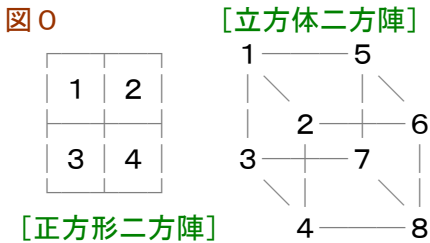


# 第6章：多次元超立体魔方陣の研究：摂田 寛二

## 第2節：四次元超立体二方陣の構成と分解

0. 前稿に引続いて多次元の超立体魔方陣を研究する。今度は、二方陣である。正方形でも立方体でも「二方陣」と言うのは、今までは魔方陣として無意味であった。



というのは、全行全列全柱の定和が成り立ちようがないからである。与えられた約束（連続する自然数を全個重複なく用いる）を厳守しなければならない以上、左図で  $n1+n2=n1+n3=n1+n5$  とするならば、 $n2=n3=n5$  となってしまって、この約束に反する。他のどんな条件を与えても、魔方陣らしくなくなる。魔方陣を作る意味が無いのである。二方陣は、かくして今までほとんど誰にも顧みられることが無かった。

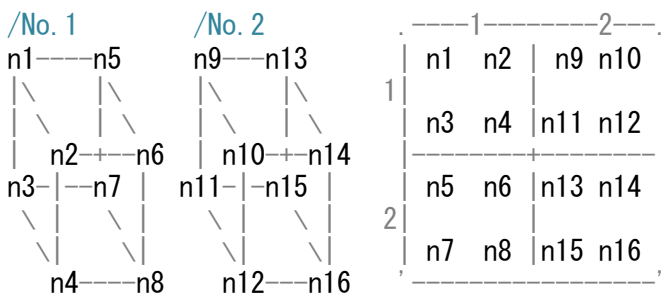
しかし、 $2 \times 2 \times 2 \times 2$  の配列を用意し、そこに四次元の超立方体魔方陣を作るのであれば、話は別だ。もしかしたら出来るかもしれない。何らかの存在意義も、あるかもしれない。

というのは、四次元二方陣は、構造的に二次元四方陣と親近性があり、小生の予感では、もしかしたら四方陣構造解明の切り札になるかも知れない存在だからである。

1. たとえ二方陣といえども、我らのゴールは「四次元超立体魔方陣」である。これを作るのは大変だ。感覚を超えたところで概念的に構成されるからである。一生懸命に頭を使わなければならない。

先ず「基本図」をどうするか。

### [基本図1] 四次元超立体二方陣と二次元四方陣



何も図が無いのでは、考察にも説明にも困惑する。低次元に展開した「分解図」を用意する。今回も、 $2 \times 8$  および  $2 \times 2 \times 4$  という2種類の展開図を用いる。

左図は2つの立方体に分解した図であり、右図は  $2 \times 2$  のブロックの中に平面二方陣をおさめたもので、左図の立体の左右方向の各面を順にベタに並べたものと思っただいてよい。

なお、右図は全体として見れば、「正方形四方陣」になっている。

2. 四次元空間をどう扱えば良いであろうか。

四次元と言うからには、座標軸が4本ある。上の基本図では、それがどう表現されるか。

$n1$  を原点と仮定すると、4軸はいずれも  $\{n1, \text{その隣りの数}\}$  の2数によって構成される。 $n2, n3, n5$  を「原点の隣りの数」とすると指名することに異存はあるまい。では、もう一つの「隣りの数」はどれか。4番目の軸はどこにあるか。

「基本図」の数値の部分を各々1だけ減じて二進法分解してみよう。



既知の3軸は、 $\{n1, n2\} = \{n(0000), n(0001)\}$ ,  $\{n1, n3\} = \{n(0000), n(0010)\}$ ,  $\{n1, n5\} =$

{n(0000), n(0100)} で表現される。その特徴から見て、4軸目は {n1, n9} = {n(0000), n(1000)} と指名するのが妥当だ。位取りを無視すれば、原点のそれと1字違いの座標を持つものは、n9 以外にはもう無いからだ。

**3. 魔方陣を構成するためには、どんな条件が必要であろうか。**

用いる素材は、1~16 の自然数 16 個。全個重複なく欠落なく用いる。配列は、上図左を使い、変数名を通し番号で n1, n2, n3, …, n15, n16 とする。

(1) 超立方体でも、二方阵には違いないから、全行全列全柱定和は成り立たない。ゆえにこれを諦める。

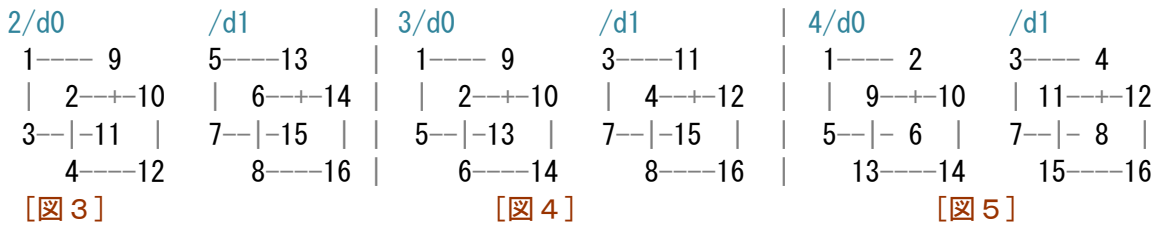
そのかわりに「頂点連結正方四数の定和」を仮定する。2つの立方体の6面ずつ全12面において、4数の和がいつも34になるように設計する。

$$\begin{aligned} n_1 + n_2 + n_3 + n_4 &= n_1 + n_2 + n_5 + n_6 = n_1 + n_3 + n_5 + n_7 = \\ n_2 + n_4 + n_6 + n_8 &= n_3 + n_4 + n_7 + n_8 = n_5 + n_6 + n_7 + n_8 = 34 \quad \dots \text{No. 1} \\ n_9 + n_{10} + n_{11} + n_{12} &= n_9 + n_{10} + n_{13} + n_{14} = n_9 + n_{11} + n_{13} + n_{15} = \\ n_{10} + n_{12} + n_{14} + n_{16} &= n_{11} + n_{12} + n_{15} + n_{16} = n_{13} + n_{14} + n_{15} + n_{16} = 34 \quad \dots \text{No. 2} \end{aligned}$$

(2) これだけでは、足りない。いかにも四次元魔方陣らしい条件をさらに付け加えなければならぬ。

ここで、数要素の並べ替えの可能性を考え、別個の展開図も用意する。例えば、次図のような並べ替えは自然である。原点 n1 の隣りに n2, n3, n5 そして n9 が来ている。

だから、これらの図中でも頂点連結正方四数の定和が成り立つようにしたい。



「拡張定和式」として、次の12本の式を加えることにしよう。

$$\begin{aligned} n_1 + n_2 + n_9 + n_{10} &= n_3 + n_4 + n_{11} + n_{12} = n_5 + n_6 + n_{13} + n_{14} = \\ n_7 + n_8 + n_{15} + n_{16} &= n_1 + n_3 + n_9 + n_{11} = n_2 + n_4 + n_{10} + n_{12} = \\ n_5 + n_7 + n_{13} + n_{15} &= n_6 + n_8 + n_{14} + n_{16} = n_1 + n_5 + n_9 + n_{13} = \\ n_2 + n_6 + n_{10} + n_{14} &= n_3 + n_7 + n_{11} + n_{15} = n_4 + n_8 + n_{12} + n_{16} = 34 \quad \dots \text{No. 3} \end{aligned}$$

(3) 補数定和と対称配置をどうするか。

$$\begin{aligned} n_1 + n_{16} &= n_2 + n_{15} = n_3 + n_{14} = n_4 + n_{13} \\ &= n_5 + n_{12} = n_6 + n_{11} = n_7 + n_{10} = n_8 + n_9 = 17 \end{aligned}$$

というペアリングは、今回も可能か。

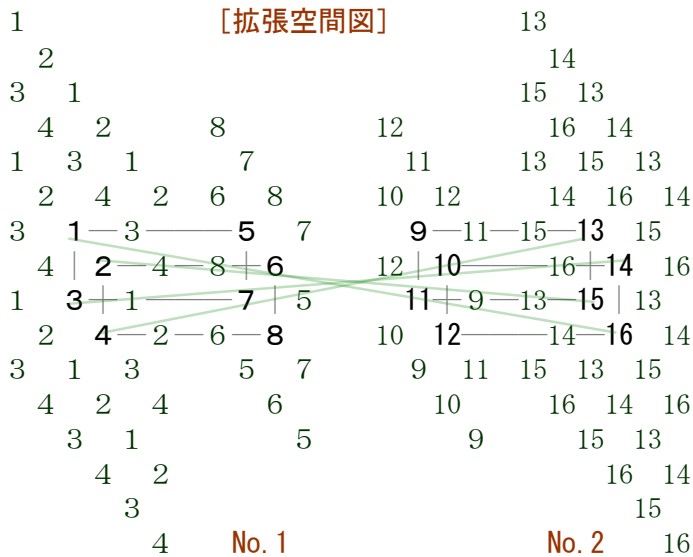
上の2図を見ても、それぞれ対称の位置に置かれている。だから、17の補数定和と対称配置をあらためて仮定することにしよう。そして他の場所に動く時でも、ペアで一緒に動くものとする。すなわち「補数対称型」の魔方陣の構成を、先ず我々は目指すことにする。

