

社会システム分析のための統合化プログラム20

－ システムダイナミクス・定積分・常微分方程式 －

福井正康・*石丸敬二・尾崎誠

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。今回は、時間の経過に従って社会システムの変化をシミュレートするシステムダイナミクスと、今後の物理学への応用の準備として、定積分と常微分方程式について基礎的なプログラムを作成した。

キーワード

社会システム分析, OR, 数学, システムダイナミクス, 定積分, 常微分方程式

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

1. はじめに

これまで、我々は社会システム分析ソフトウェア **College Analysis** において、統計学、数学、経営科学、意思決定手法などを中心にプログラムを作成してきたが、今回は、物の流れと相互作用によって様々なシステムの時間的変化をシミュレートするシステムダイナミクスと呼ばれる手法について新しくプログラムを作成した。

システムダイナミクスは 1950 年代から発展してきたシミュレーション手法で、有益性は広く認識されている¹⁾。ソフトウェアも **STELLA/iThink** (以後 **STELLA** と呼ぶ) を代表として、多く提供されており、社会システム分析には欠かせない 1 分野を形成している。元々この手法は、シミュレーション記述言語によって分析を行っていたが、コンピュータの発達によって、グラフィックによる記述が一般的になり、見た目にも分かり易い表現で分析が可能となった。我々は、このグラフィック表現をグラフィックエディタによって描画することを考え²⁾、**College Analysis** への導入を検討した。導入に当っては参考文献 1) の **STELLA** の画面を参考にしたが、これまでのグラフィックエディタの利用法との相違がないようにした。また、外界を表すクラウドについては、開発当初は導入していたが、グラフィック画面の簡単化のために省略している。

グラフィックエディタにおける画面の構成要素は、ボックスとラインである。そのため、フローとバルブのように一体化した考え方はない。また、ボックス間を結ぶラインは、方向が同じものは 1 本だけしか定義されない。この点が **STELLA** との最も大きな違いであるが、これはラインの間に節点と呼ばれるボックスを 1 つ挟むことで解決した。

システムダイナミクスを取り扱う際の、グラフィックエディタでのボックスは、ストック、バルブ、コンバータであり、ラインは、フロー (直線矢印) とコントロール (左右円弧矢印) である。ボックスにおけるデータは、数値や数式の形で、ボックスの **value** の中に保存される。数式の表示では、通常の数学関数の他に、ファイルデータや n 期前のデータなどの参照が可能である。

さて、我々は、従来の分析に加えて、3D ビューアやピクチャビューアが強力な表現手段となるような、物理教育で利用できるシミュレーションプログラムを作成して行くことを決めた。そのための準備として、ここでは数学的基礎となる、定積分と常微分方程式の解法³⁾ のプログラムを作り、今後活用できるようにしておきたい。定積分は、1 次元と 2 次元について、常微分方程式は、一般的な 2 階常微分方程式の初期値問題と 2 階線形常微分方程式の境界値問題を扱う。方程式の入力は数式をそのまま入力できるようにし、結果の表示は 3D ビューアやピクチャビューアを活用して、様々な視点から眺められるようにした。物理シミュレーションそのものについては、この論文のシリーズとは方向性が異なるので、別に新しいシリーズを作って紹介して行く予定である^{4,5)}。

