

# **College Analysis**

## リファレンスマニュアル

福井正康

福山平成大学経営学部経営学科

## はじめに

College Analysis は、主に大学の文系学部と一般教育の数理系授業で使用することを目的とした統合型フリーソフトウェアです。世の中には、優秀な分析ソフトがたくさん存在しますが、その中であって、いろいろな分析を気軽に利用できるお手軽感（利用者の方がおっしゃってました）がこのソフトの持ち味です。そのため分析を深く追求して行かれる数理系の研究者の方には全くお勧めできません。ただ、予算の少ない文系学部で授業を担当されている理系出身の研究者の方などには役に立つかも知れません。作者自身そういった立場だからです。これからも少しずつ分析を増やし、面白いソフトにして行こうと思います。

さて、この本についてですが、これまでは「このソフトにはマニュアルがない」として、紀要論文や作成中のテキストをマニュアル代わりに使っていただいております。しかし、古い紀要論文には最新の情報を書き込めない、テキストは問題等考えているとなかなか進まないということから、最近少し行き詰っておりました。そこでこの問題を緩和するために、この度リファレンスマニュアルを作ってみようと考え始めました。新しい分析は紀要論文を書くたびにその内容を使って追加し、機能変更をした場合はその都度更新する。そのため、以下のような考え方でマニュアル作りを始めたいと思います。

- できるだけ最新の情報に更新して行く。
- 分野ごとの分冊にしてページ数を別にする。
- 分冊の右肩に「分析名／分野名」を付け、まとめて並べると本になるようにする。
- 訂正が容易なように、レイアウトにこだわらず、図はすべて行内にする。
- 図の番号などは、変更が楽なように、分冊の章ごとに付ける。

内容が数学的になりすぎていたり、やたらと式に番号が付いていたり、数式エディタのバージョンが違っていたり、統一性に欠ける部分があると思いますが、古い論文を徐々に改訂しながら最新のものを作って行こうと思いますのでご理解下さい。

福井正康

# College Analysis リファレンスマニュアル

－ ツール －

## 目次

1. 共通メニュー.....	1
2. グリッドエディタ .....	4
3. テキストエディタ .....	14
4. グラフィックエディタ .....	16
5. 出力 .....	26
6. 3Dビューア .....	31

## 1. 共通メニュー

College Analysis のメニューには、各フォームに付属する個別メニューとどんな場合でも利用できる共通メニューがある。ここでは後者の共通メニューについて説明する。共通メニューの内容について表 1 に示す。

表 1 共通メニュー

メニュー	サブメニュー
エディタ表示	グリッドエディタ, テキストエディタ, グラフィックエディタ QDA プロジェクト プログラム終了
分析	基本統計, 多変量解析, 数学, OR, 意思決定支援他, 教育・科学他 質的研究 0 値補正, 見学会
ヘルプ	色パターン, 数式内利用可能関数 謝辞

College Analysis が扱うデータ編集ツール (エディタ) には、グリッドエディタ、テキストエディタ、グラフィックエディタ及び、質的研究プログラム開発用の QDA プロジェクトがある。共通メニューにはこれらの「エディタ表示」の機能がついているが、テキストエディタとグラフィックエディタは分析メニューからも呼び出すことができる。

「分析」は、それぞれのサブメニューの中にさらにメニューが含まれているが、詳細は別に記す。「0 値補正」とは、数値計算に含まれる誤差のうち、特に重要な、0 と認識する範囲を示す数値を与えるものである。図 1 に 0 値補正の画面を示す。

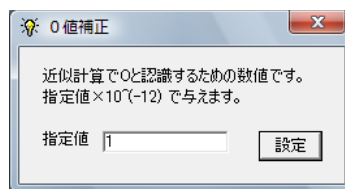


図 1 0 値補正

ここで、デフォルトは  $10^{-12}$  になっているが、この値を変更する際に用いられる。すべての分析でこの値を使っているわけではないが、線形計画法はこれを利用している代表例である。

「見学会」は、作者が体験入学会などでたまに利用するメニューで、実行画面は図 2 のようになっている。これは、適合度検定を、小学生から高校生にやってみせるためのプログラムである。

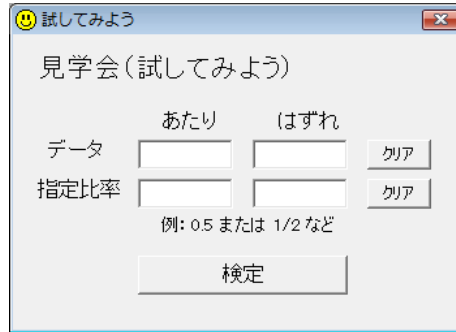


図 2 見学会

[ヘルプ] メニューの「色パターン」は、3D ビューアや2D フラクタル等で使われる、色パターンの一覧である。図 3 にその画面を示す。

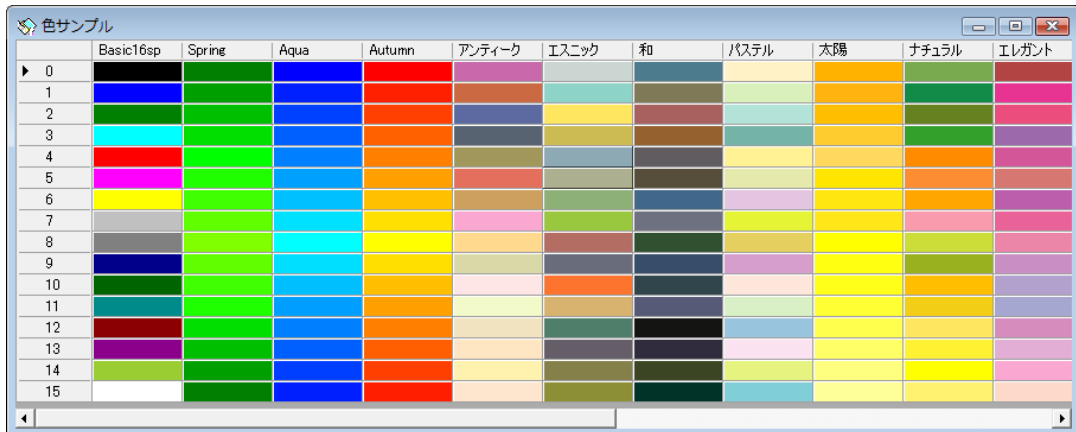


図 3 色パターン

「数式内利用可能関数」サブメニューは、College Analysis で使える数式について簡単に表したものである。図 4 にその画面を示す。

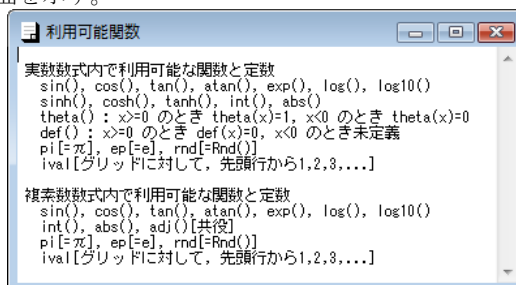


図 4 数式内利用可能関数

ここでは、実数式と複素数式に分けて表示してある。複素数式での値の主値が取られている。関数は今のところ1パラメータのものだけを定義している。

「謝辞」サブメニューは、プログラム開発で利用した参考書、協力していただいた方々の名前が書いてある。図5にその画面を示す。

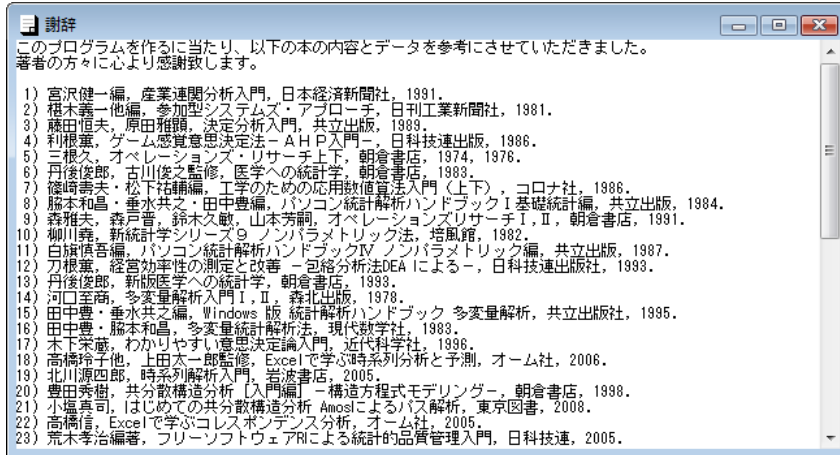


図5 謝辞

図の中では見えないが、最後の緒方昭先生と田口賢士先生には教わるが多かった。改めてお礼を述べたい。

## 2. グリッドエディタ

エディタは各分析データの入力や加工を共通に行う部分で、分析ソフトウェアの使い易さを左右する重要な役割を担う。このシステムでも個々の分析プログラムと同様に開発初期から幾度となく改定を行ってきた。現在のエディタ画面を図1に示す。

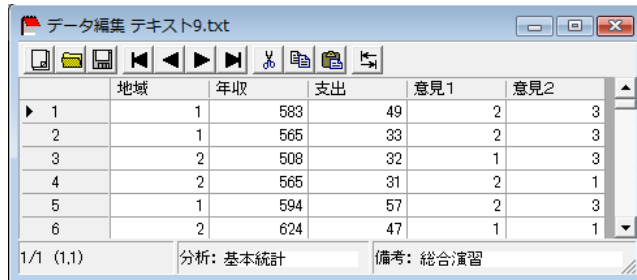


図1 エディタ画面

図1のメニューバーに並ぶメニューは初期のものとは大差ないが、メニューから呼び出されるサブメニューについては、種類がかなり増えている。それらのうちグリッドエディタに固有のものを表1に与える。

表1 グリッドエディタのメニュー

メニュー	サブメニュー
ファイル	新規作成, 開く, 上書き保存, 名前を付けて保存 ページ設定, 現在シート印刷 終了
編集	元に戻す 全コピー 全貼り付け, 先頭列付全貼り付け 切り取り, コピー, 貼り付け, 列追加貼り付け, 消去 シート名入力, 列名入力, 行名入力 行列交換, 行列名揃え, 対称行列設定, 空白除去
挿入／削除	シート追加, シート挿入, シート削除 列追加, 列挿入, 列削除 行追加, 行挿入, 行削除
表示	先頭シート, 前シート, 次シート, 最終シート 書式変更, グリッド幅

