

フラクタル構造を利用した地域情報のモデル化について

中 村 有 一

Modeling of Regional Information by using Fractal Structure

Yuichi Nakamura

地域間の情報の流れを分析する場合、エントロピー・モデルやクラスター分析などの手法が、従来利用されてきたが、通話ODデータのモデル化においてはうまくいかない場合がある。このようなデータをモデル化するにあたり、データの持つフラクタル構造に着目し、それを利用することにより、新たな手法を開発できる可能性がある。本論文では、このための基礎的な研究として、データに潜在しているフラクタル的な構造を人工的に再現する方法について考察した。フラクタル画像圧縮の分野で利用されている反復関数系を通話ODデータに適用することにより、フラクタル的特徴を持ったデータを生成させることが可能となった。

Entropy model and cluster analysis are used as a method of analysing the inter-regional information flow data, such as phone-call data. But, it is difficult to apply to some complicated data. In such cases, it is effective to consider the data characteristics of fractal structure. This paper describes the basic method for modeling the inter-regional phone-call data by using IFS(iterated function system) which is used in the field of fractal image compression. The modeling method is able to generate artificial data similar to the phone-call data.

地域情報、情報流動量、ODデータ、エントロピー・モデル、距離、クラスター分析、フラクタル、アフィン変換、反復関数系、コラージュ定理

Regional information, Information flow, OD(origin-destination) data, Entropy model, Distance, Cluster analysis, Fractal, Affine transformation, Iterated function, System(IFS), Collage theorem

(原稿受領日 2000. 10. 16)

I. はじめに

地域間の情報流動量のデータを分析する場合、いろいろな分析手法が、従来から使われてきた。たとえば、グラビティ・モデルやエントロピー・モデルなどが使われることがある。⁽³⁾ これらは一般には、「空間的相互作用モデル」と呼ばれているものである。これらのモデルに共通する考え方は、地域の規模と地域間の距離や移動コス

トに相当する量をもとにして地域間の流動量を説明するものである。この種のモデルは、ある程度、現実のデータに当てはまるものであるが、大都市の周辺など、都市が連続して分布しているような地域では、説明しきれない成分が現れる。^{(7),(8)} また、東京-大阪間などの遠距離の大都市間などの流通量などもモデルからはずれることが知られている。⁽⁵⁾ このような距離やコストによるモデル化によらず、地域間の流通量を説

明することはできないかという視点から、本論文では、地域情報のフラクタル的な構造に着目し、この性質を利用して、新たな分析モデルを提案しようと試みている。

対象とするデータは、たとえば図1のようなデータである。これは、NTTの公表している「平成10年度の都道府県間の通話回数のデータ」である。この図は、行列の形になっているODデータを3次元のグラフに表したもので、行列の対角成分にあたる要素(すなわち地域内の通話量)が非常に大きく、他の成分が相対的に少なくなっていることに特徴がある。対角成分をのぞいたデータについてもグラフに表してみた。その結果が、図2である。この図をみると、フラクタル的な構造が、現れていることが予想できる。たとえば、東京の周辺に現れる山を縮小して平行移動したものが大阪や名古屋の周辺に現れる山を近似できると考えることができる。

II. 反復関数系によるODデータの生成

自然界にあるフラクタル構造をもつデータ、たとえばシダの葉の形などは、「反復関数系(Iterated Function System: IFS)」によって、簡単なデータから生成できることが知られている。反復関数系による2次元の画像の生成には、以下のようなアフィン変換が使われる。

$$w\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \quad \dots(1)$$

この変換には、「拡大・縮小」、「回転」、「平行移動」という3つの要素が含まれているが、フラクタルデータを生成するためには、「縮小写像」である必要がある。ODデータの場合には、回転の要素はないので、縮小と平行移動だけに限定される。さらに、通話回数などのデータでは対称性が高いという点を考慮すると、縦方向と横方向の縮小率はほぼ同じと仮定することができる。(ここで対称性が高いということは、A地域

からB地域への通話量が、B地域からA地域への通話量にほぼ等しいことを意味している。)したがって、アフィン変換の式は、以下のように特殊化してよいことになる。(縮小写像であるため、 $a < 1$ となる。)

$$w\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & 0 \\ 0 & a \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} \quad (0 < a < 1) \quad \dots(2)$$

ODデータをフラクタル的なデータとして近似的に生成する場合に、前提として考えておくべき点がある。それはODデータには、もともといくつかの異質な成分が混ざっているという仮定である。これは今までの通話量の分析から得られた仮定で、ここでは必要最低限という意味で、以下の3つの成分に分けて考えていくことにする。^{(4),(5),(6)}

1. 対角成分の生成

ODデータを近似的に生成する場合、対角要素は他の地域間のデータとは、質的に異なる点がある。また、量的にも桁違いに大きいので、これは他の成分とは別に扱う必要がある。この成分については、対角要素に沿った方向に平行移動するアフィン変換を使うことにする。すなわち(2)式において、以下の条件が成り立っているものとする。

$$e = f \quad \dots(3)$$

2. 近接作用の近似

次に異なる地域間の関係であるが、近接作用と遠隔作用は、異質なものであると仮定した。これはもし同一の反復関数系で記述しようとしても破綻を来してしまうためである。ここでいう近接作用は、たとえば東京の周辺の地域で、通話量が距離とともに減衰していくような場合に見られる。これは、大きい山をその周辺に縮小