2. システムダイナミクス

メニュー [分析-OR-システムダイナミクス] を選択すると、図 2.1 のようなシステムダイナミクスの実行メニューが表示される。

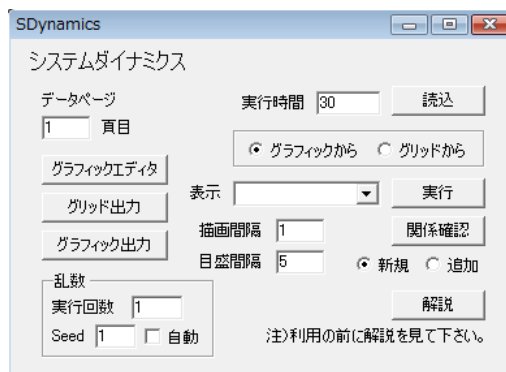


図 2.1 システムダイナミクス実行メニュー

最初にこのメニューで「グラフィックエディタ」ボタンをクリックし、グラフィックエディタの機能を活用して²⁾、図 2.2 のように入力する。

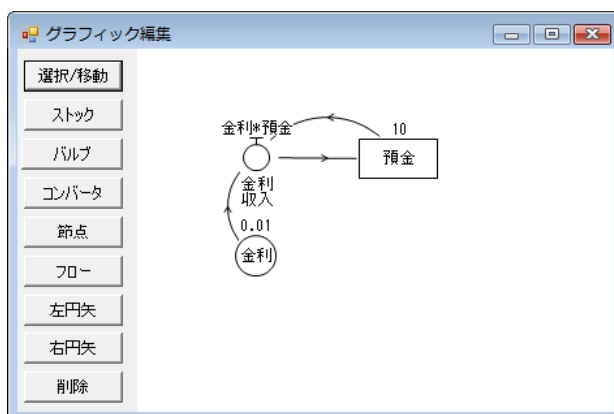


図 2.2 グラフィックエディタ画面

描かれた図は金利収入による預金の増加モデルである¹⁾。預金は「ストック」と呼ばれる貯蔵庫、金利は「コンバータ」と呼ばれる影響を表す記号、金利収入は「バルブ」と呼ばれる制御機能を表す記号で表示されている。金利収入から預金への直線は「フロー」と呼ばれ、ものの流れ（この場合は金銭の流れ）を表している。バルブはこのフローを制御している。金利から金利収入、または預金から金利収入への円弧は「コントロール」と呼ばれ、コンバータやストックから金利収入への影響を表している。図の中では、金利から金利収入への円弧は「左円矢」、預金から金利収入へは「右円矢」を利用している。二つの円弧は進行方向左側の円弧か、右側の円弧かの違いだけで同じものであり、見た目にバランスの取れた方を利用すればよい。

ボックス名の入力にはボックスをダブルクリックし、数値や数式を表す value の入力には右クリックで表示されるテキストボックスに入力する。図 2.2 では、金利の value は「0.01」、金利収入の value は「金利*預金」、預金の value は初期値の 10（万円）に設定している。

グラフィックエディタメニュー [表示-ボックス値表示[ON/OFF]] で表示のモードを切り替えると、図 2.3 のようにボックスの上の value 値を非表示にすることもできる。通常、システムダイナミクスでは ON の設定になっている。

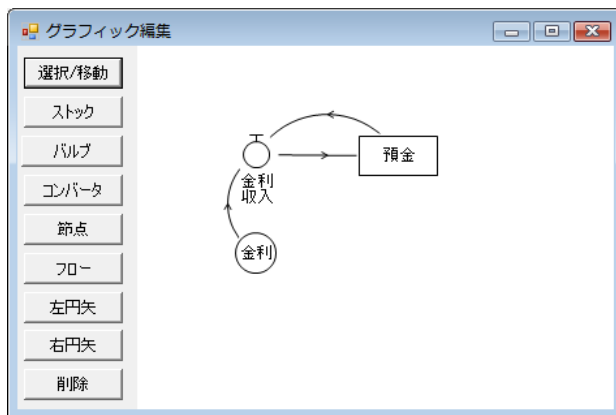


図 2.3 ボックス値非表示モード

描画されたデータは、「グリッド出力」ボタンをクリックしてグリッドエディタに保存することができる。分析の実行は、「グラフィックから」、「グリッドから」ラジオボタンで、「グラフィックデータを直接利用するか、保存したグリッドデータを利用するか」を選択する。通常は「グラフィックから」にしておき、必要に応じてグリッドエディタに保存する。新規の場合やデータの構造を変えた場合は、「読込」ボタンでプログラム中にデータを読み込み、「表示」コンボボックスで、出力するデータを選択し、「実行」ボタンをクリックする。データの構造を変えず、value のみ変更した場合は、そのまま「実行」ボタンをクリックしてもよい。「表示」コンボボックスで預金を選んだ実行結果を図 2.4 に示す。

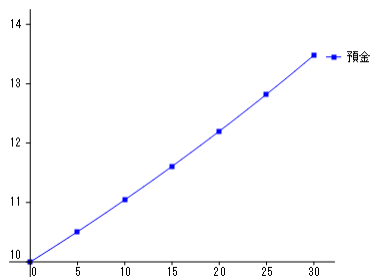


図 2.4 預金表示結果

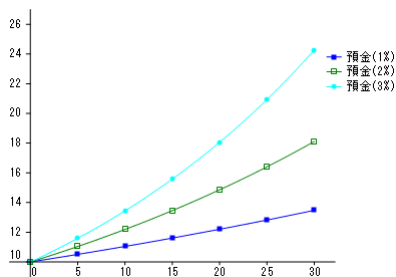


図 2.5 金利を変化させた試行

数式や数値の評価で、コンバータとバルブは実行時点で計算を行うが、ストックに関しては、1期前のデータを参照している。これは参照がループに陥るのを防ぐためである。

金利を、0.01, 0.02, 0.03 と変えた場合の比較は、value を変更しながら、メニュー下の「追加」ラジオボタンを選択してグラフを描画する。その結果を図 2.5 に示す。ここで凡例は変更してある。