以下、それぞれのメニュー・サブメニューの詳細を示す。

### 2.1 ファイルメニュー

サブメニュー「開く」と「名前を付けて保存」については、Visual Basic のコモンダイアログを利

用しており、標準的なソフトウェアと同様の操作でファイルを指定することができる。

サブメニュー「新規作成」では、図 1.1 に示される入力画面が現れる。

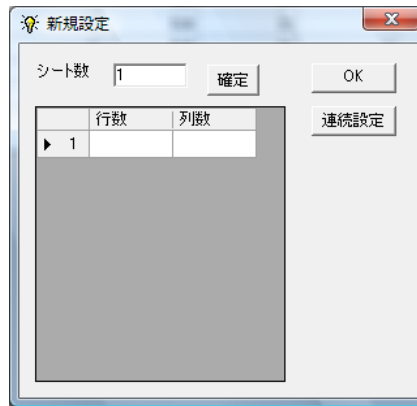


図 1.1 新規作成画面

複数のシートを設定できるので、最初に「シート数」を指定する。「確定」ボタンで、それに合わせてグリッドが表示されるが、シート毎に行数と列数を指定できるようになっている。同じ数値を連続的に指定する場合は、最初の数値だけ入力して、「連続設定」のコマンドボタンをクリックする。すべてのデータを入力し終わったら、「OK」ボタンで終了すると、エディタ画面に空欄の表が表示される。何も入力せずに「OK」ボタンをクリックすると 1 行 1 列のシートが表示される。全貼付け等はこれのできるので、そのまま「OK」することが多い。

サブメニュー「上書き保存」は編集中のデータをファイルに上書きする。もちろん、データに変更があった場合は、確認のメッセージボックスが開き、誤った保存を防ぐ。

サブメニュー「現在シート印刷」は現在表示されている機能をプリンタに出力するものであるが、あまり満足できるものではない。基本的に、表示される入出力データはワープロなどに貼り付けることを前提としており、印刷機能は確認の意味程度になっている。

## 2.2 編集メニュー

このメニューには様々なサブメニューがある。まず、「全コピー」と「全貼付け」であるが、「全コピー」は、固定行・固定列も含めて 1 つのシート全体をクリップボードにコピーする。ワープロや表計算ソフトに貼り付ける場合には、この全コピー機能が便利である。「全貼付け」は全コピーした 1 枚のシートを他のシートに貼り付ける際に用いる。複数シートの場合、特に固定行・固定列が同じ場合があり、この貼付けが便利である。

この全貼り付けを利用する例として、Excel からのデータ貼り付けを説明する。College Analysis の統計処理では Excel からのデータ貼り付けを勧めている。例えば、Excel で図 2.1 のように選択し、

コピーする。左端の数字の部分(最左列)は空欄でも構わない(場合によってはなくても問題でない)。上の変数名は空欄でもよいが、1行分は確保しておくことが望ましい。

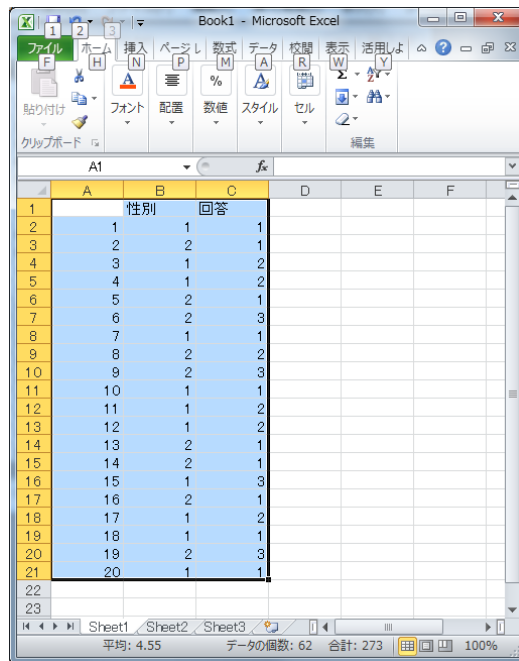


図 2.1 Excel からのコピー

その後は、図 2.2 のように College Analysis 側の [編集－全貼り付け] で、最上行は変数名、最左列は行名として張り付けることができる(最左列がない場合は、[編集－先頭列付全貼り付け] で、行名が空欄になって貼り付け可能である)。

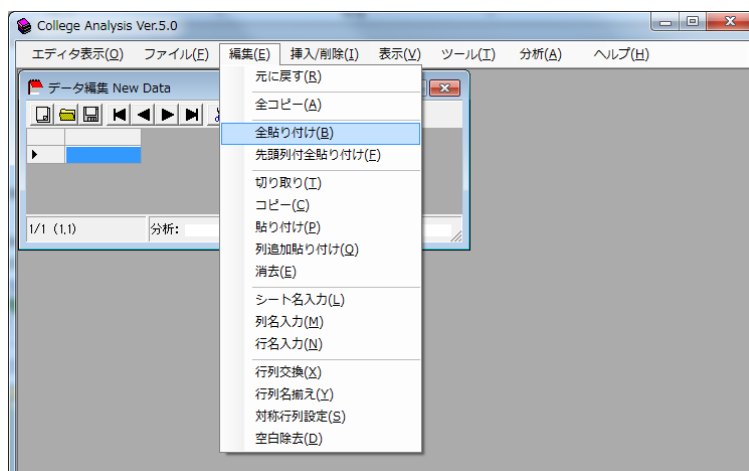


図 2.2 College Analysis での貼り付け

また、最上行も最左列もない場合は、図 2.1 の「編集－貼り付け」で、自動的に空欄の変数名と行名を付けて貼り付け可能である。この場合、「範囲を拡張して貼り付けますか？」のメッセージが出る場合があるが、「はい」を選択する。

列名や行名を含まない College Analysis 内のセルについては、「切り取り」、「コピー」、「貼付け」、「消去」のサブメニューを用いる。これらは通常の表計算ソフトウェアと同様、範囲を選択して実行する。データの消去については、「Delete」キーを用いてもよい。サブメニュー「列追加貼付け」は、列を最後に追加して貼り付ける機能である。データを新しい列を作って貼付けようとする場合、役に立つ。

各シートには、名前を付けることができる。これはサブメニュー「シート名編集」を選択して、ダイアログボックス内で指定する。そのシート名はタイトルバーの部分に「シート名/ファイル名」の形で表示される。固定行に表示される列と行の名前は、「列名編集」、「行名編集」のサブメニューで入力できる。図 2.3 に列名の入力画面を示す。



図 2.3 列名入力画面

現在の列数に応じてグリッドが表示され、列名を入力する。行名の入力も同様の画面で行う。特に行名の場合、連続的なレコード番号を付けることもあるので、「連番」というコマンドボタンを用意している。これにより、1 からレコード数まで、番号が自動的に入力される。「列名編集」においては、「連番」は var1 から var 列数のように、番号の前に「var」という文字列が付く。これは特に統計ソフトウェアなどで利用されるだろう。すべてのデータを一新する場合には、「全消去」コマンドを利用する。

サブメニュー「行列交換」はグリッドの行と列を転置して表示する。「行列名揃え」は ISM 等のデータ入力時に利用されるが、行名と列名が同じ場合に、入力の省力化のために用いられる。「対称行列設定」は、三角行列の状態に入力されたデータから、対称行列を作る入力ツールである。

サブメニュー「元に戻す」は、データを1段階前に戻す。ただ、どの時点に戻すかで、もう少し議論が必要である。

## 2.3 挿入／削除メニュー

シートの枚数や列数・行数を変更する場合はすべてこのメニューを用いる。この機能は通常編集メニューの中にあることが多いが、サブメニューが多くなりすぎるので独立させた。シートの枚数の変更には、「シート追加」、「シート挿入」、「シート削除」サブメニューが利用される。

「シート追加」は新しいシートが最後のシートとして1枚追加され、「シート挿入」は現在開いているシートの前に1枚挿入される。また、「シート削除」は現在開いているシートを1枚削除する。シート単位の処理については、まとめて行うことが少ないと思われるので1シートずつ行うようにしている。

列数の変更もサブメニュー「列追加」、「列挿入」、「列削除」を用いて行う。「列挿入」は現在選択している列数だけ、その位置に挿入する。複数列を選択していない場合は、何列分挿入するかダイアログボックスで聞く。「列削除」については、現在選択している列を削除する。また、「列追加」は予めダイアログボックスで列数を聞いて、最後の列として追加する。行についても同様の操作で変更する。

「行列交換」はグリッドの行と列を転置して表示する。「行列名揃え」はISM等のデータ入力時に利用されるが、行名と列名が同じ場合に、入力の省力化のために用いられる。「対称行列設定」は、三角行列の状態に入力されたデータから、対称行列を作る入力ツールである。

## 2.4 表示メニュー

データは複数シートに格納されるので、各シートを選択する機能が必要である。これには、サブメニュー「先頭シート」、「前シート」、「次シート」、「最終シート」を用いる。

エディタ画面の列幅はマウスにより自由に変えることができるが、一度に全体を変更したり、小数点以下桁数を設定する必要がある場合は、サブメニュー「書式」を用いる。図4.1に設定画面を示す。

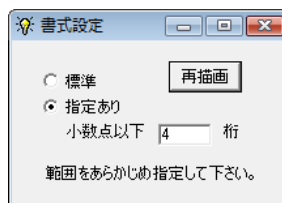


図 4.1 データ書式設定画面

グリッドエディタ上でデータの小数点桁数を指定する範囲を選択し、設定画面を適当な値にして、「再描画」ボタンをクリックする。グリッドエディタは書式設定すると元の値は（1回の「元に戻す」

機能を除いて) 残らないので、注意する必要がある。著者自身はあまり使うことはない。

逆によく利用する機能がサブメニュー「グリッド幅」の調整である。図 5 に設定画面を示す。

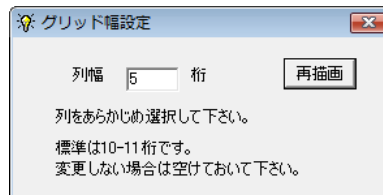


図 5 グリッド幅設定画面

横長のグリッドエディタを見易くする場合などに利用される。

## 2.5 ツールメニュー

### 並べ替え・検索・置換

行の並べ替えには、サブメニュー「並べ替え」を用いる。設定画面は図 5.1 に示す。

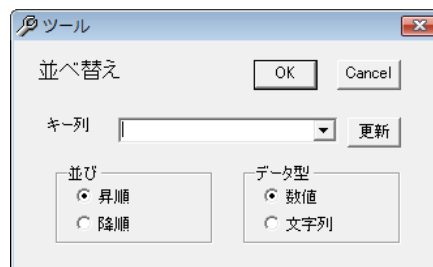


図 5.1 並べ替え画面

並べ替えのキー列を指定し、昇順か降順及び、数値としてか文字列としてかを選択して実行する。キー列のデータの値が同じ場合、エディタ上の行の順番は保持されるので、キー列が2つ以上必要な場合には下位のキーから順番にキー変数を変えて並べ替えを行えばよい。

