補足［超立方体の頂点と線分についてのアルゴリズム］

システム工学概論

6. Cによる数値計算と可視化（発展編） から抜粋の補足資料

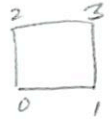
山田泰司

taiji@aihara.co.jp

株式会社あいはら 研究開発チーム

$n=2$次元 正方形

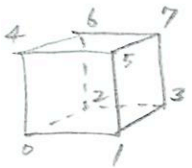
頂点番号



y	x	頂点番号
0	0	0
0	1	1
1	0	2
1	1	3

$n=3$次元 立方体

頂点番号



z	y	x	頂点番号
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

$n=4, 5\cdots$ 超立方体は…

立方体の頂点と線分（続き）

立方体の頂点と線分を渡すルーチン：

頂点の座標値について
下限値と上限値が与えられた時…

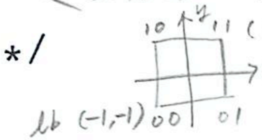
```
void cube3(double lb[3], double ub[3],
           double vertices[8][3], int segments[12*2])
{
  int i, j;
  static int segs[12*2] = { /* 線分の数: 12 */
    0,4, 2,6, 1,5, 3,7,
    0,2, 4,6, 1,3, 5,7,
    0,1, 4,5, 2,3, 6,7,
  };
  /* for (i=0; i<12*2; i++) segments[i] = segs[i]; */
  for (i=0; i<3; i++)
    for (j=0; j<12/3; j++) {
      segments[(12/3*i+j)*2+0] = segs[(12/3*i+j)*2+0];
      segments[(12/3*i+j)*2+1] = segs[(12/3*i+j)*2+1];
    }
  for (i=0; i<8; i++) { /* 頂点の数: 8 */
    vertices[i][0] = ((i>>0)%2) ? ub[0] : lb[0];
    vertices[i][1] = ((i>>1)%2) ? ub[1] : lb[1];
    vertices[i][2] = ((i>>2)%2) ? ub[2] : lb[2];
  }
}
```

線分のテーブルの任意の超立方体で得たい

頂点の座標値を得るコード

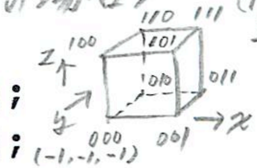
線分：頂点番号のペアを表わすテーブルが得られているなら「コピー」するだけ！

＜正方形の場合＞



i=0〜3の頂点
iのx ⇒ (i>>0)%2が 0|lb 1|ub のx成分
iのy ⇒ (i>>1)%2が 0|lb 1|ub のy成分

＜立方体の場合＞



i=0〜7の頂点
iのx ⇒ (i>>0)%2が 0|lb 1|ub のx成分
iのy ⇒ (i>>1)%2が 0|lb 1|ub のy成分
iのz ⇒ (i>>2)%2が 0|lb 1|ub のz成分

n=3
頂点の数 $nv = 2^3 = 8$
線分の数 $ns = 2^{3-1} \times 3 = 12$

立方体の頂点と線分（続き）

n=2 正方形の頂点と線分の数
$nv = 2^2 = 4$
$ns = 2^{2-1} \times 2 = 4$

任意の n
$nv = 2^n$
$ns = 2^{n-1} \times n$

立方体の頂点と線分を求めるルーチン：

```
void cube3(double lb[3], double ub[3],
           double vertices[8][3], int segments[12*2])
{
  int i, j;

  for (i=0; i<3; i++)
    for (j=0; j<12/3; j++) { /* 線分の数: 2^2*3 */
      int v0 = 0, v1 = 0, k;
      for (k=0; k<3; k++) {
        v0 = ((i>k)?((j>>k)%2):(i==k)?0:((j>>(k-1))%2)) + 2*v0;
        v1 = ((i>k)?((j>>k)%2):(i==k)?1:((j>>(k-1))%2)) + 2*v1;
      }
      segments[(12/3*i+j)*2+0] = v0;
      segments[(12/3*i+j)*2+1] = v1;
    }
  for (i=0; i<1<<3; i++) /* 頂点の数: 2^3 */
    for (j=0; j<3; j++)
      vertices[i][j] = ((i>>j)%2) ? ub[j] : lb[j];
}
```