(4) 各展開図で「汎対角線定和」はどう実現するか。

先ず「汎対角線」はどこにあるか。今までは大抵主対角線に平行に走っていた。それを下の拡張空間図でさがすことにしよう。

n1 + n16 と平行に走る n2 + nX をさがす。n2 + n15 が見つかる。n2 から斜め上に走って追跡しても、n2 + n15 しか見つからない。汎対角線は、どれも4方向に走るのだが、どの方向に行っても、見つかる相手は、n15 である。それは、n1 でも n3 でも n4 でも n5 でも …… みな同じで、結局のところ汎対角線は、全て17の補数各1ペアによってまかなわれて構成されている。

つまり、対称性と汎性が同じ形で実現しているわけで、これはこの方阵特有の簡潔で美しい構造だ。

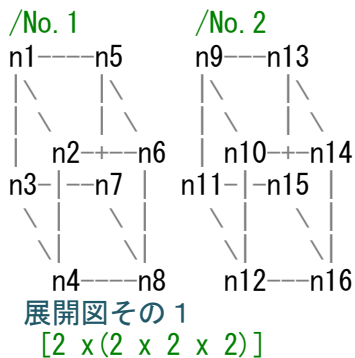


汎対角線の問題は避けては通れない。何時必要になるかわからないから。……超立体で「汎対角線」を空間的・視覚的に追うのは、難しい。しかし、諦めるなかれ。まずは、主対角線からさがす。今まで大抵は 17 の補数ペア (の和) の形になっていた。  
 $n1 + n16$  はどこにあるか。横長の直方体の対角線上にある。No. 1 と No. 2 の立方体は、元は同じものであるから、 $n8$  と対応している  $n16$  を  $n1$  が相手に選ぶのは、当然である。  
 $n2 + n15$ ,  $n3 + n14$ ,  $n4 + n13$  も直方体の対角線上にあり、極めて自然な形におさまっている。主対角線は、そのまま採用してよい。

4. 前の 1 で示したもう一つの二次元展開図についても詳しく触れておこう。

下図のように  $2 \times 2 \times 2 \times 2$  の超立方体は、 $2 \times 2 \times (2 \times 2)$  という形にも分解図を書くことができる。縦横  $2 \times 2$  のブロックに  $2 \times 2$  の正方形二方陣を配置するのだが、二つの立方体の左右方向の 4 面をベタに並べた形になっている。というよりも、もっと大きな眼で捉えれば、全体を正方形四方陣に書き換えた形と見ることができる。

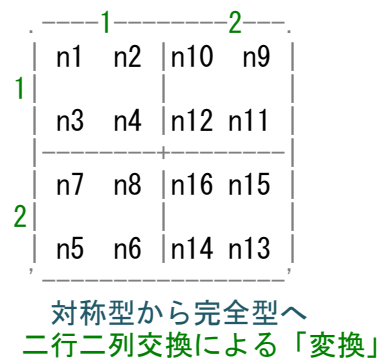
[基本図 2-0]



[図 2-1]



[図 2-2]



要するに、四次元二方陣は、二次元四方陣に書き換えることができるのである。我々は、この「書き換え」の可能性に注目し、以後「次元変換法」として研究する。何がどうなるのかは、後で詳しく検討しよう。

5. いよいよ具体的な方陣構成にかかろう。

- (1) 基本図は、どれを用いても構わないのだが、今は前頁の図 5 をメインに採用する。
- (2) 24 本の頂点連結正方四数の拡張定和式と 8 ペアの補数定和式・対称配置を仮定する。
- (3) リストを整形出力する際の不等式はどう設定するか。

超立体二方陣の場合、反転・回転形の重複計数を防ぐために不等式を念入りに立てると、解は結局 1 つしか出て来なくなる。それでは何を意味しているかわからないので、今は、戦術的に不等式を立てるのを一時的に止めて、あえて反転・回転形を封じないことにする。

四次元を代表する 4 つの座標軸が、どう動くかをつぶさに調べたいという目的もあるから。……

原始解は 384 個も出て来る。以下にリストの最初の  $n1 = 1$  の 24 例をお見せしよう。

その前にリストの見方を少し説明しよう。最初の 2 個は、小立方体に展開した「分解図」。それを二次元四方陣に並べた展開図が真ん中にある。これは実は「対称四方陣用」になっている。右側の四方陣は、それを型変換した図で、「完全四方陣用」になっている。

図6 \* リストの見方: 基本図 \*

BD/d0	/d1	SC/	CC/
1---- 2	3---- 4	1 2 3 4	1 2 4 3
9---+10	11---+12	5 6 7 8	5 6 8 7
5-- - 6	7-- - 8	9 10 11 12	13 14 16 15
13----14	15----16	13 14 15 16	9 10 12 11
[四次元超立体二方陣図]		[対称四方陣]	[完全四方陣]

\*\* 四次元超立体二方陣の解のリスト (n1=1の24例) \*\*

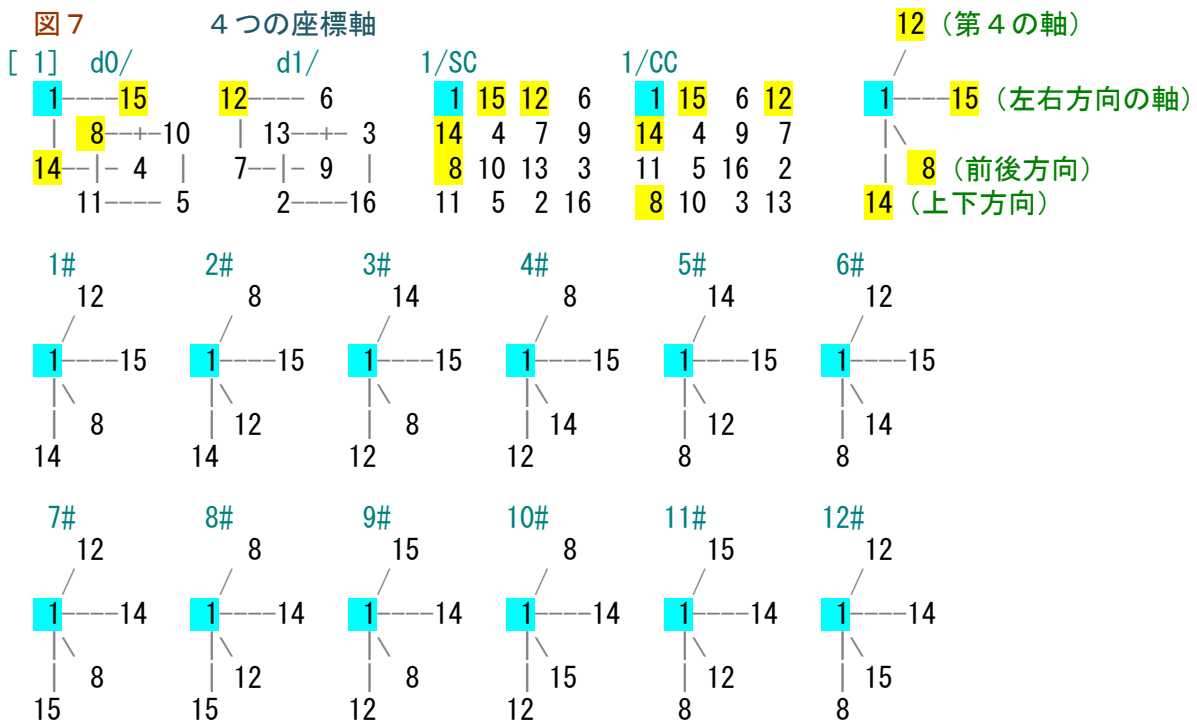
1/d0	/d1	SC/	CC/
1----15	12---- 6	1 15 12 6	1 15 6 12
8---+10	13---+ 3	14 4 7 9	14 4 9 7
14-- - 4	7-- - 9	8 10 13 3	11 5 16 2
11---- 5	2----16	11 5 2 16	8 10 3 13
2/d0	/d1	SC/	CC/
1----15	8----10	1 15 8 10	1 15 10 8
12---+ 6	13---+ 3	14 4 11 5	14 4 5 11
14-- - 4	11-- - 5	12 6 13 3	7 9 16 2
7---- 9	2----16	7 9 2 16	12 6 3 13
3/d0	/d1	SC/	CC/
1----15	14---- 4	1 15 14 4	1 15 4 14
8---+10	11---+ 5	12 6 7 9	12 6 9 7
12-- - 6	7-- - 9	8 10 11 5	13 3 16 2
13---- 3	2----16	13 3 2 16	8 10 5 11
4/d0	/d1	SC/	CC/
1----15	8----10	1 15 8 10	1 15 10 8
14---+ 4	11---+ 5	12 6 13 3	12 6 3 13
12-- - 6	13-- - 3	14 4 11 5	7 9 16 2
7---- 9	2----16	7 9 2 16	14 4 5 11
5/d0	/d1	SC/	CC/
1----15	14---- 4	1 15 14 4	1 15 4 14
12---+ 6	7---+ 9	8 10 11 5	8 10 5 11
8-- -10	11-- - 5	12 6 7 9	13 3 16 2
13---- 3	2----16	13 3 2 16	12 6 9 7
6/d0	/d1	SC/	CC/
1----15	12---- 6	1 15 12 6	1 15 6 12
14---+ 4	7---+ 9	8 10 13 3	8 10 3 13
8-- -10	13-- - 3	14 4 7 9	11 5 16 2
11---- 5	2----16	11 5 2 16	14 4 9 7
7/d0	/d1	SC/	CC/
1----14	12---- 7	1 14 12 7	1 14 7 12
8---+11	13---+ 2	15 4 6 9	15 4 9 6
15-- - 4	6-- - 9	8 11 13 2	10 5 16 3
10---- 5	3----16	10 5 3 16	8 11 2 13
8/d0	/d1	SC/	CC/
1----14	8----11	1 14 8 11	1 14 11 8
12---+ 7	13---+ 2	15 4 10 5	15 4 5 10
15-- - 4	10-- - 5	12 7 13 2	6 9 16 3
6---- 9	3----16	6 9 3 16	12 7 2 13
9/d0	/d1	SC/	CC/
1----14	15---- 4	1 14 15 4	1 14 4 15
8---+11	10---+ 5	12 7 6 9	12 7 9 6

