

# 社会システム分析のための統合化プログラム 16

## － 3Dビューアとその応用 －

福井正康，尾崎誠，石丸敬二\*

福山平成大学経営学部経営学科

\*福山大学経済学部経済学科

### 概要

我々は教育分野での利用を目的に社会システム分析に用いられる様々な手法を統合化したプログラム College Analysis を作成してきた。これまでは統計解析や経営科学に関するプログラムが中心であったが、今後は少し方向性を広げて、大学教養課程の数理教育でも利用できる楽しいプログラムも加えて行きたいと思う。この論文では、グラフィックを3D表示するために作成した3Dビューアの詳細とそれを利用するプログラムの概要について説明する。

### キーワード

College Analysis, 社会システム分析, 統計, OR, 意思決定, 3D, グラフィックス

URL: <http://www.heisei-u.ac.jp/ba/fukui/>

## 1. はじめに

我々はこれまで College Analysis において、統計解析、経営科学、意思決定手法などを中心にプログラム化を進めて来たが、プログラムの一応の完成を機に、方向性を広げて、大学教養課程における数理系授業に役立つ楽しいプログラムも加えて行きたいと思うようになった。その最初の題材に選んだものが「3D表現」である。

最近の3D映画の大ヒットで、我々は3Dという言葉に慣れてきたが、これは平面に投影された立体図形に奥行きを感じさせるしきみを表す言葉である。3Dを実現する方法は数多くあり、映画の中でも、1915年以降、アナグリフ（赤青メガネ方式）、偏光メガネ、アクティブシャッター方式など（これらの中にもまだ様々な方法がある）多くの方式が採用されている。さらにパソコンの世界でも、最近の3Dテレビの発達と相まって、3Dパソコンが販売されるようになり、3Dへの道が開かれ始めている。ただ現実には3Dパソコンはまだ普及しておらず、我々もこれを利用していない。そこで我々は表示装置に頼らず、誰でも利用可能なアナグリフ方式で3D画像を取り扱うことにした。

我々は College Analysis で1変量関数のグラフを描くプログラムを作成していたが、これを2変量に拡張して、立体的なグラフ描画を考えた。その際、マウスで図形を動かしてリアルタイムにどこからでも図形を眺めることができることを目指した。これが可能になると、グラフの各点を右目用と左目用に動かしてアナグリフにすることはさほど難しいことではない。但し、この合成には様々な方法があり、リアルタイムで動かすには多少の工夫が必要である。この論文ではこの方法についても言及したい。

立体的なグラフ描画が可能になるとプログラムの動作スピードなどの確認ができるので、今後の利用への可能性が見えてくる。まずグラフィック表示部分を独立させて3Dビューアと名付け、様々なプログラムから呼び出して使うようにする。これによって画像の取り扱いが単一になり、デバッグなども楽になる。

データ形式を整理するために取り組んだ応用プログラムは、3Dモデルを表示するプログラムである。これは既にデータとなっているグラフィック画像を表示させるだけのもので、簡単なものである。しかし、グラフィックデータの構造を考える上でこのプログラムは役に立った。グラフィックは、基本的に、点、線、多角形（他に円や文字列もある）で構成される。構造の詳細については後に述べる。

3Dビューアを利用するプログラムとしては、3Dフラクタルビューア、カオスビューア、くるくるエディット、面白グラフと開発を続けて行ったが、面白グラフでは3Dビューアに渡すデータ構造に問題が生じてくる。現在もまだ少し迷っている部分もあるが、これについては考察のところで述べる。これらのプログラムは今後改良されて行く可能性を持っているので、ここでは概略のみに留め、詳細はもう少し完成度が上がった段階で別に紹介することにする。一つ一つのプログラムは専用のグラフィックソフトに比べると簡単なものであるが、今後の開発者のために情報を開示しておく。

## 2. 3Dビューア

この章では3Dビューアのデータ構造と動作について、2変量関数グラフのデータを元に説明する<sup>1)</sup>。2変量関数グラフから3Dビューアを用いて  $z = \sin(x) + \cos(y)$  のグラフを描くと初期画面は図2.1のようになる。

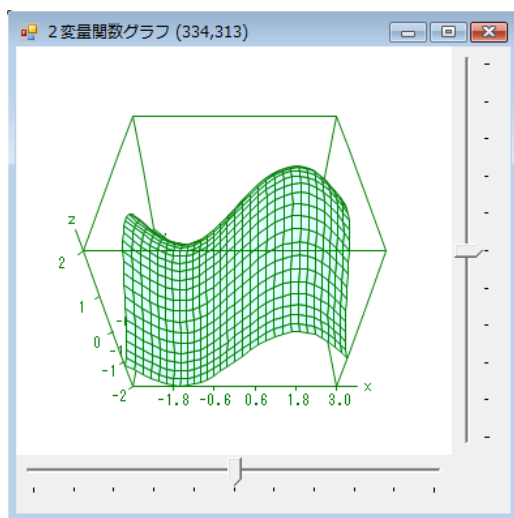


図 2.1 2変量関数グラフの出力画面

この画面上でマウスをドラッグすると、図2.2や図2.3のような別角度からの図が見られる。

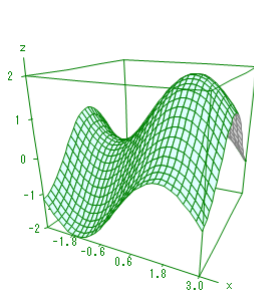


図 2.2 2変量関数グラフ別角度1

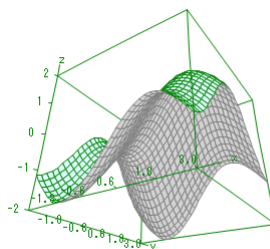


図 2.3 2変量関数グラフ別角度2

マウスの左ボタン押してを上下に動かすと、物体の中心周りに  $z$  軸を前後に傾けることができ、左右に動かすと、 $z$  方向周りの回転となる。マウスの右ボタンを押して左右に動かすと  $y$  方向周りの回転で、上下に動かすと  $x$  方向周りの回転となる。通常は左ボタンだけですべての角度から図形を眺められる。これ以外にマウスのホイールによって図までの距離を変えることができる。極端に近づけると図の中まで入り、向こう側へ通り抜けることもできる。ただ常に図の中心に近付くので、図の上と右のスライダーによって図を上下左右に移動してもよい。