データの検索にはサブメニュー「検索」を用いる。図 5.2 にその実行画面を示す。

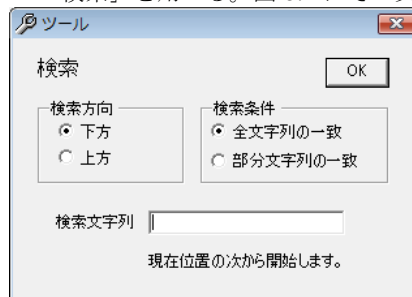


図 5.2 検索画面

文字列は現在のカーソル位置から、上か下に検索される。

指定したデータの一括変更にはサブメニュー「置換」を利用する。その設定画面を図 5.3 に示す。

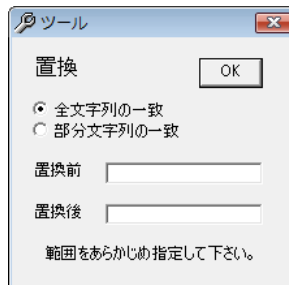


図 5.3 置換画面

これは、例えば数字や記号で入力された性別データなどを、「男」と「女」に変更するような場合に用いる。予め置換検索の範囲を設定して実行する。範囲は複数列に渡ってもよいが、飛び飛びの領域には対応していない。

## 2.6 データ発生サブメニュー

サブメニュー「データ発生」をクリックすると図 6.1 のような画面が表示される。

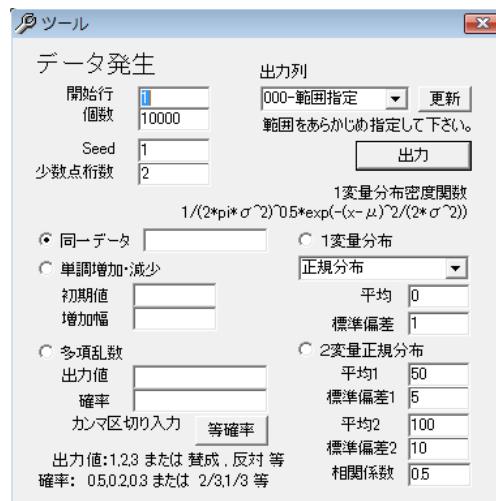


図 6.1 データ発生画面

データ発生はまず出力列を指定する。グリッド中の列を指定することもできるし、新たに列を追加して出力することもできる。列の指定はグリッド中を選択して「範囲指定」としてもよいし、列指定

のコンボボックスで選択指定してもよい。出力データの個数はデフォルトで 10000 個であるが、行数が少ない場合は行数が限度である。乱数発生に再現性を与えるため、乱数の「Seed」を与えて発生させる。乱数の小数点以下桁数も指定する。

発生乱数の種類としては、図 5.4 にあるように、同一データ、単調増加・減少、多項分布、1 変量分布、2 変量正規分布があるが、それらのパラメータの値を指定して選択する。1 変量分布は分布の数が多いため、コンボボックスから分布の名前を選択して実行する。2 変量正規分布は、散布図を描くときに役に立つが、新規追加の場合は 2 列追加される。講義資料などの作成には便利なメニューである。

## 2.7 計算サブメニュー

既存のデータから、計算式を利用して新たなデータを作り出すには、図 7.1 のサブメニュー「計算」を用いる。

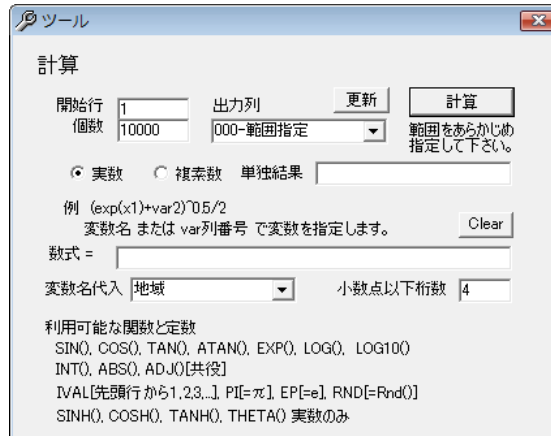


図 7.1 計算画面

計算はデータ発生と同様に、まず出力列を指定する。グリッド中の列を指定することもできるし、新たに列を追加して出力することもできる。列の指定はグリッド中を選択して「範囲指定」としてもよいし、列指定のコンボボックスで選択指定してもよい。出力データの個数はデフォルトで 10000 個であるが、行数が少ない場合は、行数が限度である。

計算は、「実数」と「複素数」を選択できる。複素数の表式は例えば、「 $3.25+2.63i$ 」のように虚数部分には「i」を付ける。出力列で「範囲指定」にして、1 つのセルだけが選択されている場合は、単独の計算とみなされ、「単独結果」テキストボックスに計算結果が表示される。

計算式の指定は通常の Basic 言語にほぼ準拠している。計算に利用できる関数は現在のところ、図 7.1 のメニューの下部に示された関数である。詳しくはメニュー [ヘルプ>数式内利用可能関数] で表示される画面にも書かれている。

計算にある列の値を利用する場合、列名が決まっている場合は列名をそのまま用いてもよいし、var4のように「var」の後に（1から始まる）列番号を付けて指定してもよい。大文字と小文字は区別しない。データを出力する列は「出力列」コンボボックスから選択する。また、「変数名代入」コンボボックスで指定して、変数名を代入してもよい。

## 2.8 データ結合

データ結合は、特に2次元より多い次元の分割表の作成に用いられる。サブメニュー「データ結合」を選択すると図8.1に示される画面が表示される。

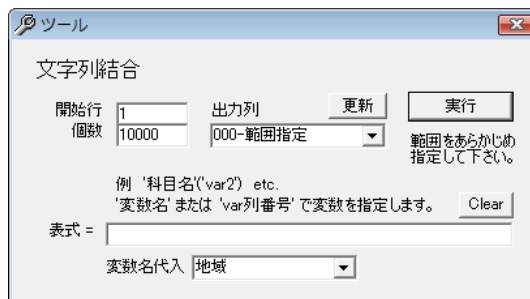


図 8.1 文字列結合画面

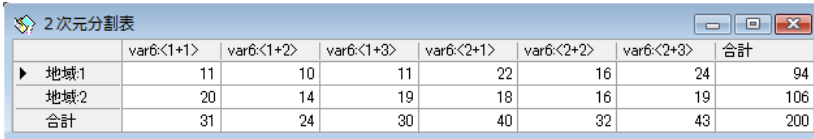
計算はデータ発生や計算と同様に、まず出力列を指定する。グリッド中の列を指定することもできるし、新たに列を追加して出力することもできる。列の指定はグリッド中を選択して「範囲指定」としてもよいし、列指定のコンボボックスで選択指定してもよい。出力データの個数はデフォルトで10000個であるが、行数が少ない場合は、当然行数が限度である。

例えば図1のデータ画面で、表式として、「<'意見1'+ '意見2'>」のように指定し、出力列を「新規追加」に変更して、実行すると図8.2のようになる。

	地域	年取	支出	意見1	意見2	
▶ 1		1	583	49	2	<2+3>
2		1	565	33	2	<2+3>
3		2	508	32	1	<1+3>
4		2	565	31	2	<2+1>
5		1	594	57	2	<2+3>
6		2	624	47	1	<1+1>
7		1	617	48	2	<2+1>
8		1	458	53	2	<2+3>

図 8.2 文字列の結合

これを元に、「地域」と空白の var6 変数を選択して、基本統計の質的データの集計のところで述べる分割表を描くと、図 8.3 のようになる。



	var6:<1+1>	var6:<1+2>	var6:<1+3>	var6:<2+1>	var6:<2+2>	var6:<2+3>	合計
▶ 地域1	11	10	11	22	16	24	94
地域2	20	14	19	18	16	19	106
合計	31	24	30	40	32	43	200

図 8.3 2次元分割表の拡張

これは3次元分割表に対応している。これを使うとさらに次数の高い分割も可能である。

### 3. テキストエディタ

テキストエディタは、グリッドエディタへの入力補助として利用されることが多い。図1にテキストエディタのシンプルな画面を示す。

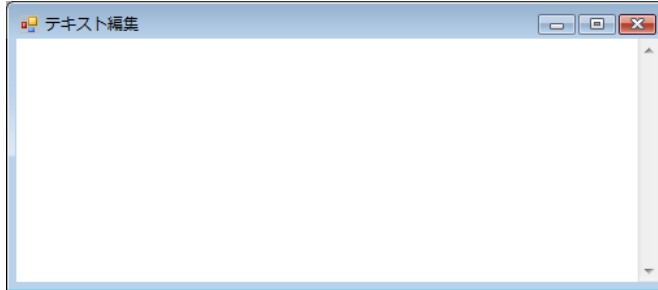


図1 テキストエディタ

テキストエディタには表1のようなメニューがあるが、説明は不用であろう。

表1 テキストエディタのメニュー

メニュー	サブメニュー
ファイル	新規作成, 開く, 上書き保存, 名前を付けて保存 ページ設定, 印刷 終了
編集	全コピー 切り取り, コピー, 貼り付け 検索

この他に、テキストボックスで標準に利用できるショートカットキーなども利用できる。

テキストボックスの利用法を、後に述べる OR 中の線形計画法を元に説明する。図2は線形計画法の分析画面である。「テキストボックス」ボタンで、図1のテキストボックスを表示し、図3のように線形計画問題の式を入力する。

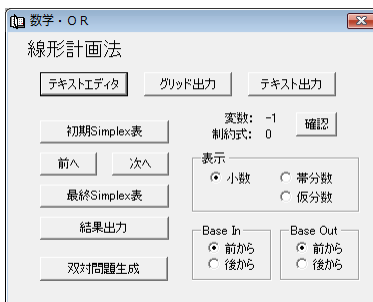


図2 線形計画法分析画面



図3 テキスト入力画面

分析画面の「グリッド出力」ボタンをクリックすると、図 4 のように式がグリッド形式に変換され、グリッドエディタに出力される。

The screenshot shows a window titled "データ編集 New Data" with a toolbar and a grid editor. The grid contains the following data:

	x1	x2		
▶	5	6		MAX
	1	3	<=	60
	3	4	<=	100
	2	1	<=	50

At the bottom of the window, there are fields for "1/1 (1,1)", "分析:", and "備考:".