立方体の線分（頂点番号のペア）テーブルを生成するアルゴリズム
☞ 次頁参照

頂点の座標値を得るコードの改良版

立方体の頂点と線分

$n=3$

線分：頂点番号のペア

xy面

zyx	v_0	zyx	v_1
000	0	100	4
010	2	110	6
001	1	101	5
011	3	111	7

$v_1 - v_0 = 100_2 = 4$

xz面

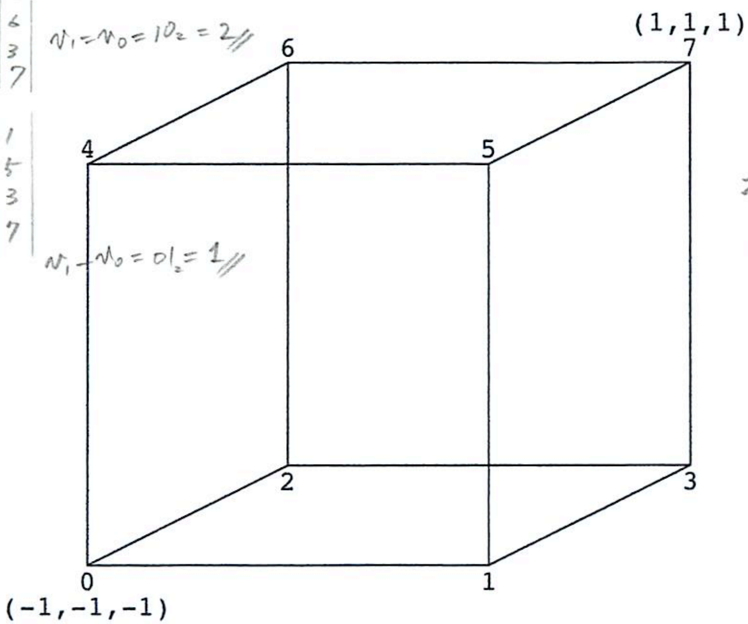
zyx		zyx	
000	0	010	2
100	4	110	6
001	1	011	3
101	5	111	7

$v_1 - v_0 = 10_2 = 2$

yz面

zyx		zyx	
000	0	001	1
100	4	101	5
010	2	011	3
110	6	111	7

$v_1 - v_0 = 01_2 = 1$



線分

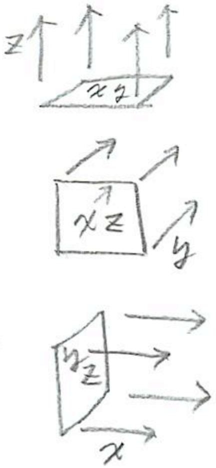
$\overbrace{n-1}^{1つ低い}$ 次元の空間から その頂点個分の線分が余次元方向に結ばれる。

次元方向 n

$i=0$ $j=0\sim3$ ${}^{nS}/_{n}$ 本の頂点からの線分

$i=1$ 〃 〃

$i=2$ 〃 〃



線分：

(0,4), (2,6), (1,5), (3,7),

(0,2), (4,6), (1,3), (5,7),

(0,1), (4,5), (2,3), (6,7),

$n=2$ 正方形

線分：頂点番号のペア

x面

yx	v_0	yx	v_1
00	0	10	2
01	1	11	3

$v_1 - v_0 = 10_2 = 2$

y面

yx		yx	
00	0	01	1
10	2	11	3

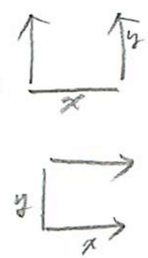
$v_1 - v_0 = 01_2 = 1$

正方形

2次元方向

$i=0$ $j=0\sim1$ $4/2 = {}^{nS}/_{n}$ 本の頂点からの線分

$i=1$ 〃 〃



超立方体の頂点と線分（続き）

超立方体の頂点と線分を求めるルーチン：

任意のnについて
線分テーブルを生成
するアルゴリズム
☞ 次頁参照

```
void cube(int n, double lb[], double ub[],
          double **vertices, int *segments)
{
  long i, j, nv = 1<<n, ns = (1<<(n-1))*n;

  for (i=0; i<n; i++)
    for (j=0; j<ns/n; j++) { /* 線分の数: 2^(n-1)*n */
      int v0 = 0, v1 = 0, k;
      for (k=0; k<n; k++) {
        v0 = ((i>k)?((j>>k)%2):(i==k)?0:((j>>(k-1))%2)) + 2*v0;
        v1 = ((i>k)?((j>>k)%2):(i==k)?1:((j>>(k-1))%2)) + 2*v1;
      }
      segments[(ns/n*i+j)*2+0] = v0;
      segments[(ns/n*i+j)*2+1] = v1;
    }
  for (i=0; i<nv; i++) /* 頂点の数: 2^n */
    for (j=0; j<n; j++)
      vertices[i][j] = ((i>>j)%2) ? ub[j] : lb[j];
}
```