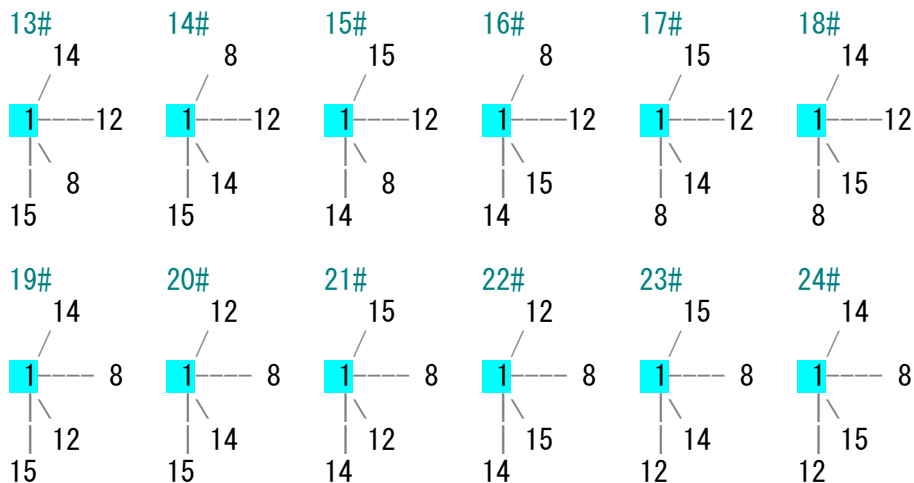
12-- - 7	6-- - 9	8 11 10 5	13 2 16 3
13---- 2	3----16	13 2 3 16	8 11 5 10
10/d0	/d1	SC/	CC/
1----14	8----11	1 14 8 11	1 14 11 8
15---+ 4	10---+ 5	12 7 13 2	12 7 2 13
12-- - 7	13-- - 2	15 4 10 5	6 9 16 3
6---- 9	3----16	6 9 3 16	15 4 5 10
11/d0	/d1	SC/	CC/
1----14	15---- 4	1 14 15 4	1 14 4 15
12---+ 7	6---+ 9	8 11 10 5	8 11 5 10
8-- -11	10-- - 5	12 7 6 9	13 2 16 3
13---- 2	3----16	13 2 3 16	12 7 9 6
12/d0	/d1	SC/	CC/
1----14	12---- 7	1 14 12 7	1 14 7 12
15---+ 4	6---+ 9	8 11 13 2	8 11 2 13
8-- -11	13-- - 2	15 4 6 9	10 5 16 3
10---- 5	3----16	10 5 3 16	15 4 9 6
13/d0	/d1	SC/	CC/
1----12	14---- 7	1 12 14 7	1 12 7 14
8---+13	11---+ 2	15 6 4 9	15 6 9 4
15-- - 6	4-- - 9	8 13 11 2	10 3 16 5
10---- 3	5----16	10 3 5 16	8 13 2 11
14/d0	/d1	SC/	CC/
1----12	8----13	1 12 8 13	1 12 13 8
14---+ 7	11---+ 2	15 6 10 3	15 6 3 10
15-- - 6	10-- - 3	14 7 11 2	4 9 16 5
4---- 9	5----16	4 9 5 16	14 7 2 11
15/d0	/d1	SC/	CC/
1----12	15---- 6	1 12 15 6	1 12 6 15
8---+13	10---+ 3	14 7 4 9	14 7 9 4
14-- - 7	4-- - 9	8 13 10 3	11 2 16 5
11---- 2	5----16	11 2 5 16	8 13 3 10
16/d0	/d1	SC/	CC/
1----12	8----13	1 12 8 13	1 12 13 8
15---+ 6	10---+ 3	14 7 11 2	14 7 2 11
14-- - 7	11-- - 2	15 6 10 3	4 9 16 5
4---- 9	5----16	4 9 5 16	15 6 3 10
17/d0	/d1	SC/	CC/
1----12	15---- 6	1 12 15 6	1 12 6 15
14---+ 7	4---+ 9	8 13 10 3	8 13 3 10
8-- -13	10-- - 3	14 7 4 9	11 2 16 5
11---- 2	5----16	11 2 5 16	14 7 9 4
18/d0	/d1	SC/	CC/
1----12	14---- 7	1 12 14 7	1 12 7 14
15---+ 6	4---+ 9	8 13 11 2	8 13 2 11
8-- -13	11-- - 2	15 6 4 9	10 3 16 5
10---- 3	5----16	10 3 5 16	15 6 9 4
19/d0	/d1	SC/	CC/
1---- 8	14----11	1 8 14 11	1 8 11 14
12---+13	7---+ 2	15 10 4 5	15 10 5 4
15-- -10	4-- - 5	12 13 7 2	6 3 16 9
6---- 3	9----16	6 3 9 16	12 13 2 7

20/d0	/d1	SC/	CC/
1---- 8	12----13	1 8 12 13	1 8 13 12
14---+11	7---+ 2	15 10 6 3	15 10 3 6
15-- -10	6-- - 3	14 11 7 2	4 5 16 9
4---- 5	9----16	4 5 9 16	14 11 2 7
21/d0	/d1	SC/	CC/
1---- 8	15----10	1 8 15 10	1 8 10 15
12---+13	6---+ 3	14 11 4 5	14 11 5 4
14-- -11	4-- - 5	12 13 6 3	7 2 16 9
7---- 2	9----16	7 2 9 16	12 13 3 6
22/d0	/d1	SC/	CC/
1---- 8	12----13	1 8 12 13	1 8 13 12
15---+10	6---+ 3	14 11 7 2	14 11 2 7
14-- -11	7-- - 2	15 10 6 3	4 5 16 9
4---- 5	9----16	4 5 9 16	15 10 3 6
23/d0	/d1	SC/	CC/
1---- 8	15----10	1 8 15 10	1 8 10 15
14---+11	4---+ 5	12 13 6 3	12 13 3 6
12-- -13	6-- - 3	14 11 4 5	7 2 16 9
7---- 2	9----16	7 2 9 16	14 11 5 4
24/d0	/d1	SC/	CC/
1---- 8	14----11	1 8 14 11	1 8 11 14
15---+10	4---+ 5	12 13 7 2	12 13 2 7
12-- -13	7-- - 2	15 10 4 5	6 3 16 9
6---- 3	9----16	6 3 9 16	15 10 5 4

上の解では、先ず 17 の補数ペアがすべて対称の位置にあり、全行全列全柱の定和こそ成り立たないが、2つの小立方体の全 12 面で頂点連結正方四数の定和が成り立つことを確かめてほしい。

さらに四次元そのものを直接に表現する4つの座標軸の内容に注目し、解によって「1の隣りの数」がどう配置されるかをチェックしてほしい。





[Count = 24]

内容から見て、4軸が平等な資格で現れ、数学的対称性が高いことに気付かれよう。そして、これらが確かに「四次元超立体二方阵」であることに納得していただければ。

### 6. 四次元二方阵と二次元四方陣との関係をもう少し具体的に詳しく調べて見よう。

(1)  $2 \times 2 \times 4$  の分解図によって出来た**対称四方陣**を詳しく調べてみよう。

[ 1 ]	d0/	d1/	1/SC	3 13 8 10	3 13 8 10	/PD	図 8
1----	15	12----	6	1 15 12 6	6 12 1 15	6 12 1 15	
8--+-	10	13--+-	3	14 4 7 9	9 7 14 4	9 7 14 4	
14-- -	4	7-- -	9	8 10 13 3	16 2 11 5	16 2 11 5	
11----	5	2----	16	11 5 2 16	3 13 8 10	3 13 8 10	← 拡張方阵空間
[四次元超立体二方阵図]		[対称四方陣]		6 12 1 15	6 12 1 15	[完全四方陣]	

行和：  $1 + 15 + 12 + 6 = 14 + 4 + 7 + 9 = 8 + 10 + 13 + 3 = 11 + 5 + 2 + 16 = K$ ； および  
 列和：  $1 + 14 + 8 + 11 = 15 + 4 + 10 + 5 = 12 + 7 + 13 + 2 = 6 + 9 + 3 + 16 = K$ ； は、  
 元の超立方体の各面の「頂点連結正方四数の定和」および拡張定和式から来ている。  
 補数の配置は、二次元面でも、めでたく点対称の位置に来ている。