図 4 グリッド出力

計算はこの式を元に行われる。また、図 4 の形式から、図 3 のテキスト形式にも「テキスト出力」ボタンで変換可能である。テキストエディタは、多目的線形計画法の数式入力や、フラクタルビューア 2D・3D などのプログラム入力に使われるが、詳細な説明はそれぞれのソフトの解説箇所に譲る。

## 4. グラフィックエディタ

### 4.1 概要

グラフィックエディタを作成するに当たり、以下のような基本仕様を考えた。

1. 画面は箱型や楕円形の図形（以後これをボックスと総称する）とその間に引かれた関係を表す直線や曲線（以後これをラインと総称する）で構成される。
2. ラインは方向性を持つものと持たないものがある。
3. 分析手法に応じてボックスやラインの形を変える。
4. データはできるだけ簡略化し、グリッド（表）エディタと1対1で変換できる。

これまで我々が取り扱ってきた分析において、分析の元となるデータと関係する図（階層構造図）はすべて1～3の枠組みに収まる。また4. は共分散構造分析において1つのデータに複数の構造モデルを対応させるために考えたもので、グリッドエディタに元データと一緒にグラフィックデータを保存することにより、AMOS などのソフトに見られるようなファイルの増加を抑え、ファイル管理を容易にする。

グラフィックエディタはピクチャーボックスとその上にある文字列編集用のテキストボックスからなる。テキストボックスは必要なとき以外には見えなくなっている。ピクチャーボックス上に描かれた図形は、ボックスの座標位置だけを管理しており、ラインについては始点と終点のボックスの番号から描画の際に求める。テキストボックスはボックスやラインの名前を変更する場合のみ利用される。

システムとして見た場合のグラフィックエディタの位置づけは、データの階層構造入力を補助し、階層構造出力をより分かりやすく表示するためのツールである。階層構造の多くは数学的に行列で表されるので、グラフィックエディタで表される図形は位置情報を持った行列データである。そのためグリッドエディタと1対1の対応関係にすることが可能である。テキストエディタでもこの互換性を意識しており、線形計画法や多目的線形計画法では分析メニューを介してグリッドエディタと対応関係にある。一部の例外を除いて、分析の中心はあくまでグリッドエディタのデータ形式で、グラフィックエディタやテキストエディタはその補助である。図 1.1 にシステムのサブルーチン構造の概要を示す。

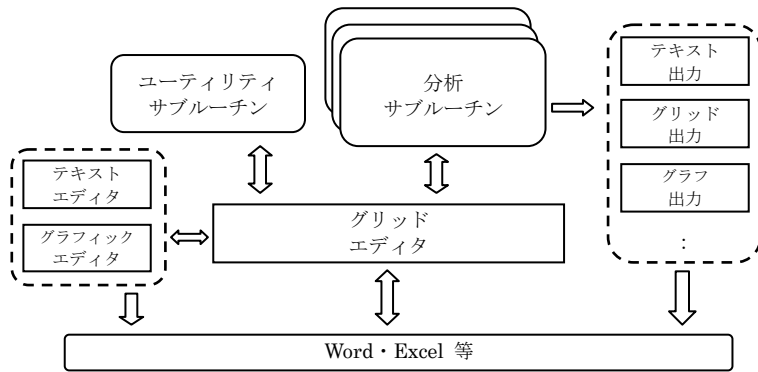


図 1.1 College Analysis のサブルーチン構造

データの入力用として、グリッドエディタ、テキストエディタ、グラフィックエディタがあり、出力用には、グリッド出力、テキスト出力、グラフ出力、グラフィックエディタ、ピクチャー出力、3Dビューアがある。グラフィックエディタは、入力用としても、出力用としても利用される。

グラフ出力とグラフィックエディタの違いは、前者が単なる画像であるのに比べて、グラフィックエディタはマウスで移動可能な図形要素の集まりである。グラフ出力は表示される Window のサイズによって伸縮可能であるが、グラフィックエディタには、図形要素を追加していく可能性があることから、伸縮機能は付けていない。

## 4.2 グラフィックエディタのプログラム構造

グラフィックエディタの2つの要素であるボックスとラインは図 2.1a と図 2.1b のようなデータ構造体で表される（変数名は実際のプログラムとは異なる）。

```

Structure BoxData
  Dim mode As Integer      ' ボックスの種類
  Dim name As String       ' 名前（表示文字列）
  Dim left As Single      ' 左端の x 座標
  Dim top As Single       ' 左端の y 座標
  Dim width As Single     ' 幅
  Dim height As Single    ' 高さ
  Dim value As String     ' ボックスに付属する値
End Structure
Dim bdata() as BoxData    ' ボックスデータの配列

```

図 2.1a ボックスのデータ構造

```

Structure LineData
  Dim mode As Integer      ’ ラインの種類
  Dim name As String      ’ 名前 (表示文字列)
  Dim start As Integer    ’ 開始点のボックスの配列番号
  Dim dest As Integer     ’ 終了点のボックスの配列番号
  Dim value As String     ’ ラインに付属する値
End Structure
Dim ldata() as LineData  ’ ラインデータの配列
    
```

図 2.1b ラインのデータ構造

ボックスとラインの種類は固有の番号で与えられる。ボックスの名前は長い文字列として使われる場合があるので、ボックスの種類によっては改行記号を「&」として複数行の指定が可能となっている。またそれぞれに付属する値は、数値の場合もあれば、分数のような表記もあるので文字列型にしている。ラインデータは開始点と終了点がボックスの番号だけで指定されており、グラフィック座標の値は持っていない。

グラフィックエディタ内部でのデータ構造は図 2 の形であるが、データをグリッドエディタに保存する場合は、表 2.1 の行列形式で保存される。

表 2.1 グリッドエディタへのデータの保存

	名前	・・・	名前	種類	順番	Left	Top	Width	Height	Value
名前				1	0	120	20	60	30	0
・ ・ ・		(ラインデータは 開始点の行、終了 点の列の位置に) 名前;種類;番号; 開始点;終了点; 値 (の形で入力)		・ ・ ・	・ ・ ・	・ ・ ・	・ ・ ・	・ ・ ・	・ ・ ・	・ ・ ・
名前				11	19	625	595	60	30	0

網かけの部分が、ラインデータを保存する部分であり、その他のところにボックスデータを保存する。図 2.1 のデータ形式と表 2.1 のデータ形式を入れ替えることによって、グラフィックエディタとグリッドエディタでデータの相互変換を行う。

次の章ではこれらのデータ形式で与えられるグラフィックエディタの具体的な利用法と各分析<sup>1)・3)</sup>における利用例を与える。

### 4.3 グラフィックエディタの利用法と利用例

ここではグラフィックエディタの基本的な利用法を初期設定画面のサンプルをもとに説明する。図 3.1 に初期設定画面の描画例を示す。

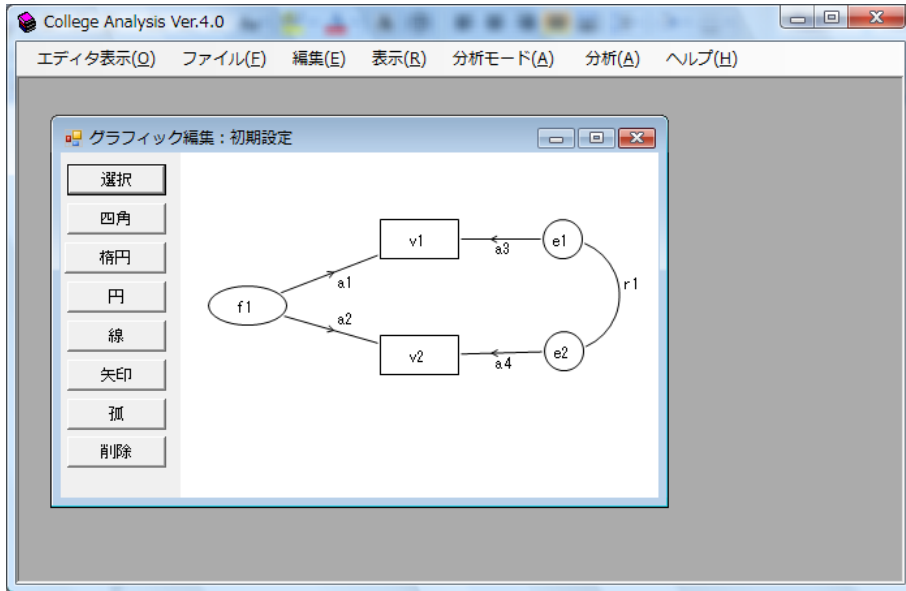


図 3.1 初期画面描画例

グラフィックエディタは、左側のボタンと上のメニューバーとピクチャーボックスからなる。描画は左側のコマンドボタンを選んでそれぞれの描画モードを決め、画面上をクリックする。四角、楕円、円の大きさや形は予め決定されている。ボックスやライン上をダブルクリックするとテキストボックスが現れ、文字列を変更できるようになるが、文字列の長さに応じてボックスの大きさは自動調整される。矢印はコマンドボタンを選んで、ボックスからボックスへマウスをドラッグさせて引く。ボックスをドラッグすると移動するが、それに伴って矢印も再描画される。選択・移動モードのとき、ボックスを囲むようにドラッグすると、複数のボックスが選択でき（赤色表示）、それらをまとめて移動させることもできる。操作に失敗した場合も最大 10 回は元に戻せる。これらの処理は一般的な画像ソフトと似ているので、何度か試せば特に戸惑うことはないであろう。

#### 共分散構造分析

以後は具体的に各分析について実行画面を示しながら特徴的な部分を説明する。メニュー「分析－多変量解析等－共分散構造分析」を選択すると図 3.2 のようなメニュー画面が表示される。「グラフ

「グラフィックエディタ」コマンドボタンでグラフィックエディタを表示させ、構造図を描いたものが図 3.2a である。メニュー「エディタ表示－グラフィックエディタ」を選択した後、メニュー「分析モード－共分散構造分析」を選んでも同じ処理ができる。

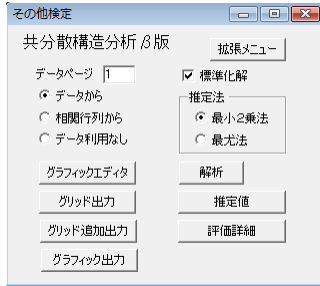


図 3.2a エディタ編集画面

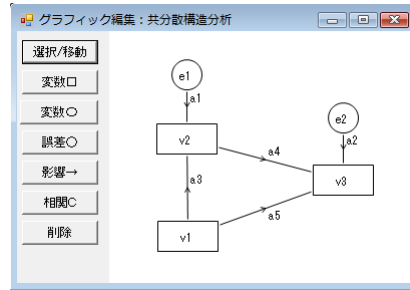


図 3.2b 構造図

ボックスの種類には観測・構造変数を与える四角、潜在・構造変数を与える楕円、潜在・誤差変数を与える円があり、ラインには影響を与える矢印と相関を与える円弧がある。このデータを「グリッド出力」コマンドボタンでグリッドへ貼り付けると図 3.2c のようになる。グラフィックエディタのメニュー「編集－グリッド出力」を使っても同じ処理ができる。また、「グリッド追加出力」コマンドボタンやメニュー「編集－グリッド追加出力」でグリッドエディタの最終頁にデータを追加することもできる。

	e2	e3	v1	v2	v3	種類	順番	Left	Top	Width	Height	Value
e2				a4.22.3.3:1.0...		12	3	70	32	30	30	0
e3					a5.22.4.4:2.0...	12	4	323	131	30	30	0
v1				a1.22.0.0:1.0...	a3.22.2.0:2.0...	1	0	53	187	60	30	0
v2					a2.22.1.1:2.0...	1	1	54	97	60	30	0
v3						1	2	199	131	60	30	0

図 3.2c グラフィックデータのグリッドへの貼り付け

変数名（ボックス名）の v1, v2, v3 は図 3.2d のグリッドエディタにある観測変数のデータ名と一致させている。構造データの画面でメニューの「解析」ボタンをクリックした後、「推定値」ボタンをクリックすると図 3.2b の矢印に相当するパラメータ名が推定値に変更されて図 3.2e のように表示される。

