-nm[9]-nm[10];
if((0<a)&&(a<26)){
    b=LSM-nm[20]-nm[19]-nm[17]-nm[16];
    if(a+b==CC){
        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){stp32();}
            uflg[b]=0; uflg[a]=0;}}}
}
/* Set n12=LSM-n2-n7-n17-n22 & n14 *
void stp32(){
short a,b;
a=LSM-nm[2]-nm[7]-nm[17]-nm[22];
if((0<a)&&(a<26)){
    b=LSM-nm[24]-nm[19]-nm[9]-nm[4];
    if(a+b==CC){
        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){ansprint();}
            uflg[b]=0; uflg[a]=0;}}}
}
// 
/* Print the Answers *
void ansprint(){
short n;
cnt++;
if(nm[1]==1){cnt11++;}
if(nm[3]==1){cnt13++;}
cnt2++;
if(cnt2==1){
    for(n=1;n<26;n++){nt[cnt3][n]=nm[n];}
    cntr[cnt3]=cnt;
    cnt3++; if(cnt3==5){prans(cnt3); cnt3=0;}}
}
// 
/* Print the Answers *
void prans(short x){
short l,15,m,n;
for(n=0;n<x;n++){
    printf("%15ld#",cntr[n]);
    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",
               nt[m][l5+1],nt[m][l5+2],nt[m][l5+3],nt[m][l5+4],nt[m][l5+5]);
        if(m+1<x){printf(" ");}
    }
    printf("\n");
}
//
```

次のリストは、その結果である。何しろ総数が多いので、間引きして表示してある。

#### \* 補数対称型五方陣の標準解のリスト（抜粋）：2015年版 \*

1#	41#	77#	113#	245#
1 23 5 24 12	1 22 6 24 12	1 21 7 24 12	1 20 8 24 12	1 20 9 23 12
8 15 19 17 6	8 21 16 17 3	8 17 23 11 6	7 23 16 15 4	7 24 11 18 5
22 16 13 10 4	19 11 13 15 7	22 10 13 16 4	21 9 13 17 5	22 10 13 16 4
20 9 7 11 18	23 9 10 5 18	20 15 3 9 18	22 11 10 3 19	21 8 15 2 19
14 2 21 3 25	14 2 20 4 25	14 2 19 5 25	14 2 18 6 25	14 3 17 6 25
337#				
509#				
677#				
1073#				
1553#				
1 18 10 24 12	1 17 11 24 12	1 17 12 24 11	1 16 14 23 11	1 16 15 21 12
11 23 4 21 6	7 16 21 18 3	7 23 10 21 4	4 19 20 17 5	9 22 7 24 3
19 17 13 9 7	20 22 13 4 6	20 18 13 8 6	24 18 13 8 2	18 20 13 6 8
20 5 22 3 15	23 8 5 10 19	22 5 16 3 19	21 9 6 7 22	23 2 19 4 17
14 2 16 8 25	14 2 15 9 25	15 2 14 9 25	15 3 12 10 25	14 5 11 10 25
1881#				
2417#				
2801#				
3345#				
3861#				
1 14 16 23 11	1 11 17 24 12	1 10 18 24 12	1 9 19 24 12	1 8 20 24 12
6 18 22 17 2	8 23 21 7 6	7 21 22 9 6	18 23 5 15 4	9 21 16 15 4
19 21 13 5 7	22 10 13 16 4	23 15 13 11 3	10 20 13 6 16	19 23 13 3 7
24 9 4 8 20	20 19 5 3 18	20 17 4 5 19	22 11 21 3 8	22 11 10 5 17
15 3 10 12 25	14 2 9 15 25	14 2 8 16 25	14 2 7 17 25	14 2 6 18 25
4525#				
4969#				
5829#				
6601#				
7793#				
1 8 21 24 11	1 6 22 24 12	1 5 23 24 12	1 5 24 23 12	2 23 4 25 11
12 22 19 3 9	16 19 18 9 3	15 16 19 6 9	11 22 16 9 7	8 14 20 16 7
20 10 13 16 6	11 21 13 5 15	18 22 13 4 8	20 18 13 8 6	21 17 13 9 5
17 23 7 4 14	23 17 8 7 10	17 20 7 10 11	19 17 10 4 15	19 10 6 12 18
15 2 5 18 25	14 2 4 20 25	14 2 3 21 25	14 3 2 21 25	15 1 22 3 24
14601#				
21805#				
27597#				
33193#				
37313#				
3 21 4 25 12	4 23 2 25 11	5 23 1 24 12	6 22 1 24 12	7 21 1 24 12
7 15 16 18 9	7 17 20 16 5	6 15 19 17 8	7 16 21 18 3	6 15 23 17 4
24 20 13 6 2	18 14 13 12 8	22 16 13 10 4	15 17 13 9 11	16 18 13 8 10
17 8 10 11 19	21 10 6 9 19	18 9 7 11 20	23 8 5 10 19	22 9 3 11 20
14 1 22 5 23	15 1 24 3 22	14 2 25 3 21	14 2 25 4 20	14 2 25 5 19
41081#				
44129#				
46533#				
48065#				
8 20 1 24 12	9 19 1 24 12	10 18 1 24 12	11 17 1 24 12	
3 22 17 16 7	3 22 15 20 5	4 15 21 19 6	3 18 19 20 5	
21 11 13 15 5	18 16 13 10 8	17 23 13 3 9	16 22 13 4 10	
19 10 9 4 23	21 6 11 4 23	20 7 5 11 22	21 6 7 8 23	
14 2 25 6 18	14 2 25 7 17	14 2 25 8 16	14 2 25 9 15	

[Solution Counts (n1=1/All) = 7792/48544] OK!

ずいぶんと多数あるものだ。この5万個近い総数は、近年ようやくコンピュータによって確定されたものだ。このくらい多いと、手計算では余程の「推測」が必要になる。

しかし、解が多過ぎても、扱いに何かと困ることが多い。分類や構造研究にとかく支障を来しがちだ。できることなら、少數の「基本解」を何とか見つけたいものだ。

そこで、変換法のさらなる研究が必要になる。

4. 変換法には、内部変換法と外部変換法の2種類がある。後者は、変換後の解が他の一定

の解集合に属するようになるものを言うが、前者は自分自身の属する解集合の他の解に変換するものを言う。及ぼす結果があくまで内部に限られるので、そう言う。

我々の欲するのは、内部変換法である。これには、分類と整理という副産物がある。

一般に**補数対称型五方陣**では、要素の対称配置ゆえに、下図のような行列の交換移動が可能である。

対角線の内容が変わってしまうので、全然別個の方陣になるような気がするが、定和には変わりが無く、補数対称方陣には違いない。自分自身の解集合に戻る。行列の内容がいつも同じなので、兄弟方陣に違いない。これを同じ系列の解に分類して全然差し支えない。

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
C1	C2	C3	C4	C5

#### \* 行列の交換可能性を示す概念図 \*

- (1) 縦の C2列と C4列をそっくり交換できる。
  - (2) 横の R2行と R4行をそっくり交換できる。
  - (3) 縦の C1列と C5列をそっくり交換できる。
  - (4) 横の R1行と R5行をそっくり交換できる。
  - (5) 縦の外側と内側を 2列ごと交換できる。  
(C1↔C2) & (C4↔C5)
  - (6) 横の外側と内側を 2行ごと交換できる。  
(R1↔R2) & (R4↔R5)
- この 6種類の交換移動を組み合わせてもよい。  
同種の変換を 2度繰り返すのは意味が無い。  
元に戻るだけだから。

では、全ての行列交換とその組み合せによって出来る派生方陣は、いったい幾つ可能なのだろうか。いまさら鏡映反転形や回転形は、カウントしない。その他の可能性を求めている。この際、細かく調べてみよう。

では、なぜ交換ができるのか。具体的に横の R2行と R4行を交換する場合で、調べよう。

[原形 : 補数表現]				
1	2	3	4	5
R2	6	7	8	9
11	12	13	12	11
R4	10	9	8	7
5	4	3	2	1

[交換後 : 補数表現]				
1	2	3	4	5
R4	10	9	8	7
11	12	13	12	11
R2	6	7	8	9
5	4	3	2	1

行和・列和には変わりが無い。  
5項の順序に変化はあるが、内容には、変化が無い。従って、左図で行列の定和が成り立つなら、右図でも成り立つ、と言える。

問題は、2本の主対角線だ。  
左図では、  

$$\begin{aligned} 1 + 7 + 13 + \underline{9} + \underline{1} \\ = (1 + \underline{1}) + (7 + \underline{7}) + 13 \\ = 26 + 26 + 13 = 65 \\ 5 + 9 + 13 + \underline{9} + \underline{5} \\ = (5 + 5) + (9 + \underline{9}) + 13 \\ = 26 + 26 + 13 = 65 \end{aligned}$$

右図では、  

$$\begin{aligned} 1 + \underline{9} + 13 + 9 + \underline{1} \\ = (1 + \underline{1}) + (9 + \underline{9}) + 13 \\ 5 + \underline{7} + 13 + 7 + \underline{5} \\ = (5 + \underline{5}) + (7 + \underline{7}) + 13 = 26 + 26 + 13 = 65 \end{aligned}$$

(0)	(1)	(2)
1 2 3 4 5	1 2 3 4 5	21 22 23 24 25
6 7 8 9 10	16 17 18 19 20	6 7 8 9 10
11 12 13 14 15	11 12 13 14 15	11 12 13 14 15
16 17 18 19 20	6 7 8 9 10	16 17 18 19 20
21 22 23 24 25	21 22 23 24 25	1 2 3 4 5
(3)	(4)=(1)+(3)	(5)=(2)+(3)
1 4 3 2 5	1 4 3 2 5	21 24 23 22 25
6 9 8 7 10	16 19 18 17 20	6 9 8 7 10
11 14 13 12 15	11 14 13 12 15	11 14 13 12 15
16 19 18 17 20	6 9 8 7 10	16 19 18 17 20
21 24 23 22 25	21 24 23 22 25	1 4 3 2 5

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

例えば、上図のように内側の(2-4)交換と外側の(1-5)交換を組み合わせてみよう。  
縦横2方向に分けて2次元に組み合わせてある。

(1), (3), (4)は、相互に別の形と見てよいが、他を良く見ると、反転形・回転形が幾つも見つかるではないか。そもそも(1)と(2)が反転形だ。(3)と(6)も反転形だ。(4)と(5)も、(4)と(7)も反転形だ。(4)と(8)は回転形だ。

これでは、原形から8個もの異形を作り出したとは、とても言えない。

内側(2-4)交換と外側(1-5)交換とは、同時に使えないことがわかる。

(0)	(1)	(2)
1 2 3 4 5	1 2 3 4 5	6 7 8 9 10
6 7 8 9 10	16 17 18 19 20	1 2 3 4 5
11 12 13 14 15	11 12 13 14 15	11 12 13 14 15
16 17 18 19 20	6 7 8 9 10	21 22 23 24 25
21 22 23 24 25	21 22 23 24 25	16 17 18 19 20
(3)	(4)=(1)+(3)	(5)=(2)+(3)
1 4 3 2 5	1 4 3 2 5	6 9 8 7 10
6 9 8 7 10	16 19 18 17 20	1 4 3 2 5
11 14 13 12 15	11 14 13 12 15	11 14 13 12 15
16 19 18 17 20	6 9 8 7 10	21 24 23 22 25
21 24 23 22 25	21 24 23 22 25	16 19 18 17 20
(6)	(7)=(1)+(6)	(8)=(2)+(6)
2 1 3 5 4	2 1 3 5 4	7 6 8 10 9
7 6 8 10 9	17 16 18 20 19	2 1 3 5 4
12 11 13 15 14	12 11 13 15 14	12 11 13 15 14
17 16 18 20 19	7 6 8 10 9	22 21 23 25 24
22 21 23 25 24	22 21 23 25 24	17 16 18 20 19

では、内側(2-4)交換と内外(1-2)&(4-5)交換とは、同時に使えるか。

上図を見てほしい。今度は、良さそうだ。全部が別個の解になっている。

これに(1)と(2)の合成、(3)と(6)の合成を加えると、全部で $4 \times 4 = 16$ 通りの変換法が可能だ。概念図で表すと、次に示す通りになる。

どの場合でも、主対角線の定和が成り立つことを「直視」して確かめてほしい。

#### \* 補数対称型五方陣の内部変換法システム：16タイプの概念図 \*

[原型]	0/T	1/T	2/T	3/T
1 2 3 4 5	1 2 3 4 5	6 7 8 9 10	16 17 18 19 20	
6 7 8 9 10	16 17 18 19 20	1 2 3 4 5	1 2 3 4 5	
11 12 13 14 15	11 12 13 14 15	11 12 13 14 15	11 12 13 14 15	
16 17 18 19 20	6 7 8 9 10	21 22 23 24 25	21 22 23 24 25	
21 22 23 24 25	21 22 23 24 25	16 17 18 19 20	6 7 8 9 10	
4/T	5/T	6/T	7/T	
1 4 3 2 5	1 4 3 2 5	6 9 8 7 10	16 19 18 17 20	
6 9 8 7 10	16 19 18 17 20	1 4 3 2 5	1 4 3 2 5	
11 14 13 12 15	11 14 13 12 15	11 14 13 12 15	11 14 13 12 15	
16 19 18 17 20	6 9 8 7 10	21 24 23 22 25	21 24 23 22 25	
21 24 23 22 25	21 24 23 22 25	16 19 18 17 20	6 9 8 7 10	

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

[Count = 16]

この変換法の存在によって、1個の解が常に15個の派生解を生み、16個ずつで各グループを作ることがわかった。当然こうしたグループは、 $48544 \div 16 = 3034$  個あることになる。そして、各グループから1個ずつ代表となる解を出せば、「基本解」が3034個出来るはずだ。

では、その基本解をコンピュータにどうやって見つけ出させるか？

ここは、例によってリスト出力用の不等式を与えて、選別させるしか無い。例えば、 $n_1 < n_5 < n_{21}$ ;  $n_1 < n_{25}$ ; に加えて、試しに  $n_1 < n_2 < n_4$ ;  $n_1 < n_6 < n_{16}$ ; としてみる。

#### \* 対称五方陣の基本解を求める実験 \*

1#	11#	20#	29#	62#
1 23 5 24 12	1 22 6 24 12	1 21 7 24 12	1 20 8 24 12	1 20 9 23 12
8 15 19 17 6	8 21 16 17 3	8 17 23 11 6	7 23 16 15 4	7 24 11 18 5