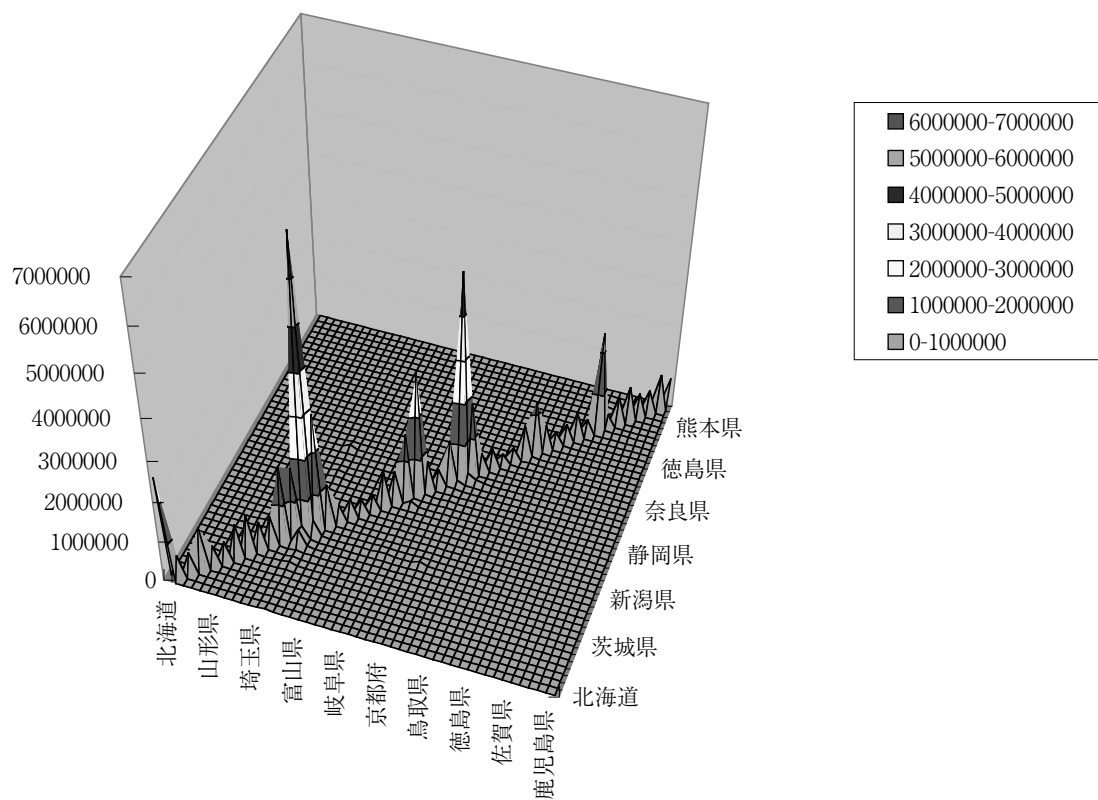


図1 平成10年度の都道府県間通話回数のグラフ

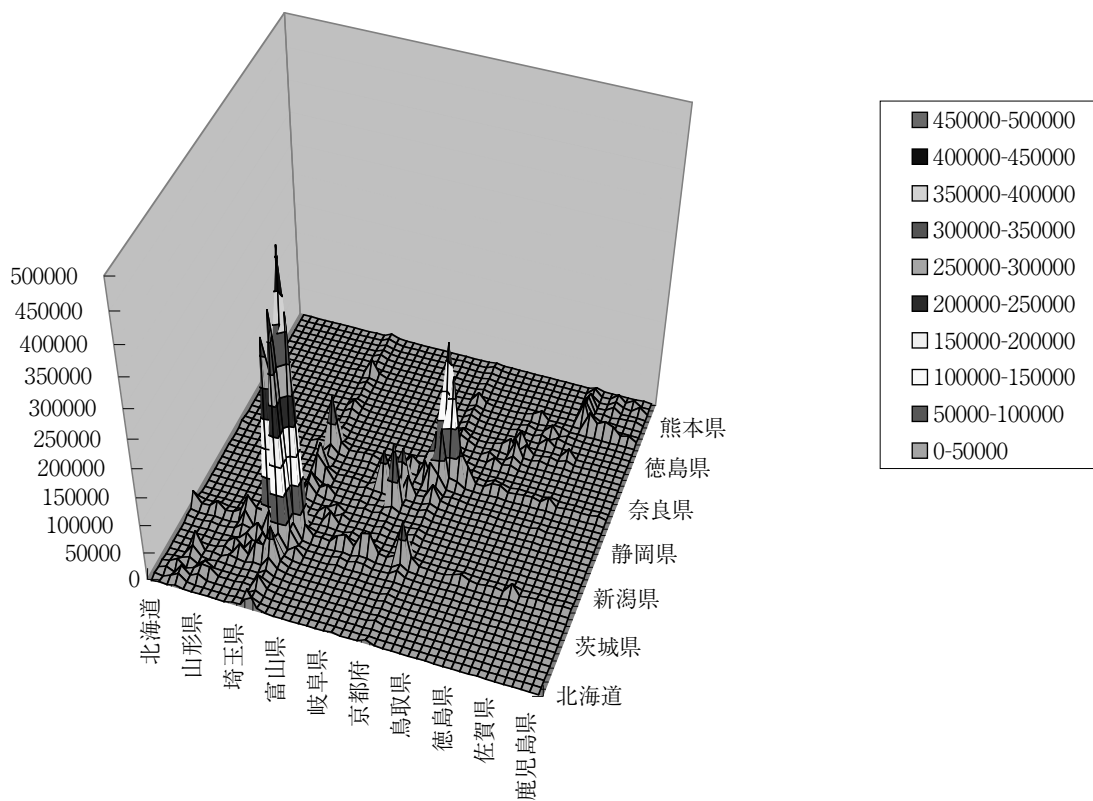


図2 通話回数の対角成分をのぞくデータのグラフ

コピーしていくことによって、近似することができる。この結果は、ほぼ指数関数的に減衰していくので、エントロピー・モデルの式とも類似性がある。

3. 遠隔作用の近似

ここでいう遠隔作用とは、たとえば東京と大阪のように、遠く離れた都市間の関係を説明するのに有効と考えられる。従来のグラビティ・モデルやエントロピー・モデルのような距離による減衰式では、距離に依存しにくい成分は十分説明できない。このような成分を説明するために、ここでは遠隔作用として別に扱うことにする。なお遠隔作用については、反復関数系の繰り返し回数を多くしても意味がないので、この場合には、繰り返しを浅くして、必要最低限にとどめることにする。

以上の3つの成分をそれぞれ生成したあと、合成することにより、最終的なODデータを近似するモデルを作成する。以下、仮想的なODデータを、上の方法で生成してみよう。

まず、初期データとなる山を表すデータであるが、ここではピラミッドの形をした関数を利用することにする。これを「ピラミッド関数」と呼ぶことにしよう。図3は、標準的なピラミッド関数を示したものである。このピラミッド関数は、値がゼロより大きい部分は、以下の式で表される。

$$z = 1 - (|x| + |y|) \quad \dots(4)$$

上の式を一般化して、実際に使うピラミッド関数を求めると、以下のような式が得られる。

$$P_{hg}(x, y) = \max(h - g(|x| + |y|), 0) \quad \dots(5)$$

ここで、hは、ピラミッドの高さを表し、gは、ピラミッドの稜線の傾きを表している。なお、gは関

数ではなくパラメータである点に注意する。

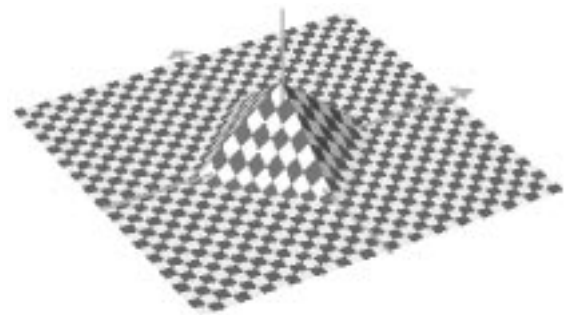


図3 ピラミッド関数の概形

III. 反復関数系の構成

実際にデータを生成するにあたって、次に必要となるのは各成分ごとに具体的な反復関数系の式を構成することである。

1. 対角成分の生成

ここでは純粋な意味での対角成分ではなく、行列の対角線近くにある成分という意味である。以下のような2個の反復関数系を使ってデータを生成することにする。

$$w_1 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0.3 & 0 \\ 0 & 0.3 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad \dots(6)$$

$$w_2 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0.7 & 0 \\ 0 & 0.7 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} 15 \\ 15 \end{pmatrix} \quad \dots(7)$$

このようにいちいちアフィン変換の式を書くのは煩雑になるので、今後は表1のようなパラメータの表の形で表すことにする。

表1 対角成分を生成する反復関係系のパラメータ

	a	e	f
w ₁	0.3	0	0
w ₂	0.7	15	15

アフィン変換の対象とする領域は、50×50の正方形領域とする。これは都道府県の個数(47)に近くて、きりのいい数字ということで選んだものである。実際の計算では、整数値の点(格

子上の点) でしか値を持たないので、領域を狭くすると計算が粗くなりすぎるし、逆に領域を広くすると複雑になるので、適当なところで50にしたという経緯もある。

2つの写像を図に表したのが図4である。この図で、外側の大きな正方形が対象とする領域であり、内部にある w_1, w_2 と書かれた2つの正方形が、写像により縮小された領域を表している。30%, 70%と書かれているのは、縮小率をわかりやすく示したものである。

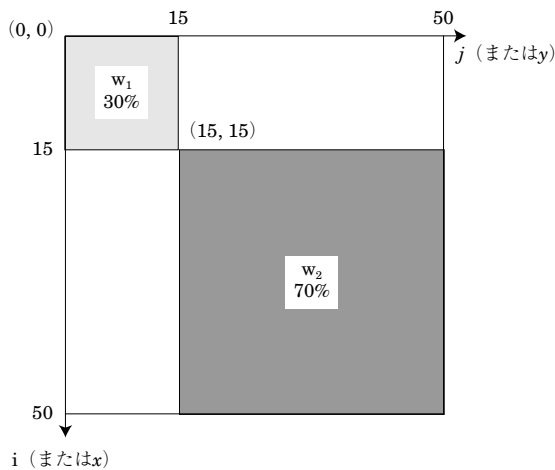


図4 対角要素を生成する反復関数系

2. 近接成分の生成

近接効果の成分の生成には、中心部の山を縮小コピーしていく写像と、周辺に拡散していく写像の、2種類の写像を考える。この2種類の写像の相対的な重みの違いにより、結果が異なってくるので、ここでは以下の(A), (B)という2つのタイプの反復関数系を試してみた。