図は実際の大きさに関係なく、初期画面では、常に中心を真ん中にして、画面内に適当な大きさで

表示される。それ以降の移動によっては当然画面からはみ出すこともある。図は表示される対象によって、3Dモデルのようにすべての軸が同一単位で表示されたり、3Dグラフのように軸の縮尺が違って表示されることがある。この2変量関数の場合は後者である。

次に3Dビューアのメニューについて説明する。メニューは、保存などの「ファイル」、コピーなどの「編集」の他に、特徴的なものは「表示」、「色」、「アクション」である。「表示」のメニューを図 2.4 に示す。

この中で「図形サイズ」は最初に表示される図形の大きさを示す。図形の描画3次元空間は仮想的に  $1000 \times 1000 \times 1000$  のサイズであり、その中での図形のサイズ（最大軸方向）を与える。標準は 700 である。

「マウスホイール移動」は1回の操作音で近づく図形の距離（仮想的な空間での距離は表示される数値にホイール移動量の 120 をかけたものである）を与える。

ここで少し仮想空間の取り方について説明しておく。図形は先に述べた通り  $(0, 0, -500)$  の点を中心に「図形サイズ」で指定される大き

さで表示されている。初期の視点は  $z$  軸方向 1500 であり、図形が投影される面は  $z = 500$  の位置の  $1000 \times 1000$  の大きさの平面である。ビューアに表示される画像は、視点と物体の点を結ぶ直線がこの平面と交わる点に描かれる。マウスホイールによって物体が近づく際、投影面より視点に近付いてもよい。但し、物体の描画要素の一部が視点を超えるとその描画要素は表示しないようにする。このようにすることで物体を内部から見ることもできるようになる。

次に「2D」、「3D」、「3D カラー」であるが、「2D」は通常表示である。「3D」はモノクロの3D表示である。これは2枚のビットマップ上に左目用の画像と右目用の画像をシアンと赤で描画してこれを合成する。左右の画像のずれは、視差 0 の位置から図形上の点までの距離に  $\pm 0.03$  をかけて与えている。この数値は試行錯誤で得られた見易い数値である。また、視差 0 の位置は  $z = -1000$  の平面にしている。これによって、最初に表示されたとき、図形はディスプレイから少し浮き出るように表示される。

次に左右の画像の合成である。白の背景にシアンと赤の2色であるので、左右の画像が交わる点のマスクを作り、シアン画面に背景色の白を透明にして、赤画面を合成する。その後、マスクの合成部分を黒にして背景色の白を透明にして、シアンと赤の合成画面の上に合成するとき綺麗なモノクロのアナグリフが出来上がる。実際試してみると左右の分離はかなり良い。しかし、この処理には少し計算時間がかかり、スピードの遅いパソコンでは図形の動きがなめらかにならない。そこで綺麗なア

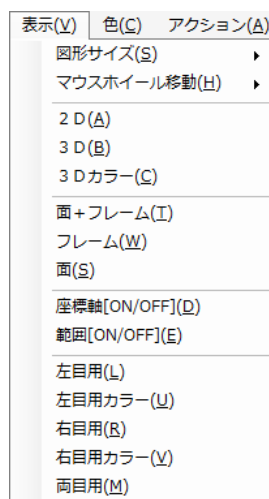


図 2.4 「表示」メニュー

ナグリフは「3D カラー」に譲って、スピードを重視した処理を行うことにした。即ち、赤の色を薄くした画面にシアンの色を半透明にして重ねるという方法である。この方法では多少残像が残るが、図形の動きは申し分なく、また図形のギザギザを中間色で補正したなめらかな描画モードでも効果を発揮する。

最後に「3D カラー」のアナグリフは図形の立体感だけでなく、色までも合成してしまう人間の脳の能力を活かした方法である。まず左目用の画面から赤成分を取り除き、青と緑の成分だけ残す。次に右目用の画像から青と緑の成分を取り除き、赤成分だけを残す。それらを点ごとに重ね合わせると、色を感じる「3D カラー」のアナグリフが出来上がる。但し、これは1点ずつ処理を行うので、画面が大きくなると画素数に応じて図形描画のスピードはかなり落ちる。これら3つの方式はパソコンの性能や用途に応じて使い分ける必要がある。3つの方式は描画の状態を保持したまま、メニューによって切り替えることができる。紙面上では3Dに見えないが、3つの方式で表した画像を図2.5a、図2.5b、図2.5cに示す。