i: 次元分nのループ

j: ns/n 個のループ

線分テーブル生成

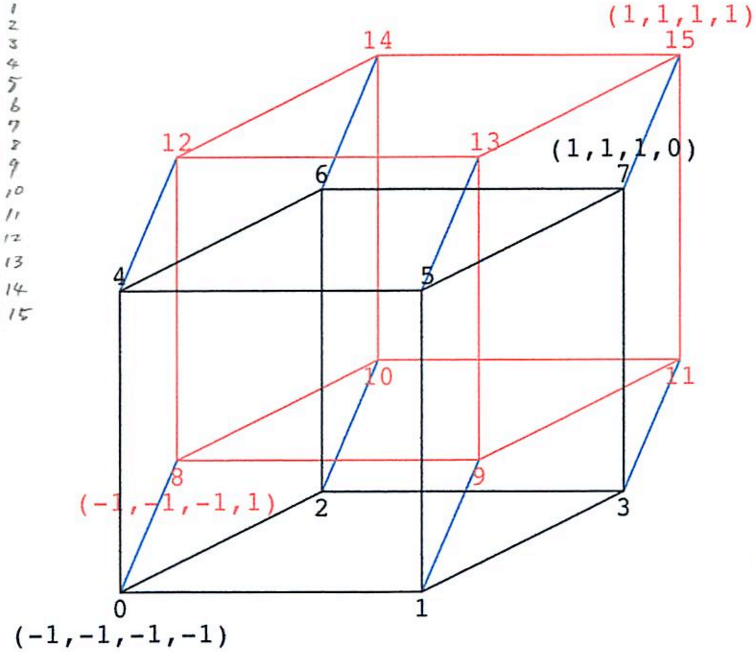
頂点の座標値

n次元の超立方体の

超立方体の頂点と線分

$n=4$ 超立方体
頂点番号

wzyx	
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15



$n=4$
線分：頂点番号のペア

xyz 面

wzyx	v_0	wzyx	v_1
0000	0	1000	8
0100	4	1100	12
0010	2	1010	10
0110	6	1110	14
0001	1	1001	9
0101	5	1101	13
0011	3	1011	11
0111	7	1111	15

$v_1 - v_0 = 1000_2 = 8$

xyw 面

wzyx	v_0	wzyx	v_1
0000	0	0100	4
1000	8	1100	12
0010	2	0110	6
1010	10	1110	14
0001	1	0101	5
1001	9	1101	13
0011	3	0111	7
1011	11	1111	15

$v_1 - v_0 = 100_2 = 4$

xzw 面

wzyx	v_0	wzyx	v_1
0000	0	0010	2
1000	8	1010	10
0100	4	0110	6
1100	12	1110	14
0001	1	0011	3
1001	9	1011	11
0101	5	0111	7
1101	13	1111	15

$v_1 - v_0 = 10_2 = 2$

yzw 面

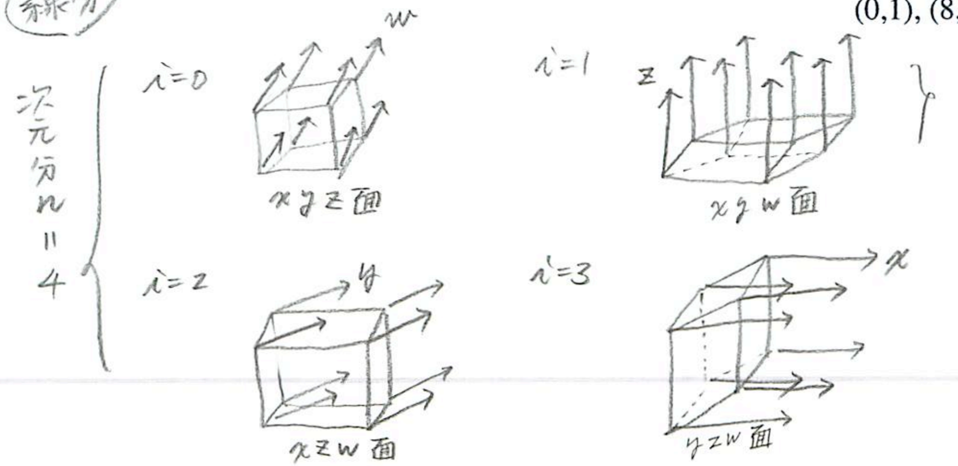
wzyx	v_0	wzyx	v_1
0000	0	0001	1
1000	8	1001	9
0100	4	0101	5
1100	12	1101	13
0010	2	0011	3
1010	10	1011	11
0110	6	0111	7
1110	14	1111	15

$v_1 - v_0 = 01_2 = 1$

線分：

(0,8), (4,12), (2,10), (6,14), (1,9), (5,13), (3,11), (7,15)
(0,4), (8,12), (2,6), (10,14), (1,5), (9,13), (3,7), (11,15)
(0,2), (8,10), (4,6), (12,14), (1,3), (9,11), (5,7), (13,15)
(0,1), (8,9), (4,5), (12,13), (2,3), (10,11), (6,7), (14,15)

線分



n_S/n 本の頂点からの線分

$2^{4-1} \times 4/4 = 2^3 = 8$