22 16 13 10 4	19 11 13 15 7	22 10 13 16 4	21 9 13 17 5	22 10 13 16 4
20 9 7 11 18	23 9 10 5 18	20 15 3 9 18	22 11 10 3 19	21 8 15 2 19
14 2 21 3 25	14 2 20 4 25	14 2 19 5 25	14 2 18 6 25	14 3 17 6 25
85#	128#	170#	269#	389#
1 18 10 24 12	1 17 11 24 12	1 17 12 24 11	1 16 14 23 11	1 16 15 21 12
11 23 4 21 6	7 16 21 18 3	7 23 10 21 4	4 19 20 17 5	9 22 7 24 3
19 17 13 9 7	20 22 13 4 6	20 18 13 8 6	24 18 13 8 2	18 20 13 6 8
20 5 22 3 15	23 8 5 10 19	22 5 16 3 19	21 9 6 7 22	23 2 19 4 17
14 2 16 8 25	14 2 15 9 25	15 2 14 9 25	15 3 12 10 25	14 5 11 10 25
471#	605#	701#	837#	966#
1 14 16 23 11	1 11 17 24 12	1 10 18 24 12	1 9 19 24 12	1 8 20 24 12
6 18 22 17 2	8 23 21 7 6	7 21 22 9 6	18 23 5 15 4	9 21 16 15 4
19 21 13 5 7	22 10 13 16 4	23 15 13 11 3	10 20 13 6 16	19 23 13 3 7
24 9 4 8 20	20 19 5 3 18	20 17 4 5 19	22 11 21 3 8	22 11 10 5 17
15 3 10 12 25	14 2 9 15 25	14 2 8 16 25	14 2 7 17 25	14 2 6 18 25
1132#	1243#	1458#	1651#	1949#
1 8 21 24 11	1 6 22 24 12	1 5 23 24 12	1 5 24 23 12	2 23 4 25 11
12 22 19 3 9	16 19 18 9 3	15 16 19 6 9	11 22 16 9 7	8 14 20 16 7
20 10 13 16 6	11 21 13 5 15	18 22 13 4 8	20 18 13 8 6	21 17 13 9 5
17 23 7 4 14	23 17 8 7 10	17 20 7 10 11	19 17 10 4 15	19 10 6 12 18
15 2 5 18 25	14 2 4 20 25	14 2 3 21 25	14 3 2 21 25	15 1 22 3 24
3651#	5358#	6569#	7560#	8132#
3 21 4 25 12	4 23 2 25 11	5 23 1 24 12	6 22 1 24 12	7 21 1 24 12
7 15 16 18 9	7 17 20 16 5	6 15 19 17 8	7 16 21 18 3	8 11 23 17 6
24 20 13 6 2	18 14 13 12 8	22 16 13 10 4	15 17 13 9 11	16 22 13 4 10
17 8 10 11 19	21 10 6 9 19	18 9 7 11 20	23 8 5 10 19	20 9 3 15 18
14 1 22 5 23	15 1 24 3 22	14 2 25 3 21	14 2 25 4 20	14 2 25 5 19
8498#	8691#	8781#	8836#	
8 20 1 24 12	9 20 1 23 12	10 18 1 24 12	11 17 1 24 12	
10 19 9 23 4	11 18 7 24 5	11 20 9 22 3	16 21 3 19 6	
11 21 13 5 15	10 22 13 4 16	7 21 13 5 19	4 18 13 8 22	
22 3 17 7 16	21 2 19 8 15	23 4 17 6 15	20 7 23 5 10	
14 2 25 6 18	14 3 25 6 17	14 2 25 8 16	14 2 25 9 15	

[Solution Counts (n1=1/All) = 1948/8846] OK!

$n_1 = 1$  の解が 1948 個出て来たのは成果だが、全体としては使えない。総数が 3034 個にならない。 $n_1$ の値が 2 以上になったときに、扱いが少々困る問題が生じる。1 の来る場所がバラバラで、整理し難い。別のタイプの「基本形」を仮定する必要を感じる。

いろいろと試した結果、基礎になる標準解のリストのスタイルを改め、 $n_3 = 1$  から始めて数え出すのが良いとわかった。そして、基本解のための不等式は、 $n_3 < n_{23}; n_3 < n_{11} < n_{15}$ ; の他に  $n_3 < n_8 < n_{18}; n_{11} < n_{12} < n_{14}$ ; と立てて、 $n_1$  に関しては全く条件を付けない。

こうすると、全ての反転・回転・行列交換が封じられて、基本解だけが求まるのである。以下がそのリストだ。標準解と基本解のリストを連続して示そう。

\* 対称五方陣の標準解の新しいリスト :  $n_3 = 1$  から始めて数え切るスタイル : 2015 年版 \*

1#	53#	117#	165#	269#
3 18 1 24 19	4 17 1 23 20	5 16 1 24 19	6 15 1 19 24	7 10 1 24 23
21 15 9 16 4	21 12 19 11 2	22 12 17 11 3	22 9 21 10 3	21 15 17 8 4
12 20 13 6 14	10 18 13 8 16	8 20 13 6 18	12 18 13 8 14	12 20 13 6 14
22 10 17 11 5	24 15 7 14 5	23 15 9 14 4	23 16 5 17 4	22 18 9 11 5
7 2 25 8 23	6 3 25 9 22	7 2 25 10 21	2 7 25 11 20	3 2 25 16 19
469#				
761#				
981#				
1281#				
1633#				
8 10 1 24 22	9 10 1 21 24	10 7 1 23 24	11 6 1 23 24	12 6 1 22 24
19 20 9 14 3	19 22 15 6 3	20 15 17 8 5	19 16 17 8 5	19 18 15 10 3
11 21 13 5 15	12 8 13 18 14	12 22 13 4 14	12 22 13 4 14	9 21 13 5 17
23 12 17 6 7	23 20 11 4 7	21 18 9 11 6	21 18 9 10 7	23 16 11 8 7
4 2 25 16 18	2 5 25 16 17	2 3 25 19 16	2 3 25 20 15	2 4 25 20 14
2077#				
2405#				
2845#				
3205#				
3617#				
14 5 1 23 22	15 3 1 24 22	16 3 1 21 24	17 3 1 20 24	18 2 1 21 23
17 15 19 8 6	16 19 17 5 8	15 17 19 8 6	15 18 21 4 7	15 16 19 6 9
10 24 13 2 16	12 20 13 6 14	12 22 13 4 14	12 16 13 10 14	12 22 13 4 14
20 18 7 11 9	18 21 9 7 10	20 18 7 9 11	19 22 5 8 11	17 20 7 10 11
4 3 25 21 12	4 2 25 23 11	2 5 25 23 10	2 6 25 23 9	3 5 25 24 8
4025#				
4849#				
5433#				
6081#				
6737#				
19 2 1 20 23	20 2 1 23 19	21 2 1 18 23	22 2 1 19 21	23 2 1 18 21
15 21 9 12 8	14 21 11 9 10	11 19 17 12 6	14 23 11 9 8	12 22 15 9 7
10 22 13 4 16	8 22 13 4 18	10 22 13 4 16	6 16 13 10 20	6 16 13 10 20
18 14 17 5 11	16 17 15 5 12	20 14 9 7 15	18 17 15 3 12	19 17 11 4 14
3 6 25 24 7	7 3 25 24 6	3 8 25 24 5	5 7 25 24 4	5 8 25 24 3
7681#				
8689#				
17309#				
23157#				
29177#				
24 3 1 15 22	3 16 2 25 19	1 20 3 22 19	1 18 4 23 19	1 20 5 22 17
7 21 17 14 6	21 17 8 15 4	21 17 15 10 2	21 15 10 17 2	23 11 19 10 2
10 18 13 8 16	12 20 13 6 14	12 8 13 18 14	12 20 13 6 14	8 14 13 12 18
20 12 9 5 19	22 11 18 9 5	24 16 11 9 5	24 9 16 11 5	24 16 7 15 3
4 11 25 23 2	7 1 24 10 23	7 4 23 6 25	7 3 22 8 25	9 4 21 6 25
33689#				
38109#				
41085#				
43829#				
46033#				
1 15 6 24 19	1 17 7 22 18	1 15 8 20 21	1 15 9 22 18	1 14 10 22 18
22 17 18 5 3	20 23 5 15 2	23 19 4 17 2	21 24 7 10 3	21 17 6 19 2
12 10 13 16 14	12 10 13 16 14	12 16 13 10 14	12 6 13 20 14	11 23 13 3 15
23 21 8 9 4	24 11 21 3 6	24 9 22 7 3	23 16 19 2 5	24 7 20 9 5
7 2 20 11 25	8 4 19 9 25	5 6 18 11 25	8 4 17 11 25	8 4 16 12 25
47585#				
1 16 11 18 19				
21 17 3 22 2				
12 20 13 6 14				
24 4 23 9 5				
7 8 15 10 25				

[Solution Counts (n3=1/All) = 8688/48544] OK!

\* 対称五方陣の基本解の新しいリスト : n3 = 1 から始めて数え切るスタイル : 2015 年版 \*

1#	3#	11#	15#	24#
3 20 1 22 19	4 17 1 23 20	5 16 1 24 19	6 17 1 19 22	7 18 1 24 15
21 18 9 15 2	24 18 7 11 5	22 20 9 11 3	24 18 3 15 5	20 21 9 12 3
10 12 13 14 16	10 12 13 14 16	8 12 13 14 18	10 12 13 14 16	4 10 13 16 22
24 11 17 8 5	21 15 19 8 2	23 15 17 6 4	21 11 23 8 2	23 14 17 5 6
7 4 25 6 23	6 3 25 9 22	7 2 25 10 21	4 7 25 9 20	11 2 25 8 19
39#	61#	83#	125#	168#
8 15 1 24 17	9 10 1 22 23	10 9 1 21 24	11 8 1 22 23	12 9 1 19 24
23 20 7 10 5	24 19 11 6 5	23 20 11 7 4	20 24 7 9 5	22 21 11 8 3
4 12 13 14 22	8 12 13 14 18	8 12 13 14 18	10 12 13 14 16	6 10 13 16 20
21 16 19 6 3	21 20 15 7 2	22 19 15 6 3	21 17 19 2 6	23 18 15 5 4
9 2 25 11 18	3 4 25 16 17	2 5 25 17 16	3 4 25 18 15	2 7 25 17 14
224#	270#	328#	367#	417#
14 7 1 22 21	15 8 1 17 24	16 7 1 21 20	17 5 1 20 22	18 3 1 19 24
23 20 9 2 11	23 20 7 10 5	24 18 9 3 11	19 24 3 8 11	20 21 9 4 11
8 10 13 16 18	4 12 13 14 22	4 12 13 14 22	10 12 13 14 16	10 12 13 14 16
15 24 17 6 3	21 16 19 6 3	15 23 17 8 2	15 18 23 2 7	15 22 17 5 6
5 4 25 19 12	2 9 25 18 11	6 5 25 19 10	4 6 25 21 9	2 7 25 23 8
455#	567#	632#	720#	815#
19 4 1 18 23	20 5 1 15 24	21 3 1 16 24	22 3 1 15 24	23 4 1 16 21
16 24 11 9 5	23 17 7 4 14	17 22 11 8 7	19 20 9 5 12	14 24 11 7 9
6 12 13 14 20	8 10 13 16 18	6 12 13 14 20	8 10 13 16 18	6 8 13 18 20
21 17 15 2 10	12 22 19 9 3	19 18 15 4 9	14 21 17 6 7	17 19 15 2 12
3 8 25 22 7	2 11 25 21 6	2 10 25 23 5	2 11 25 23 4	5 10 25 22 3
935#	1087#	1092#	1106#	1116#
24 3 1 15 22	3 19 2 21 20	4 15 2 25 19	5 17 2 25 16	6 14 2 21 22
17 21 7 6 14	22 18 10 14 1	23 20 8 9 5	20 22 8 12 3	25 18 10 9 3
8 10 13 16 18	9 11 13 15 17	10 12 13 14 16	7 11 13 15 19	7 11 13 15 19
12 20 19 5 9	25 12 16 8 4	21 17 18 6 3	23 14 18 4 6	23 17 16 8 1
4 11 25 23 2	6 5 24 7 23	7 1 24 11 22	10 1 24 9 21	4 5 24 12 20
1140#	1160#	1192#	1226#	1285#
7 15 2 21 20	8 11 2 21 23	9 12 2 23 19	10 8 2 20 25	11 7 2 20 25
25 17 12 8 3	25 20 4 9 7	20 25 4 11 5	22 21 12 7 3	21 23 8 9 4
4 10 13 16 22	10 12 13 14 16	8 10 13 16 18	9 11 13 15 17	10 12 13 14 16
23 18 14 9 1	19 17 22 6 1	21 15 22 1 6	23 19 14 5 4	22 17 18 3 5
6 5 24 11 19	3 5 24 15 18	7 3 24 14 17	1 6 24 18 16	1 6 24 19 15
1328#	1341#	1375#	1418#	1462#
12 9 2 22 20	14 5 2 19 25	15 5 2 20 23	16 11 2 14 22	17 4 2 23 19
19 25 8 10 3	23 20 10 4 8	17 25 12 7 4	25 18 6 7 9	20 21 8 1 15
5 11 13 15 21	9 11 13 15 17	8 10 13 16 18	3 5 13 21 23	10 12 13 14 16
23 16 18 1 7	18 22 16 6 3	22 19 14 1 9	17 19 20 8 1	11 25 18 5 6
6 4 24 17 14	1 7 24 21 12	3 6 24 21 11	4 12 24 15 10	7 3 24 22 9
1505#	1547#	1611#	1651#	1718#
18 4 2 16 25	19 3 2 21 20	20 7 2 11 25	21 3 2 17 22	22 6 2 10 25
23 19 6 5 12	15 25 12 4 9	17 21 8 16 3	15 25 8 10 7	14 23 8 15 5
9 11 13 15 17	8 10 13 16 18	4 12 13 14 22	6 12 13 14 20	7 9 13 17 19
14 21 20 7 3	17 22 14 1 11	23 10 18 5 9	19 16 18 1 11	21 11 18 3 12
1 10 24 22 8	6 5 24 23 7	1 15 24 19 6	4 9 24 23 5	1 16 24 20 4
1771#	1842#	2009#	2012#	2022#
23 4 2 11 25	25 3 2 17 18	2 19 3 21 20	4 15 3 25 18	5 18 3 20 19
16 21 12 9 7	15 21 4 6 19	22 18 9 15 1	24 20 5 9 7	24 17 11 12 1