(A) 周辺部の成分が中心部よりも相対的に大きい場合

表2、図5で表される反復関数系がこの例である。

(B) 周辺部の成分が中心部よりも相対的に小さい場合

表2 対角成分を生成する反復関数系のパラメータ (タイプA)

	a	e	f
w_1	0.8	5	5
w_2	0.5	0	12.5
w_3	0.5	12.5	0
w_4	0.5	12.5	25
w_5	0.5	25	12.5

(注意) e, fの値は、実際の計算では四捨五入して整数にしている。

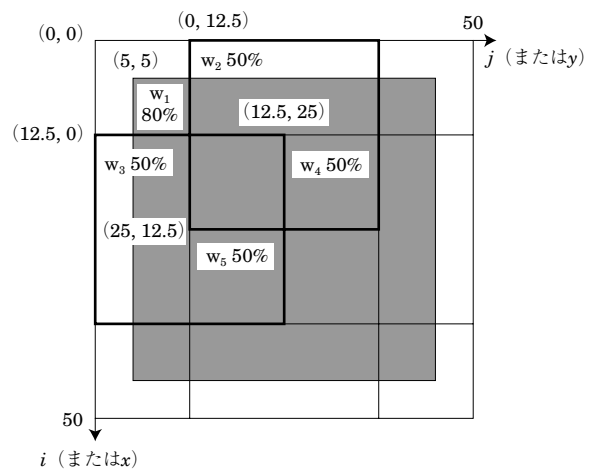


図5 近接効果の成分を生成する反復関数系 (タイプA)

表3、図6で表される反復関数系がこの例である。

3. 遠隔成分の生成

遠隔効果に関わる成分の生成には、表4および図7のような反復関数系を用いる。ここで、 w_2 と w_3 が遠隔作用に対応する写像であるが、現実には遠隔作用が発生するためには、対角成分もそれなりに大きく、全体として大きな影響力を持っているので、その対角成分にあたるものとして w_1 という写像を用いた。

表3 対角成分を生成する反復関数系のパラメータ (タイプB)

	a	e	f
w_1	0.8	5	5
w_2	0.3	0	17.5
w_3	0.3	17.5	0
w_4	0.3	17.5	35
w_5	0.3	35	17.5

(注意) e, f の値は、実際の計算では四捨五入して整数にしている。

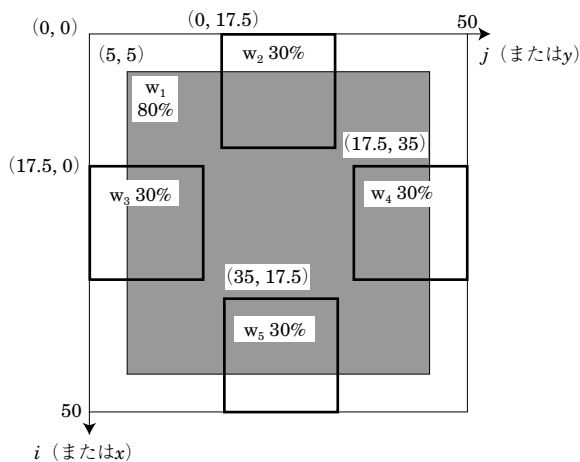


図6 近接効果の成分を生成する反復関数系 (タイプB)

IV. 仮想的データの生成結果

以上で一通り計算のための準備が整ったので、実際に仮想的なデータを生成させてみよう。まず成分ごとに生成させて、個別に考察した上で、最後にすべての成分を合成させるという手順をとる。

1. 対角成分の生成結果

図4の反復関数系を1回適用したのが図8で、5回繰り返し適用したのが図9である。この結果をみると、対角線上に自己相似的な山を作ることができていることがわかる。

表4 遠隔成分を生成する反復関数系のパラメータ

	a	e	f
w_1	0.5	0	0
w_2	0.2	7.5	40
w_3	0.2	40	7.5

(注意) e, f の値は、実際の計算では四捨五入して整数にしている。

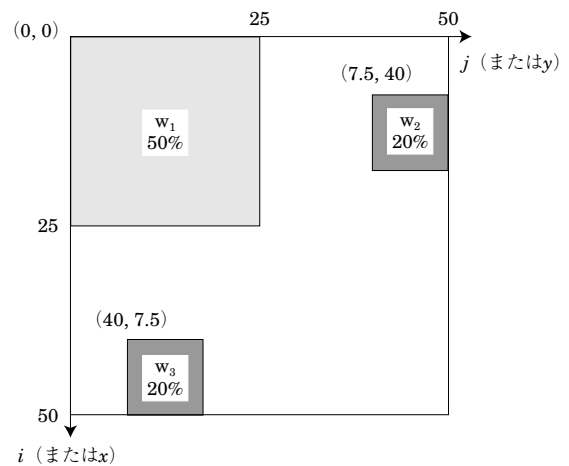


図7 遠隔効果の成分を生成する反復関数系

2. 近接効果の生成結果

近接効果については、以下の(A),(B)の2通りの場合についてデータを生成した。

(A) 周辺部の成分が中心部よりも相対的に大きい場合

図10が1回目の結果、図11, 図12は5回目の繰り返し後の結果である。なお図12は、真上から見た等高線図の形で表されている。この場合には、植物の葉のように複雑に拡散していく様子が見られる。また、繰り返しを続けていくと、中心部が相対的に陥没してくるという現象が発生している。中心部にデータを配分しているにもかかわらず、このような現象が起こることは興味深い。周辺部の力が

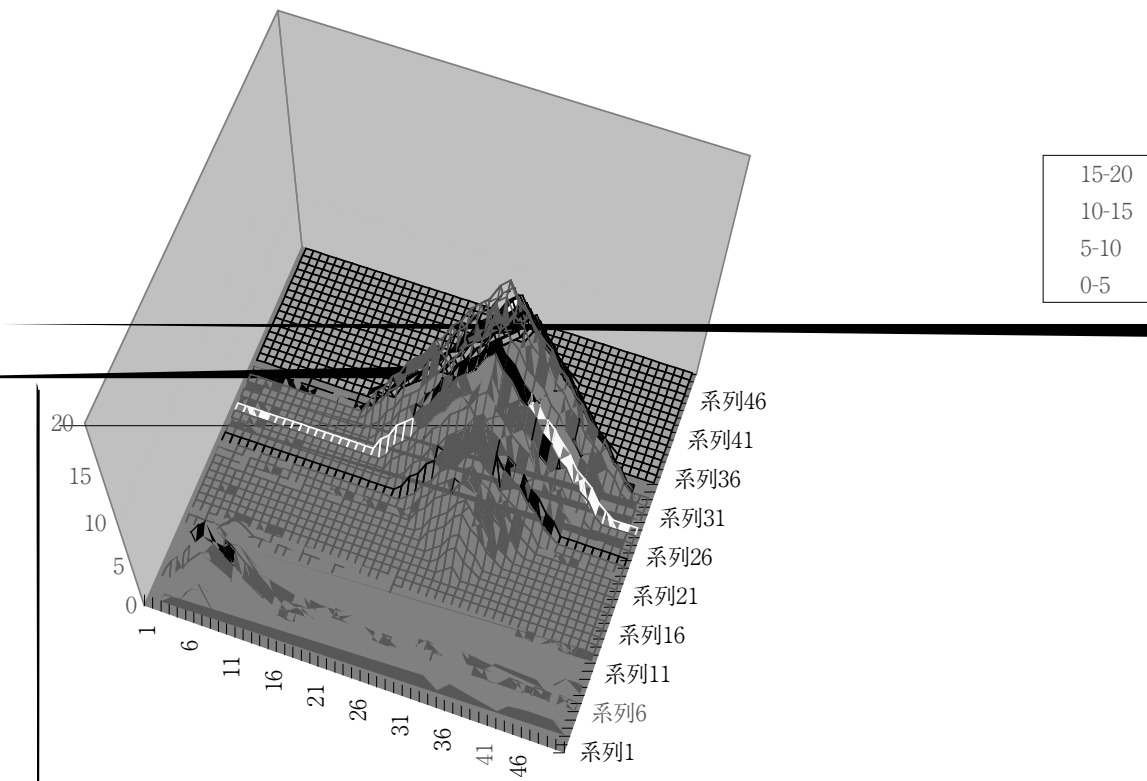


図8 対角成分の生成 (step=1)