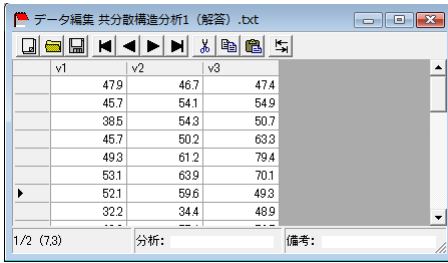


図 3.2d 観測変数の実測値

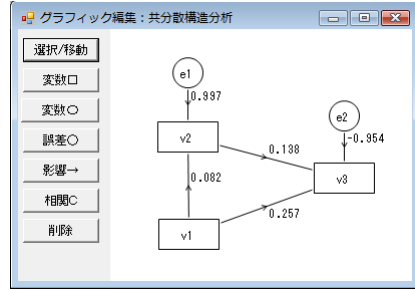


図 3.2e パラメータ推定値

図 3.2e は、推定値がラインの value 変数に代入され、メニュー「表示-パラメータ値」のパラメータ表示モードに自動的に変更されたものである。共分散構造分析ではこのようにしてグラフィックエディタが利用される。

### ISM

ISM (Interpretive Structure Modeling) では構造図の出力に利用される。分析メニュー画面は図 3.3a で与えられる。各種の構造図は右側のボタンで表示される。例えば相互到達可能な要素をひとまとめでにした構造図 (我々はこれを縮約構造図と呼ぶ) は図 3.3b のように表される。

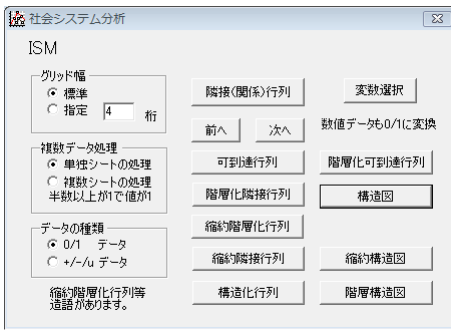


図 3.3a ISM 分析メニュー

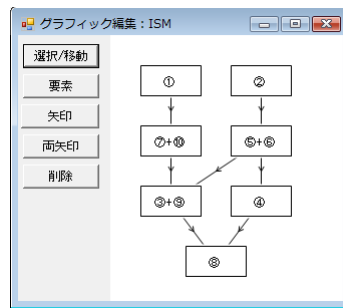


図 3.3b 縮約構造図

ボックスには要素を与える四角があり、ラインには一方向の影響を与える矢印と相互の影響を与える両方向矢印がある。図 3.3b は見た目に分り易く表示された例であるが、例えばすべての隣接的影響を描く構造図の場合などは図 3.3c のようにボックスが重なって表示される場合がある。このようなときは利用者の判断でボックスを移動させて図 3.3d のように分かり易くしておく必要がある。ラインは自動的についてくるのでこれは特に困難ではない。必要であれば分かり易く配置した時点でグリッドに保存しておく。

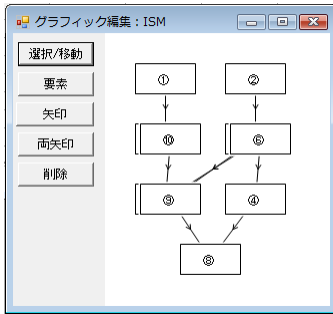


図 3.3c 出力された構造図

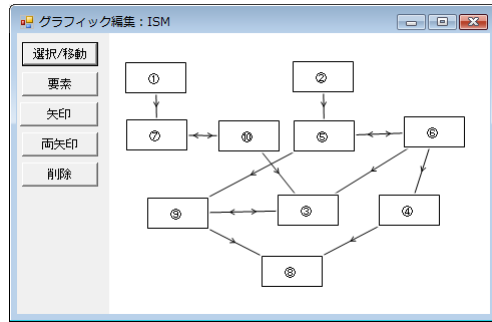


図 3.3d 整理された構造図

今の段階では出力された状態はあまり分かり易くないが、表示のアルゴリズムの改良によって整理された形に近づけられる可能性がある。

### AHP

階層的意思決定手法である AHP (Analytic Hierarchy Process) のメニュー画面は図 3.4a である。階層構造を表す構造図のデータはグリッドエディタからでもグラフィックエディタからも入力できる。グラフィックエディタのデータは、分析メニューの「グリッド出力」ボタンからでも、グラフィックエディタのメニュー「編集→グリッド出力」からでもグリッドエディタに出力可能であるが、変数の表示順が分析メニューからだとは階層順、グラフィックエディタのメニューからだとは変数名昇順となる。

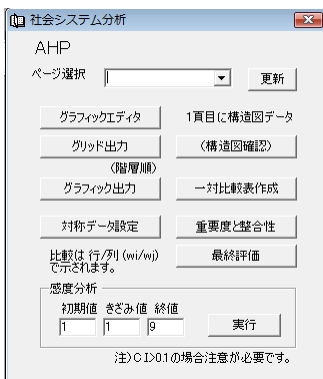


図 3.4a AHP 分析メニュー

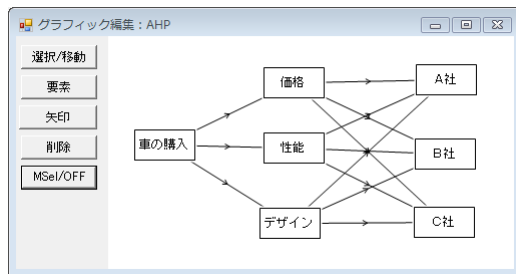


図 3.4b 構造図入力

この分析におけるグラフィックエディタのボタンの特徴は「MSel/OFF」ボタンで、これをクリック

すると他のボタンは利用不可能になり、ボタン名も「MSel/ON」となる。これは一度に複数の矢印を引くモードで、例えば、価格、性能、デザインボックスを複数選択し、車の購入ボックスをクリックすれば、後者から前者 3 つに矢印が引かれる。再度「MSel/ON」ボタンをクリックすると通常モードに戻る。ただ分析を実行するという観点からは、グラフィックエディタを使って入力するより、直接グリッドエディタを用いた方が効率的かも知れない。

## デシジョンツリー

デシジョンツリーは多段階意思決定手法を分かり易く図に表現したもので、複数の意思決定によって確率の期待値を求める方法である。この分析手法はまず図を描くのが大変であるので、グラフィックエディタは非常に役に立つ。分析メニューを図 3.5a に示すが、大変シンプルなものである。グラフィックエディタを用いて図を描き、グリッドエディタに取り込み、「グラフィック表示」ボタンをクリックした画面が図 3.5b である。

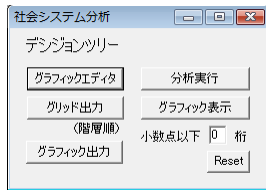


図 3.5a 分析メニュー

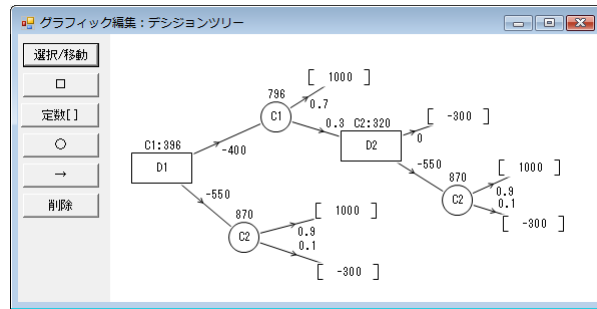


図 3.5b デシジョンツリー結果画面

図 7b で与えられる [ ] 中の数値は利得と呼ばれるが、これも一つのボックスである。また、四角や円のボックスの上に数値などが付いているが、これもデシジョンツリー用の Value 値のついたボックスである。

## PERT

スケジュール管理に使われる基本的な手法である PERT を学ぶ際には、アローダイアグラムの作成が一番難しい。現在開発中のプログラムのメニュー画面とそのデータをそれぞれ図 3.6a、図 3.6b に示す。図 3.6b の中では作業名、先行作業、所要日数が利用され、分析メニューの「アローダイアグラム」ボタンで結果が表示されるが、表示のアルゴリズムの検討が不十分なため、ISM のときと同様の手直しが必要である。

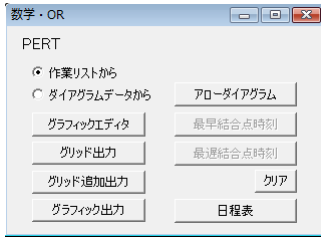


図 3.6a PERT 分析メニュー

作業名	先行作業	所要日数	作業内容	
1	A	7	設計	
2	B	A	3	地盤工事
3	C	B	5	基礎工事
4	D	A	6	資材調達
5	E	C	3	屋根工事
6	F	C/D	6	外壁・防水...
7	G	E	4	床面工事
8	H	F/G	5	内壁工事
9	I	E	3	ガス・水道工...
10	J	H/I	2	電気工事
11	K	F/G	10	仕上工事

図 3.6b PERT 用データ

自動出力された図のノード (円形のボックス) の位置を動かし、手直しをした画面が図 3.6c である。

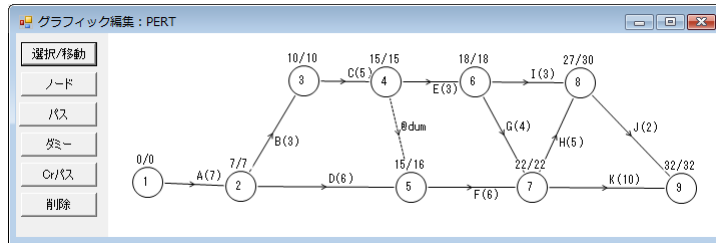


図 3.6c PERT 図結果画面

矢印の名前の括弧内の値はラインの Value である。また、ノードの上にある分数表示は、分子のところが最早結合点時刻と呼ばれ、次の仕事を始められる最早の時刻で、分母のところは最遅結合点時刻と呼ばれ、いつまで次の仕事を待てるかを表す時刻である。この分数形式もノードの Value である。Value の型を文字列型にしたのはこのような場合に対応させるためである。

### 特性要因図

品質管理の七つ道具の 1 つである特性要因図 (フィッシュボーン図) はこれまでのボックスをラインで結ぶものとは違った形となっている。図 3.7a に品質管理の分析メニューを、図 3.7b に特性要因図の入力画面を示す。