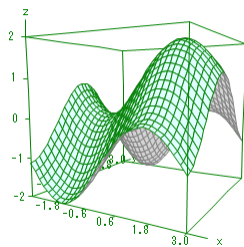


図 2.5a 2D 形式

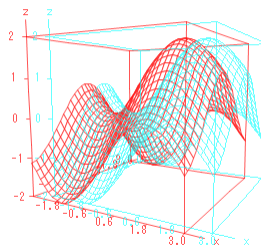


図 2.5b 3D 形式

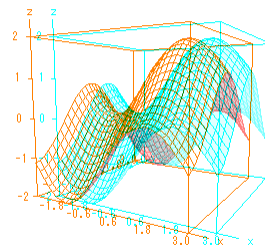


図 2.5c 3D カラー形式

ここで3Dの描画要素について説明を行う。描画はこの描画要素単位で行われる。描画要素は、点、線、多角形、円、文字列であり、データは以下のような構造体の配列で与えられる。

```
Public Structure pointset
    Dim no As Integer      ' 図形要素の構成点数
    Dim gp() As gpoint     ' 描画要素の座標配列
    Dim c As Integer       ' 図形の色値
    Dim r As Single        ' 円の描画などの半径
    Dim st As String       ' 文字列表示の場合の文字列
    Dim cfl As Color       ' ColorForeLine
    Dim cfs As Color       ' ColorForeSurface
    Dim cbl As Color       ' ColorBackLine
    Dim cbs As Color       ' ColorBackSurface
End Structure
```

ここに、gpoint は構造体で、その形式は以下の通りである。

```
Public Structure gpoint
```

```

Dim x1 As Double
Dim x2 As Double
Dim x3 As Double
Dim dist As Double
Dim dp As Double      ' 前後判定のための z/10 とした値

```

End Structure

pointset 構造体は、描画の 1 単位を表す。no は描画の構成点数を与える。例えば、点では 1、線では 2、三角形で 3 となる。例外として、円の場合は-1、文字列の場合は-99 を設定している。gpoint 構造体は、x1, x2, x3 で構成要素の 1 点の座標を与え、dist は視差 0 の位置からの z 方向の距離を表す。dp は極端に z 方向に長い棒グラフなどの z 方向の位置を与えるために用いるので、一般にはほとんど利用しない。

pointset 構造体の c は図形の色を以下のように数値化したものである。

$$c = B \text{ 値} + 256 \times G \text{ 値} + 256^2 \times R \text{ 値}$$

r は描画要素が円（3Dでは球のように見える）の場合の半径で、st は描画要素が文字列の場合の表示文字列である。描画の面には表と裏があり、cf1 は表面のフレーム、cfs は表面の面の色を表す。cb1 は裏面のフレーム、cbs は裏面の面の色を表す。描画が点と線の場合 cf1 以外は利用しない。

図 2.4 の「表示」メニューで、「面+フレーム」、「フレーム」、「面」は、上で述べた面とフレームの表示・非表示を決める。図 2.6a、図 2.6b、図 2.6c にそれぞれの形式の画面を示す。

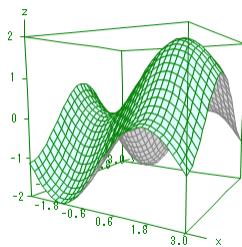


図 2.6a 面+フレーム表示

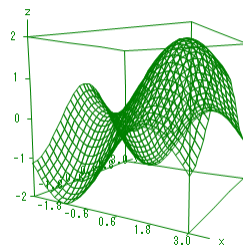


図 2.6b フレーム表示

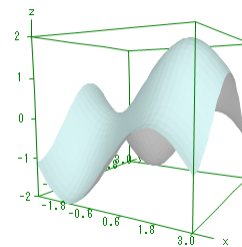


図 2.6c 面表示

2D の場合と同様に 3D カラー表示ではすべての場合があるが、3D 表示では面表示がない。

図 2.6c で見るように、平面には裏表があり、面の角度によって陰影があるが、これは描画要素の微小平面向きに法線ベクトルを考え、その向きによって表裏を決め、その z 方向との角度によって、色に濃淡を付けている。但し、法線ベクトルは、単純に元の図形上に立てるのではなく、x-y 方向を投影面に投影する処理を行った後に立てている。この処理を行う前に法線ベクトルを設定すると、法線ベクトルが z 軸に垂直に近い場合に裏表の判定で間違いが生じる。

描画要素の表示は、単純に描画要素の中心の z 成分の値でソートし、視点からの距離の遠い順に表示している。この方法だと描画要素が微小な場合には問題がないが、要素が大きくなると要素の中心

と端が離れて、陰面処理に問題が生じる。現在これについて特に問題となるのは、縦横の長さの異なる3D棒グラフの場合であるが、見る方向が通常、横か斜め上ということから、縦横のスケール変換を行って同サイズにして、処理している。また、座標軸については、経験的に誤描画が少なくなるように、軸を3分割して表示している。もう少し多く分割すると安全であるが、描画があまりきれいでなくなることもあり、この分割数を選択している。高速で優れた描画法の適用は今後の課題である。

図2.4のメニューで、「座標軸[ON/OFF]」は座標軸の表示・非表示のモード変更である。また、「範囲[ON/OFF]」は座標軸上に数値を描くかわりに、右上に座標軸の最小値と最大値を表示するモード変更である。図2.7に座標軸の非表示画面、図2.8に範囲の表示画面を示す。

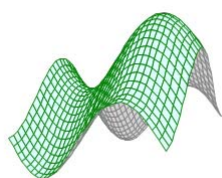


図 2.7 座標軸の非表示

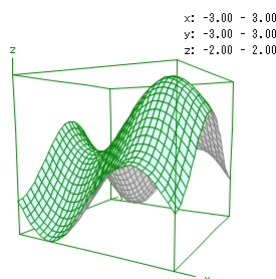


図 2.8 範囲の表示

このようなグラフでは座標軸を非表示にすることはないが、一般の3Dモデルの描画では、軸を描かないことが一般的である。

図2.4のメニューの下から5つは左右の目の画像を別々に表示するモードである。これらは2D表示では違いがない。3D表示では、左目用と右目用で図2.9a(シアン)と図2.9b(赤)のようになる。但し表示位置を明らかにするために、図形に枠を付けている。