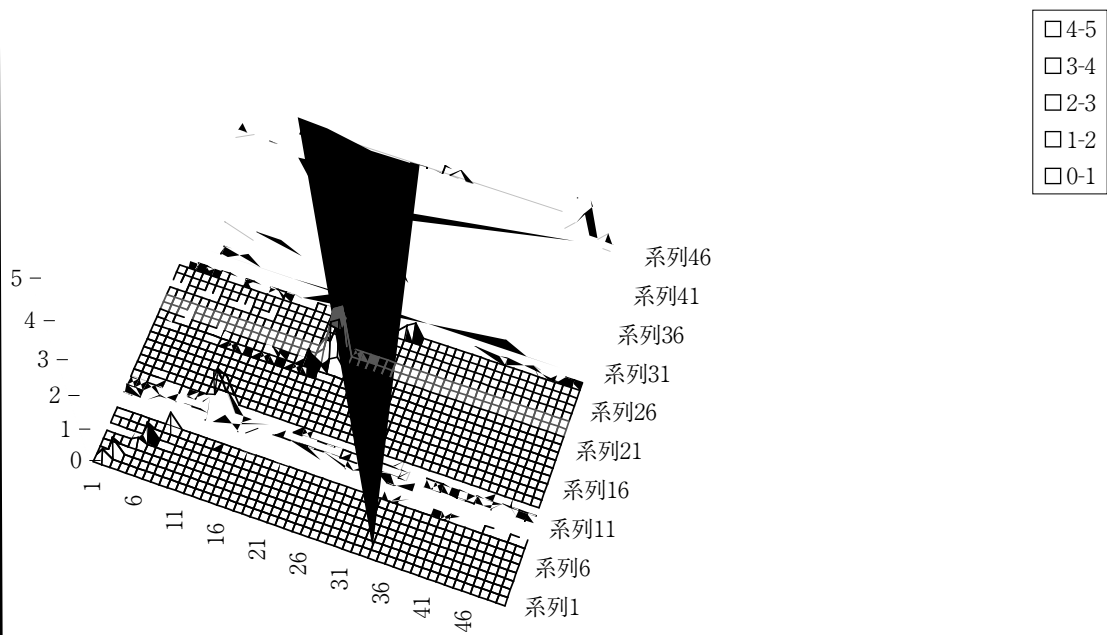


図9 対角成分の生成 (step=5)

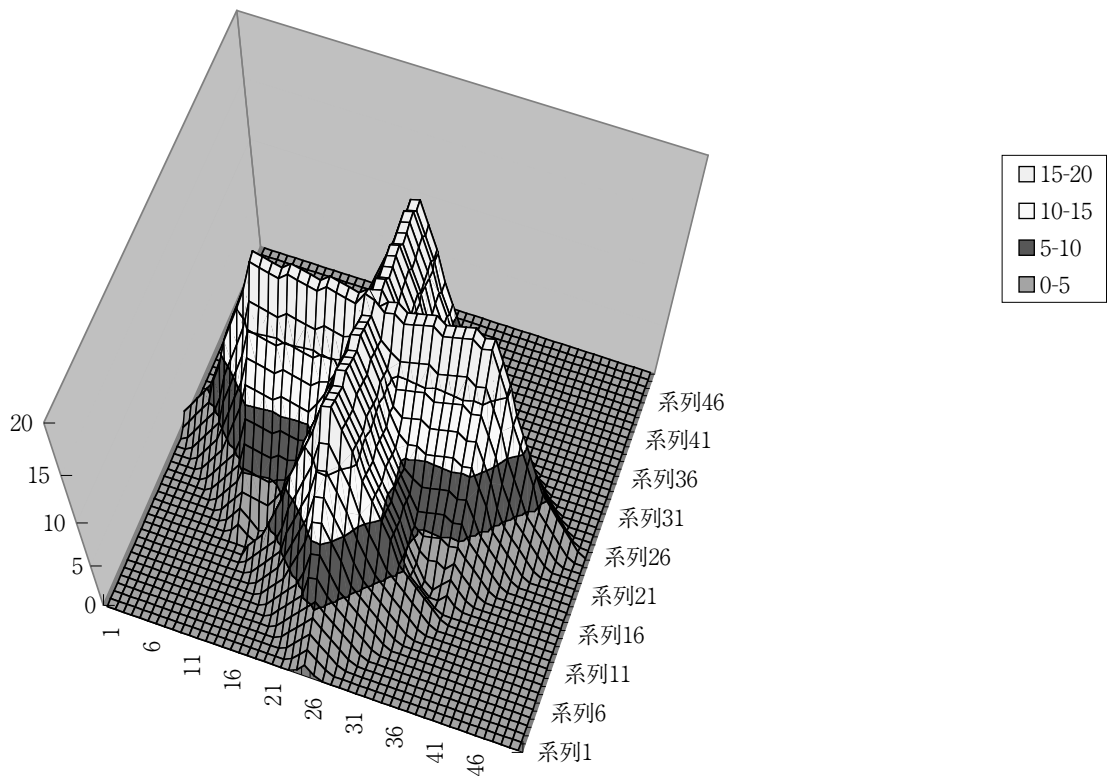


図10 近接効果(A)の生成データ (step=1)

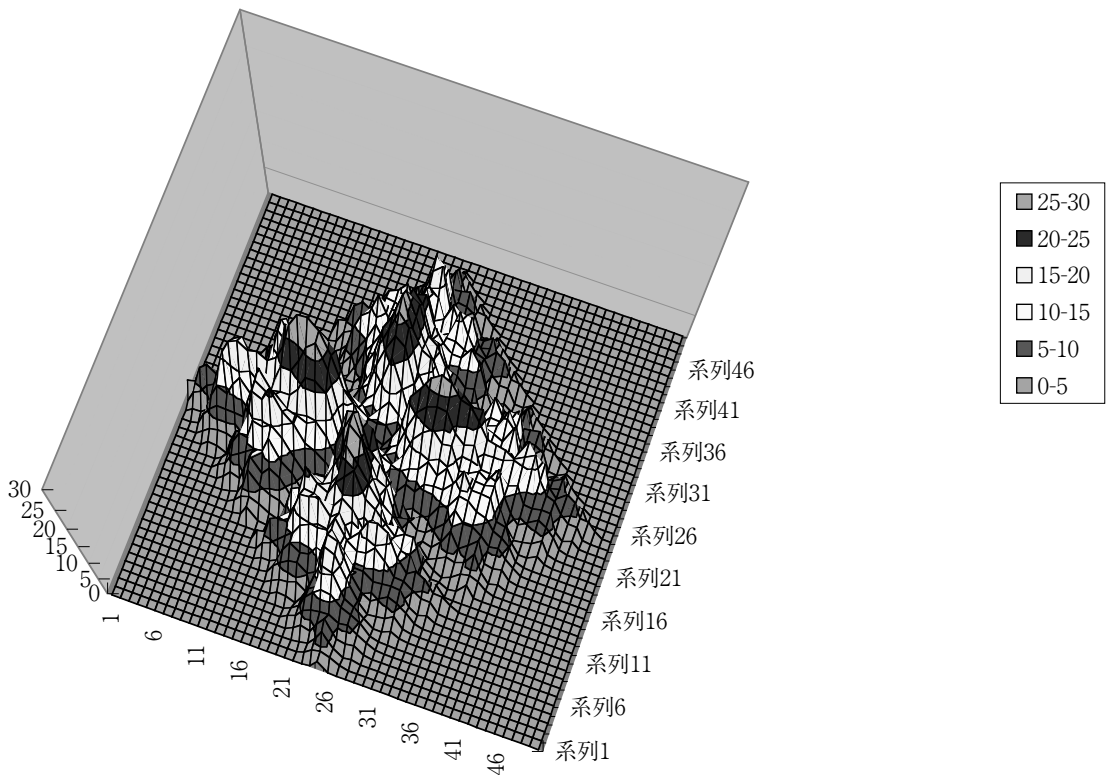


図11 近接効果(A)の生成データ (step=5)