6 8 13 18 20	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16	4 10 13 16 22
19 17 14 5 10	7 20 22 5 11	25 11 17 8 4	19 17 21 6 2	25 14 15 9 2
1 15 24 22 3	8 9 24 23 1	6 5 23 7 24	8 1 23 11 22	7 6 23 8 21

2030#	2047#	2063#	2072#	2082#
6 11 3 21 24	7 12 3 22 21	8 15 3 25 14	9 14 3 21 18	10 9 3 24 19
25 19 9 8 4	20 24 9 11 1	24 20 5 7 9	19 24 11 10 1	25 20 5 4 11
10 12 13 14 16	8 10 13 16 18	4 10 13 16 22	4 6 13 20 22	8 12 13 14 18
22 18 17 7 1	25 15 17 2 6	17 19 21 6 2	25 16 15 2 7	15 22 21 6 1
2 5 23 15 20	5 4 23 14 19	12 1 23 11 18	8 5 23 12 17	7 2 23 17 16
2087#	2091#	2106#	2122#	2140#
11 8 3 22 21	12 11 3 21 18	14 7 3 17 24	15 4 3 18 25	16 9 3 12 25
19 25 9 10 2	25 19 9 2 10	25 18 11 5 6	20 24 5 9 7	24 19 5 11 6
6 12 13 14 20	4 6 13 20 22	4 10 13 16 22	10 12 13 14 16	4 8 13 18 22
24 16 17 1 7	16 24 17 7 1	20 21 15 8 1	19 17 21 2 6	20 15 21 7 2
5 4 23 18 15	8 5 23 15 14	2 9 23 19 12	1 8 23 22 11	1 14 23 17 10
2156#	2183#	2214#	2235#	2249#
17 4 3 16 25	18 5 3 15 24	19 4 3 18 21	20 1 3 17 24	21 1 3 18 22
19 24 5 11 6	17 25 7 12 4	24 16 9 1 15	19 21 11 4 10	19 20 9 2 15
8 12 13 14 18	6 10 13 16 20	6 12 13 14 20	8 12 13 14 18	10 12 13 14 16
20 15 21 2 7	22 14 19 1 9	11 25 17 10 2	16 22 15 5 7	11 24 17 6 7
1 10 23 22 9	2 11 23 21 8	5 8 23 22 7	2 9 23 25 6	4 8 23 25 5
2292#	2327#	2376#	2412#	2414#
22 2 3 17 21	24 1 3 17 20	25 2 3 14 21	1 20 4 21 19	2 17 4 23 19
14 25 11 7 8	16 22 7 5 15	18 19 9 4 15	23 17 8 15 2	25 18 6 11 5
6 10 13 16 20	8 12 13 14 18	6 10 13 16 20	10 12 13 14 16	10 12 13 14 16
18 19 15 1 12	11 21 19 4 10	11 22 17 7 8	24 11 18 9 3	21 15 20 8 1
5 9 23 24 4	6 9 23 25 2	5 12 23 24 1	7 5 22 6 25	7 3 22 9 24
2426#	2436#	2442#	2449#	2458#
3 17 4 21 20	5 16 4 25 15	6 16 4 24 15	7 16 4 23 15	8 20 4 24 9
24 18 10 12 1	19 23 12 9 2	25 17 12 3 8	18 25 6 14 2	12 25 10 15 3
7 11 13 15 19	6 8 13 18 20	5 7 13 19 21	5 9 13 17 21	5 7 13 19 21
25 14 16 8 2	24 17 14 3 7	18 23 14 9 1	24 12 20 1 8	23 11 16 1 14
6 5 22 9 23	11 1 22 10 21	11 2 22 10 20	11 3 22 10 19	17 2 22 6 18
2459#	2470#	2473#	2489#	2516#
9 8 4 25 19	10 18 4 9 24	11 6 4 19 25	12 5 4 19 25	14 5 4 17 25
24 21 6 3 11	25 12 6 19 3	21 23 10 9 2	20 24 10 8 3	20 24 8 10 3
10 12 13 14 16	5 11 13 15 21	8 12 13 14 18	9 11 13 15 17	7 11 13 15 19
15 23 20 5 2	23 7 20 14 1	24 17 16 3 5	23 18 16 2 6	23 16 18 2 6
7 1 22 18 17	2 17 22 8 16	1 7 22 20 15	1 7 22 21 14	1 9 22 21 12
2526#	2539#	2560#	2573#	2584#
15 5 4 24 17	16 2 4 18 25	17 3 4 16 25	18 15 4 16 12	19 1 4 17 24
19 25 6 3 12	19 23 12 5 6	24 19 6 5 11	25 9 6 2 23	21 20 10 3 11
8 10 13 16 18	9 11 13 15 17	8 12 13 14 18	5 7 13 19 21	8 12 13 14 18
14 23 20 1 7	20 21 14 3 7	15 21 20 7 2	3 24 20 17 1	15 23 16 6 5
9 2 22 21 11	1 8 22 24 10	1 10 22 23 9	14 10 22 11 8	2 9 22 25 7
2605#	2627#	2656#	2693#	2709#
20 1 4 17 23	21 1 4 15 24	23 1 4 19 18	24 3 4 9 25	25 2 4 11 23
19 21 12 2 11	17 23 10 8 7	17 21 10 2 15	15 21 12 10 7	18 19 10 6 12
8 10 13 16 18	6 12 13 14 20	6 12 13 14 20	6 8 13 18 20	5 9 13 17 21
15 24 14 5 7	19 18 16 3 9	11 24 16 5 9	19 16 14 5 11	14 20 16 7 8
3 9 22 25 6	2 11 22 25 5	8 7 22 25 3	1 17 22 23 2	3 15 22 24 1
2747#	2750#	2762#	2767#	2770#
1 20 5 24 15	2 17 5 23 18	3 16 5 22 19	4 17 5 25 14	7 8 5 22 23
22 19 9 12 3	20 22 7 15 1	25 18 9 11 2	19 23 11 10 2	20 24 11 9 1
8 10 13 16 18	10 12 13 14 16	6 12 13 14 20	6 8 13 18 20	10 12 13 14 16
23 14 17 7 4	25 11 19 4 6	24 15 17 8 1	24 16 15 3 7	25 17 15 2 6
11 2 21 6 25	8 3 21 9 24	7 4 21 10 23	12 1 21 9 22	3 4 21 18 19

2773#	2777#	2779#	2787#	2792#
9 6 5 22 23	11 22 5 18 9	12 19 5 4 25	14 19 5 3 24	15 16 5 22 7
19 25 11 8 2	25 3 7 6 24	20 11 9 23 2	25 4 11 17 8	23 8 9 1 24
10 12 13 14 16	10 12 13 14 16	8 10 13 16 18	6 10 13 16 20	6 12 13 14 20
24 18 15 1 7	2 20 19 23 1	24 3 17 15 6	18 9 15 22 1	2 25 17 18 3
3 4 21 20 17	17 8 21 4 15	1 22 21 7 14	2 23 21 7 12	19 4 21 10 11
2795#	2799#	2810#	2816#	2826#
16 19 5 1 24	17 2 5 16 25	18 15 5 4 23	19 3 5 14 24	20 8 5 9 23
22 3 11 20 9	19 23 11 8 4	14 17 7 25 2	16 25 11 9 4	7 24 11 22 1
8 12 13 14 18	6 12 13 14 20	6 10 13 16 20	6 8 13 18 20	10 12 13 14 16
17 6 15 23 4	22 18 15 3 7	24 1 19 9 12	22 17 15 1 10	25 4 15 2 19
2 25 21 7 10	1 10 21 24 9	3 22 21 11 8	2 12 21 23 7	3 17 21 18 6
2830#	2843#	2859#	2869#	2881#
22 2 5 19 17	23 8 5 4 25	24 1 5 12 23	25 2 5 10 23	3 15 6 25 16
16 23 11 1 14	24 7 9 10 15	17 22 7 8 11	19 17 11 4 14	21 22 12 8 2
6 8 13 18 20	6 12 13 14 20	6 10 13 16 20	6 8 13 18 20	7 9 13 17 19
12 25 15 3 10	11 16 17 19 2	15 18 19 4 9	12 22 15 9 7	24 18 14 4 5
9 7 21 24 4	1 22 21 18 3	3 14 21 25 2	3 16 21 24 1	10 1 20 11 23
2883#	2886#	2890#	2896#	2900#
4 11 6 25 19	10 21 6 4 24	11 19 6 4 25	12 18 6 24 5	14 2 6 22 21
21 24 8 9 3	25 3 12 18 7	21 9 10 23 2	22 9 10 1 23	19 25 10 3 8
10 12 13 14 16	9 11 13 15 17	8 12 13 14 18	7 11 13 15 19	9 11 13 15 17
23 17 18 2 5	19 8 14 23 1	24 3 16 17 5	3 25 16 17 4	18 23 16 1 7
7 1 20 15 22	2 22 20 5 16	1 22 20 7 15	21 2 20 8 14	5 4 20 24 12
2902#	2910#	2926#	2936#	2942#
15 1 6 19 24	21 2 6 22 14	22 8 6 10 19	23 12 6 2 22	24 1 6 12 22
21 23 8 4 9	16 23 8 1 17	24 5 12 1 23	10 17 8 25 5	21 16 8 3 17
10 12 13 14 16	7 11 13 15 19	9 11 13 15 17	7 11 13 15 19	7 11 13 15 19
17 22 18 3 5	9 25 18 3 10	3 25 14 21 2	21 1 18 9 16	9 23 18 10 5
2 7 20 25 11	12 4 20 24 5	7 16 20 18 4	4 24 20 14 3	4 14 20 25 2
2950#	2966#	2968#	2970#	2974#
25 3 6 7 24	3 24 7 22 9	4 25 7 9 20	9 20 7 4 25	11 4 7 23 20
21 11 12 4 17	12 21 11 20 1	21 8 11 23 2	21 8 11 23 2	24 21 9 1 10
8 10 13 16 18	8 10 13 16 18	10 12 13 14 16	10 12 13 14 16	8 12 13 14 18
9 22 14 15 5	25 6 15 5 14	24 3 15 18 5	24 3 15 18 5	16 25 17 5 2
2 19 20 23 1	17 4 19 2 23	6 17 19 1 22	1 22 19 6 17	6 3 19 22 15
2975#	2977#	2981#	2983#	2991#
14 22 7 1 21	21 8 7 4 25	22 5 7 11 20	24 3 7 6 25	25 2 7 8 23
23 2 9 20 11	9 20 11 23 2	24 8 9 1 23	9 22 11 18 5	21 11 9 4 20
8 10 13 16 18	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16
15 6 17 24 3	24 3 15 6 17	3 25 17 18 2	21 8 15 4 17	6 22 17 15 5
5 25 19 4 12	1 22 19 18 5	6 15 19 21 4	1 20 19 23 2	3 18 19 24 1
2997#	2999#	3003#	3005#	3011#
3 25 8 22 7	12 2 8 22 21	3 24 9 22 7	4 8 9 25 19	5 7 9 20 24
10 20 12 21 2	20 25 10 3 7	25 5 11 6 18	21 24 11 6 3	23 22 11 8 1
9 11 13 15 17	9 11 13 15 17	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16
24 5 14 6 16	19 23 16 1 6	8 20 15 21 1	23 20 15 2 5	25 18 15 4 3
19 4 18 1 23	5 4 18 24 14	19 4 17 2 23	7 1 17 18 22	2 6 17 19 21
3013#	3019#	3021#	3027#	3029#
8 4 9 21 23	20 7 9 5 24	21 4 9 8 23	22 6 9 3 25	25 3 9 4 24
20 25 11 7 2	8 22 11 23 1	7 25 11 20 2	24 5 11 7 18	8 21 11 19 6
10 12 13 14 16	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16
24 19 15 1 6	25 3 15 4 18	24 6 15 1 19	8 19 15 21 2	20 7 15 5 18
3 5 17 22 18	2 21 17 19 6	3 18 17 22 5	1 23 17 20 4	2 22 17 23 1

[Solution Counts (n3=1/All) = 1086/3034] OK!

\*\* Calculated and Listed by Kanji Setsuda on  
Jan. 25, 2015 with Mac OSX 10.10.1 & Xcode 6.1.1 \*\*

$n_3 = 1$  のタイプの基本解が 1086 個、それ以外の解が 1948 個あるとわかった。

ただし、このリストの欠点は、 $n_1 = 1$  のタイプの基本解が幾つあるのかがわかりにくいこと。前のリストでは、1948 個あるとわかったのに、とてもそんなにあるようには見えない。

何故か。解の形が反転・回転してしまっているのである。行列の移動も起きている。

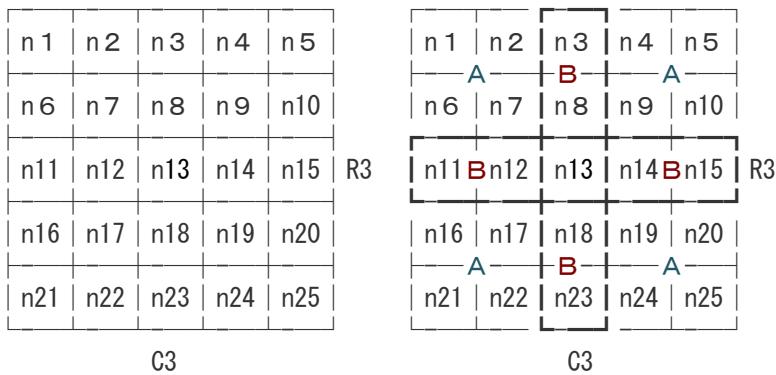
もし、何種類かの変換法システムを仮定すれば、 $n_1 = 1$  のタイプに書き変わる基本解は、実際には多数ある。いつでもホーム・ポジションに戻せるのだ。結局のところ  $n_3 = 1$  以外のタイプの 1948 個は、全てが  $n_1 = 1$  のタイプに変換することができるとわかった。

奇数次の魔方陣では、いつもこの辺がわかりにくい。しばしば議論になるところだ。

3 次の場合は、 $n_1 = 1$  のタイプは無い。中央の行列に 1 が来る形が、基本なのだ。

しかし、どうしてそうなるのか。何故に  $n_3$  や  $n_8$  や  $n_{18}$  や  $n_{11}$  や  $n_{12}$  や  $n_{14}$  など、中央の行列でのみ不等式を立てるとうまく行くのか？

## 5. ここで奇数次の補数対称型方陣特有の「排他的領域」のことを少し考えておこう。



第4分節のどの変換でも、横 R3 行 {n11, n12, n13, n14, n15} と縦 C3 列 {n3, n8, n13, n18, n23} の構成要素は、全く変わっていないことにお気づきだろうか。方陣を時計回りに、あるいは反時計回りに 90 度回することで、縦 C3 と横 R3 を交換することはできるが、行列の構成内容を他の場所のものと自由に取り換えることはできない。

縦 C3 と横 R3 の構成要素の移動は、その中だけに限られる。今まで考えたような行列の簡単な交換などでは、他の場所に自由に移すことはできないのだ。逆に他の場所からここへ移すこともできない。

他方、上図の領域 A {n1, n2, n6, n7}, {n4, n5, n9, n10}, {n16, n17, n21, n22}, {n19, n20, n24, n25} にあったものは、この領域内でならば自由に移動が可能である。従って 1 が  $n_1$  以外にあっても、この A 内にあるなら、いつでも 1 をホームポジションの  $n_1$  に戻すことができる。つまり、この場合は  $n_1 = 1$  (即ち  $n_{25}=25$ ) の解を「標準形」として、変換法を考えることができる。