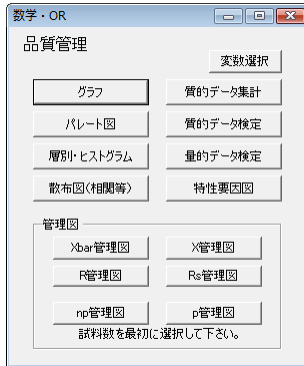


図 3.7a 品質管理分析メニュー

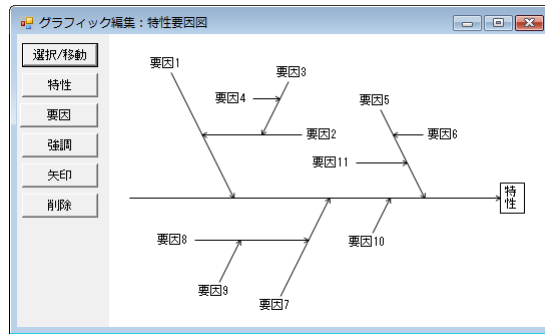


図 3.7b 特性要因図入力画面

特性要因図はボックスをラインで直接結ぶ形ではないが、ラインの傾きとその開始点、終了点となるボックスが決まっているため、それらのボックスの位置とボックスとラインの次数（階層）だけでラインの先頭座標は計算できる。このことから、ボックスとラインの次数をそれぞれの Value に保存しておけば、データ構造は我々のグラフィックエディタの枠内に収まる。ラインを描画する際に再帰処理を使って座標を求めるため、通常の描画より時間がかかるが、実際の作業上は全く問題にならない。ラインは、最初に描く特性ボックス（右端）以外、ボックスからラインにマウスをドラッグすることで描画する。

以上我々のプログラムでのグラフィックエディタの利用例であるが、これら以外にも多くの利用法が考えられ、我々のシステムには欠かせないツールとなった。

グラフィックエディタには多くの図形が登録できるが、現在のシステムでは図形の登録番号があまりシステム化されていない。当初、1～9 を四角形、11～19 を楕円と円、21～29 を矢印と直線、31～39 を円弧と規定していたが、矢印などは予想よりも多く、かなりの部分を使っている。そのため特性要因図などには別の数値を当てはめており、登録番号の管理がくずれている。また表示スピードも予想以上であるので、もっと多くの図形を考えることも可能となった。著者らはこれらの番号を100番単位に置き換えるように拡張して行く予定であるが、どのような図形が必要になるのか十分に吟味する必要がある。今後新しい図形を登録しながら、よりシステム化された番号の登録法を考えて行きたい。

## 5. 出力

College Analysis の出力には、グリッド出力、テキスト出力、グラフ出力、グラフィックエディター、ピクチャービューア、3D ビューアがある。ここではこれらの画面と主な機能について紹介する。

### 5.1 グリッド出力

グリッド出力は、最も一般的な出力である。図 1.1 にその例を示す。

	度数	相対度数(%)	累積度数	累積相対度数(%)
▶ 200<=x<400	1	0.5	1	0.5
400<=x<600	85	42.5	86	43
600<=x<800	103	51.5	189	94.5
800<=x<1000	5	2.5	194	97
1000<=x<1200	3	1.5	197	98.5
1200<=x<1400	2	1	199	99.5
1400<=x<1600	1	0.5	200	100

図 1.1 グリッド出力

グリッド出力はメニューを持っているが、表 1.1 にそれを示す。

表 1.1 グリッド出力のメニュー

メニュー	サブメニュー
ファイル	名前を付けて保存 ページ設定, 印刷 終了
編集	全コピー, コピー 行列交換, エディタ貼付, エディタ追加
表示	書式変更, グリッド幅

メニューは殆どグリッドエディタのメニューに含まれるものだが、「編集」メニューの「エディタ貼付」と「エディタ追加」はグリッド出力だけの機能である。「エディタ貼付」はデータをそのままグリッドエディタの現在のページに貼り付け、「エディタ追加」はグリッドエディタに新しいページを追加し、データをそのまま貼り付ける。

### 5.2 テキスト出力

テキスト出力も多くの分析に利用されている。図 2.1 にその例を示す。

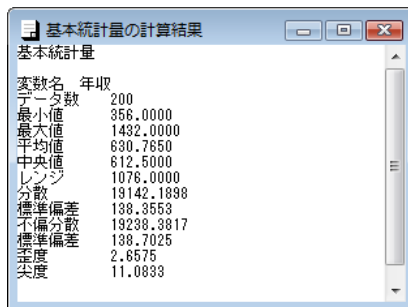


図 2.1 テキスト出力

テキストエディタには表 2.1 のようなメニューがある。

表 2.1 テキストエディタのメニュー

メニュー	サブメニュー
ファイル	名前を付けて保存 ページ設定, 印刷 終了
編集	全コピー, 切り取り, コピー, 貼り付け

### 5.3 グラフ出力

グラフ出力も多くの分析から汎用的に利用される出力形式である。グラフ出力には、多くのグラフを描くプログラムが含まれており、新しいグラフが必要になると、番号を付けて追加している。これまで登録されたグラフは以下の通りである。

1. 散布図
2. デンドログラム
3. AHP 構造図
4. ヒストグラム
5. 棒グラフ
6. 積重ね棒グラフ
7. 折れ線グラフ
8. 円グラフ
9. 横棒グラフ
10. 積重ね横棒グラフ
11. 関数グラフ
12. 待ち行列グラフ
13. 横帯グラフ
14. 3D 棒グラフ
15. 多重散布図
16. 横折れ線グラフ
17. パレート図
18. QC 管理図
19. 2変数関数グラフ
20. レーダーチャート
21. 箱ひげ図
22. トルネードチャート
23. 2次元パラメータ関数グラフ

図 3.1 にグラフ出力の例を示す。

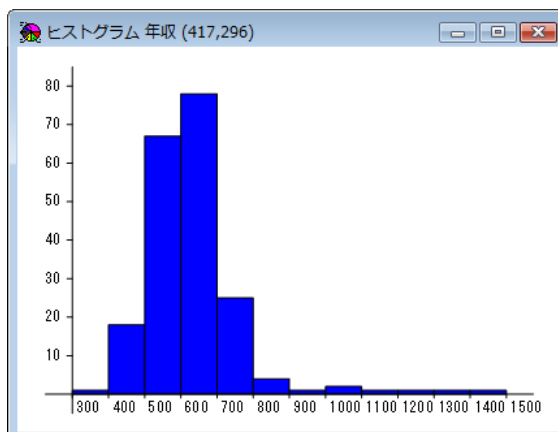


図 3.1 グラフ出力

グラフ出力には表 3.1 のようなメニューがある。

表 3.1 グラフ出力のメニュー

メニュー	サブメニュー
ファイル	名前を付けて保存 ページ設定, 印刷 終了
編集	画面コピー 項目名変更, データ・凡例名変更, 項目名位置変更 データ表示
設定	%表示 [ON/OFF], データラベル [ON/OFF], 線グラフデータポイント [ON/OFF], 凡例 [ON/OFF], 回帰直線 (近似曲線) [ON/OFF], 正規楕円半径 (非表示, $\sigma$ , $1.5\sigma$ , $2\sigma$ ), x 軸文字縦書き [ON/OFF] 軸設定

メニュー「ファイル」のサブメニュー「名前を付けて保存」は、ビットマップ、gif、jpeg 形式の中から選んで保存する。

メニュー「編集」の「画面コピー」は、表示されたグラフをクリップボードにコピーし、その他のソフトに貼り付けられるようにする。「項目名変更」は、グラフに渡すデータの行名の変更である。例えば、棒グラフでは横軸の項目名、円グラフでは分割領域の項目名である。「データ・凡例名変更」はグラフに渡すデータの列名の変更である。例えば、積み重ね棒グラフでは凡例名である。「項目名位置変更」は、円グラフの文字位置を変えるためのものである。これらは、すべてのグラフで使えるというのではなく、使えない場合はその旨のメッセージが表示される。サブメニュー「データ表示」はグラフを描くためのデータがグリッド出力で表示される。Excel 等でもっときれいなグラフを描きたい場合には便利な機能である。

メニュー「設定」の中のサブメニューは様々なグラフで使える機能の集合体である。グラフによって使えるものと使えないものがあり、使えない場合はその旨表示される。それぞれのグラフで使えるものを試してもらいたい。

#### 5.4 グラフィックエディタ

グラフィックエディタは、データ作成用だけでなく、構造図出力用としても利用される。図 4.1 は意思決定手法 ISM の構造図の出力例である。

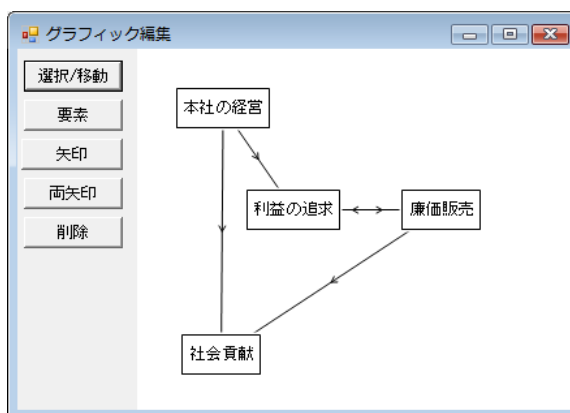


図 4.1 グラフィックエディタ

これについては前節で詳しく述べたので、ここでは省略する。

#### 5.5 ピクチャビューア

ピクチャビューアは、グラフではなく、むしろ絵や写真を表示するための出力環境である。図 5.1 にフラクタルビューア 2D で作られたピクチャビューア表示の例を示す。図の名前はマンデルブロ集合である。

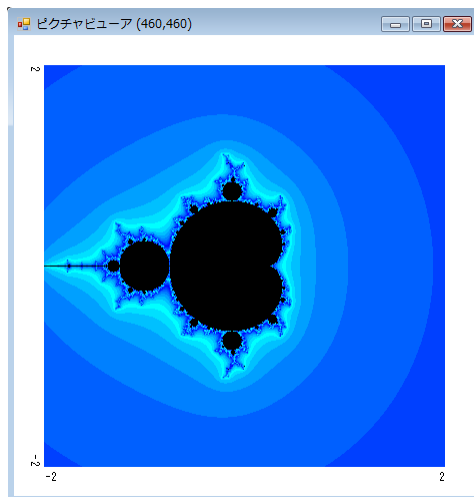


図 5.1 ピクチャビューア

ピクチャビューアには、名前を付けて保存と画面コピーの機能しかない。画像は、ビットマップ、gif、jpeg 形式から選んで保存できる。

## 5.6 3D ビューア

3Dビューアは3次元図形表示用に開発されたビューアで、多くのプログラムから利用されている。図 6.1 に使用例を示す。

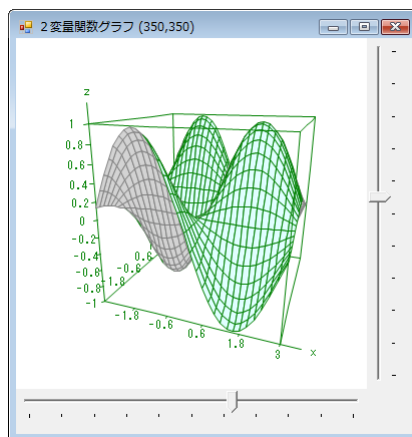


図 6.1 3D ビューア

3Dビューアには多くのメニューがあるので、次章で詳しく説明する。

## 6. 3Dビューア

### 6.1 3Dビューアの概要

この節では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。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1.1のようになる。