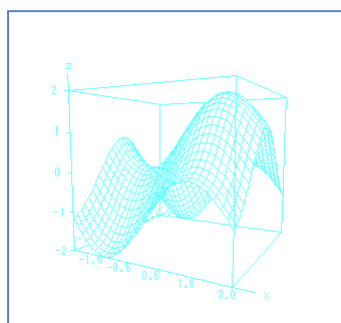


図 2.9a 3D 表示左目用 (シアン)

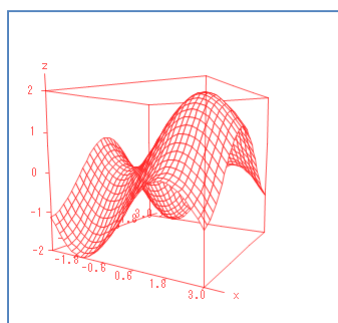


図 2.9b 3D 表示右目用 (赤)

3D カラー表示では、左目用と右目用ではそのままの色で図2.10aと図2.10bのように表示される。

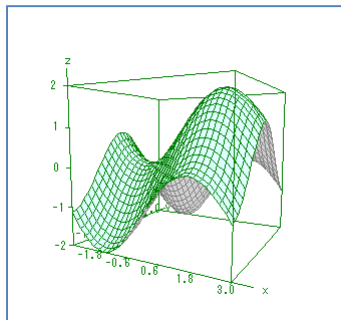


図 2.10a 3D カラー表示左目用

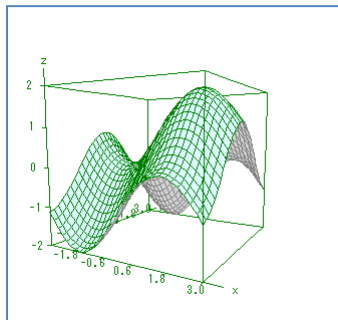


図 2.10b 3D カラー表示右目用

3D カラー表示で、左目用カラーと右目用カラーでは、色を合成する前の赤とシアン画面を（紙面では分りにくいと思われるが）表示する。

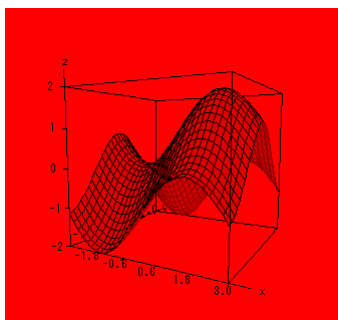


図 2.11a 3D カラー表示左目用カラー

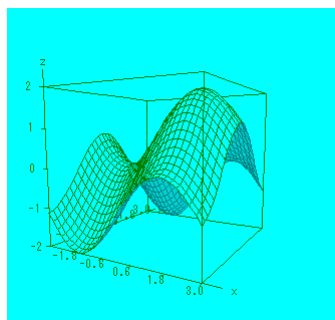


図 2.11b 3D カラー表示右目用カラー

「色」メニューについては、図 2.12 のようになる。

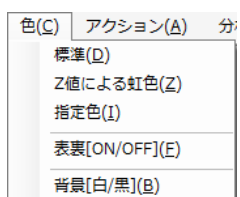


図 2.12 色メニュー

元の 2D 表示で、「標準」、「Z 値による虹色」、「指定色」メニューを選択すると、図 2.13a、図 2.13b、図 2.13c のように表示される。「指定色」は元のプログラムで指定された色で表面を色付けし、それを暗くした色で裏面を色付けしている。



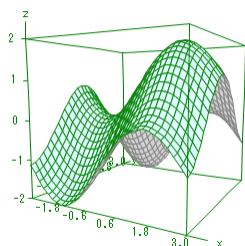


図 2.13a 標準

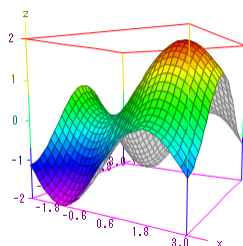


図 2.13b Z 値による虹色

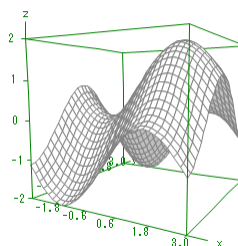


図 2.13c 指定色

もちろんこれらの色付けに対して、表示メニューで与えた設定は有効である。

図 2.12 の色メニューの「表裏[ON/OFF]」は、図形描画の際の表裏の差をなくすモードへの切り替えである。また、「背景[白/黒]」は、図形描画の背景色の白と黒の切り替えである。図 2.14 に虹色モードでの裏表なし、図 2.15 に虹色モードでの背景黒の画面を示す。