ところが、領域 B {n3, n8}, {n18, n23}, {n11, n12}, {n14, n15} にある 1 は、この中でしか動けない。従って、この場合のホームポジションを  $n_1$  に仮定することはできない。

かわりに  $n_3$  をホームポジションとして仮定する。そして、 $n_3 = 1$  (即ち  $n_{23}=25$ ) の解を「標準形」として、変換法を考えるのである。

変換法を考える時に、こうした 2 種類の「標準形」がどうしても必要になってしまう。基本解でも標準解でも、同じである。これは、構造上の要請である。

試みに、2 つのタイプ別に基本解を求め、リストの上で合成してみた。下表が、その結果である。大幅に間引きしてあるが、特徴は何とかつかめよう。

\* 対称五方陣の基本解 3034 個 :  $n_1 = 1$  &  $n_3 = 1$  のタイプ別に求めた \*

1#	11#	20#	29#	62#
1 23 5 24 12	1 22 6 24 12	1 21 7 24 12	1 20 8 24 12	1 20 9 23 12
8 15 19 17 6	8 21 16 17 3	8 17 23 11 6	7 23 16 15 4	7 24 11 18 5
22 16 13 10 4	19 11 13 15 7	22 10 13 16 4	21 9 13 17 5	22 10 13 16 4
20 9 7 11 18	23 9 10 5 18	20 15 3 9 18	22 11 10 3 19	21 8 15 2 19
14 2 21 3 25	14 2 20 4 25	14 2 19 5 25	14 2 18 6 25	14 3 17 6 25
85#	128#	170#	269#	389#
1 18 10 24 12	1 17 11 24 12	1 17 12 24 11	1 16 14 23 11	1 16 15 21 12
11 23 4 21 6	7 16 21 18 3	7 23 10 21 4	4 19 20 17 5	9 22 7 24 3
19 17 13 9 7	20 22 13 4 6	20 18 13 8 6	24 18 13 8 2	18 20 13 6 8
20 5 22 3 15	23 8 5 10 19	22 5 16 3 19	21 9 6 7 22	23 2 19 4 17
14 2 16 8 25	14 2 15 9 25	15 2 14 9 25	15 3 12 10 25	14 5 11 10 25
471#	605#	701#	837#	966#
1 14 16 23 11	1 11 17 24 12	1 10 18 24 12	1 9 19 24 12	1 8 20 24 12
6 18 22 17 2	8 23 21 7 6	7 21 22 9 6	18 23 5 15 4	9 21 16 15 4
19 21 13 5 7	22 10 13 16 4	23 15 13 11 3	10 20 13 6 16	19 23 13 3 7
24 9 4 8 20	20 19 5 3 18	20 17 4 5 19	22 11 21 3 8	22 11 10 5 17
15 3 10 12 25	14 2 9 15 25	14 2 8 16 25	14 2 7 17 25	14 2 6 18 25
1132#	1243#	1458#	1651#	
1 8 21 24 11	1 6 22 24 12	1 5 23 24 12	1 5 24 23 12	
12 22 19 3 9	16 19 18 9 3	15 16 19 6 9	11 22 16 9 7	
20 10 13 16 6	11 21 13 5 15	18 22 13 4 8	20 18 13 8 6	
17 23 7 4 14	23 17 8 7 10	17 20 7 10 11	19 17 10 4 15	
15 2 5 18 25	14 2 4 20 25	14 2 3 21 25	14 3 2 21 25	
1949#	1951#	1959#	1963#	1972#
3 20 1 22 19	4 17 1 23 20	5 16 1 24 19	6 17 1 19 22	7 18 1 24 15
21 18 9 15 2	24 18 7 11 5	22 20 9 11 3	24 18 3 15 5	20 21 9 12 3
10 12 13 14 16	10 12 13 14 16	8 12 13 14 18	10 12 13 14 16	4 10 13 16 22
24 11 17 8 5	21 15 19 8 2	23 15 17 6 4	21 11 23 8 2	23 14 17 5 6
7 4 25 6 23	6 3 25 9 22	7 2 25 10 21	4 7 25 9 20	11 2 25 8 19
1987#	2009#	2031#	2073#	2116#
8 15 1 24 17	9 10 1 22 23	10 9 1 21 24	11 8 1 22 23	12 9 1 19 24
23 20 7 10 5	24 19 11 6 5	23 20 11 7 4	20 24 7 9 5	22 21 11 8 3
4 12 13 14 22	8 12 13 14 18	8 12 13 14 18	10 12 13 14 16	6 10 13 16 20
21 16 19 6 3	21 20 15 7 2	22 19 15 6 3	21 17 19 2 6	23 18 15 5 4
9 2 25 11 18	3 4 25 16 17	2 5 25 17 16	3 4 25 18 15	2 7 25 17 14
2172#	2218#	2276#	2315#	2365#
14 7 1 22 21	15 8 1 17 24	16 7 1 21 20	17 5 1 20 22	18 3 1 19 24
23 20 9 2 11	23 20 7 10 5	24 18 9 3 11	19 24 3 8 11	20 21 9 4 11
8 10 13 16 18	4 12 13 14 22	4 12 13 14 22	10 12 13 14 16	10 12 13 14 16
15 24 17 6 3	21 16 19 6 3	15 23 17 8 2	15 18 23 2 7	15 22 17 5 6
5 4 25 19 12	2 9 25 18 11	6 5 25 19 10	4 6 25 21 9	2 7 25 23 8
2403#	2515#	2580#	2668#	2763#
19 4 1 18 23	20 5 1 15 24	21 3 1 16 24	22 3 1 15 24	23 4 1 16 21
16 24 11 9 5	23 17 7 4 14	17 22 11 8 7	19 20 9 5 12	14 24 11 7 9
6 12 13 14 20	8 10 13 16 18	6 12 13 14 20	8 10 13 16 18	6 8 13 18 20
21 17 15 2 10	12 22 19 9 3	19 18 15 4 9	14 21 17 6 7	17 19 15 2 12
3 8 25 22 7	2 11 25 21 6	2 10 25 23 5	2 11 25 23 4	5 10 25 22 3
2883#				
24 3 1 15 22				
17 21 7 6 14				
8 10 13 16 18				
12 20 19 5 9				
4 11 25 23 2				

[Solution Counts (n1=1/n3=1/All) = 1948/1086/3034] OK!

\*\* Calculated and Listed by Kanji Setsuda on  
Jan. 25, 2015 with Mac OSX 10.10.1 & Xcode 6.1.1 \*\*

うまく出来たように見えるが、こうした「ダブル・スタンダード」は数学的に決して好ましいことではない。違った方法で数えたものを、後で無造作に加えてまとめるのは無謀だ。2つの計数方法の間で必ずしも均質性が保証されているとは限らず、和に意味が無い。特に変換法に利用しようとする場合に、結果に混乱が生じる心配がある。

$n3 < n23; n3 < n11 < n15; n3 < n8 < n18; n11 < n12 < n14;$  がリスト出力用の不等式として最適であり、しかも1つの基準にのみ従うとしたら、 $n3 = 1$ をスタートにして中央の行列から決めて行くしかない。前の14~16頁で求めた基本解のリストの方を、正式に採用する。

前のリストに戻って、実際に数値1が現れる領域を調べ直し計測して見たら、領域Aに現れた解が1948個、1が領域Bに現れた解が1086個あった。

なぜ中央の行列が重要なのか。理由として、次のようなものが考えられる。

- (1) 奇数次の対称方陣では、平面でも立体でも、概ね中央の行列／面が支配的である。
- (2) 行列の交換規則がどれも中央の行列を対象としていないために、ここに大小判断の不等式を置くと、公平性が保たれる。行列の交換可能性をうまく封じることができる。
- (3) 行列の交換は、どれも対称移動だ。その対称軸が中央の行列の位置にある。従って、交換に注文を付けたい時には、対称軸上で条件を考えるのが良い。

## 6. 基本解を3034個としたが、その検証作業をどうするか。

ここは、幾つかの変換法を用いて、その3034個から全体の48544個を実際に再構成して見せなければならない。さもなければ、声高に「基本解」とは言いづらい。

### \* 補数対称型五方陣の変換法システム：16タイプの概念図（再録） \*

[原型]	0/T	1/T	2/T	3/T
1 2 3 4 5	1 2 3 4 5	6 7 8 9 10	16 17 18 19 20	
6 7 8 9 10	16 17 18 19 20	1 2 3 4 5	1 2 3 4 5	
11 12 13 14 15	11 12 13 14 15	11 12 13 14 15	11 12 13 14 15	
16 17 18 19 20	6 7 8 9 10	21 22 23 24 25	21 22 23 24 25	
21 22 23 24 25	21 22 23 24 25	16 17 18 19 20	6 7 8 9 10	
4/T	5/T	6/T	7/T	
1 4 3 2 5	1 4 3 2 5	6 9 8 7 10	16 19 18 17 20	
6 9 8 7 10	16 19 18 17 20	1 4 3 2 5	1 4 3 2 5	
11 14 13 12 15	11 14 13 12 15	11 14 13 12 15	11 14 13 12 15	
16 19 18 17 20	6 9 8 7 10	21 24 23 22 25	21 24 23 22 25	
21 24 23 22 25	21 24 23 22 25	16 19 18 17 20	6 9 8 7 10	
8/T	9/T	10/T	11/T	
2 1 3 5 4	2 1 3 5 4	7 6 8 10 9	17 16 18 20 19	
7 6 8 10 9	17 16 18 20 19	2 1 3 5 4	2 1 3 5 4	
12 11 13 15 14	12 11 13 15 14	12 11 13 15 14	12 11 13 15 14	
17 16 18 20 19	7 6 8 10 9	22 21 23 25 24	22 21 23 25 24	
22 21 23 25 24	22 21 23 25 24	17 16 18 20 19	7 6 8 10 9	
12/T	13/T	14/T	15/T	
4 1 3 5 2	4 1 3 5 2	9 6 8 10 7	19 16 18 20 17	
9 6 8 10 7	19 16 18 20 17	4 1 3 5 2	4 1 3 5 2	
14 11 13 15 12	14 11 13 15 12	14 11 13 15 12	14 11 13 15 12	
19 16 18 20 17	9 6 8 10 7	24 21 23 25 22	24 21 23 25 22	
24 21 23 25 22	24 21 23 25 22	19 16 18 20 17	9 6 8 10 7	

この16通りの変換法は、このままで良い。変更する必要は全然無い。

では、実際に基本解第1番を対象にしてこの変換法を実行してみよう。そして、その結果から、次に何が必要かを考えよう。

テストの結果を元のリストと比較し、48544個の中に同じ解があるかどうかを検索した。

以下のリストを見てほしい。

### \*\* 変換法のテストの結果 \*\*

[1/F]

9/S	11/S	45611/S	0/S
3 20 1 22 19	3 20 1 22 19	21 18 9 15 2	24 11 17 8 5
21 18 9 15 2	24 11 17 8 5	3 20 1 22 19	3 20 1 22 19
10 12 13 14 16	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16
24 11 17 8 5	21 18 9 15 2	7 4 25 6 23	7 4 25 6 23
7 4 25 6 23	7 4 25 6 23	24 11 17 8 5	21 18 9 15 2
22/S	24/S	45585/S	0/S
3 22 1 20 19	3 22 1 20 19	21 15 9 18 2	24 8 17 11 5
21 15 9 18 2	24 8 17 11 5	3 22 1 20 19	3 22 1 20 19
10 14 13 12 16	10 14 13 12 16	10 14 13 12 16	10 14 13 12 16
24 8 17 11 5	21 15 9 18 2	7 6 25 4 23	7 6 25 4 23
7 6 25 4 23	7 6 25 4 23	24 8 17 11 5	21 15 9 18 2
4857/S	4855/S	45172/S	0/S
20 3 1 19 22	20 3 1 19 22	18 21 9 2 15	11 24 17 5 8
18 21 9 2 15	11 24 17 5 8	20 3 1 19 22	20 3 1 19 22
12 10 13 16 14	12 10 13 16 14	12 10 13 16 14	12 10 13 16 14
11 24 17 5 8	18 21 9 2 15	4 7 25 23 6	4 7 25 23 6
4 7 25 23 6	4 7 25 23 6	11 24 17 5 8	18 21 9 2 15
0/S	0/S	0/S	0/S
22 3 1 19 20	22 3 1 19 20	15 21 9 2 18	8 24 17 5 11
15 21 9 2 18	8 24 17 5 11	22 3 1 19 20	22 3 1 19 20
14 10 13 16 12	14 10 13 16 12	14 10 13 16 12	14 10 13 16 12
8 24 17 5 11	15 21 9 2 18	6 7 25 23 4	6 7 25 23 4
6 7 25 23 4	6 7 25 23 4	8 24 17 5 11	15 21 9 2 18

行列がどう動いたかを、具体的にたどってほしい。変換法が適正に実行されていることを確かめてほしい。

上図の解の n/は、前もって求めておいた標準解リストの番号である。同じものが見つかった時には、緑色の番号表示にしてある。見つからなかった解には、赤く 0/と表示した。

元のリストに無い解が 7 個もある。何故そうなるのか？

変換の結果、リストの条件である  $n_3 < n_{23}; n_3 < n_{11} < n_{15}$ ; の範囲を逸脱してしまったからである。48544 個は、いわゆる「原始解」ではなかった。8 分の 1 に整理され正規化された「標準解」であった。だから、該当する解が見つからなかったのである。

どうすべきか。今さら 38万8千余個の原始解を相手にするわけには行かない。

ならば、今回変換後に得られた各解を「清書」してはどうか。反転・回転の操作を必要なだけ行って、上の不等式の範囲内に収まるように変形するのである。そうして元のリストと比較して見てはどうだろうか。

そこで、次のような 7通りの変形を考える。

#### \* 中央の行列を対象にした反転・回転 \*

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

上図の 0/Tは無変換の「原形」。1/T, 2/T, 3/Tは 90度ずつの回転形である。4/Tは 0/Tを中心

央の列の  $3 + 8 + 13 + 18 + 23$  を対称軸として反転させた形。5/T, 6/T, 7/Tは 90度ずつの回転形である。結果的には、いつもの8個と同じパターンの異形が生じる。

この変形操作を用いて「清書」する。 $n_3 < n_{23}; n_3 < n_{11} < n_{15}$  の範囲に収まるように回転・反転を繰り返すのである。

次にわが力作プログラムの全体を示そう。やるべきことは、

- (1) 対称五方陣の標準解 48544個のリストを作つて配列にセーブする。
- (2) その中から 3034個の基本解を選び出し、別の配列にセーブする。
- (3) 基本解の1個1個について、16通りの変換を施す。
- (4) 必要があれば、清書する。
- (5) 出来たものを元のリストと比較しながら、出力する。
- (6) 対応する番号をモニターして、重複・欠落の有無を最後に報告させる。

## 7. プログラムも結果のリストも、長大なものになる。

特に解のリストの方は、大幅に間引いて概略を示すしかない。あとは、類推と想像によって補いながら読んでいただきたい。