(2) 対称四方陣から「型変換」して得た**完全四方陣**も調べよう（上図右）。

行和列和の出自： (1) の対称型と同じ。

主対角線の内容：  $1 + 4 + 16 + 13 = (1 + 16) + (4 + 13) = 17 + 17 = K$ ；

$8 + 5 + 9 + 12 = (8 + 9) + (5 + 12) = 17 + 17 = K$ ；

汎対角線：  $1 + 7 + 16 + 10 = (1 + 16) + (7 + 10) = 17 + 17 = K$ ；

$15 + 9 + 2 + 8 = (15 + 2) + (9 + 8) = 17 + 17 = K$ ；  $15 + 14 + 2 + 3 = (15 + 2) + (14 + 3) = 17 + 17 = K$

$6 + 7 + 11 + 10 = (6 + 11) + (7 + 10) = 17 + 17 = K$ ；  $6 + 4 + 11 + 13 = (6 + 11) + (4 + 13) = 17 + 17 = K$

$12 + 14 + 5 + 3 = (12 + 5) + (14 + 3) = 17 + 17 = K$ ；

というふうに、汎対角線（主対角線も含む）8本の上に全ての補数ペアが確かに並んでいて、定和 $K (=34)$ を実現しているのである。

(3) さらに「完全四方陣」では、任意の  $2 \times 2$  の正方連結4数の和が成り立つ。

その出自：  $1 + 15 + 14 + 4 = 11 + 5 + 8 + 10 = 6 + 12 + 9 + 7 =$

$16 + 2 + 3 + 13 = 14 + 4 + 11 + 5 = 9 + 7 + 16 + 2 = K$ ； は、

元の超立方体の各面での「頂点連結正方四数の定和」および拡張定和式から来ている。また、

$15 + 6 + 4 + 9 = 4 + 9 + 5 + 16 = 5 + 16 + 10 + 3 = K$ ； も、同様だ。

言い換えれば、二次元の対称四方陣も完全四方陣も、その特徴の出自をことごとくこの**四次元超立体二方阵**に求めることができるのである。

かくのごとく四次元超立体二方阵の解は、その各々が二次元の対称四方陣と完全四方陣にみごとに一対一対応していて、全てを緊密に結び付けているのである。

平面四方陣においては、昔から異種間で [1] 解の総数の一致，[2] 型変換が可能，[3] 汎四方陣・完全四方陣・正方連結汎四方陣の「奇跡の一致」が知られていた。随分と不思議がられていたのだが、これらを成り立たせる根拠・理由として、**四次元超立体二方陣**の存在を指名することができるのではないだろうか。

**7. いや、待て。厳密な意味では、これは「一対一対応」ではないかもしれない。**

何故なら四次元超立体二方陣の解は、本質的には1つしかないからである。4軸を平等に扱えるなど、数学的対称性が高いので、4つの異形をいつでも1つの基本解にまとめることができる。整理を徹底して行くと、結局のところ大基本解1個に絞られる。

今は、あえて方陣を反転・回転させて異形・派生形を生み出している。

ところが、それらの異形・派生形をそのまま二次元四方陣に変換すると、しばしば別解になる。幾つもの基本解が出来る。これでは、厳密な一対一対応にはならない。あえて言うなら、「一対多対応」なのである。

しかし、これは弱点とは言えないのかも知れない。

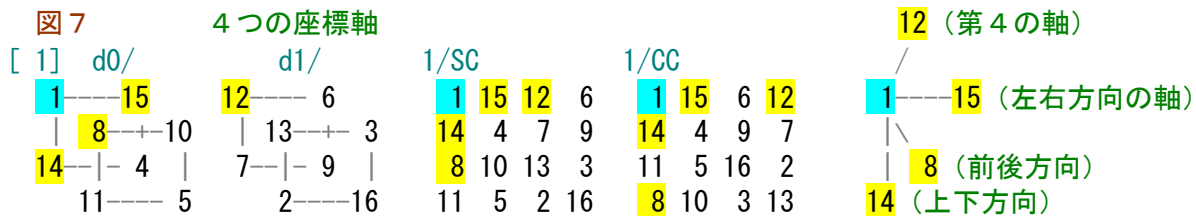
具体的に4軸の内容に踏み込んで調べよう。

四次元系では原点1の隣りには決まって(15, 14, 12, 8)の4数が4軸上に現れる。

この4数はどれも平等に扱われ、どの4軸上にも公平に現れる。

その「場合の数」は、並び方なら、4個から4個を取る順列だから、 $4P_4=24$ 通り。

絞り込んだ大基本解の数は、 $4C_4=1$ 個になる。

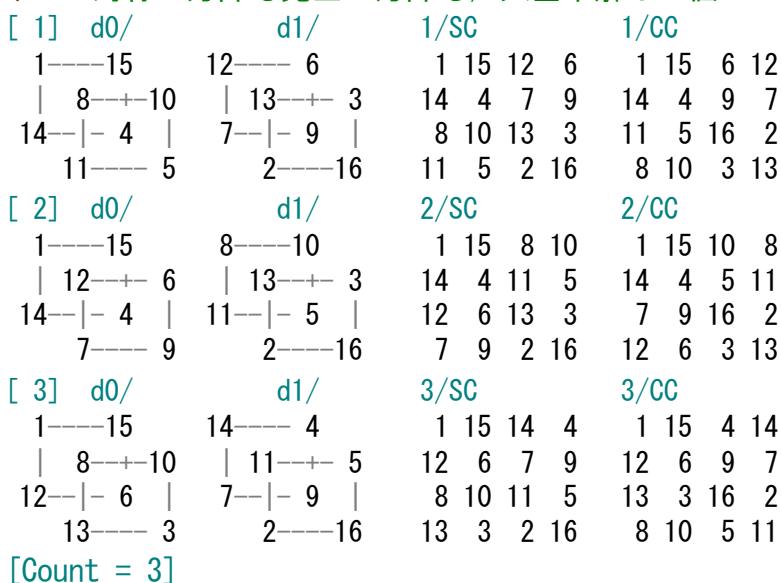


1	15	12	*	1	14	8	*	ところが、二次元系では x, y の2軸しか識別しない。
14	*	*	*	15	*	*	*	整理の対称も2軸のみ。しかも、2数ずつ2組に別れて
8	*	*	*	12	*	*	*	2軸に登場する。したがって、4個から2個を取る組み合わせ
*	*	*	*	*	*	*	*	の数値は、 $4C_2 = (4 \times 3) \div (2 \times 1) = 6$