「関係確認」ボタンをクリックすると、図 2.6 のようなボックス間の関係がまとめて表示される。資料などに使用する際に利用する。

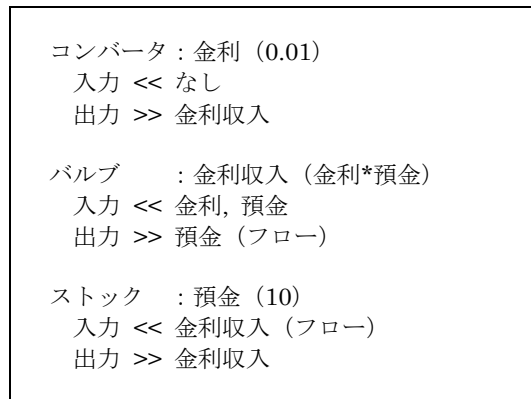


図 2.6 ボックス間の関係

次はアルコールの吸収・排出問題の例である¹⁾。図 2.7a にグラフィックエディタの画面、図 2.7b にその画面をコピーしたものを示す。

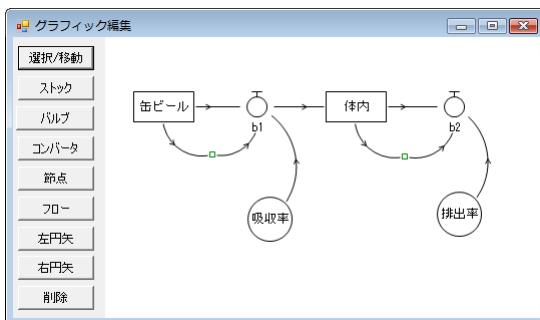


図 2.7a グラフィックエディタ画面

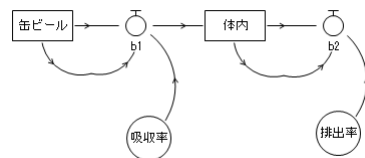


図 2.7b グラフィックデータのコピー

これは缶ビールからバルブ b1 に、体内からバルブ b2 に、フローとコントロールの 2 本のラインが入る例であるが、グラフィックエディタの仕様から、同じボックス間に同方向の複数のラインを引くことができず、間にグラフィックエディタの左側にボタンのある「節点」を入れている。節点はコピー画像では消えている。しかし、この節点が気になる場合は、図 2.8a のようにボックスの内部に節点を持って行く。コピーすると図 2.8b のようにあたかも節点がなかったかのように見える。

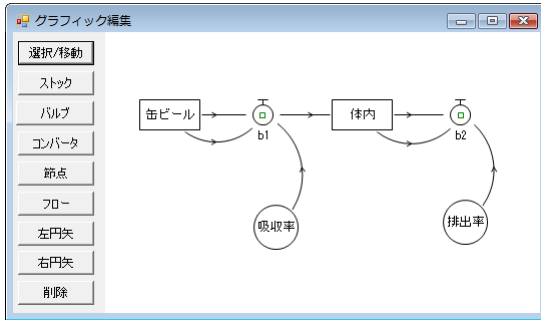


図 2.8a グラフィックエディタ画面

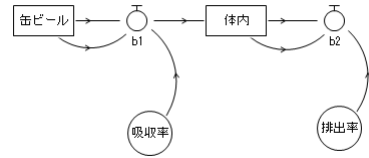


図 2.8b グラフィックデータのコピー

図 2.9 に value 値を表示したグラフィックデータ、図 2.10 に「実行時間」を 100、「目盛間隔」を 10 とした実行結果を示す（見易いように、排出率の値は参考文献と変えてある）。節点があっても、それについての設定は不要である。

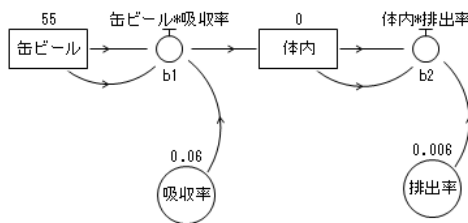


図 2.9 value 値付きグラフィックデータ

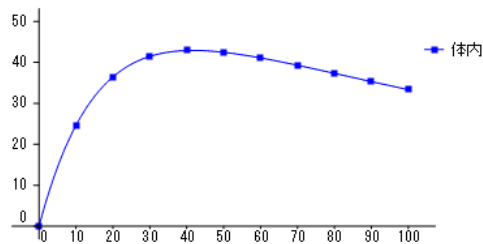


図 2.10 実行結果

次にファイルのデータを利用する在庫管理問題の例を考える⁶⁾。この問題はシステムダイナミクスで考えられる、乱数やファイル入力、時間の遅延の問題を扱うには最適である。図 2.11 がそのモデル画面である。ファイルのデータは出庫データで、このモデルのデータが入っているファイルの 1 ページ目に図 2.12 のように含まれている。