```
/**> Self-Complementary Magic Squares of Order 5: ***
 **> Transform 3040 Fundamental into 48544 Standard **
 **> File: 'SCMS55Trnsf.c' Dictated by Kanji Setsuda **
 **> on Feb.20, 2005; Revised on Jan.26, 2015 **
 **> with Mac OSX 10.10.1 & Xcode 6.1.1 **
 //
#include <stdio.h>
//
long int cnt;
long int tcnt[17];
short cnt2, cnt3;
short LSM, CC;
short nm[26], uflg[26];
short cnm[26], dnm[26];
short ft[3035][26];
short tsol[48545][26];
short nt[17][26];
short refnmb[48545];
//
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);
void rcrdsol(void);
//
void prtrns(void), pr4ans(void);
void trnsf1(void), trnsf2(void), trnsf3(void);
void trnsf4(void), trnsf5(void), trnsf6(void);
void refrot(void), reflect3(void), reflect1(void), rotate(void);
void fndon(void), prmon(void);
//
/**> Main Program */
int main(){
    long m;
    short n;
    printf("\n");
    printf("/**> Self-Complementary Magic Squares of Order 5: ***\n");
    printf("/**> Find the 3040 Fundamental among 48544 Standard **\n");
    for(n=0;n<26;n++){nm[n]=0; uflg[n]=0;}
    for(m=0;m<48545;m++){refnmb[m]=0;}
    LSM=65; CC=26; cnt=0; cnt3=0;
    nm[13]=13; uflg[13]=1;
    stp01(); /* Calculations for Type: n3=1 *
    uflg[13]=0; nm[13]=0;
```