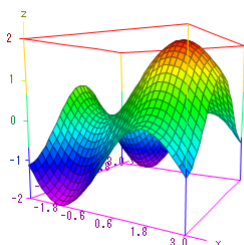


図 2.14 裏表なし

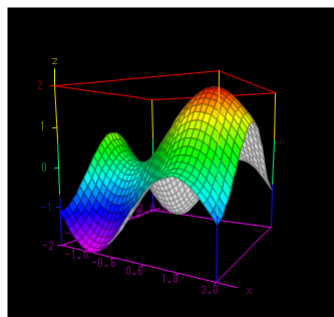


図 2.15 背景黒

最後に「アクション」メニューについて説明する。アクションメニューは、回転を除き、動きを楽しむメニューであり、何かの役に立つというものではない。人間の目は動きがあるとより立体感を感じる傾向があるらしく、「右回転」や「左回転」メニューを使って図形を連続的に回転させると、2D でも 3D でもより図形の形を認識し易くなる。「X-Wave」、「Y-Wave」、「XY-Wave」は図 2.17a に見るようにそれぞれの方向へ図形がゆれ、「Bound」は図形がゴムのよう<sup>1</sup>に z 方向に伸び縮みする。「Crash」は図 2.17b のように図形の各要素がばらばらになり、また元に戻る。「Zoom」は急激な前後への移動で、3D の効果を高める。最後に「Neon」は現在設定されている色パターンで、流れるように図形を塗り替える。

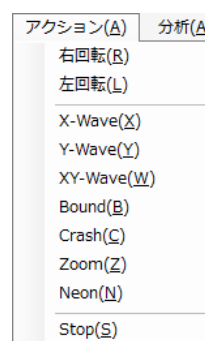


図 2.16 アクションメニュー

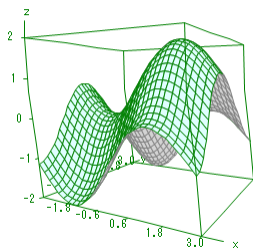


図 2.17a X-Wave

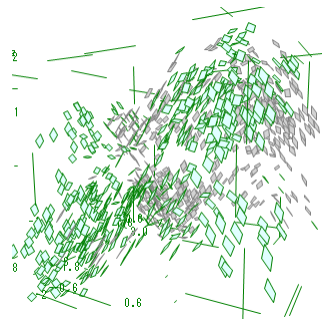


図 2.17b Crash

これらはグラフの効果としては何の意味もないが、純粋に楽しんでもらいたい。

### 3. 3Dビューアの利用

この章では3Dビューアを利用するプログラムの概略を紹介して、全体像を明らかにするが、各節の詳細はプログラムの完成度が上がった時点で別に報告する。

#### 3.1 2変量関数グラフ

3Dビューアを利用するプログラムの中で最初に取り組んだものが2変量関数グラフである。このようなグラフを表示するプログラムは多く存在するが、我々は完全に自由な方向と距離からグラフを見ることができることを目指した。図 3.1.1 にメニュー画面を示す。

図 3.1.1 2変量関数グラフメニュー

基本的な利用法は、数式のテキストボックスに例で与えられるような数式を書き込み、「グラフ描画」ボタンをクリックするだけである。このグラフの描画結果は2章の例になっている。また、このプログラムは図 3.1.2 のように、テキストボックスに書かれた複数の数式を同時にグラフ化できる。また、

グラフの停留点（極大、極小、鞍点、変曲点など）を求めることもでき、 $x$ - $y$  座標を指定して接平面の式を求めることもできる。これらの機能を利用すると図 3.1.3 のように指定された点の接平面を描くこともできる。機能の詳細については別に説明する。

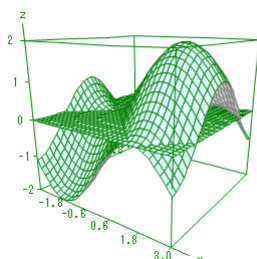


図 3.1.2 複数グラフの表示 1

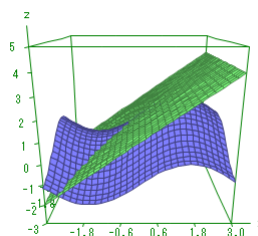


図 3.1.3 複数グラフの表示 2

### 3.2 3Dモデルビューア

これは 3D で与えられたデータから、立体図形を描画する基本的なプログラムである。しかし、現在は描画ツールがないため、有効活用はされていない。メニューも現在は図 3.2.1 に与えられる単純なもので、既存のデータを用いて図 3.2.2 や図 3.2.3 のようなグラフィックモデルを表示する。

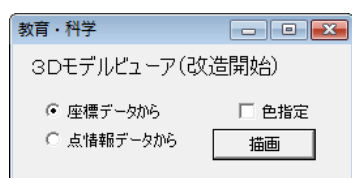


図 3.2.1 3Dモデルビューアメニュー

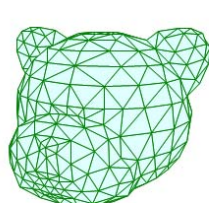


図 3.2.2 サンプル 1

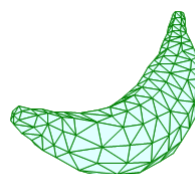


図 3.2.3 サンプル 2

このプログラムでは今後、複数のモデルを組み合わせて表示し、それらを自由に動かせるようにして行きたい。3D ビューアの作成の際には大いに利用したが、まだ始めたばかりで今後に変更に予定されているプログラムである。

### 3.3 3Dグラフ

これはよく利用される 3 次元グラフを表示するプログラムである。現在は棒グラフと散布図だけである。メニューも図 3.3.1 に示すように非常に簡単である。実行結果の 3D 棒グラフを図 3.3.2 に、3D 散布図を図 3.3.3 に示す。