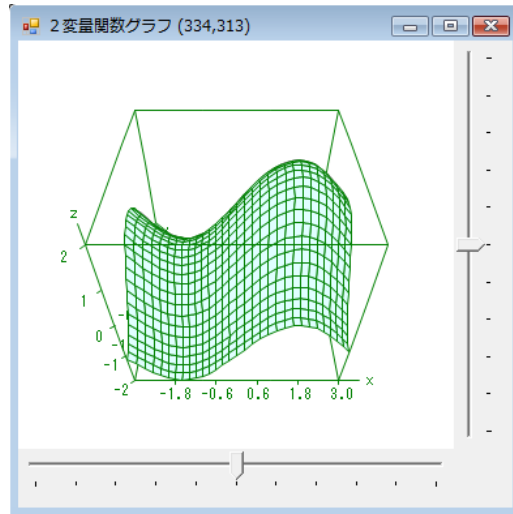


図 1.1 2変量関数グラフの出力画面

この画面上でマウスをドラッグすると、図1.2や図1.3のような別角度からの図が見られる。

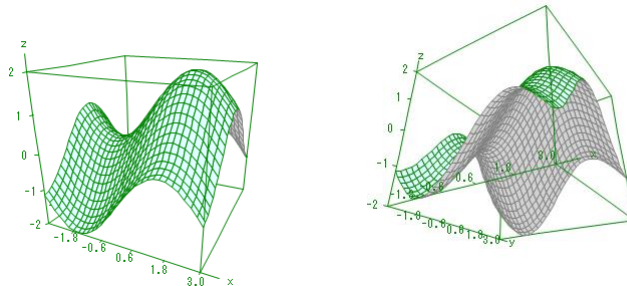


図 1.2 2変量関数グラフ別角度1    図 1.3 2変量関数グラフ別角度2

マウスの左ボタン押し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と表示される。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1.4に示す。

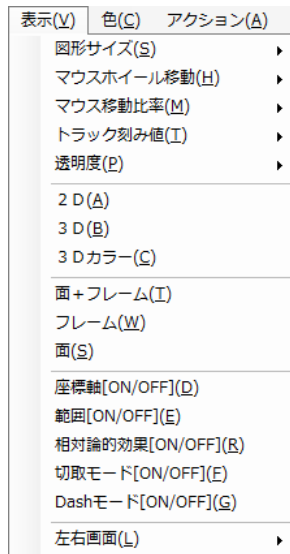


図 1.4 「表示」メニュー

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。

る。

「切取モード[ON/OFF]」については、画面に近づいて行った際に、ある距離を超えた時点で、面や線を非表示にする機能である。これは丁度、雑誌などで見る内部図解のように、部分的に切り取って内部が見えるような感じになる。

「Dash モード」については、立体図形の後ろの線を点線に変える機能である。但し、凸図形に対しては正しく表示されるが、凹凸のある図形では、面の表面と裏面の機能を利用しているため、正しく表示されるとは限らない。

「相対論的效果」については、物体が光速に近いスピードで動くときの見え方をシミュレートするもので、教育・科学の部に章を設けて解説している。

次に「2D」、「3D」、「3D カラー」であるが、「2D」は通常の表示である。「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画面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的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1.5a、図1.5b、図1.5cに示す。

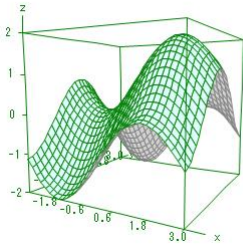


図 1.5a 2D 形式

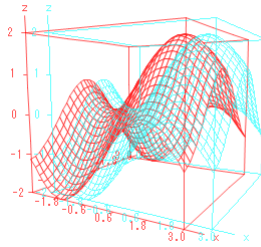


図 1.5b 3D 形式

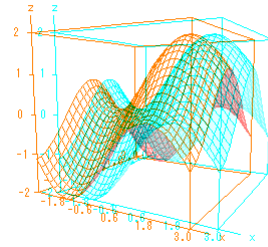


図 1.5c 3D カラー形式

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 1.6a、図 1.6b、図 1.6c にそれぞれの形式の画面を示す。

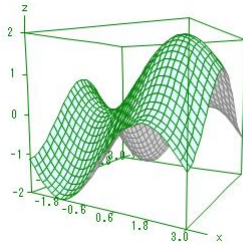


図 1.6a 面+フレーム表示

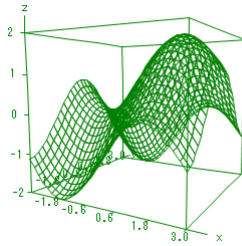


図 1.6b フレーム表示

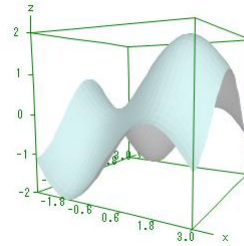


図 1.6c 面表示

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 1.6c 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表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 1.4 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 1.7 に座標軸の非表示画面、図 1.8 に範囲の表示画面を示す。

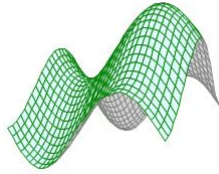


図 1.7 座標軸の非表示

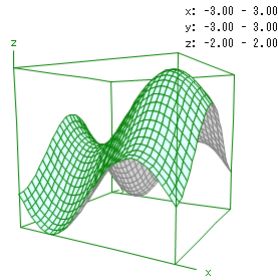


図 1.8 範囲の表示

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 1.4 のメニューの一番下の「左右画面」には5つのサブメニューがあり、これらは左右の目の画像を別々に表示する機能である。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 1.9a (シアン) と図 1.9b (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。

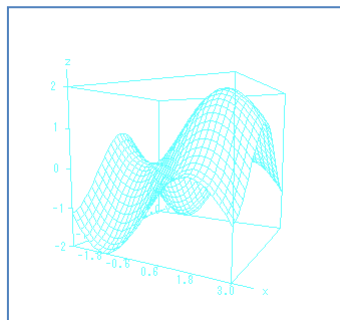


図 1.9a 3D 表示左目用 (シアン)

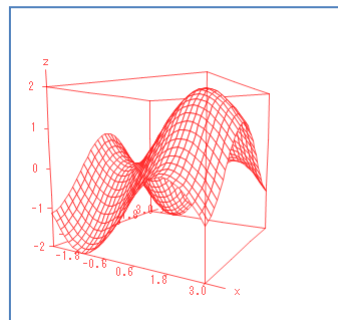


図 1.9b 3D 表示右目用 (赤)

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。

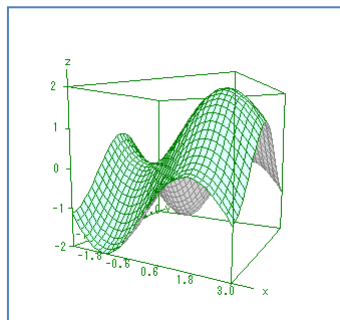


図 1.10a 3D カラー表示左目用

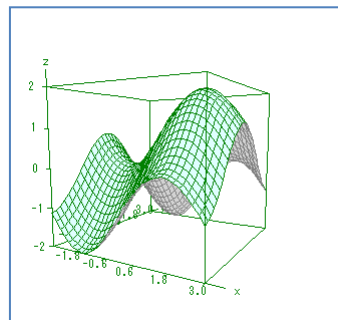


図 1.10b 3D カラー表示右目用

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ヤンの画面を（紙面

では分りにくいと思われるが) 表示する。

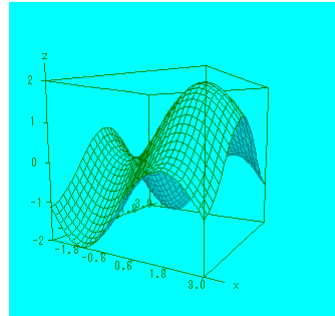
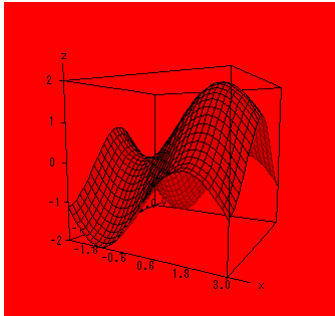


図 2.11a 3D カラー表示左目用カラー 図 2.11b 3D カラー表示右目用カラー

「色」メニューについては、図 1.12 のようになる。

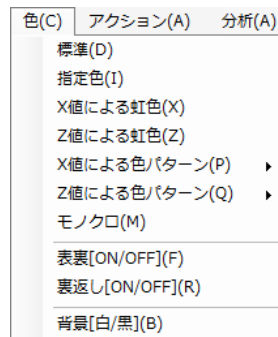


図 1.12 色メニュー

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 1.13a、図 1.13b、図 1.13c 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ている。

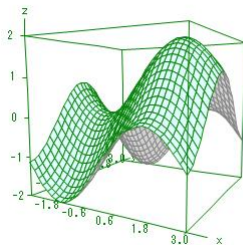


図 1.13a 標準

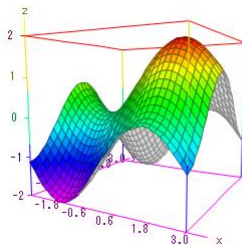


図 1.13b Z 値による虹色

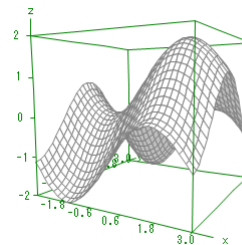


図 1.13c 指定色

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 1.12 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 1.14 に虹色モ

ードでの裏表なし、図 1.15 に虹色モードでの背景黒の画面を示す。

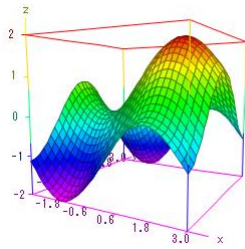


図 1.14 裏表なし

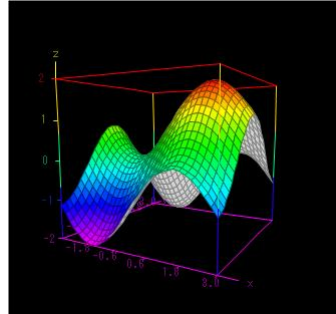


図 1.15 背景黒

その他のメニューについては実際選んでもらえば理解できる。

最後に図 1.16 の「アクション」メニューについて説明する。

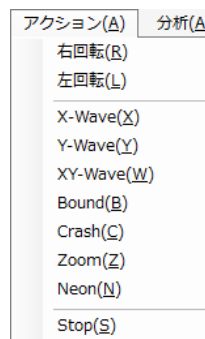


図 1.16 アクションメニュー

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 1.17a 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に z 方向に伸び縮みする。「Crash」は図 1.17b 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。