```

printf(" [Count = %d/F, %ld/S]\n",cnt3,cnt);
printf("\n");
printf("** Reconstruct 48544 Solutions by Transforming the 3034 Fundamentals **\n");
prtrns();
printf("\n");
printf(" . . . . .\n");
printf("\n");
printf("* Monitor List of Solution Correspondence between Old and New *\n");
prmon();
printf(" [Count = %ld] OK!\n",cnt);
printf("\n");
printf("** Calculated and Listed by Kanji Setsuda on\n");
printf(" Jan.26, 2015 with MacOSX 10.10.1 & Xcode 6.1.1 **\n");
printf("\n");
return 0;
}
// 
/** Begin The Search 2 *
/** Normalizing Inequality Conditions: n3<n11<n15 and n3<n23;
/** Define Level 1: *
/** Set N3 & n23 & n3<n23 *
void stp01(){
short a,b;
for(a=1;a<13;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[3]=a; nm[23]=b;
uflg[a]=1; uflg[b]=1;
stp02();
uflg[b]=0; uflg[a]=0;}
}
}
/** Set N1 & n25 *
void stp02(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[1]=a; nm[25]=b;
uflg[a]=1; uflg[b]=1;
stp03();
uflg[b]=0; uflg[a]=0;}
}
}
/** Set N2 & n24 *
void stp03(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[2]=a; nm[24]=b;
uflg[a]=1; uflg[b]=1;
stp04();
uflg[b]=0; uflg[a]=0;}
}
}
/** Set N4 & n22 *
void stp04(){
short a,b;
for(a=1;a<26;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[4]=a; nm[22]=b;
uflg[a]=1; uflg[b]=1;
stp05();
uflg[b]=0; uflg[a]=0;}
}
}
/** Set n5=LSM-n1-n2-n3-n4 & n21 *

```

```

void stp05(){
    short a,b;
    a=LSM-nm[1]-nm[2]-nm[3]-nm[4];
    if((0<a)&&(a<26)){
        b=LSM-nm[25]-nm[24]-nm[23]-nm[22];
        if(a+b==CC){
            if((uflg[a]==0)&&(uflg[b]==0)){
                nm[5]=a; nm[21]=b;
                uflg[a]=1; uflg[b]=1;
                stp06();
                uflg[b]=0; uflg[a]=0;}}}
    }
    /* Define Level 2: *
    /* Set N6 & n20 */
void stp06(){
    short a,b;
    for(a=1;a<26;a++){b=CC-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[6]=a; nm[20]=b;
            uflg[a]=1; uflg[b]=1;
            stp07();
            uflg[b]=0; uflg[a]=0;}}
    }
    /* Set N16 & n10 */
void stp07(){
    short a,b;
    for(a=25;a>0;a--) {b=CC-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[16]=a; nm[10]=b;
            uflg[a]=1; uflg[b]=1;
            stp08();
            uflg[b]=0; uflg[a]=0;}}
    }
    /* Set n11=LSM-n1-n6-n16-n21 & n15 & n3<n11<n15 */
void stp08(){
    short a,b;
    a=LSM-nm[1]-nm[6]-nm[16]-nm[21];
    if((nm[3]<a)&&(a<26)){
        b=LSM-nm[25]-nm[20]-nm[10]-nm[5];
        if((a<b)&&(a+b==CC)){
            if((uflg[a]==0)&&(uflg[b]==0)){
                nm[11]=a; nm[15]=b;
                uflg[a]=1; uflg[b]=1;
                stp09();
                uflg[b]=0; uflg[a]=0;}}}
    }
    /* Define Level 2: *
    /* Set N7 & n19 */
void stp09(){
    short a,b;
    for(a=1;a<26;a++){b=CC-a;
        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 N9 & n17 */
void stp10(){
    short a,b;
    for(a=25;a>0;a--) {b=CC-a;
        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){stp11();}
    uflg[b]=0; uflg[a]=0;
}
}