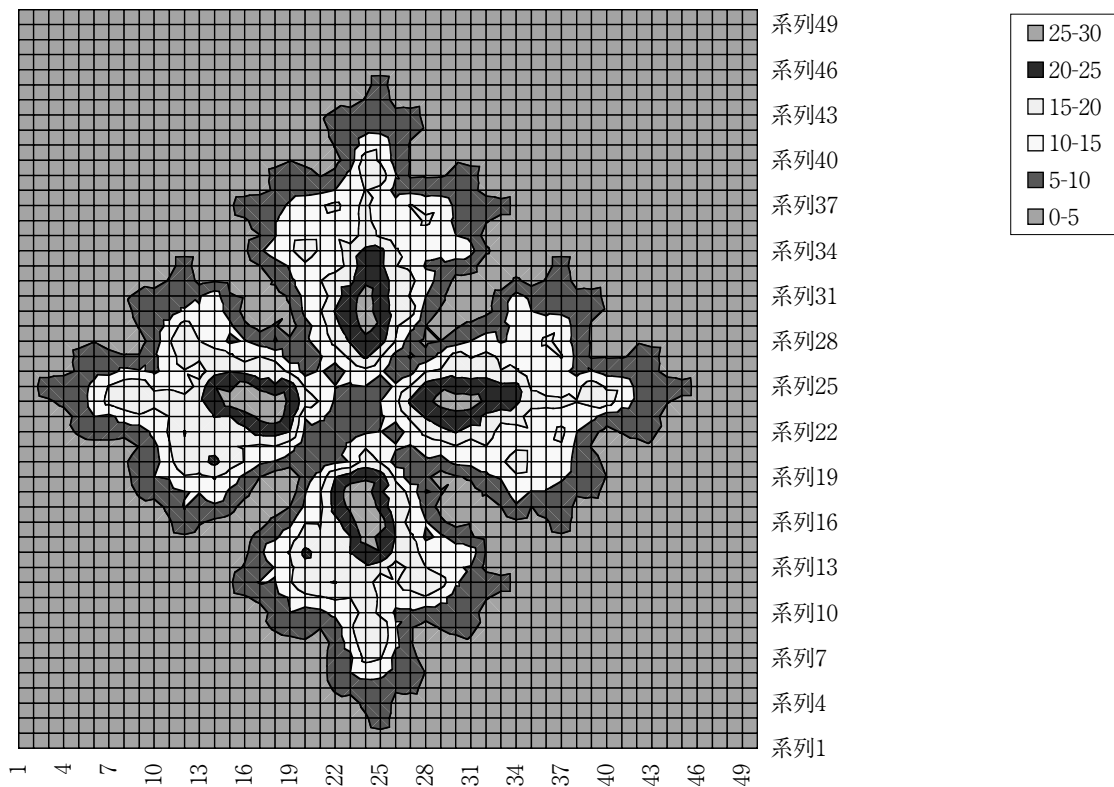


図12 近接効果(A)の生成データの等高線図 (step=5)

相対的に強いために、中心部が空洞化しているものと考えられる。

(B) 周辺部の成分が中心部よりも相対的に小さい場合

図13～図16は、近接効果の成分を、図6の反復関数系を使って生成したものである。このうち図13は、1回目の繰り返しの結果、図14は、5回目の繰り返しの結果である。図15は、図14を等高線図の形に表したもので、図16は、図15の中心付近の断面図である。

図のように、簡単なデータから複雑な構造を生成できるという点は、注目に値する。また中心部の山が残っているという点が、タイプAの場合と異なっている。このように反復関数系の少しの違いが、繰り返しの積み重ねによって、結果として大きな差を生み出すことにも注意すべきである。

通常はこちらのBタイプのほうが、中心部のおちこみがないという点で、Aタイプよりも現

実のデータに近いものと思われる。なお、Aタイプ、Bタイプとも、結果は微妙に対称性が崩れているが、これは反復関数系のパラメータの小数点以下を四捨五入したことによるもので、本質的な問題ではない。

3. 遠隔効果の生成結果

図17～図19は、遠隔効果の反復関数系(図7)を使って生成した結果である。図17, 18, 19は、それぞれ1回目、2回目、3回目の繰り返し後のデータである。ここでは5回目まで計算すると、山の幅が狭くなりすぎて、メッシュの中に埋もれてしまう。データとしてすべてゼロになってしまっは意味がなくなるので、3回までにしてある。

遠隔効果については、結果はそれほど複雑ではないが、東京-大阪間の実際の通話データと比較しても、それらしい形が得られているとみていいだろう。

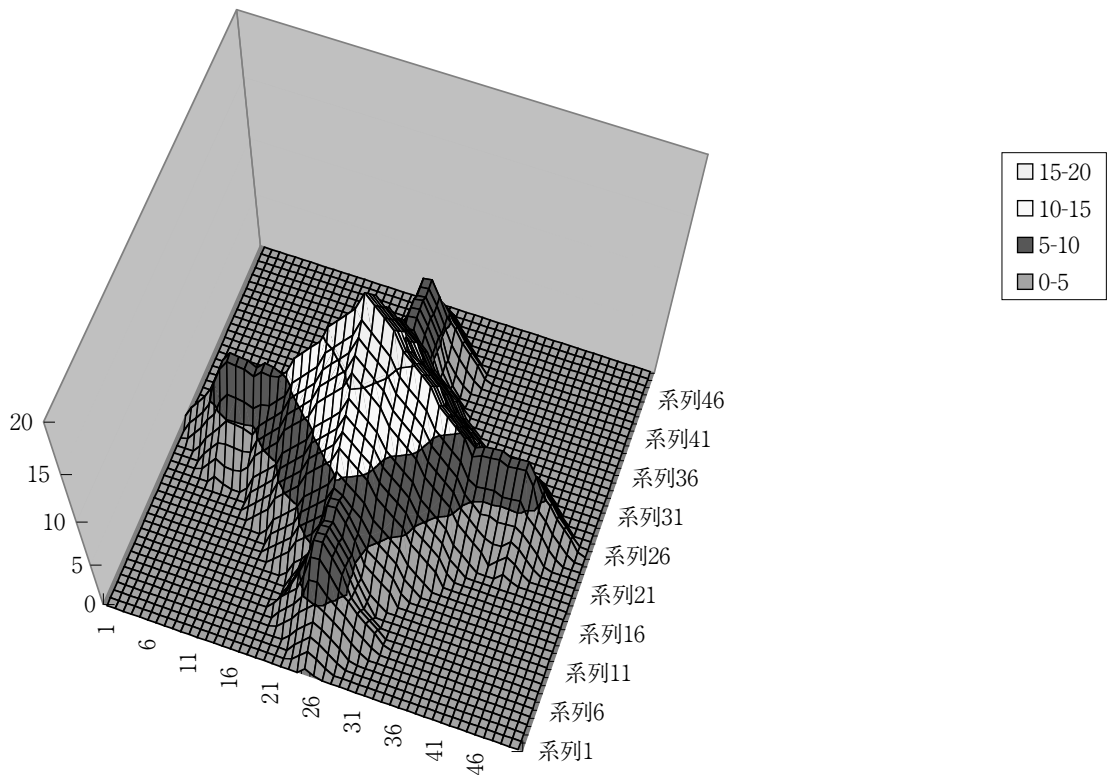


図 13 近接効果(B)の生成データ (step=1)