**図9**

しかし、左図のような場合は、反転形と見なして2個でなく1個にまとめて数えるのが普通だから、 $6 \div 2 = 3$ 個。これが、大基本解の総数となる。

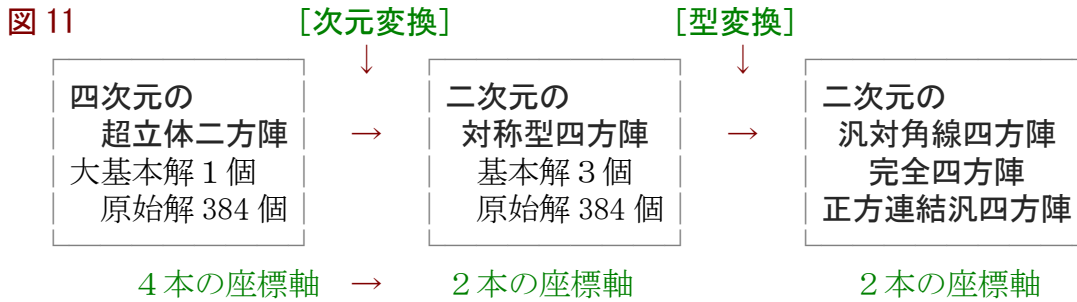
**図10 対称四方陣も完全四方陣も、大基本解は3個：**



低次元では、言わば「対称性の一部喪失」によって、基本解の数が増えてしまうのだ。



ちなみに、上の3個の基本解を得る方法は、リスト整形出力用の不等式を用いる。



ここで、対称四方陣と完全四方陣の原始解 384 個のリストを掲げておく。

**\*\* 四次元超立体二方陣から次元変換して得た二種類の四方陣の原始解384個のリスト \*\***

補数対称四方陣 SC/解番号/CC 正方連結完全四方陣

SC/ 1/CC		SC/ 2/CC		SC/ 3/CC	
1 15 12 6	1 15 6 12	1 15 8 10	1 15 10 8	1 15 14 4	1 15 4 14
14 4 7 9	14 4 9 7	14 4 11 5	14 4 5 11	12 6 7 9	12 6 9 7
8 10 13 3	11 5 16 2	12 6 13 3	7 9 16 2	8 10 11 5	13 3 16 2
11 5 2 16	8 10 3 13	7 9 2 16	12 6 3 13	13 3 2 16	8 10 5 11
SC/ 4/CC		SC/ 5/CC		SC/ 6/CC	
1 15 8 10	1 15 10 8	1 15 14 4	1 15 4 14	1 15 12 6	1 15 6 12
12 6 13 3	12 6 3 13	8 10 11 5	8 10 5 11	8 10 13 3	8 10 3 13
14 4 11 5	7 9 16 2	12 6 7 9	13 3 16 2	14 4 7 9	11 5 16 2
7 9 2 16	14 4 5 11	13 3 2 16	12 6 9 7	11 5 2 16	14 4 9 7
SC/ 7/CC		SC/ 8/CC		SC/ 9/CC	
1 14 12 7	1 14 7 12	1 14 8 11	1 14 11 8	1 14 15 4	1 14 4 15
15 4 6 9	15 4 9 6	15 4 10 5	15 4 5 10	12 7 6 9	12 7 9 6
8 11 13 2	10 5 16 3	12 7 13 2	6 9 16 3	8 11 10 5	13 2 16 3
10 5 3 16	8 11 2 13	6 9 3 16	12 7 2 13	13 2 3 16	8 11 5 10
SC/ 10/CC		SC/ 11/CC		SC/ 12/CC	
1 14 8 11	1 14 11 8	1 14 15 4	1 14 4 15	1 14 12 7	1 14 7 12
12 7 13 2	12 7 2 13	8 11 10 5	8 11 5 10	8 11 13 2	8 11 2 13
15 4 10 5	6 9 16 3	12 7 6 9	13 2 16 3	15 4 6 9	10 5 16 3
6 9 3 16	15 4 5 10	13 2 3 16	12 7 9 6	10 5 3 16	15 4 9 6
SC/ 13/CC		SC/ 14/CC		SC/ 15/CC	
1 12 14 7	1 12 7 14	1 12 8 13	1 12 13 8	1 12 15 6	1 12 6 15
15 6 4 9	15 6 9 4	15 6 10 3	15 6 3 10	14 7 4 9	14 7 9 4
8 13 11 2	10 3 16 5	14 7 11 2	4 9 16 5	8 13 10 3	11 2 16 5
10 3 5 16	8 13 2 11	4 9 5 16	14 7 2 11	11 2 5 16	8 13 3 10
SC/ 16/CC		SC/ 17/CC		SC/ 18/CC	
1 12 8 13	1 12 13 8	1 12 15 6	1 12 6 15	1 12 14 7	1 12 7 14
14 7 11 2	14 7 2 11	8 13 10 3	8 13 3 10	8 13 11 2	8 13 2 11
15 6 10 3	4 9 16 5	14 7 4 9	11 2 16 5	15 6 4 9	10 3 16 5
4 9 5 16	15 6 3 10	11 2 5 16	14 7 9 4	10 3 5 16	15 6 9 4
SC/ 19/CC		SC/ 20/CC		SC/ 21/CC	
1 8 14 11	1 8 11 14	1 8 12 13	1 8 13 12	1 8 15 10	1 8 10 15
15 10 4 5	15 10 5 4	15 10 6 3	15 10 3 6	14 11 4 5	14 11 5 4
12 13 7 2	6 3 16 9	14 11 7 2	4 5 16 9	12 13 6 3	7 2 16 9
6 3 9 16	12 13 2 7	4 5 9 16	14 11 2 7	7 2 9 16	12 13 3 6

補数対称四方陣 SC/解番号/CC 正方連結完全四方陣

SC/ 22/CC		SC/ 23/CC		SC/ 24/CC	
1 8 12 13	1 8 13 12	1 8 15 10	1 8 10 15	1 8 14 11	1 8 11 14
14 11 7 2	14 11 2 7	12 13 6 3	12 13 3 6	12 13 7 2	12 13 2 7
15 10 6 3	4 5 16 9	14 11 4 5	7 2 16 9	15 10 4 5	6 3 16 9
4 5 9 16	15 10 3 6	7 2 9 16	14 11 5 4	6 3 9 16	15 10 5 4
SC/ 25/CC		SC/ 26/CC		SC/ 27/CC	
2 16 11 5	2 16 5 11	2 16 7 9	2 16 9 7	2 16 13 3	2 16 3 13
13 3 8 10	13 3 10 8	13 3 12 6	13 3 6 12	11 5 8 10	11 5 10 8
7 9 14 4	12 6 15 1	11 5 14 4	8 10 15 1	7 9 12 6	14 4 15 1
12 6 1 15	7 9 4 14	8 10 1 15	11 5 4 14	14 4 1 15	7 9 6 12
SC/ 28/CC		SC/ 29/CC		SC/ 30/CC	
2 16 7 9	2 16 9 7	2 16 13 3	2 16 3 13	2 16 11 5	2 16 5 11
11 5 14 4	11 5 4 14	7 9 12 6	7 9 6 12	7 9 14 4	7 9 4 14
13 3 12 6	8 10 15 1	11 5 8 10	14 4 15 1	13 3 8 10	12 6 15 1
8 10 1 15	13 3 6 12	14 4 1 15	11 5 10 8	12 6 1 15	13 3 10 8
SC/ 31/CC		SC/ 32/CC		SC/ 33/CC	
2 13 11 8	2 13 8 11	2 13 7 12	2 13 12 7	2 13 16 3	2 13 3 16
16 3 5 10	16 3 10 5	16 3 9 6	16 3 6 9	11 8 5 10	11 8 10 5
7 12 14 1	9 6 15 4	11 8 14 1	5 10 15 4	7 12 9 6	14 1 15 4
9 6 4 15	7 12 1 14	5 10 4 15	11 8 1 14	14 1 4 15	7 12 6 9
SC/ 34/CC		SC/ 35/CC		SC/ 36/CC	
2 13 7 12	2 13 12 7	2 13 16 3	2 13 3 16	2 13 11 8	2 13 8 11
11 8 14 1	11 8 1 14	7 12 9 6	7 12 6 9	7 12 14 1	7 12 1 14
16 3 9 6	5 10 15 4	11 8 5 10	14 1 15 4	16 3 5 10	9 6 15 4
5 10 4 15	16 3 6 9	14 1 4 15	11 8 10 5	9 6 4 15	16 3 10 5
SC/ 37/CC		SC/ 38/CC		SC/ 39/CC	
2 11 13 8	2 11 8 13	2 11 7 14	2 11 14 7	2 11 16 5	2 11 5 16
16 5 3 10	16 5 10 3	16 5 9 4	16 5 4 9	13 8 3 10	13 8 10 3
7 14 12 1	9 4 15 6	13 8 12 1	3 10 15 6	7 14 9 4	12 1 15 6
9 4 6 15	7 14 1 12	3 10 6 15	13 8 1 12	12 1 6 15	7 14 4 9
SC/ 40/CC		SC/ 41/CC		SC/ 42/CC	
2 11 7 14	2 11 14 7	2 11 16 5	2 11 5 16	2 11 13 8	2 11 8 13
13 8 12 1	13 8 1 12	7 14 9 4	7 14 4 9	7 14 12 1	7 14 1 12
16 5 9 4	3 10 15 6	13 8 3 10	12 1 15 6	16 5 3 10	9 4 15 6
3 10 6 15	16 5 4 9	12 1 6 15	13 8 10 3	9 4 6 15	16 5 10 3
SC/ 43/CC		SC/ 44/CC		SC/ 45/CC	
2 7 13 12	2 7 12 13	2 7 11 14	2 7 14 11	2 7 16 9	2 7 9 16
16 9 3 6	16 9 6 3	16 9 5 4	16 9 4 5	13 12 3 6	13 12 6 3
11 14 8 1	5 4 15 10	13 12 8 1	3 6 15 10	11 14 5 4	8 1 15 10
5 4 10 15	11 14 1 8	3 6 10 15	13 12 1 8	8 1 10 15	11 14 4 5
SC/ 46/CC		SC/ 47/CC		SC/ 48/CC	
2 7 11 14	2 7 14 11	2 7 16 9	2 7 9 16	2 7 13 12	2 7 12 13
13 12 8 1	13 12 1 8	11 14 5 4	11 14 4 5	11 14 8 1	11 14 1 8
16 9 5 4	3 6 15 10	13 12 3 6	8 1 15 10	16 9 3 6	5 4 15 10
3 6 10 15	16 9 4 5	8 1 10 15	13 12 6 3	5 4 10 15	16 9 6 3
SC/ 49/CC		SC/ 50/CC		SC/ 51/CC	
3 16 10 5	3 16 5 10	3 16 6 9	3 16 9 6	3 16 13 2	3 16 2 13
13 2 8 11	13 2 11 8	13 2 12 7	13 2 7 12	10 5 8 11	10 5 11 8
6 9 15 4	12 7 14 1	10 5 15 4	8 11 14 1	6 9 12 7	15 4 14 1
12 7 1 14	6 9 4 15	8 11 1 14	10 5 4 15	15 4 1 14	6 9 7 12

.....