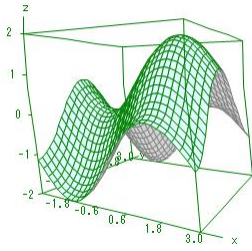


図 1.17a X-Wave

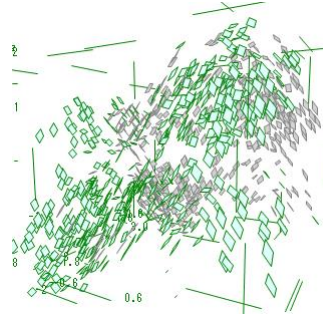


図 1.17b Crash

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。

## 6.2 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。

### 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 2.1 にメニュー画面を示す。

図 2.1 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 2.2 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 2.3 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。

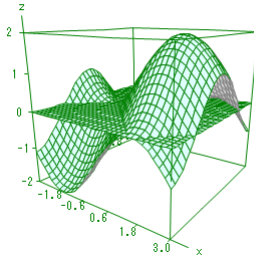


図 2.2 複数グラフの表示 1

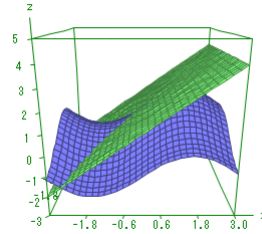


図 2.3 複数グラフの表示 2

### 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 2.4 に与えられる単純なもので、既存のデータを用いて図 2.5 や図 2.6 のようなグラフィックモデルを表示する。

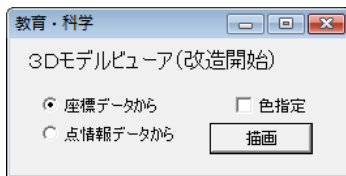


図 2.4 3Dモデルビューアメニュー

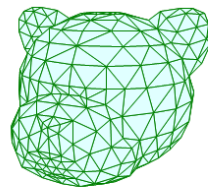


図 2.5 サンプル 1

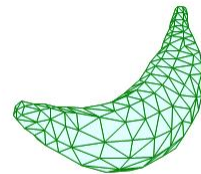


図 2.6 サンプル 2

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大きな変更が予定されているプログラムである。

### 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 2.7 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 2.8 に、3D 散布図を図 2.9 に示す。

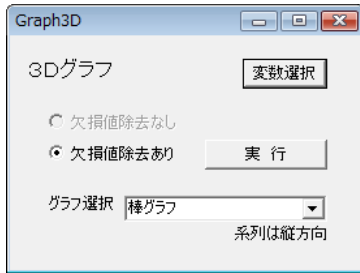


図 2.7 3Dグラフメニュー

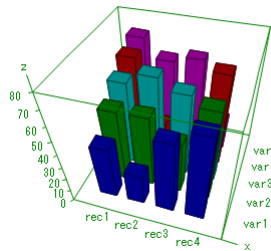


図 2.8 3D棒グラフ

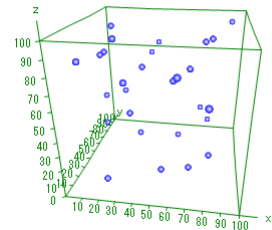


図 2.9 3D散布図

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。

### フラクタルビューア3D

3D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3Dのメニュー画面を図2.10に示す。

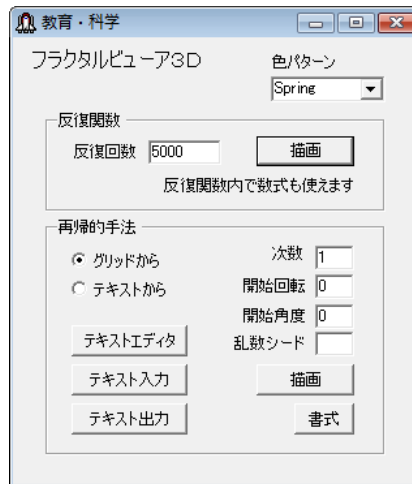


図 2.10 フラクタルビューア3Dメニュー画面

図 2.11 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 2.12 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	60
		0	0	0.5	0

図 2.11 反復関数のデータ



図 2.12 反復関数による描画

また、図 2.13 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、「回数」テキストボックスで与えられた回数のフラクタルが表示される。図 2.14 は回数 1, 2, 3 の結果で、図 2.15 のサンプル 1 は回数 7 の結果である。図 2.16、図 2.17 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 2.13 再帰的手法のデータ

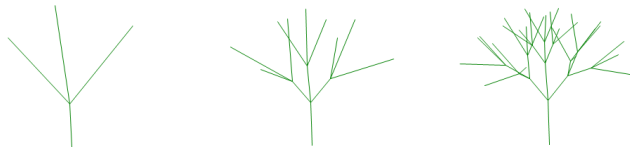


図 2.14 再帰的手法の描画過程



図 2.15 サンプル 1

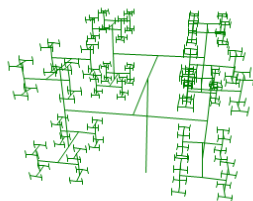


図 2.16 サンプル 2

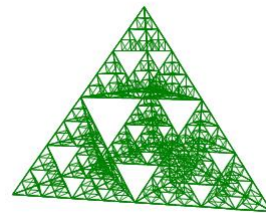


図 2.17 サンプル 3

### 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2.18に示す。これは現在開発を進めているが、現段階では図2.19のローレンツ写像のアトラクタと図2.20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。

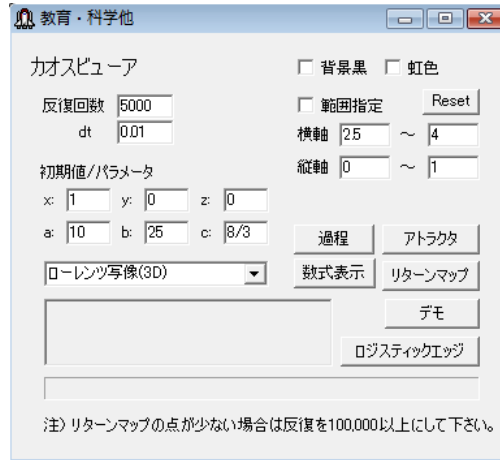


図 2.18 カオスビューアメニュー画面

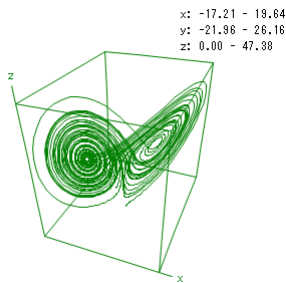


図 2.19

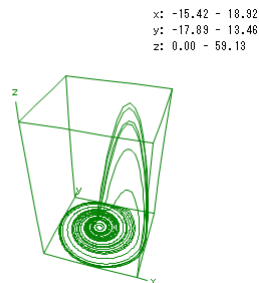


図 2.20

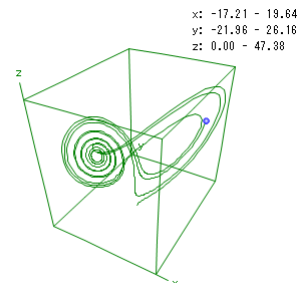


図 2.21

### 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2.22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2.23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。

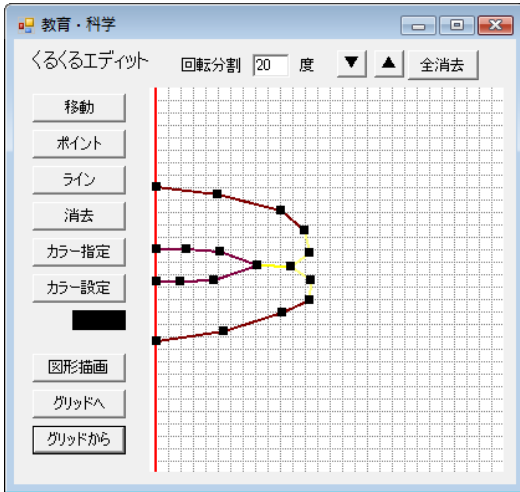


図 2.22 くるくるエディット画面

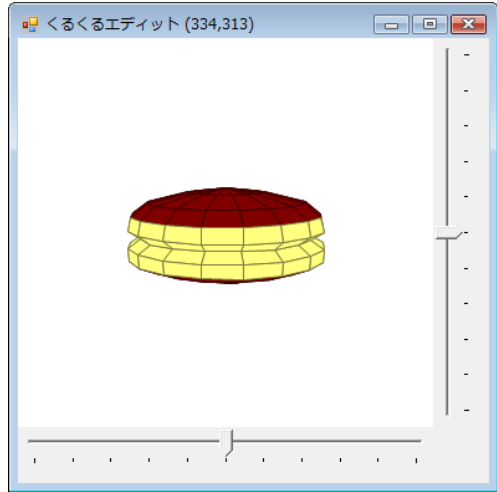


図 2.23 くるくるエディット実行結果

### 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 2.24 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 2.25 から図 2.27b 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 2.26a と図 2.27a 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。

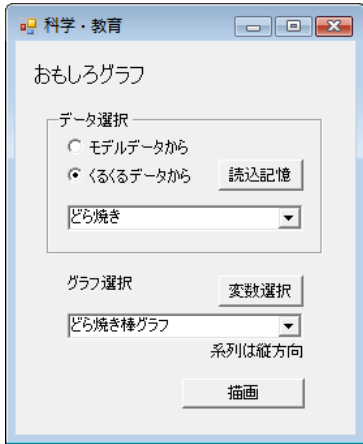


図 2.24 おもしろグラフメニュー

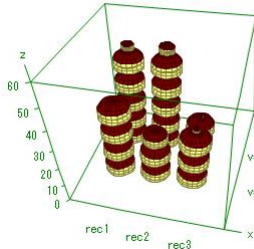


図 2.25 どら焼き棒グラフ

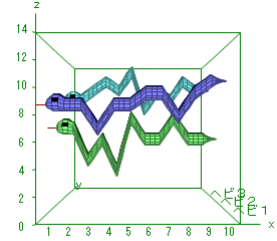


図 2.26a ヘビ線グラフ 1

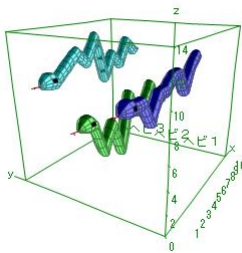


図 2.26b ヘビ線グラフ 2

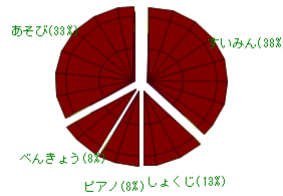


図 2.27a アップルパイグラフ 1

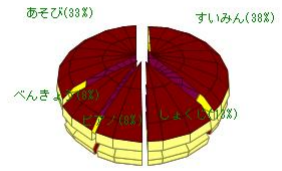


図 2.27b アップルパイグラフ 2

### 6.3 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 3.1 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 3.1 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（ヘビ線）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 3.1 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 datamodel

	<pre> 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 </pre>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

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。

2節で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多いので、このあたりで打ち止め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もできるようになる。

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、メビュー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。