/* Set n8=LSM-n6-n7-n9-n10 & n18 */
void stp11(){
short a,b;
a=LSM-nm[6]-nm[7]-nm[9]-nm[10];
if((0<a)&&(a<26)){
    b=LSM-nm[20]-nm[19]-nm[17]-nm[16];
    if(a+b==CC){
        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){stp12();}
            uflg[b]=0; uflg[a]=0;}}}
}

/* Set n12=LSM-n2-n7-n17-n22 & n14 */
void stp12(){
short a,b;
a=LSM-nm[2]-nm[7]-nm[17]-nm[22];
if((0<a)&&(a<26)){
    b=LSM-nm[24]-nm[19]-nm[9]-nm[4];
    if(a+b==CC){
        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){rcrdsol();}
            uflg[b]=0; uflg[a]=0;}}}
}

// Record the full Data to the Table
// and Select the Fundamental Solutions **
void rcrdsol(void){
short n;
for(n=1;n<26;n++){tsol[cnt][n]=nm[n];}
cnt++;
if((nm[3]<nm[8])&&(nm[8]<nm[18])&&(nm[11]<nm[12])&&(nm[12]<nm[14])){
    for(n=1;n<26;n++){ft[cnt3][n]=nm[n];}
    ft[cnt3][0]=cnt3+1; cnt3++;
}
}

// Print the Transformations *
void prtrns(){
short m,n,p;
for(m=0;m<cnt3;m++){
    for(n=1;n<26;n++){cnm[n]=ft[m][n]; nt[0][n]=cnm[n]; }
    trnsf4(); for(n=1;n<26;n++){nt[1][n]=dnm[n];}
    trnsf5(); for(n=1;n<26;n++){nt[2][n]=dnm[n];}
    trnsf6(); for(n=1;n<26;n++){nt[3][n]=dnm[n];}
    }

    for(p=0;p<4;p++){
        for(n=1;n<26;n++){cnm[n]=nt[p][n];}
        trnsf1(); for(n=1;n<26;n++){nt[p+4][n]=dnm[n];}
        trnsf2(); for(n=1;n<26;n++){nt[p+8][n]=dnm[n];}
        trnsf3(); for(n=1;n<26;n++){nt[p+12][n]=dnm[n];}}
}

refrot();
fnodon();
if(m%300==0){
    printf("[%d/F]\n",m+1);
}
}

```

```

        pr4ans();
    }
}

//*
/* Find the Old Number *
void fndon(){
long l;
short mtc=0;
short m,n;
for(m=0;m<16;m++){tcnt[m]=0;}
for(m=0;m<16;m++){
    for(l=0;l<48544;l++){mtc=0;
        for(n=1;n<26;n++){
            if(nt[m][n]==tsol[l][n]){mtc++;}
            else{break;}
        }
        if(mtc==25){tcnt[m]=l+1; refnmb[l+1]++; break;}
    }
    if(mtc<25){refnmb[0]++;}
}
}

//*
/* Print the 4 Answers *
void pr4ans(){
short k,k4,l,l5,m,n;
for(k=0;k<4;k++){k4=k*4;
    printf("%14ld/S%15ld/S%15ld/S%15ld/S\n",
           tcnt[k4],tcnt[k4+1],tcnt[k4+2],tcnt[k4+3]);
    for(l=0;l<5;l++){l5=l*5;
        for(m=0;m<4;m++){
            printf(" ");
            for(n=1;n<6;n++){printf("%3d",nt[k4+m][l5+n]);}
            if(m+1<4){printf(" ");}
        }
        printf("\n");
    }
}
}

//*
/* Print the Reference Monitor *
void prmon(){
int m;
printf(" ???: %d",refnmb[0]);
for(m=1;m<48545;m++){
    if(m%32==1){printf("\n%6d:",m);}
    printf("%2d",refnmb[m]);
}
printf("\n");
}

//*
/* Transform: Type 1 *
void trnsf1(){
dnm[1]=cnm[1]; dnm[2]=cnm[4]; dnm[3]=cnm[3]; dnm[4]=cnm[2]; dnm[5]=cnm[5];
dnm[6]=cnm[6]; dnm[7]=cnm[9]; dnm[8]=cnm[8]; dnm[9]=cnm[7]; dnm[10]=cnm[10];
dnm[11]=cnm[11]; dnm[12]=cnm[14]; dnm[13]=cnm[13]; dnm[14]=cnm[12]; dnm[15]=cnm[15];
dnm[16]=cnm[16]; dnm[17]=cnm[19]; dnm[18]=cnm[18]; dnm[19]=cnm[17]; dnm[20]=cnm[20];
dnm[21]=cnm[21]; dnm[22]=cnm[24]; dnm[23]=cnm[23]; dnm[24]=cnm[22]; dnm[25]=cnm[25];
}
}

//*
/* Transform: Type 2 *
void trnsf2(){
dnm[1]=cnm[2]; dnm[2]=cnm[1]; dnm[3]=cnm[3]; dnm[4]=cnm[5]; dnm[5]=cnm[4];
dnm[6]=cnm[7]; dnm[7]=cnm[6]; dnm[8]=cnm[8]; dnm[9]=cnm[10]; dnm[10]=cnm[9];
dnm[11]=cnm[12]; dnm[12]=cnm[11]; dnm[13]=cnm[13]; dnm[14]=cnm[15]; dnm[15]=cnm[14];
}

```

```

dnm[16]=cnm[17]; dnm[17]=cnm[16]; dnm[18]=cnm[18]; dnm[19]=cnm[20]; dnm[20]=cnm[19];
dnm[21]=cnm[22]; dnm[22]=cnm[21]; dnm[23]=cnm[23]; dnm[24]=cnm[25]; dnm[25]=cnm[24];
}
// 
/* Transform: Type 3 */
void trnsf3(){
    dnm[1]=cnm[4];    dnm[2]=cnm[1];    dnm[3]=cnm[3];    dnm[4]=cnm[5];    dnm[5]=cnm[2];
    dnm[6]=cnm[9];    dnm[7]=cnm[6];    dnm[8]=cnm[8];    dnm[9]=cnm[10];   dnm[10]=cnm[7];
    dnm[11]=cnm[14];   dnm[12]=cnm[11];   dnm[13]=cnm[13];   dnm[14]=cnm[15];   dnm[15]=cnm[12];
    dnm[16]=cnm[19];   dnm[17]=cnm[16];   dnm[18]=cnm[18];   dnm[19]=cnm[20];   dnm[20]=cnm[17];
    dnm[21]=cnm[24];   dnm[22]=cnm[21];   dnm[23]=cnm[23];   dnm[24]=cnm[25];   dnm[25]=cnm[22];
}
// 
/* Transform: Type 4 */
void trnsf4(){
    dnm[1]=cnm[1];    dnm[2]=cnm[2];    dnm[3]=cnm[3];    dnm[4]=cnm[4];    dnm[5]=cnm[5];
    dnm[6]=cnm[16];   dnm[7]=cnm[17];   dnm[8]=cnm[18];   dnm[9]=cnm[19];   dnm[10]=cnm[20];
    dnm[11]=cnm[11];   dnm[12]=cnm[12];   dnm[13]=cnm[13];   dnm[14]=cnm[14];   dnm[15]=cnm[15];
    dnm[16]=cnm[6];    dnm[17]=cnm[7];    dnm[18]=cnm[8];    dnm[19]=cnm[9];    dnm[20]=cnm[10];
    dnm[21]=cnm[21];   dnm[22]=cnm[22];   dnm[23]=cnm[23];   dnm[24]=cnm[24];   dnm[25]=cnm[25];
}
// 
/* Transform: Type 5 */
void trnsf5(){
    dnm[1]=cnm[6];    dnm[2]=cnm[7];    dnm[3]=cnm[8];    dnm[4]=cnm[9];    dnm[5]=cnm[10];
    dnm[6]=cnm[1];    dnm[7]=cnm[2];    dnm[8]=cnm[3];    dnm[9]=cnm[4];    dnm[10]=cnm[5];
    dnm[11]=cnm[11];   dnm[12]=cnm[12];   dnm[13]=cnm[13];   dnm[14]=cnm[14];   dnm[15]=cnm[15];
    dnm[16]=cnm[21];   dnm[17]=cnm[22];   dnm[18]=cnm[23];   dnm[19]=cnm[24];   dnm[20]=cnm[25];
    dnm[21]=cnm[16];   dnm[22]=cnm[17];   dnm[23]=cnm[18];   dnm[24]=cnm[19];   dnm[25]=cnm[20];
}
// 
/* Transform: Type 6 */
void trnsf6(){
    dnm[1]=cnm[16];   dnm[2]=cnm[17];   dnm[3]=cnm[18];   dnm[4]=cnm[19];   dnm[5]=cnm[20];
    dnm[6]=cnm[1];    dnm[7]=cnm[2];    dnm[8]=cnm[3];    dnm[9]=cnm[4];    dnm[10]=cnm[5];
    dnm[11]=cnm[11];   dnm[12]=cnm[12];   dnm[13]=cnm[13];   dnm[14]=cnm[14];   dnm[15]=cnm[15];
    dnm[16]=cnm[21];   dnm[17]=cnm[22];   dnm[18]=cnm[23];   dnm[19]=cnm[24];   dnm[20]=cnm[25];
    dnm[21]=cnm[6];    dnm[22]=cnm[7];    dnm[23]=cnm[8];    dnm[24]=cnm[9];    dnm[25]=cnm[10];
}
// 
/* Reflect & Rotate */
void refrot(){
    short m,n;
    for(m=0;m<16;m++){
        for(n=1;n<26;n++){cnm[n]=nt[m][n]; dnm[n]=nt[m][n];}
        while((cnm[3]>cnm[11])||(cnm[3]>cnm[15])||(cnm[3]>cnm[23])){rotate();}
        if(cnm[11]>cnm[15]){reflect3();}
        for(n=1;n<26;n++){nt[m][n]=dnm[n];}
    }
}
// 
/* Mirror Reflection with Respect to n3+n8+n13+n18+n23 */
void reflect3(){
    dnm[1]=cnm[5];    dnm[2]=cnm[4];    dnm[3]=cnm[3];    dnm[4]=cnm[2];    dnm[5]=cnm[1];
    dnm[6]=cnm[10];   dnm[7]=cnm[9];    dnm[8]=cnm[8];    dnm[9]=cnm[7];    dnm[10]=cnm[6];
    dnm[11]=cnm[15];   dnm[12]=cnm[14];   dnm[13]=cnm[13];   dnm[14]=cnm[12];   dnm[15]=cnm[11];
    dnm[16]=cnm[20];   dnm[17]=cnm[19];   dnm[18]=cnm[18];   dnm[19]=cnm[17];   dnm[20]=cnm[16];
    dnm[21]=cnm[25];   dnm[22]=cnm[24];   dnm[23]=cnm[23];   dnm[24]=cnm[22];   dnm[25]=cnm[21];
}
// 
/* Mirror Reflection with Respect to n1+n7+n13+n19+n25 */
void reflect1(){
    dnm[1]=cnm[1];    dnm[2]=cnm[6];    dnm[3]=cnm[11];   dnm[4]=cnm[16];   dnm[5]=cnm[21];
    dnm[6]=cnm[2];    dnm[7]=cnm[7];    dnm[8]=cnm[12];   dnm[9]=cnm[17];   dnm[10]=cnm[22];
    dnm[11]=cnm[3];   dnm[12]=cnm[8];    dnm[13]=cnm[13];   dnm[14]=cnm[18];   dnm[15]=cnm[23];
}

```