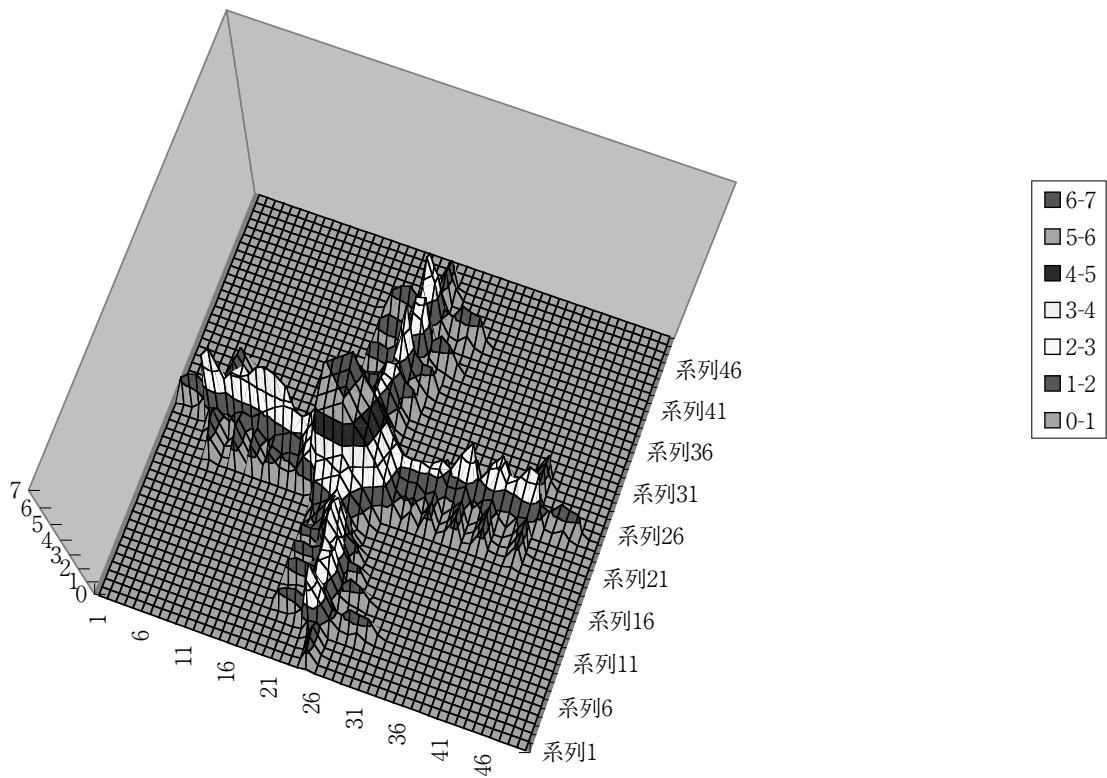


図 14 近接効果(B)の生成データ (step=5)

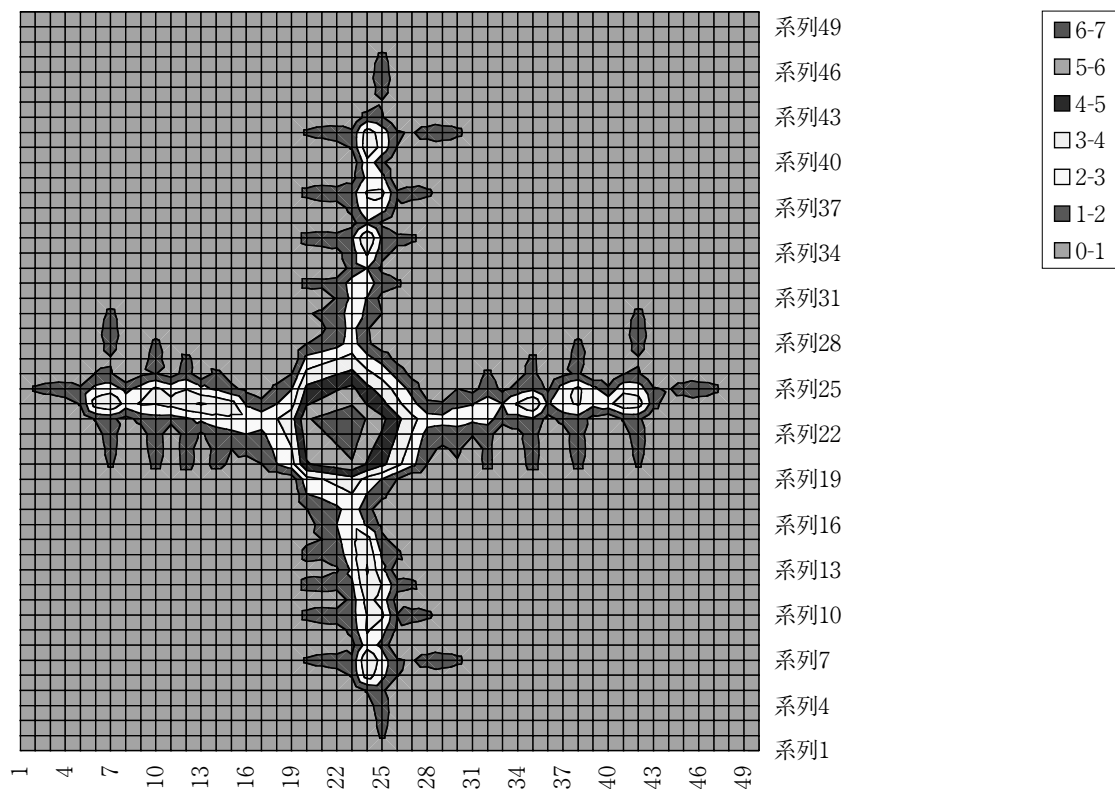


図 15 近接効果(B)の生成データの等高線図 (step=5)

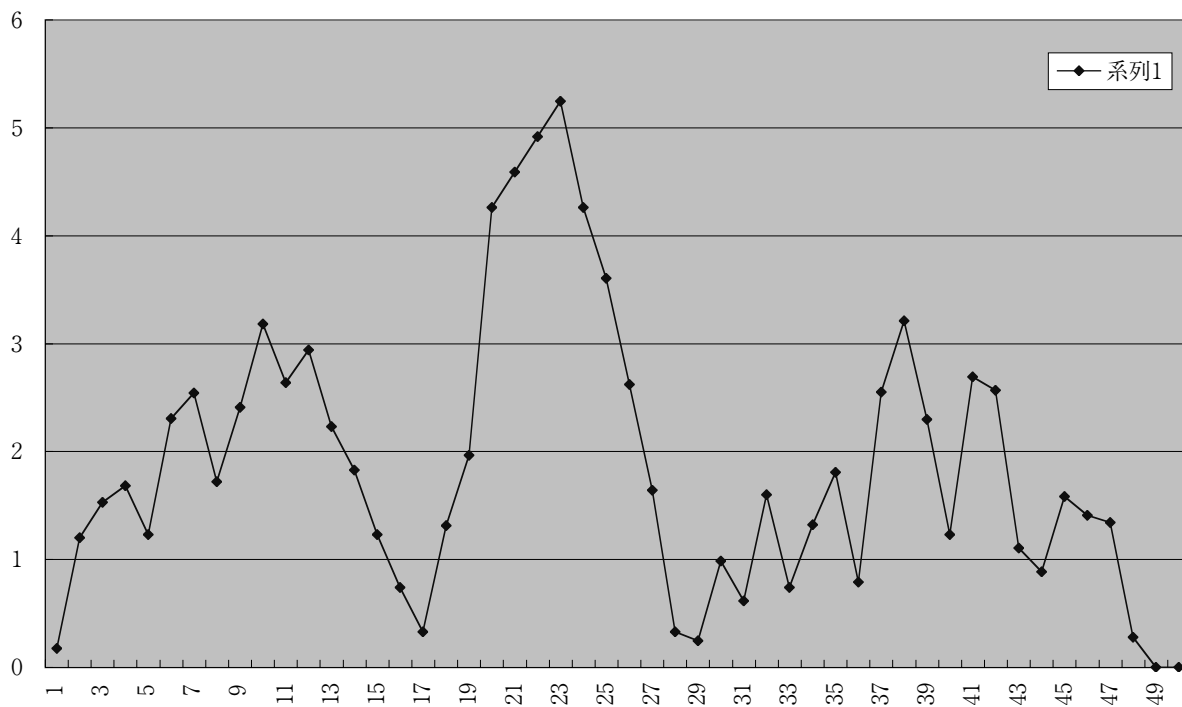


図 16 近接効果(B)の生成データの断面図 (step=5, line=25)