図 3.3.1 3 D グラフメニュー

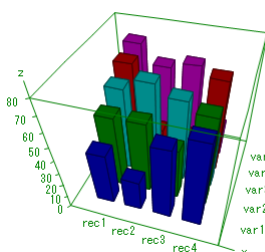


図 3.3.2 3 D 棒グラフ

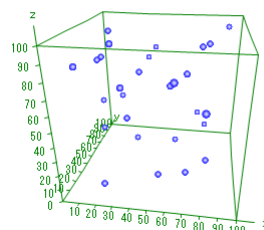


図 3.3.3 3 D 散布図

このグラフには軸目盛間隔の設定などの基本的な機能が欠けている部分があり、今後機能追加とグラフの種類の充実が必要である。これからのプログラムである。

### 3.4 フラクタルビューア 3 D

3 Dビューアを利用してこれまであまり目にしていない図形を表示する題材としてフラクタルは大変興味深い。フラクタルを表示する方法は多く存在するが、ここでは反復関数を用いる方法と再帰的方法を取り上げた。フラクタルビューア 3 D のメニュー画面を図 3.4.1 に示す。

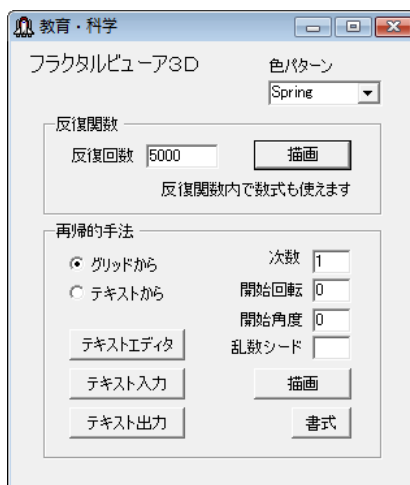


図 3.4.1 フラクタルビューア 3 D メニュー画面

図 3.4.2 のような写像行列と定数及び、それを採用する確率のデータを与えて、反復関数の「描画」ボタンをクリックすると、図 3.4.3 のような 3 次元ギャスケットが表示される。

	確率	a1	a2	a3	a0
▶ 初期値		1	1	1	1
	0.25	0.5	0	0	0
		0	0.5	0	0
		0	0	0.5	0
	0.25	0.5	0	0	60
		0	0.5	0	0
		0	0	0.5	0
	0.25	0.5	0	0	30
		0	0.5	0	60
		0	0	0.5	0
	0.25	0.5	0	0	30
		0	0.5	0	30
		0	0	0.5	60

図 3.4.2 反復関数のデータ



図 3.4.3 反復関数による描画

また、図 3.4.4 のような、フラクタル記述言語（テキストエディタでも利用可能）を用いて、再帰処理の手続きを記述し、再帰的手法の「描画」ボタンをクリックすると、「次数」テキストボックスで与えられた次数のフラクタルが表示される。図 3.4.5 は次数 1, 2, 3 の結果で、図 3.4.6 のサンプル 1 は次数 7 の結果である。図 3.4.7、図 3.4.8 はその他のサンプルであるが、後者は反復関数のサンプルの再帰処理版である。

▶	go	0.4
	separate	
	turn	-30
	frac	0.6
	return	
	rotate	60
	turn	30
	frac	0.6
	return	
	rotate	-60
	turn	30
	frac	0.6
	return	

図 3.4.4 再帰的手法のデータ



図 3.4.5 再帰的手法の描画過程



図 3.4.6 サンプル 1

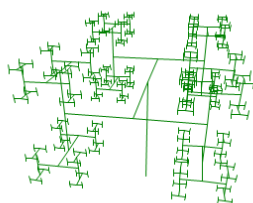


図 3.4.7 サンプル 2

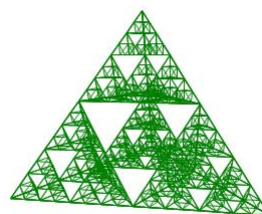


図 3.4.8 サンプル 3

### 3.5 カオスビューア

フラクタルと関係の深いカオスの画像を見るためのプログラムがカオスビューアである。ここでは

3次元の微分方程式を用いたカオスのアトラクタを表示する際に3Dビューアが利用される。カオスビューアのメニュー画面を図3.5.1に示す。これは現在開発を進めているが、現段階では図3.5.2のローレンツ写像のアトラクタと図3.5.3のレスラー写像のアトラクタを「アトラクタ」ボタンをクリックすることで見ることができる。これらのアトラクタがどのような過程で作られるかを示す動画を、アトラクタを表示した後「過程」ボタンをクリックすることで見ることができる。