```

dnm[16]=cnm[4]; dnm[17]=cnm[9]; dnm[18]=cnm[14]; dnm[19]=cnm[19]; dnm[20]=cnm[24];
dnm[21]=cnm[5]; dnm[22]=cnm[10]; dnm[23]=cnm[15]; dnm[24]=cnm[20]; dnm[25]=cnm[25];
}
//
/** Rotation by 90 degrees *
void rotate(){
short n;
dnm[1]=cnm[21]; dnm[2]=cnm[16]; dnm[3]=cnm[11]; dnm[4]=cnm[6]; dnm[5]=cnm[1];
dnm[6]=cnm[22]; dnm[7]=cnm[17]; dnm[8]=cnm[12]; dnm[9]=cnm[7]; dnm[10]=cnm[2];
dnm[11]=cnm[23]; dnm[12]=cnm[18]; dnm[13]=cnm[13]; dnm[14]=cnm[8]; dnm[15]=cnm[3];
dnm[16]=cnm[24]; dnm[17]=cnm[19]; dnm[18]=cnm[14]; dnm[19]=cnm[9]; dnm[20]=cnm[4];
dnm[21]=cnm[25]; dnm[22]=cnm[20]; dnm[23]=cnm[15]; dnm[24]=cnm[10]; dnm[25]=cnm[5];
for(n=1;n<26;n++){cnm[n]=dnm[n];}
}
//

```

### <実験結果の報告>

\* 対称五方陣の基本解／標準解のリストを作つて保存する \*

[Count = 3034/F, 48544/S]

\* 基本解3034個から変換法によって標準解48544個を再構成した結果（抜粋）\*

[1/F]

9/S	11/S	45611/S	45612/S
3 20 1 22 19	3 20 1 22 19	21 18 9 15 2	21 18 9 15 2
21 18 9 15 2	24 11 17 8 5	3 20 1 22 19	7 4 25 6 23
10 12 13 14 16	10 12 13 14 16	10 12 13 14 16	10 12 13 14 16
24 11 17 8 5	21 18 9 15 2	7 4 25 6 23	3 20 1 22 19
7 4 25 6 23	7 4 25 6 23	24 11 17 8 5	24 11 17 8 5
22/S	24/S	45585/S	45586/S
3 22 1 20 19	3 22 1 20 19	21 15 9 18 2	21 15 9 18 2
21 15 9 18 2	24 8 17 11 5	3 22 1 20 19	7 6 25 4 23
10 14 13 12 16	10 14 13 12 16	10 14 13 12 16	10 14 13 12 16
24 8 17 11 5	21 15 9 18 2	7 6 25 4 23	3 22 1 20 19
7 6 25 4 23	7 6 25 4 23	24 8 17 11 5	24 8 17 11 5
4857/S	4855/S	45172/S	45171/S
20 3 1 19 22	20 3 1 19 22	18 21 9 2 15	18 21 9 2 15
18 21 9 2 15	11 24 17 5 8	20 3 1 19 22	4 7 25 23 6
12 10 13 16 14	12 10 13 16 14	12 10 13 16 14	12 10 13 16 14
11 24 17 5 8	18 21 9 2 15	4 7 25 23 6	20 3 1 19 22
4 7 25 23 6	4 7 25 23 6	11 24 17 5 8	11 24 17 5 8
5252/S	5250/S	45094/S	45093/S
20 19 1 3 22	20 19 1 3 22	18 2 9 21 15	18 2 9 21 15
18 2 9 21 15	11 5 17 24 8	20 19 1 3 22	4 23 25 7 6
12 16 13 10 14	12 16 13 10 14	12 16 13 10 14	12 16 13 10 14
11 5 17 24 8	18 2 9 21 15	4 23 25 7 6	20 19 1 3 22
4 23 25 7 6	4 23 25 7 6	11 5 17 24 8	11 5 17 24 8

[301/F]

2633/S	2635/S	10710/S	10748/S
15 20 1 22 7	15 20 1 22 7	12 15 2 19 17	12 19 2 15 17
12 23 5 16 9	17 10 21 3 14	23 20 8 4 10	23 4 8 20 10
2 8 13 18 24	2 8 13 18 24	5 1 13 25 21	5 25 13 1 21
17 10 21 3 14	12 23 5 16 9	16 22 18 6 3	16 6 18 22 3
19 4 25 6 11	19 4 25 6 11	9 7 24 11 14	9 11 24 7 14
2709/S	2711/S	10709/S	10747/S
15 22 1 20 7	15 22 1 20 7	12 15 2 19 17	12 19 2 15 17
12 16 5 23 9	17 3 21 10 14	16 22 18 6 3	16 6 18 22 3
2 18 13 8 24	2 18 13 8 24	5 1 13 25 21	5 25 13 1 21
17 3 21 10 14	12 16 5 23 9	23 20 8 4 10	23 4 8 20 10
19 6 25 4 11	19 6 25 4 11	9 7 24 11 14	9 11 24 7 14

5144/S				5138/S				32852/S				32850/S			
20	15	1	7 22	20	15	1	7 22	23	12	5	9 16	23	12	5	9 16
23	12	5	9 16	10	17	21	14 3	20	15	1	7 22	4	19	25	11 6
8	2	13	24 18	8	2	13	24 18	8	2	13	24 18	8	2	13	24 18
10	17	21	14 3	23	12	5	9 16	4	19	25	11 6	20	15	1	7 22
4	19	25	11 6	4	19	25	11 6	10	17	21	14 3	10	17	21	14 3
4918/S				4912/S				32809/S				32807/S			
20	7	1	15 22	20	7	1	15 22	23	9	5	12 16	23	9	5	12 16
23	9	5	12 16	10	14	21	17 3	20	7	1	15 22	4	11	25	19 6
8	24	13	2 18	8	24	13	2 18	8	24	13	2 18	8	24	13	2 18
10	14	21	17 3	23	9	5	12 16	4	11	25	19 6	20	7	1	15 22
4	11	25	19 6	4	11	25	19 6	10	14	21	17 3	10	14	21	17 3

[601/F]

5256/S				5255/S				19770/S				19768/S			
20	19	1	4 21	20	19	1	4 21	17	10	3	24 11	17	10	3	24 11
17	10	3	24 11	15	2	23	16 9	20	19	1	4 21	5	22	25	7 6
8	12	13	14 18	8	12	13	14 18	8	12	13	14 18	8	12	13	14 18
15	2	23	16 9	17	10	3	24 11	5	22	25	7 6	20	19	1	4 21
5	22	25	7 6	5	22	25	7 6	15	2	23	16 9	15	2	23	16 9
4874/S				4873/S				20006/S				20004/S			
20	4	1	19 21	20	4	1	19 21	17	24	3	10 11	17	24	3	10 11
17	24	3	10 11	15	16	23	2 9	20	4	1	19 21	5	7	25	22 6
8	14	13	12 18	8	14	13	12 18	8	14	13	12 18	8	14	13	12 18
15	16	23	2 9	17	24	3	10 11	5	7	25	22 6	20	4	1	19 21
5	7	25	22 6	5	7	25	22 6	15	16	23	2 9	15	16	23	2 9
4636/S				4635/S				18309/S				18310/S			
19	20	1	21 4	19	20	1	21 4	10	17	3	11 24	10	17	3	11 24
10	17	3	11 24	2	15	23	9 16	19	20	1	21 4	22	5	25	6 7
12	8	13	18 14	12	8	13	18 14	12	8	13	18 14	12	8	13	18 14
2	15	23	9 16	10	17	3	11 24	22	5	25	6 7	19	20	1	21 4
22	5	25	6 7	22	5	25	6 7	2	15	23	9 16	2	15	23	9 16
4684/S				4683/S				18289/S				18290/S			
19	21	1	20 4	19	21	1	20 4	10	11	3	17 24	10	11	3	17 24
10	11	3	17 24	2	9	23	15 16	19	21	1	20 4	22	6	25	5 7
12	18	13	8 14	12	18	13	8 14	12	18	13	8 14	12	18	13	8 14
2	9	23	15 16	10	11	3	17 24	22	6	25	5 7	19	21	1	20 4
22	6	25	5 7	22	6	25	5 7	2	9	23	15 16	2	9	23	15 16

[901/F]

7492/S				7494/S				23477/S				23491/S			
23	19	1	20 2	23	19	1	20 2	5	23	4	24 9	5	24	4	23 9
5	18	11	14 17	9	12	15	8 21	18	19	10	6 12	18	6	10	19 12
4	10	13	16 22	4	10	13	16 22	11	1	13	25 15	11	25	13	1 15
9	12	15	8 21	5	18	11	14 17	14	20	16	7 8	14	7	16	20 8
24	6	25	7 3	24	6	25	7 3	17	2	22	3 21	17	3	22	2 21
7555/S				7557/S				23476/S				23490/S			
23	20	1	19 2	23	20	1	19 2	5	23	4	24 9	5	24	4	23 9
5	14	11	18 17	9	8	15	12 21	14	20	16	7 8	14	7	16	20 8
4	16	13	10 22	4	16	13	10 22	11	1	13	25 15	11	25	13	1 15
9	8	15	12 21	5	14	11	18 17	18	19	10	6 12	18	6	10	19 12
24	7	25	6 3	24	7	25	6 3	17	2	22	3 21	17	3	22	2 21
4746/S				4745/S				46927/S				46873/S			
19	23	1	2 20	19	23	1	2 20	18	19	10	6 12	18	6	10	19 12
18	5	11	17 14	12	9	15	21 8	5	23	4	24 9	5	24	4	23 9
10	4	13	22 16	10	4	13	22 16	11	1	13	25 15	11	25	13	1 15
12	9	15	21 8	18	5	11	17 14	17	2	22	3 21	17	3	22	2 21
6	24	25	3 7	6	24	25	3 7	14	20	16	7 8	14	7	16	20 8

4036/S					4035/S					46928/S					46874/S				
19	2	1	23	20	19	2	1	23	20	18	19	10	6	12	18	6	10	19	12
18	17	11	5	14	12	21	15	9	8	17	2	22	3	21	17	3	22	2	21
10	22	13	4	16	10	22	13	4	16	11	1	13	25	15	11	25	13	1	15
12	21	15	9	8	18	17	11	5	14	5	23	4	24	9	5	24	4	23	9
6	3	25	24	7	6	3	25	24	7	14	20	16	7	8	14	7	16	20	8

[1201/F]

9692/S					9690/S					32810/S					32851/S				
9	15	2	25	14	9	15	2	25	14	23	9	5	12	16	23	12	5	9	16
23	20	8	4	10	16	22	18	6	3	20	15	7	1	22	20	1	7	15	22
5	7	13	19	21	5	7	13	19	21	8	2	13	24	18	8	24	13	2	18
16	22	18	6	3	23	20	8	4	10	4	25	19	11	6	4	11	19	25	6
12	1	24	11	17	12	1	24	11	17	10	14	21	17	3	10	17	21	14	3
9869/S					9867/S					32808/S					32849/S				
9	25	2	15	14	9	25	2	15	14	23	9	5	12	16	23	12	5	9	16
23	4	8	20	10	16	6	18	22	3	4	25	19	11	6	4	11	19	25	6
5	19	13	7	21	5	19	13	7	21	8	2	13	24	18	8	24	13	2	18
16	6	18	22	3	23	4	8	20	10	20	15	7	1	22	20	1	7	15	22
12	11	24	1	17	12	11	24	1	17	10	14	21	17	3	10	17	21	14	3
11213/S					11214/S					40058/S					39884/S				
15	9	2	14	25	15	9	2	14	25	20	15	7	1	22	20	1	7	15	22
20	23	8	10	4	22	16	18	3	6	23	9	5	12	16	23	12	5	9	16
7	5	13	21	19	7	5	13	21	19	8	2	13	24	18	8	24	13	2	18
22	16	18	3	6	20	23	8	10	4	10	14	21	17	3	10	17	21	14	3
1	12	24	17	11	1	12	24	17	11	4	25	19	11	6	4	11	19	25	6
11263/S					11264/S					40055/S					39881/S				
15	14	2	9	25	15	14	2	9	25	20	15	7	1	22	20	1	7	15	22
20	10	8	23	4	22	3	18	16	6	10	14	21	17	3	10	17	21	14	3
7	21	13	5	19	7	21	13	5	19	8	2	13	24	18	8	24	13	2	18
22	3	18	16	6	20	10	8	23	4	23	9	5	12	16	23	12	5	9	16
1	17	24	12	11	1	17	24	12	11	4	25	19	11	6	4	11	19	25	6

[1501/F]

12593/S					12591/S					26326/S					26325/S				
17	25	2	6	15	17	25	2	6	15	19	3	4	21	18	19	3	4	21	18
19	3	4	21	18	8	5	22	23	7	17	25	2	6	15	11	20	24	1	9
10	12	13	14	16	10	12	13	14	16	10	12	13	14	16	10	12	13	14	16
8	5	22	23	7	19	3	4	21	18	11	20	24	1	9	17	25	2	6	15
11	20	24	1	9	11	20	24	1	9	8	5	22	23	7	8	5	22	23	7
12163/S					12161/S					26542/S					26541/S				
17	6	2	25	15	17	6	2	25	15	19	21	4	3	18	19	21	4	3	18
19	21	4	3	18	8	23	22	5	7	17	6	2	25	15	11	1	24	20	9
10	14	13	12	16	10	14	13	12	16	10	14	13	12	16	10	14	13	12	16
8	23	22	5	7	19	21	4	3	18	11	1	24	20	9	17	6	2	25	15
11	1	24	20	9	11	1	24	20	9	8	23	22	5	7	8	23	22	5	7
16986/S					16989/S					23334/S					23333/S				
25	17	2	15	6	25	17	2	15	6	3	19	4	18	21	3	19	4	18	21
3	19	4	18	21	5	8	22	7	23	25	17	2	15	6	20	11	24	9	1
12	10	13	16	14	12	10	13	16	14	12	10	13	16	14	12	10	13	16	14
5	8	22	7	23	3	19	4	18	21	20	11	24	9	1	25	17	2	15	6
20	11	24	9	1	20	11	24	9	1	5	8	22	7	23	5	8	22	7	23
16843/S					16846/S					23318/S					23317/S				
25	15	2	17	6	25	15	2	17	6	3	18	4	19	21	3	18	4	19	21
3	18	4	19	21	5	7	22	8	23	25	15	2	17	6	20	9	24	11	1
12	16	13	10	14	12	16	13	10	14	12	16	13	10	14	12	16	13	10	14
5	7	22	8	23	3	18	4	19	21	20	9	24	11	1	25	15	2	17	6
20	9	24	11	1	20	9	24	11	1	5	7	22	8	23	5	7	22	8	23

[1801/F]

<b>15710/S</b>	<b>15711/S</b>	<b>24367/S</b>	<b>24245/S</b>
23 15 2 6 19	23 15 2 6 19	10 23 4 7 21	10 7 4 23 21
10 17 8 25 5	21 1 18 9 16	17 15 12 20 1	17 20 12 15 1
4 12 13 14 22	4 12 13 14 22	8 2 13 24 18	8 24 13 2 18
21 1 18 9 16	10 17 8 25 5	25 6 14 11 9	25 11 14 6 9
7 20 24 11 3	7 20 24 11 3	5 19 22 3 16	5 3 22 19 16
<b>15428/S</b>	<b>15429/S</b>	<b>24368/S</b>	<b>24246/S</b>
23 6 2 15 19	23 6 2 15 19	10 23 4 7 21	10 7 4 23 21
10 25 8 17 5	21 9 18 1 16	25 6 14 11 9	25 11 14 6 9
4 14 13 12 22	4 14 13 12 22	8 2 13 24 18	8 24 13 2 18
21 9 18 1 16	10 25 8 17 5	17 15 12 20 1	17 20 12 15 1
7 11 24 20 3	7 11 24 20 3	5 19 22 3 16	5 3 22 19 16
<b>11538/S</b>	<b>11537/S</b>	<b>42697/S</b>	<b>42699/S</b>
15 23 2 19 6	15 23 2 19 6	17 10 8 5 25	17 10 8 5 25
17 10 8 5 25	1 21 18 16 9	15 23 2 19 6	20 7 24 3 11
12 4 13 22 14	12 4 13 22 14	12 4 13 22 14	12 4 13 22 14
1 21 18 16 9	17 10 8 5 25	20 7 24 3 11	15 23 2 19 6
20 7 24 3 11	20 7 24 3 11	1 21 18 16 9	1 21 18 16 9
<b>11378/S</b>	<b>11377/S</b>	<b>42662/S</b>	<b>42664/S</b>
15 19 2 23 6	15 19 2 23 6	17 5 8 10 25	17 5 8 10 25
17 5 8 10 25	1 16 18 21 9	15 19 2 23 6	20 3 24 7 11
12 22 13 4 14	12 22 13 4 14	12 22 13 4 14	12 22 13 4 14
1 16 18 21 9	17 5 8 10 25	20 3 24 7 11	15 19 2 23 6
20 3 24 7 11	20 3 24 7 11	1 16 18 21 9	1 16 18 21 9

[2101/F]

<b>18722/S</b>	<b>18719/S</b>	<b>28298/S</b>	<b>28213/S</b>
12 25 3 7 18	12 25 3 7 18	24 12 4 8 17	24 8 4 12 17
24 5 11 16 9	17 10 15 21 2	5 25 6 19 10	5 19 6 25 10
4 6 13 20 22	4 6 13 20 22	11 3 13 23 15	11 23 13 3 15
17 10 15 21 2	24 5 11 16 9	16 7 20 1 21	16 1 20 7 21
8 19 23 1 14	8 19 23 1 14	9 18 22 14 2	9 14 22 18 2
<b>18566/S</b>	<b>18563/S</b>	<b>28301/S</b>	<b>28216/S</b>
12 7 3 25 18	12 7 3 25 18	24 12 4 8 17	24 8 4 12 17
24 16 11 5 9	17 21 15 10 2	16 7 20 1 21	16 1 20 7 21
4 20 13 6 22	4 20 13 6 22	11 3 13 23 15	11 23 13 3 15
17 21 15 10 2	24 16 11 5 9	5 25 6 19 10	5 19 6 25 10
8 1 23 19 14	8 1 23 19 14	9 18 22 14 2	9 14 22 18 2
<b>22974/S</b>	<b>22975/S</b>	<b>34260/S</b>	<b>34212/S</b>
25 12 3 18 7	25 12 3 18 7	5 25 6 19 10	5 19 6 25 10
5 24 11 9 16	10 17 15 2 21	24 12 4 8 17	24 8 4 12 17
6 4 13 22 20	6 4 13 22 20	11 3 13 23 15	11 23 13 3 15
10 17 15 2 21	5 24 11 9 16	9 18 22 14 2	9 14 22 18 2
19 8 23 14 1	19 8 23 14 1	16 7 20 1 21	16 1 20 7 21
<b>23094/S</b>	<b>23095/S</b>	<b>34255/S</b>	<b>34207/S</b>
25 18 3 12 7	25 18 3 12 7	5 25 6 19 10	5 19 6 25 10
5 9 11 24 16	10 2 15 17 21	9 18 22 14 2	9 14 22 18 2
6 22 13 4 20	6 22 13 4 20	11 3 13 23 15	11 23 13 3 15
10 2 15 17 21	5 9 11 24 16	24 12 4 8 17	24 8 4 12 17
19 14 23 8 1	19 14 23 8 1	16 7 20 1 21	16 1 20 7 21

[2401/F]

<b>23063/S</b>	<b>23062/S</b>	<b>25687/S</b>	<b>25486/S</b>
25 17 3 2 18	25 17 3 2 18	16 25 4 8 12	16 8 4 25 12
16 11 5 19 14	12 7 21 15 10	11 17 6 24 7	11 24 6 17 7
4 6 13 20 22	4 6 13 20 22	5 3 13 23 21	5 23 13 3 21
12 7 21 15 10	16 11 5 19 14	19 2 20 9 15	19 9 20 2 15
8 24 23 9 1	8 24 23 9 1	14 18 22 1 10	14 1 22 18 10

22679/S									22678/S									25689/S									25488/S								
25	2	3	17	18	25	2	3	17	18	16	25	4	8	12	16	8	4	25	12																
16	19	5	11	14	12	15	21	7	10	19	2	20	9	15	19	9	20	2	15																
4	20	13	6	22	4	20	13	6	22	5	3	13	23	21	5	23	13	3	21																
12	15	21	7	10	16	19	5	11	14	11	17	6	24	7	11	24	6	17	7																
8	9	23	24	1	8	9	23	24	1	14	18	22	1	10	14	1	22	18	10																
20064/S									20061/S									30225/S									30226/S								
17	25	3	18	2	17	25	3	18	2	11	16	5	14	19	11	16	5	14	19																
11	16	5	14	19	7	12	21	10	15	17	25	3	18	2	24	8	23	1	9																
6	4	13	22	20	6	4	13	22	20	6	4	13	22	20	6	4	13	22	20																
7	12	21	10	15	11	16	5	14	19	24	8	23	1	9	17	25	3	18	2																
24	8	23	1	9	24	8	23	1	9	7	12	21	10	15	7	12	21	10	15																
19924/S									19921/S									30217/S									30218/S								
17	18	3	25	2	17	18	3	25	2	11	14	5	16	19	11	14	5	16	19																
11	14	5	16	19	7	10	21	12	15	17	18	3	25	2	24	1	23	8	9																
6	22	13	4	20	6	22	13	4	20	6	22	13	4	20	6	22	13	4	20																
7	10	21	12	15	11	14	5	16	19	24	1	23	8	9	17	18	3	25	2																
24	1	23	8	9	24	1	23	8	9	7	10	21	12	15	7	10	21	12	15																

### [2701/F]

28382/S									28380/S									29895/S									29922/S								
24	16	4	20	1	24	16	4	20	1	8	24	5	25	3	8	25	5	24	3																
8	15	12	7	23	3	19	14	11	18	15	16	9	6	19	15	6	9	16	19																
5	9	13	17	21	5	9	13	17	21	12	4	13	22	14	12	22	13	4	14																
3	19	14	11	18	8	15	12	7	23	7	20	17	10	11	7	10	17	20	11																
25	6	22	10	2	25	6	22	10	2	23	1	21	2	18	23	2	21	1	18																
28453/S									28451/S									29892/S									29919/S								
24	20	4	16	1	24	20	4	16	1	8	24	5	25	3	8	25	5	24	3																
8	7	12	15	23	3	11	14	19	18	7	20	17	10	11	7	10	17	20	11																
5	17	13	9	21	5	17	13	9	21	12	4	13	22	14	12	22	13	4	14																
3	11	14	19	18	8	7	12	15	23	15	16	9	6	19	15	6	9	16	19																
25	10	22	6	2	25	10	22	6	2	23	1	21	2	18	23	2	21	1	18																
25657/S									25659/S									44984/S									44966/S								
16	24	4	1	20	16	24	4	1	20	15	16	9	6	19	15	6	9	16	19																
15	8	12	23	7	19	3	14	18	11	8	24	5	25	3	8	25	5	24	3																
9	5	13	21	17	9	5	13	21	17	12	4	13	22	14	12	22	13	4	14																
19	3	14	18	11	15	8	12	23	7	23	1	21	2	18	23	2	21	1	18																
6	25	22	2	10	6	25	22	2	10	7	20	17	10	11	7	10	17	20	11																
25428/S									25430/S									44985/S									44967/S								
16	1	4	24	20	16	1	4	24	20	15	16	9	6	19	15	6	9	16	19																
15	23	12	8	7	19	18	14	3	11	23	1	21	2	18	23	2	21	1	18																
9	21	13	5	17	9	21	13	5	17	12	4	13	22	14	12	22	13	4	14																
19	18	14	3	11	15	23	12	8	7	8	24	5	25	3	8	25	5	24	3																
6	2	22	25	10	6	2	22	25	10	7	20	17	10	11	7	10	17	20	11																

### [3001/F]

42119/S									42123/S									43906/S									44008/S								
12	20	8	22	3	12	20	8	22	3	2	12	9	23	19	2	23	9	12	19																
2	25	10	21	7	19	5	16	1	24	25	20	11	4	5	25	4	11	20	5																
9	11	13	15	17	9	11	13	15	17	10	8	13	18	16	10	18	13	8	16																
19	5	16	1	24	2	25	10	21	7	21	22	15	6	1	21	6	15	22	1																
23	4	18	6	14	23	4	18	6	14	7	3	17	14	24	7	14	17	3	24																
42138/S									42142/S									43905/S									44007/S								
12	22	8	20	3	12	22	8	20	3	2	12	9	23	19	2	23	9	12	19																
2	21	10	25	7	19	1	16	5	24	21	22	15	6	1	21	6	15	22	1																
9	15	13	11	17	9	15	13	11	17	10	8	13	18	16	10	18	13	8	16																
19	1	16	5	24	2	21	10	25	7	25	20	11	4	5	25	4	11	20	5																
23	6	18	4	14	23	6	18	4	14	7	3	17	14	24	7	14	17	3	24																

42960/S				42959/S				47462/S				47457/S							
20	12	8	3	22	20	12	8	3	22	25	2	10	7	21	25	2	10	7	21
25	2	10	7	21	5	19	16	24	1	20	12	8	3	22	4	23	18	14	6
11	9	13	17	15	11	9	13	17	15	11	9	13	17	15	11	9	13	17	15
5	19	16	24	1	25	2	10	7	21	4	23	18	14	6	20	12	8	3	22
4	23	18	14	6	4	23	18	14	6	5	19	16	24	1	5	19	16	24	1
42912/S				42911/S				47518/S				47513/S							
20	3	8	12	22	20	3	8	12	22	25	7	10	2	21	25	7	10	2	21
25	7	10	2	21	5	24	16	19	1	20	3	8	12	22	4	14	18	23	6
11	17	13	9	15	11	17	13	9	15	11	17	13	9	15	11	17	13	9	15
5	24	16	19	1	25	7	10	2	21	4	14	18	23	6	20	3	8	12	22
4	14	18	23	6	4	14	18	23	6	5	24	16	19	1	5	24	16	19	1

\* Monitor List of Solution Correspondence between Old and New \*

?? : 0

```
48321: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
48353: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
48385: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
48417: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
48449: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
48481: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
48513: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
[Count = 48544] OK!
```

\*\* Calculated and Listed by Kanji Setsuda on  
Jan. 26, 2015 with MacOSX 10.10.1 & Xcode 6.1.1 \*\*

見ての通り 3034 個から 48544 個を完璧に再構成することができた。

変換法は、かくも有効である。求めた 3034 個が標準解 48544 個の「基本解」となっているという予想を具体的に検証・確認することができた。

48544=3034×16 の意味を実証することができたのである。

次の課題は、3034 個の基本解集合内部の変換法である。

そのためには構造研究が欠かせない。

ただし、3034 という数には約数が少ないので、余り楽観はできないように思う。タイプ・グループ別に解の分析・研究が新たに必要になるかもしれない。

(Jun. 7, 2000(初稿) ; Sept. 7, 2002(第二稿) ;  
Feb. 20, 2006(改定新稿) ; Jan. 26, 2015(再計算改訂稿) ;  
MacOSX 10.10.1 & Xcode 6.1.1 による計数・作表・著述 : 摂田 寛二(Kanji Setsuda))

---

E-Mail Address: jag12001@nifty.com