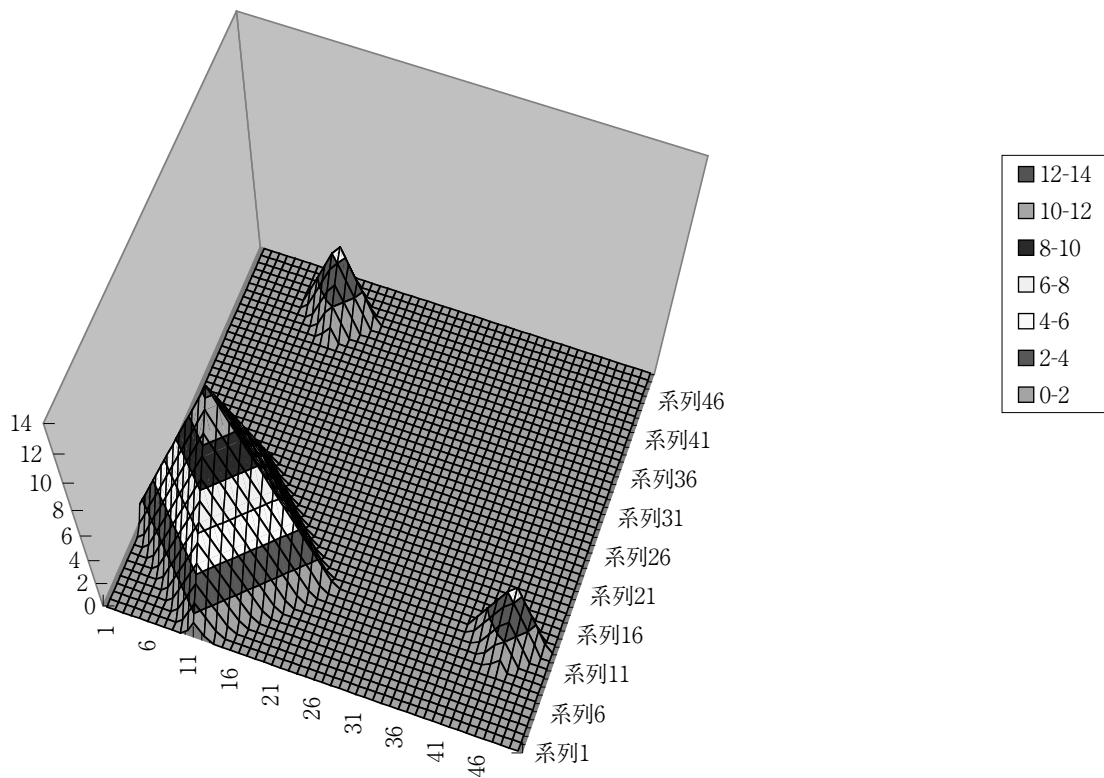


図 17 遠隔効果の生成データ (step=1)

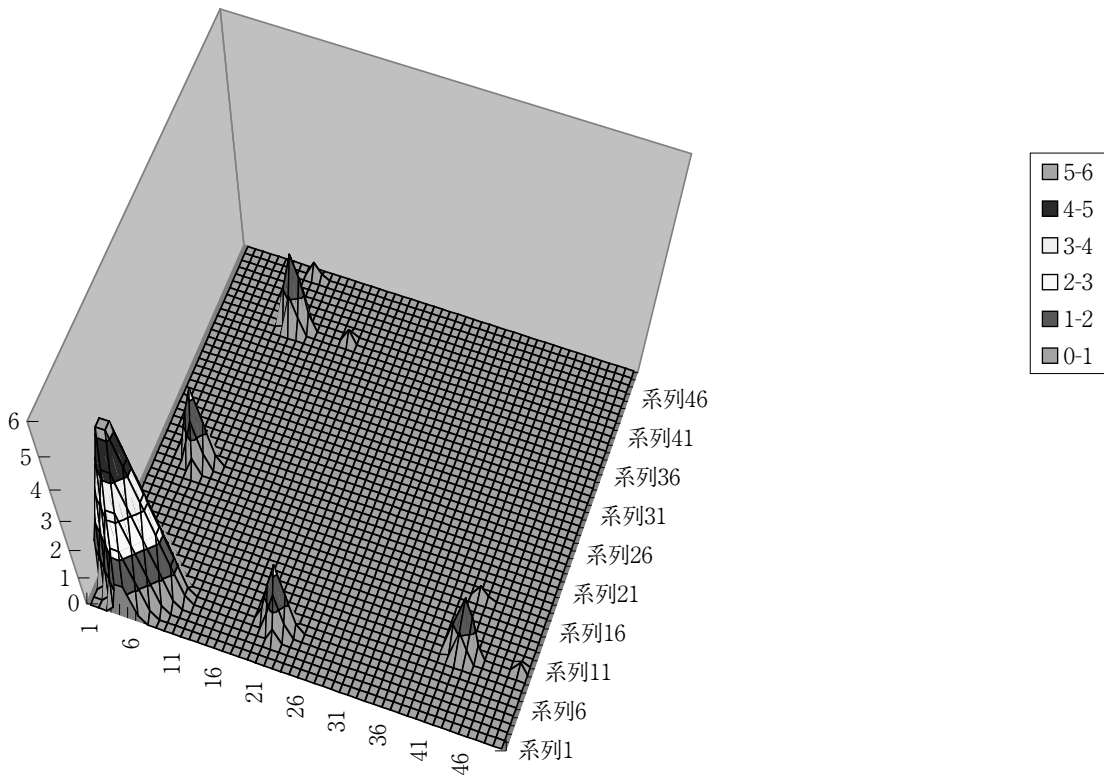


図 18 遠隔効果の生成データ (step=2)

)

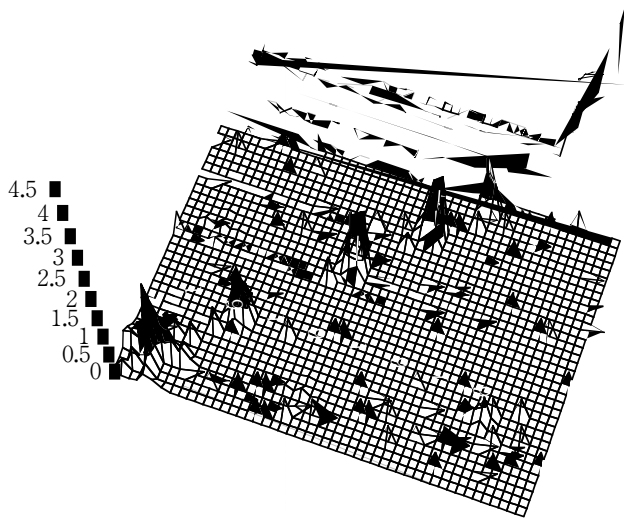
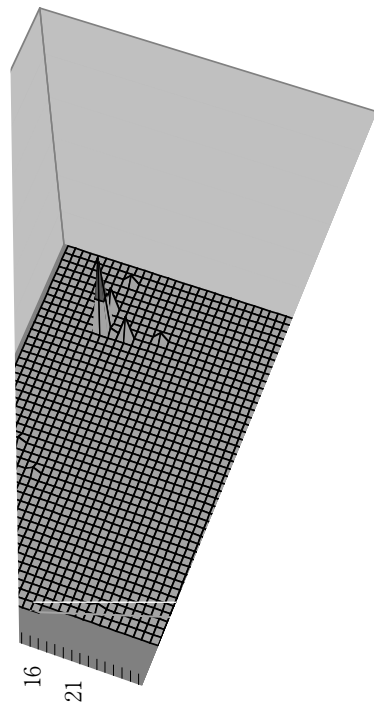