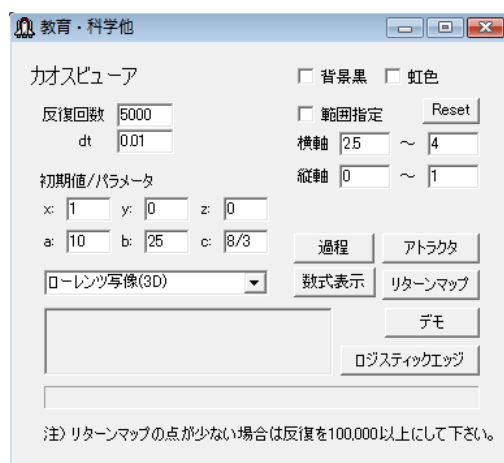


図 3.5.1 カオスビューアメニュー画面

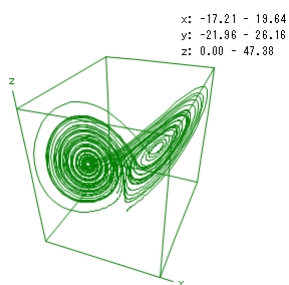


図 3.5.2 ローレンツ・アトラクタ

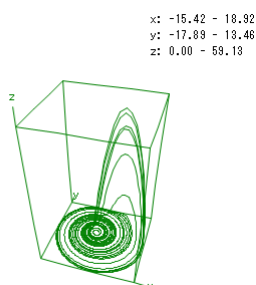


図 3.5.3 レスラー・アトラクタ

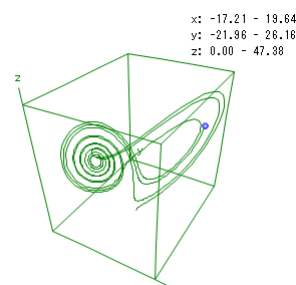


図 3.5.4 実行過程表示

### 3.6 くるくるエディット

これは後に述べるおもしろグラフのデータを作るために作成した中心軸周りの回転図形を描くプログラムである。図3.6.1にその実行画面を示す。左側の「ポイント」や「ライン」ボタンを選択して図形を描き、「図形描画」ボタンをクリックすると、左端の線を回転軸として、ラインの色を面の色とした図3.6.2のような回転図形を描く。「回転分割」を多く取って、フレームを取り去るときれいなグラフィック図形となる。くるくるエディットのデータは「グリッドへ」ボタンでグリッドに移して保存する。その際、後のおもしろグラフで利用するために、ページに図形を表す「どら焼き」など

の名前を付けておくとよい。

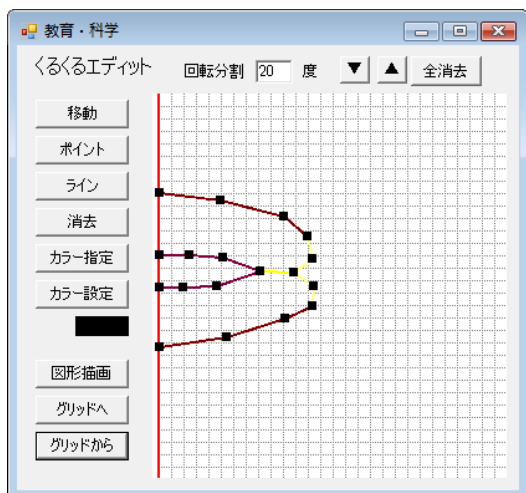


図 3.6.1 くるくるエディット画面

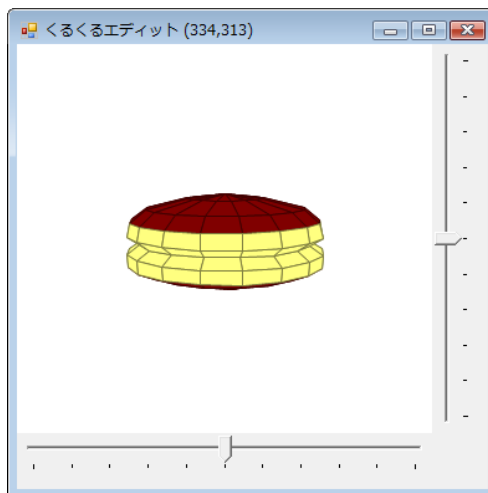


図 3.6.2 くるくるエディット実行結果

### 3.7 おもしろグラフ

おもしろグラフは統計グラフとしては殆ど役に立たないが、楽しめるグラフである。現在、種類は棒グラフ、折れ線グラフ、円（パイ）グラフに対応して、どら焼き棒グラフ、へび線グラフ、アップルパイグラフの3種類である。実行画面を図 3.7.1 に示す。どら焼き棒グラフのグラフィックデータは、モデルデータからでもくるくるエディットで作られたデータからでもよい。また、アップルパイグラフのグラフィックデータはくるくるエディットのデータを利用する。ラジオボタンで利用するモデルの種類を選択し、「読込記憶」ボタンでデータファイルを指定して、データを読み込む。その後、グラフデータのファイルを開き、「変数選択」で利用する変数を選択する。グラフの種類を選択して、「描画」ボタンをクリックするとグラフが表示される。図 3.7.2 から図 3.7.4b まではそれぞれのサンプルである。特にへび線グラフとアップルパイグラフでは、最初表示されるのは図 3.7.4a と図 3.7.5a であり、グラフのように見えるが、図形を回転させるとへびや回転物体がよく見えてくるという、だまし絵のようにになっている。