SC/ 70/CC		SC/ 71/CC		SC/ 72/CC	
3 6 10 15	3 6 15 10	3 6 16 9	3 6 9 16	3 6 13 12	3 6 12 13
13 12 8 1	13 12 1 8	10 15 5 4	10 15 4 5	10 15 8 1	10 15 1 8
16 9 5 4	2 7 14 11	13 12 2 7	8 1 14 11	16 9 2 7	5 4 14 11
2 7 11 14	16 9 4 5	8 1 11 14	13 12 7 2	5 4 11 14	16 9 7 2
SC/ 73/CC		SC/ 74/CC		SC/ 75/CC	
4 15 9 6	4 15 6 9	4 15 5 10	4 15 10 5	4 15 14 1	4 15 1 14
14 1 7 12	14 1 12 7	14 1 11 8	14 1 8 11	9 6 7 12	9 6 12 7
5 10 16 3	11 8 13 2	9 6 16 3	7 12 13 2	5 10 11 8	16 3 13 2
11 8 2 13	5 10 3 16	7 12 2 13	9 6 3 16	16 3 2 13	5 10 8 11
SC/ 76/CC		SC/ 77/CC		SC/ 78/CC	
4 15 5 10	4 15 10 5	4 15 14 1	4 15 1 14	4 15 9 6	4 15 6 9
9 6 16 3	9 6 3 16	5 10 11 8	5 10 8 11	5 10 16 3	5 10 3 16
14 1 11 8	7 12 13 2	9 6 7 12	16 3 13 2	14 1 7 12	11 8 13 2
7 12 2 13	14 1 8 11	16 3 2 13	9 6 12 7	11 8 2 13	14 1 12 7
SC/ 94/CC		SC/ 95/CC		SC/ 96/CC	
4 5 9 16	4 5 16 9	4 5 15 10	4 5 10 15	4 5 14 11	4 5 11 14
14 11 7 2	14 11 2 7	9 16 6 3	9 16 3 6	9 16 7 2	9 16 2 7
15 10 6 3	1 8 13 12	14 11 1 8	7 2 13 12	15 10 1 8	6 3 13 12
1 8 12 13	15 10 3 6	7 2 12 13	14 11 8 1	6 3 12 13	15 10 8 1
SC/ 97/CC		SC/ 98/CC		SC/ 99/CC	
5 16 10 3	5 16 3 10	5 16 4 9	5 16 9 4	5 16 11 2	5 16 2 11
11 2 8 13	11 2 13 8	11 2 14 7	11 2 7 14	10 3 8 13	10 3 13 8
4 9 15 6	14 7 12 1	10 3 15 6	8 13 12 1	4 9 14 7	15 6 12 1
14 7 1 12	4 9 6 15	8 13 1 12	10 3 6 15	15 6 1 12	4 9 7 14
SC/100/CC		SC/101/CC		SC/102/CC	
5 16 4 9	5 16 9 4	5 16 11 2	5 16 2 11	5 16 10 3	5 16 3 10
10 3 15 6	10 3 6 15	4 9 14 7	4 9 7 14	4 9 15 6	4 9 6 15
11 2 14 7	8 13 12 1	10 3 8 13	15 6 12 1	11 2 8 13	14 7 12 1
8 13 1 12	11 2 7 14	15 6 1 12	10 3 13 8	14 7 1 12	11 2 13 8
SC/118/CC		SC/119/CC		SC/120/CC	
5 4 10 15	5 4 15 10	5 4 16 9	5 4 9 16	5 4 11 14	5 4 14 11
11 14 8 1	11 14 1 8	10 15 3 6	10 15 6 3	10 15 8 1	10 15 1 8
16 9 3 6	2 7 12 13	11 14 2 7	8 1 12 13	16 9 2 7	3 6 12 13
2 7 13 12	16 9 6 3	8 1 13 12	11 14 7 2	3 6 13 12	16 9 7 2
SC/121/CC		SC/122/CC		SC/123/CC	
6 15 9 4	6 15 4 9	6 15 3 10	6 15 10 3	6 15 12 1	6 15 1 12
12 1 7 14	12 1 14 7	12 1 13 8	12 1 8 13	9 4 7 14	9 4 14 7
3 10 16 5	13 8 11 2	9 4 16 5	7 14 11 2	3 10 13 8	16 5 11 2
13 8 2 11	3 10 5 16	7 14 2 11	9 4 5 16	16 5 2 11	3 10 8 13
SC/124/CC		SC/125/CC		SC/126/CC	
6 15 3 10	6 15 10 3	6 15 12 1	6 15 1 12	6 15 9 4	6 15 4 9
9 4 16 5	9 4 5 16	3 10 13 8	3 10 8 13	3 10 16 5	3 10 5 16
12 1 13 8	7 14 11 2	9 4 7 14	16 5 11 2	12 1 7 14	13 8 11 2
7 14 2 11	12 1 8 13	16 5 2 11	9 4 14 7	13 8 2 11	12 1 14 7
SC/142/CC		SC/143/CC		SC/144/CC	
6 3 9 16	6 3 16 9	6 3 15 10	6 3 10 15	6 3 12 13	6 3 13 12
12 13 7 2	12 13 2 7	9 16 4 5	9 16 5 4	9 16 7 2	9 16 2 7
15 10 4 5	1 8 11 14	12 13 1 8	7 2 11 14	15 10 1 8	4 5 11 14
1 8 14 11	15 10 5 4	7 2 14 11	12 13 8 1	4 5 14 11	15 10 8 1

SC/145/CC		SC/146/CC		SC/147/CC	
7 14 9 4	7 14 4 9	7 14 2 11	7 14 11 2	7 14 12 1	7 14 1 12
12 1 6 15	12 1 15 6	12 1 13 8	12 1 8 13	9 4 6 15	9 4 15 6
2 11 16 5	13 8 10 3	9 4 16 5	6 15 10 3	2 11 13 8	16 5 10 3
13 8 3 10	2 11 5 16	6 15 3 10	9 4 5 16	16 5 3 10	2 11 8 13
SC/148/CC		SC/149/CC		SC/150/CC	
7 14 2 11	7 14 11 2	7 14 12 1	7 14 1 12	7 14 9 4	7 14 4 9
9 4 16 5	9 4 5 16	2 11 13 8	2 11 8 13	2 11 16 5	2 11 5 16
12 1 13 8	6 15 10 3	9 4 6 15	16 5 10 3	12 1 6 15	13 8 10 3
6 15 3 10	12 1 8 13	16 5 3 10	9 4 15 6	13 8 3 10	12 1 15 6
. . . . .					
SC/166/CC		SC/167/CC		SC/168/CC	
7 2 9 16	7 2 16 9	7 2 14 11	7 2 11 14	7 2 12 13	7 2 13 12
12 13 6 3	12 13 3 6	9 16 4 5	9 16 5 4	9 16 6 3	9 16 3 6
14 11 4 5	1 8 10 15	12 13 1 8	6 3 10 15	14 11 1 8	4 5 10 15
1 8 15 10	14 11 5 4	6 3 15 10	12 13 8 1	4 5 15 10	14 11 8 1
SC/169/CC		SC/170/CC		SC/171/CC	
8 13 10 3	8 13 3 10	8 13 1 12	8 13 12 1	8 13 11 2	8 13 2 11
11 2 5 16	11 2 16 5	11 2 14 7	11 2 7 14	10 3 5 16	10 3 16 5
1 12 15 6	14 7 9 4	10 3 15 6	5 16 9 4	1 12 14 7	15 6 9 4
14 7 4 9	1 12 6 15	5 16 4 9	10 3 6 15	15 6 4 9	1 12 7 14
SC/172/CC		SC/173/CC		SC/174/CC	
8 13 1 12	8 13 12 1	8 13 11 2	8 13 2 11	8 13 10 3	8 13 3 10
10 3 15 6	10 3 6 15	1 12 14 7	1 12 7 14	1 12 15 6	1 12 6 15
11 2 14 7	5 16 9 4	10 3 5 16	15 6 9 4	11 2 5 16	14 7 9 4
5 16 4 9	11 2 7 14	15 6 4 9	10 3 16 5	14 7 4 9	11 2 16 5
. . . . .					
SC/190/CC		SC/191/CC		SC/192/CC	
8 1 10 15	8 1 15 10	8 1 13 12	8 1 12 13	8 1 11 14	8 1 14 11
11 14 5 4	11 14 4 5	10 15 3 6	10 15 6 3	10 15 5 4	10 15 4 5
13 12 3 6	2 7 9 16	11 14 2 7	5 4 9 16	13 12 2 7	3 6 9 16
2 7 16 9	13 12 6 3	5 4 16 9	11 14 7 2	3 6 16 9	13 12 7 2
SC/193/CC		SC/194/CC		SC/195/CC	
9 16 6 3	9 16 3 6	9 16 4 5	9 16 5 4	9 16 7 2	9 16 2 7
7 2 12 13	7 2 13 12	7 2 14 11	7 2 11 14	6 3 12 13	6 3 13 12
4 5 15 10	14 11 8 1	6 3 15 10	12 13 8 1	4 5 14 11	15 10 8 1
14 11 1 8	4 5 10 15	12 13 1 8	6 3 10 15	15 10 1 8	4 5 11 14
SC/196/CC		SC/197/CC		SC/198/CC	
9 16 4 5	9 16 5 4	9 16 7 2	9 16 2 7	9 16 6 3	9 16 3 6
6 3 15 10	6 3 10 15	4 5 14 11	4 5 11 14	4 5 15 10	4 5 10 15
7 2 14 11	12 13 8 1	6 3 12 13	15 10 8 1	7 2 12 13	14 11 8 1
12 13 1 8	7 2 11 14	15 10 1 8	6 3 13 12	14 11 1 8	7 2 13 12
. . . . .					
SC/358/CC		SC/359/CC		SC/360/CC	
15 1 4 14	15 1 14 4	15 1 10 8	15 1 8 10	15 1 6 12	15 1 12 6
6 12 9 7	6 12 7 9	4 14 5 11	4 14 11 5	4 14 9 7	4 14 7 9
10 8 5 11	3 13 2 16	6 12 3 13	9 7 2 16	10 8 3 13	5 11 2 16
3 13 16 2	10 8 11 5	9 7 16 2	6 12 13 3	5 11 16 2	10 8 13 3
SC/361/CC		SC/362/CC		SC/363/CC	
16 9 3 6	16 9 6 3	16 9 2 7	16 9 7 2	16 9 5 4	16 9 4 5
5 4 10 15	5 4 15 10	5 4 11 14	5 4 14 11	3 6 10 15	3 6 15 10
2 7 13 12	11 14 1 8	3 6 13 12	10 15 1 8	2 7 11 14	13 12 1 8
11 14 8 1	2 7 12 13	10 15 8 1	3 6 12 13	13 12 8 1	2 7 14 11