図 20 3つの効果の合成結果

4. 3つの成分の合成結果

以上の計算によって得られた3つの成分は、単純に足し合わせることによっても合成することは可能であるが、ここではもう少し工夫して、以下のような手順で合成を行う。

まず、近接効果(B)の5回目のデータ(図15)を初期値とし、遠隔効果の反復関数系(表4、図7)を2回繰り返し適用する。最後に、対角成分の5回目の繰り返し結果(図9)を加える。以上の操作で作られたデータが、図20である。

この合成結果を見ると、対角成分の値が相対的に大きいので細かな構造は分かりにくくなっているが、いくつかのフラクタル的なデータの

特徴は再現されていることがわかる。まだ細かい点では改良の余地があると思われるが、フラクタル的なデータを生成するという目的は達成されたものと考えられる。

V. 反復関数系の計算プログラム

反復関数系の計算で使用したプログラムは、下のようにC言語で記述したものである。特別な機能は使っていないので、さまざまなOS環境で汎用的に使うことが可能である。ただし、このプログラムは一例を示したもので、実際にはパラメータ設定などの細かい部分は、必要に応じて変更して使っている。

```

/*
    odifs.c
    フラクタル構造を持つODデータの生成
*/
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

/* 定数の定義 */
#define N 51
#define F 11

/* 関数宣言 */
double pyramid(double, double, int, int);

/* 大域変数宣言 */
double od[N][N], odnext[N][N], odm[N][N];

main()
{
    int step, i, j, k;
    int nstep, n, m, nf;

    /* 初期データのパラメータ */

```

```
double height, grad;
int xmove, ymove;

/* 縮小写像のパラメータ */
double p[F];
int imove[F], jmove[F];
/* ODデータの大きさ */
n=50;

/* 反復計算回数 */
nstep=1;

/* 縮小写像のパラメータの設定値 */
nf=3;
p[1]=0.5; imove[1]=0; jmove[1]=0;
p[2]=0.2; imove[2]=8; jmove[2]=40;
p[3]=0.2; imove[3]=40; jmove[3]=8;

/* 初期データのパラメータ値 */
height=25.0; grad=1.0;
xmove=25; ymove=25;

/* 初期値の設定 */
for(i=1; i<=n; i++)
for(j=1; j<=n; j++)
od[i][j]=pyramid(height, grad, i-xmove, j-ymove);

/* 反復関数系の計算 */
for(step=1; step<=nstep; step++)
{
/* 作業領域の初期化 */
for(i=1; i<=n; i++)
for(j=1; j<=n; j++)
{
odnext[i][j]=0.0;
odm[i][j]=0.0;
}
for(k=1; k<=nf; k++)
```

```

{
    /* 縮小データの計算 */
    m=(int)floor(p[k]*n);
    for(i=1; i<=m; i++)
        for(j=1; j<=m; j++)
            odm[i][j]=p[k]*od[(int)ceil(i/p[k]+0.5)]
                               [(int)ceil(j/p[k]+0.5)];
    /* 平行移動と埋め込み */
    for(i=1; i<=m; i++)
        for(j=1; j<=m; j++)
            if((i+imove[k])<=n && (j+jmove[k])<=n)
                odnext[i+imove[k]][j+jmove[k]]
                    =odnext[i+imove[k]][j+jmove[k]]+odm[i][j];
    }
    /* データをコピーして次の回へ */
    for(i=1; i<=n; i++)
        for(j=1; j<=n; j++)
            od[i][j]=odnext[i][j];
}
/* データ出力 */
for(i=1; i<=n; i++)
{
    for(j=1; j<=n; j++)
        printf("%f ", od[i][j]);
    printf("_n");
}
return 0;
}

/* ピラミッド関数 */
double pyramid(double height, double grad, int x, int y)
{
    double z;

    z=height-grad*(abs(x)+abs(y));
    if(z<0) z=0.0;
    return z;
}

```


VI. おわりに

地域間の通話データを出発点として、そのフラクタル的な構造に着目し、モデル化することを試みてきたが、ある程度現実のデータと似通ったデータを生成させることが可能となった。このモデル化の中心となる考え方は、対角成分、近接効果成分、遠隔効果成分という3つの成分に分解し、それぞれを質的に違うものとして扱った点である。このように分解することにより、現実的なモデル化が容易になった。

今後の課題として考えられることは、まず現実のデータから反復関数系のパラメータを計算することにより、現実に近いモデルを作成することである。この計算には、フラクタル画像圧縮において使われている「コラージュ定理」を利用すれば可能であることが知られている。^{(1),(2)} コラージュ定理とは、簡単にいえば、「自分自身の図形をみずからの縮小コピー（これはIFSで表される）で覆った場合（これを自己被覆という）、これが元の図形と近ければ、そのIFSのアトラクタで元の図形を近似できる」というものである。ここでアトラクタとは、IFSを無限に繰り返したときの極限の図形のことである。元の図形とその自己被覆との間の近さを測るには、ハウスドルフ距離と呼ばれる特殊な距離が使われる。これは、2つの画像の食い違いを測る尺度であり、二つの画像をA, Bとすると、ハウスドルフ距離は $h(A, B)$ と表される。Tを元の画像とし、 $W(T)$ をIFSを適用した自己被覆（コラージュ）とすると、問題は、 $h(T, W(T))$ を最小化するIFSのパラメータを求める問題に帰着される。この問題は多変数の最適化問題であり、局所解に収束する可能性が高いので注意する必要がある。

以上の説明は、一般の画像データにコラージュ定理を適用する場合であるが、ODデータに

適用する場合には、ODデータに特有な性質（たとえば回転の成分がないとか）が利用できるの
で、さらに簡単にできる可能性がある。

次の課題としては、フラクタル的な性質を利用したクラスター分析の手法を開発して、今まで分析の困難であったデータについても自然なクラスタリングができるようにすることが考えられる。このような手法を通して、地域データのもつ特徴をより鮮明に描き出すことが、本研究の最終的な目標である。

参考文献

- (1) M.F. バーンズレイ・L.P. ハード（蔡東生・江守正多 訳）「マルチメディア フラクタル画像圧縮」トッパン（1995）
- (2) マリオ・ベルギア（柳原二郎・川瀬眞・陳啓浩 訳）「離散的反復関数系の理論」トッパン（1995）
- (3) 中村有一、長谷川文雄、出石宏彦「通話からみた地域間の結合に関する研究」、第18回日本都市計画学会学術研究発表会研究論文集、pp. 73-78（1983）
- (4) 中村有一、南部世紀夫、長谷川文雄「全国通話トラヒックの事務用住宅用比較研究」、1991年度日本都市計画学会論文集、pp. 877-882（1991）
- (5) 中村有一、長谷川文雄「通話等情報トラヒックの地域構造に関する一考察」、平成2年度情報通信学会年報、pp. 14-29（1991）
- (6) 中村有一、南部世紀夫、長谷川文雄「関東地方における通話の都市間交流に関する研究」、1992年度日本都市計画学会論文集、pp. 325-330（1992）
- (7) 中村有一、長谷川文雄「単位料金区域からみた通話トラヒックの基本特性に関する研究」、平成3年度情報通信学会年報、pp. 106-119（1992）
- (8) 中村有一、高嶋裕一「通話トラヒックデータに基づく単位料金区域の統合化手法に関する考察」、TIMIS JOURNAL No. 32（1993）

著者プロフィール

中村有一（なかむら ゆういち）
多摩大学経営情報学部助教授