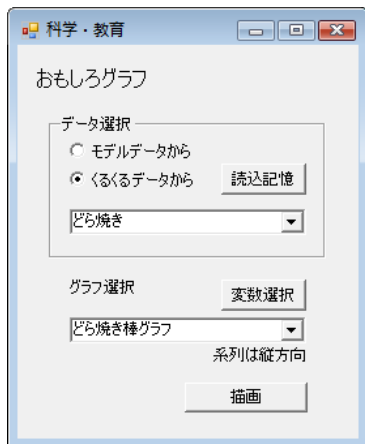


図 3.7.1 おもしろグラフメニュー

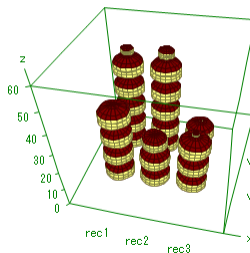


図 3.7.2 どら焼き棒グラフ

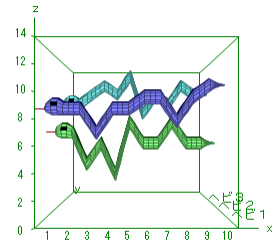


図 3.7.3a ヘビ線グラフ 1

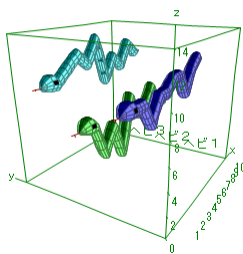


図 3.7.3b ヘビ線グラフ 2

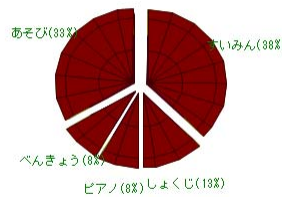


図 3.7.4a アップルパイグラフ 1

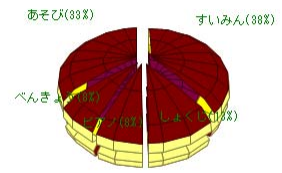


図 3.7.4b アップルパイグラフ 2

## 4. 考察

3Dビューアを他のプログラムから利用する場合、3Dビューアへのデータの渡し方について、少し統一が取れていない部分がある。例えば3Dモデルや3Dグラフの場合は、表 4.1 左のグラフィックデータそのものを渡しているが、おもしろグラフ（棒グラフとパイグラフ）の場合はグラフ描画のためのデータを渡し、グラフィックデータについては別に2章で述べた `pointset` 構造体の配列を含む表 4.1 右の `datamodel` 構造体（実際のプログラムでの名前は少し異なる）の配列（この場合は成分数1）を渡している。さらに2変量関数やおもしろグラフ（へび線）の場合は、グラフ描画用データのみを渡し、処理を3Dビューアに任せている。

表 4.1 3Dビューアへのデータ渡し

モデルデータのデータ渡し	グラフデータのデータ渡し
要素数, 色整数, 頂点1のx座標, 同y座標, ...	グラフ描画用データ（通常のグラフと同じ）
要素数, 色整数, 頂点1のx座標, 同y座標, ...	+
要素数, 色整数, 頂点1のx座標, 同y座標, ...	Structure <code>datamodel</code>



	Dim n As Integer	'model 構成点数
	Dim size As Single	'model サイズ
	Dim max() As Double	'各軸方向の最大値
	Dim min() As Double	'同最小値
	Dim mp() As g_pointset	'pointset 構造体
	End Structure	
	Public g_model() As datamodel	

現時点では、これらのうちどれかに統一しようと考えても問題が生じ、うまい方法が見つからない。統一的な方法が取れるのか、このままの形態で続けるのか今後の課題である。

3章で3Dビューアを利用するプログラムのごく簡単な説明をしたが、プログラムに応じて完成度はまちまちである。2変量関数グラフ、フラクタルビューア3Dは、かなり完成度が高い。くるくるエディットとおもしろグラフは著者の趣味の部分が大きいので、このあたりで打ち止めにしておくべきだろう。カオスビューアはもっと多くのカオスを取り扱うようにしたいし、もう少し統一性も取りたいと思う。3Dグラフはもっと多くの応用があるはずである。特に3D散布図は今まではドットの位置に垂直な棒を立てたりして分かり易くする工夫をしていたが、3Dで直接位置が把握できるようになれば面白い使い方も考えられる。例えばコレスポンデンス分析などの表示では次元を増やして3次元で位置関係を見ることができるようになる。

ここでは取り上げなかったが、3Dパナーメーカーというプログラムも作成している。これは、コピーによって切り取った画像を平面、球形、円筒形、メビウスなどに投影するプログラムであり、3Dビューアを利用しているため動かして見ることができる。但し、画素数が大きくなると処理速度が遅くなるのであまり実用的ではないが、手書きの線画では線の部分だけデータ化するので、ペイントブラシで水木先生の一反木綿などを描くとヒラヒラとよく動く。

3Dビューアの開発は、グラフィックエディタの開発<sup>2)</sup>と合わせて、College Analysisの弱点であったグラフィック機能の向上に大きな影響を与えるであろう。これらによりグラフィックの取り扱い方法や表示速度が著者らの知識となったからである。実行速度が気になるVisual Basicでも、標準的なパソコンで、描画の要素数1万程度まではリアルタイムに動かせることが分った。この経験により、これまで見栄えに問題のあった通常のグラフ表示も改良されて行くかも知れない。

3Dテレビや3Dパソコンが発売された現在、3D画像の需要は増えて行くであろう。現在はアナグリフの表示であるが、近い将来これを3Dパソコンに対応するように拡張することはさほど難しくはないはずである。そのときに備えて様々な現象を取り扱うプログラムを作成しておきたいと思う。

## 参考文献

- 1) College Analysis における「2 変量関数表示ユーティリティ」の構造と機能, 石丸敬二, 福井正康, 福山大学経済学部経済学論集, 35 巻, 2 号, (2010) 87-106.
- 2) 社会システム分析のための統合化プログラム 13 – グラフィックエディタとその応用 –, 福井正康, 石丸敬二, 尾崎誠, 宋東明, 福山平成大学経営学部経営研究, 7 号, (2011) 掲載予定.

# **Multi-purpose Program for Social System Analysis 16**

## **- 3D Viewer and its Applications -**

Masayasu FUKUI, Makoto OZAKI and Keiji ISHIMARU\*

Department of Business Administration, Faculty of Business Administration,  
Fukuyama Heisei University

\* Department of Economics, Faculty of Economics,  
Fukuyama University

### **Abstract**

We have been constructing a unified program on the social system analysis for the purpose of education. Our program has so far been mainly on statistical analysis and management science, but we will add new programs which can be used for lectures on mathematical science in general education course of universe. In this paper, we introduce multipurpose 3D graphics viewer program and some sample programs which use this “3D viewer.”

### **Keywords**

College Analysis, social system analysis, OR, statistics, analysis of covariance structures

URL: <http://www.heisei-u.ac.jp/ba/fukui/>