<p>SC/364/CC</p> <p>16 9 2 7 16 9 7 2 3 6 13 12 3 6 12 13 5 4 11 14 10 15 1 8 10 15 8 1 5 4 14 11</p>	<p>SC/365/CC</p> <p>16 9 5 4 16 9 4 5 2 7 11 14 2 7 14 11 3 6 10 15 13 12 1 8 13 12 8 1 3 6 15 10</p>	<p>SC/366/CC</p> <p>16 9 3 6 16 9 6 3 2 7 13 12 2 7 12 13 5 4 10 15 11 14 1 8 11 14 8 1 5 4 15 10</p>
<p>SC/367/CC</p> <p>16 5 3 10 16 5 10 3 9 4 6 15 9 4 15 6 2 11 13 8 7 14 1 12 7 14 12 1 2 11 8 13</p>	<p>SC/368/CC</p> <p>16 5 2 11 16 5 11 2 9 4 7 14 9 4 14 7 3 10 13 8 6 15 1 12 6 15 12 1 3 10 8 13</p>	<p>SC/369/CC</p> <p>16 5 9 4 16 5 4 9 3 10 6 15 3 10 15 6 2 11 7 14 13 8 1 12 13 8 12 1 2 11 14 7</p>
<p>SC/370/CC</p> <p>16 5 2 11 16 5 11 2 3 10 13 8 3 10 8 13 9 4 7 14 6 15 1 12 6 15 12 1 9 4 14 7</p>	<p>SC/371/CC</p> <p>16 5 9 4 16 5 4 9 2 11 7 14 2 11 14 7 3 10 6 15 13 8 1 12 13 8 12 1 3 10 15 6</p>	<p>SC/372/CC</p> <p>16 5 3 10 16 5 10 3 2 11 13 8 2 11 8 13 9 4 6 15 7 14 1 12 7 14 12 1 9 4 15 6</p>
<p>SC/373/CC</p> <p>16 3 5 10 16 3 10 5 9 6 4 15 9 6 15 4 2 13 11 8 7 12 1 14 7 12 14 1 2 13 8 11</p>	<p>SC/374/CC</p> <p>16 3 2 13 16 3 13 2 9 6 7 12 9 6 12 7 5 10 11 8 4 15 1 14 4 15 14 1 5 10 8 11</p>	<p>SC/375/CC</p> <p>16 3 9 6 16 3 6 9 5 10 4 15 5 10 15 4 2 13 7 12 11 8 1 14 11 8 14 1 2 13 12 7</p>
<p>SC/376/CC</p> <p>16 3 2 13 16 3 13 2 5 10 11 8 5 10 8 11 9 6 7 12 4 15 1 14 4 15 14 1 9 6 12 7</p>	<p>SC/377/CC</p> <p>16 3 9 6 16 3 6 9 2 13 7 12 2 13 12 7 5 10 4 15 11 8 1 14 11 8 14 1 5 10 15 4</p>	<p>SC/378/CC</p> <p>16 3 5 10 16 3 10 5 2 13 11 8 2 13 8 11 9 6 4 15 7 12 1 14 7 12 14 1 9 6 15 4</p>
<p>SC/379/CC</p> <p>16 2 5 11 16 2 11 5 9 7 4 14 9 7 14 4 3 13 10 8 6 12 1 15 6 12 15 1 3 13 8 10</p>	<p>SC/380/CC</p> <p>16 2 3 13 16 2 13 3 9 7 6 12 9 7 12 6 5 11 10 8 4 14 1 15 4 14 15 1 5 11 8 10</p>	<p>SC/381/CC</p> <p>16 2 9 7 16 2 7 9 5 11 4 14 5 11 14 4 3 13 6 12 10 8 1 15 10 8 15 1 3 13 12 6</p>
<p>SC/382/CC</p> <p>16 2 3 13 16 2 13 3 5 11 10 8 5 11 8 10 9 7 6 12 4 14 1 15 4 14 15 1 9 7 12 6</p>	<p>SC/383/CC</p> <p>16 2 9 7 16 2 7 9 3 13 6 12 3 13 12 6 5 11 4 14 10 8 1 15 10 8 15 1 5 11 14 4</p>	<p>SC/384/CC</p> <p>16 2 5 11 16 2 11 5 3 13 10 8 3 13 8 10 9 7 4 14 6 12 1 15 6 12 15 1 9 7 14 4</p>

[Count = 384/384] [OK!]

\*\* Recompiled and Listed by Kanji Setsuda on  
March 2, 2015 with MacOSX 10.10.2 & Xcode 6.1.1 \*\*

原始解 384個ではなく標準解 48個×2の解集合を求めたければ、例えば、次のような不等式を用いて、上の結果から解を選別すれば良い。

$n1 < n4$ ,  $n1 < 13$ ,  $n1 < n16$ , and  $n2 > n5$ ;

ただし、標準解どうしてもでは、対称型と完全型との対応関係は一部見失われてしまうが。

8. 今回は、あえて「ひな形変換法」には触れなかった。もちろん、いつでも使うことができるし、結果は予想通りになるが、必ずしも新しい発見・知識をもたらすものではない。だから、省いた。

四方陣の秘密解明には、今回の「次元変換法」だけでも、十分に迫り得るかと思う。

しかし、今あらためて不思議に思う。四次元超立体二方陣と言うのは何物なのか。単なる「思考実験」に終わるものでないことは、四方陣との関連で明らかだ。

想像力たくましい人ならば、この四次元超立体二方陣を、理想世界の「アイデア」と考えるかも知れない。或いは、全能の神の抱いたコンセプトと考えるかも知れない。四次元世界を直視できる完全存在＝神ならば、この魔方陣の完全なる対称性をきっとそのまま楽しめるだろうから。

それが現実世界に「現象」として押し出されて来ると、例の四方陣になる。低次元に変換する時に対称性の一部を失う。そのかわりに「個性」を獲得して、対称型・完全型・正方連結型などになる。そして、これならばわかる我々平凡なる人間を大いに楽しませてくれる（実は悩ませるだけなのかもしれないが…）。

今回の研究では、四次元二方陣と「正方連結型」との運命的な関係が印象に残った。

他の八方陣、十二方陣、十六方陣の正方連結型ではどうなっているのだろうか。

というのは、例えば八方陣は、 $64 = 2 \times 2 \times 16 = 2 \times 2 \times (2 \times 2 \times 2 \times 2)$  だから。

前節の四次元超立体三方陣から多重対称九方陣に書き換えた経験から言えば、解を見つけるスピードは最初から九方陣で探索した時に比べて断然速い。条件式が多い分だけ、解を見つけやすくなるようだ。六次元超立体二方陣と八方陣の研究に希望が持てる。

その意味でも、次元数を増やして次数を減らす知的な探求方法にはメリットがある。

(初稿 : in 2001; 再計算改訂稿 : Aug. 16, 2005; Jun. 6, 2009; Mar. 3, 2013; Mar. 2, 2015;  
計算・作表・執筆 : by 撰田 寛二(Kanji Setsuda) with MacOSX 10.10.2 and Xcode 6.1.1;)

E-Mail Address:<jag12001@nifty.com>

## << 付録 : プログラム・リスト >>

```
/** Compose the 4-Dimensional Magic Objects of Order 2 and **
/** Transform them into Two Types of MS44:'S-C' and 'C&C' **
/** File:'EC024A384.c' Dictated by Kanji Setsuda **
/** on Jan.14, 2006; Revised on March 3, 2013; **
/** Recompiled on March 2, 2015 **
/** with MacOSX 10.10.2 & Xcode 6.1.1 **
//
/** Basic Diagrams expressing Positions of Elements ***
// BD/d0      /d1      SC/      CC/
// 1---- 2      3---- 4      1 2 3 4      1 2 4 3
// | 9---+10    | 11---+12    5 6 7 8      5 6 8 7
// 5--|- 6 |    7--|- 8 |      9 10 11 12    13 14 16 15
// 13----14      15----16    13 14 15 16      9 10 12 11
//
/** Definition of Self-complementary Pairs **
// n1+n16=n2+n15=n3+n14=n4+n13=n5+n12=n6+n11=n7+n10=n8+n9=17 ...scc
//
/** The 12 Essential 'Composite Conditions' **
// n1+n2+n3+n4=C ... (1);    n1+n2+n5+n6=C ... (2);    n1+n2+n9+n10=C ... (3);
// n1+n3+n5+n7=C ... (4);    n1+n3+n9+n11=C ... (5);    n1+n5+n9+n13=C ... (6);
// n2+n4+n6+n8=C ... (7);    n2+n4+n10+n12=C ... (8);    n2+n6+n10+n14=C ... (9);
// n3+n4+n7+n8=C ... (10);   n3+n4+n11+n12=C ... (11);   n5+n6+n7+n8=C ... (12);
//
#include <stdio.h>
//
/** Variables *
short int cnt;
short CC, LSM, cnt2;
short nm[17], uflg[17];
short t[384][17];
//
/** Sub-Routines *
void stp01(void), stp02(void), stp03(void);
```

```

void stp04(void), stp05(void), stp06(void);
void stp07(void), stp08(void), stp09(void);
void stp10(void), stp11(void), stp12(void);
void ansrecrd(void);
void pr4f24(void), pr3scms2(void);
//
/* Main Program *
int main(){
short n;
printf("¥n");
printf("*** Compose the 4-Dimensional Magic Objects of Order 2 and **¥n");
printf("*** Transform them into Two Types of MS44:'S-C' and 'C&C' **¥n");
for(n=0;n<17;n++){nm[n]=0; uflg[n]=0;}
CC=17; LSM=34; cnt=0;
stp01();      /* Begin the Calculation *
pr4f24();     /* Print the Answer in 4 Forms *
printf("¥n");
printf("*** Another List: Three Comparative Pairs of Solutions **¥n");
pr3scms2();  /* Print the 3 Answers only in 2 Forms *
printf(" [Count = %d] [OK!]¥n",cnt);
printf("¥n");
printf("*** Recompiled and Listed by Kanji Setsuda on¥n");
printf("  March 2, 2015 with MacOSX 10.10.2 & Xcode 6.1.1 **¥n");
printf("¥n");
return 0;
}
//
/* Calculations *
/* Set n1 & n16 *
void stp01(){
short a,b;
for(a=1;a<17;a++){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[1]=a; nm[16]=b;
uflg[a]=1; uflg[b]=1;
stp02();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set n2 & n15 *
void stp02(){
short a,b;
for(a=16;a>0;a--){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[2]=a; nm[15]=b;
uflg[a]=1; uflg[b]=1;
stp03();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set n5 & n12 *
void stp03(){
short a,b;
for(a=16;a>0;a--){b=CC-a;
if((uflg[a]==0)&&(uflg[b]==0)){
nm[5]=a; nm[12]=b;
uflg[a]=1; uflg[b]=1;
stp04();
uflg[b]=0; uflg[a]=0;}
}
}
/* Set n6=LSM-n1-n2-n5 & n11 *
void stp04(){
short a,b;
a=LSM-nm[1]-nm[2]-nm[5];

```

```

    if((0<a)&&(a<17)){b=CC-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[6]=a; nm[11]=b;
            uflg[a]=1; uflg[b]=1;
            stp05();
            uflg[b]=0; uflg[a]=0;}}
}
/* Set n3 & n14 *
void stp05(){
    short a,b;
    for(a=16;a>0;a--){b=CC-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[3]=a; nm[14]=b;
            uflg[a]=1; uflg[b]=1;
            stp06();
            uflg[b]=0; uflg[a]=0;}}
}
}
/* Set n4=LSM-n1-n2-n3 & n13 *
void stp06(){
    short a,b;
    a=LSM-nm[1]-nm[2]-nm[3];
    if((0<a)&&(a<17)){b=CC-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[4]=a; nm[13]=b;
            uflg[a]=1; uflg[b]=1;
            stp07();
            uflg[b]=0; uflg[a]=0;}}
}
}
/* Set n7=LSM-n1-n3-n5 & n10 *
void stp07(){
    short a,b;
    a=LSM-nm[1]-nm[3]-nm[5];
    if((0<a)&&(a<17)){b=CC-a;
        if((0<b)&&(uflg[a]==0)&&(uflg[b]==0)){
            nm[7]=a; nm[10]=b;
            uflg[a]=1; uflg[b]=1;
            stp08();
            uflg[b]=0; uflg[a]=0;}}
}
}
/* Set n8=LSM-n2-n4-n6 & n9 *
void stp08(){
    short a,b,c;
    a=LSM-nm[2]-nm[4]-nm[6];
    b=LSM-nm[3]-nm[4]-nm[7];
    c=LSM-nm[5]-nm[6]-nm[7];
    if((0<a)&&(a<17)&&(a==b)&&(a==c)){b=CC-a;
        if((uflg[a]==0)&&(uflg[b]==0)){
            nm[8]=a; nm[9]=b;
            uflg[a]=1; uflg[b]=1; cnt2=0;
            stp09();
            uflg[b]=0; uflg[a]=0;}}
}
}
/* Check some Line-sums *
void stp09(){
    short sm1,sm2,sm3;
    sm1=nm[1]+nm[2]+nm[9]+nm[10];
    sm2=nm[1]+nm[3]+nm[9]+nm[11];
    sm3=nm[1]+nm[5]+nm[9]+nm[13];
    if((sm1==LSM)&&(sm2==LSM)&&(sm3==LSM)){stp10();}
}
}
/* Check some Line-sums *
void stp10(){
    short sm1,sm2,sm3;
    sm1=nm[2]+nm[4]+nm[10]+nm[12];

```



```

sm2=nm[2]+nm[6]+nm[10]+nm[14];
sm3=nm[3]+nm[4]+nm[11]+nm[12];
if((sm1==LSM)&&(sm2==LSM)&&(sm3==LSM)){ansrecrd();}
}
//
void ansrecrd(void){
short n;
t[cnt][0]=cnt+1;
for(n=1;n<17;n++){t[cnt][n]=nm[n];}
cnt++;
}
//
/* Print the Answer of Order 2^4 in 4 Forms *
void pr4f24(void){
short m,n;
short tn[17];
for(m=0;m<24;m++){
for(n=0;n<17;n++){tn[n]=t[m][n];}
printf("%3d/d0      /d1      SC/      CC/¥n", tn[0]);
printf("%4d----%2d%7d----%2d  ", tn[1], tn[2], tn[3], tn[4]);
printf(" %3d%3d%3d%3d %3d%3d%3d%3d¥n",
tn[1], tn[2], tn[3], tn[4], tn[1], tn[2], tn[4], tn[3]);
printf(" | %2d--+%2d | %2d--+%2d", tn[9], tn[10], tn[11], tn[12]);
printf(" %3d%3d%3d%3d %3d%3d%3d%3d¥n",
tn[5], tn[6], tn[7], tn[8], tn[5], tn[6], tn[8], tn[7]);
printf("%4d--l-%2d l%4d--l-%2d l", tn[5], tn[6], tn[7], tn[8]);
printf(" %3d%3d%3d%3d %3d%3d%3d%3d¥n",
tn[9], tn[10], tn[11], tn[12], tn[14], tn[16], tn[15]);
printf("%7d----%2d%7d----%2d", tn[13], tn[14], tn[15], tn[16]);
printf(" %3d%3d%3d%3d %3d%3d%3d%3d¥n",
tn[13], tn[14], tn[15], tn[16], tn[9], tn[10], tn[12], tn[11]);
}
}
//
/* Print the Solutions of Order 2^4 in 2 Forms Transformed *
void pr3scms2(void){
short m,n,p;
short tn[3][17];
for(m=0;m<cnt;m=m+3){
for(p=0;p<3;p++){for(n=0;n<17;n++){tn[p][n]=t[m+p][n];}}
printf("      SC/%3d/CC      SC/%3d/CC      SC/%3d/CC¥n",
tn[0][0], tn[1][0], tn[2][0]);
printf("%3d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d¥n",
tn[0][1], tn[0][2], tn[0][3], tn[0][4], tn[0][1], tn[0][2], tn[0][4], tn[0][3],
tn[1][1], tn[1][2], tn[1][3], tn[1][4], tn[1][1], tn[1][2], tn[1][4], tn[1][3],
tn[2][1], tn[2][2], tn[2][3], tn[2][4], tn[2][1], tn[2][2], tn[2][4], tn[2][3]);
printf("%3d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d¥n",
tn[0][5], tn[0][6], tn[0][7], tn[0][8], tn[0][5], tn[0][6], tn[0][8], tn[0][7],
tn[1][5], tn[1][6], tn[1][7], tn[1][8], tn[1][5], tn[1][6], tn[1][8], tn[1][7],
tn[2][5], tn[2][6], tn[2][7], tn[2][8], tn[2][5], tn[2][6], tn[2][8], tn[2][7]);
printf("%3d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d¥n",
tn[0][9], tn[0][10], tn[0][11], tn[0][12], tn[0][13], tn[0][14], tn[0][16], tn[0][15],
tn[1][9], tn[1][10], tn[1][11], tn[1][12], tn[1][13], tn[1][14], tn[1][16], tn[1][15],
tn[2][9], tn[2][10], tn[2][11], tn[2][12], tn[2][13], tn[2][14], tn[2][16], tn[2][15]);
printf("%3d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d%6d%3d%3d%3d%5d%3d%3d%3d¥n",
tn[0][13], tn[0][14], tn[0][15], tn[0][16], tn[0][9], tn[0][10], tn[0][12], tn[0][11],
tn[1][13], tn[1][14], tn[1][15], tn[1][16], tn[1][9], tn[1][10], tn[1][12], tn[1][11],
tn[2][13], tn[2][14], tn[2][15], tn[2][16], tn[2][9], tn[2][10], tn[2][12], tn[2][11]);
}
}
//

```