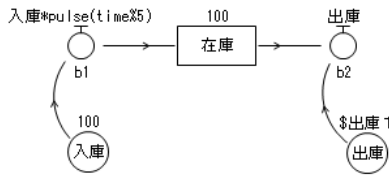


図 2.11 ファイルデータを利用した在庫の変化

	出庫1	出庫2	出庫3
1	18	34	10
2	17	25	13
3	14	30	14
4	17	24	12
5	19	41	16
6	21	20	4

図 2.12 出庫データ

入庫量は常に 100、入庫は 5 日ごとにする。パルプ b1 の式「 $\text{入庫} * \text{pulse}(\text{time} \% 5)$ 」の time 変数は経過時間で 1 から始まる。「 $\text{time} \% 5$ 」は time を 5 で割った余りを表す。pulse(x) 関数は、 $x=0$ のとき 1、その他のとき 0 の値を持つ。これを文章で表すと、「時間を 5 で割った余りが 0 のときだけ、入庫(100)で表わされる値を在庫ストックに送り込む」ということになる。出庫コンバータの式「 $\$ \text{出庫 1}$ 」で、変数名の先頭に付いた \$ マークは、実行メニューの「データページ」テキストボックスで与えたページにあるデータを参照することを意味する。変数名は「出庫 1」であり、経過時間 t に t 行目のデータを参照する。在庫量のシミュレーション結果を図 2.13 に示す。

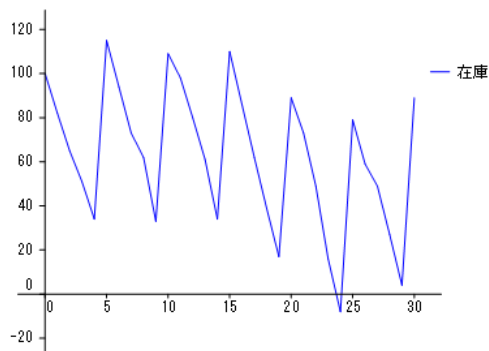


図 2.13 在庫量シミュレーション結果

グラフ画面のメニュー [設定-データポイント[ON/OFF]] で、データ点は消してある。

次は平均 20、標準偏差 5 の正規分布に従う出庫の場合の、定期発注方式のシミュレーションである。図 2.14 にモデルを示す。

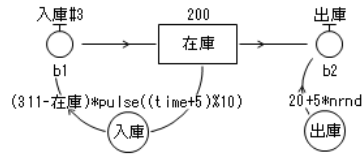


図 2.14 乱数出庫の定期発注方式

出庫は「 $20+5*nrnd$ 」で表されるが、「nrnd」は標準正規乱数であるので、出庫は平均 20、標準偏差 5 の正規乱数データとなる。入庫は 10 日ごとに発生するように pulse() 関数を用いて設定されているが、その大きさは、最大在庫 311 から現在の在庫の量を引いたものである。但し、在庫納入はバルブ b1 によって、入庫発生から 3 日後となる。b1 の式「入庫#3」は 3 日前の入庫データを参照することを意味する。ストック（在庫）はその 1 日前の値を参照するので、この式の在庫は 4 日前の値である。そのため、入庫のリードタイムは 4 日と解釈される。在庫量のシミュレーション結果を図 2.15 に示す。

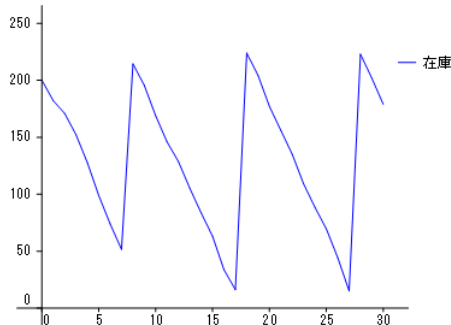


図 2.15 在庫量シミュレーション結果

乱数は、図 2.1 のメニューにあるように、Seed を決めて、同じ乱数系列を発生させることも、現在時刻から自動的に Seed を決めて毎回違う乱数系列にすることもできる。メニューの実行回数は、今後拡張するモンテカルロシミュレーション用に作った部分であるが、現在はまだ使用していない。

定量発注方式の表現は、定期発注方式より少し複雑である。在庫が発注点在庫量 (96) より少なくなった時点で発注するので、図 2.16 の入庫コンバータのような複雑な式になる。theta(x) 関数は $x \geq 0$ のとき 1、 $x < 0$ のとき 0 となる関数である。pulse() 関数と合わせて使うと、条件文などが表現できるようになる。この式を文で表現すると「在庫量が 96 より少なく、かつ 1 期前の在庫量が 96 より多い時に (初めて在庫量が 96 を下回った時に)、200 単位を発注する」となる。図 2.17 に定量発

注方式の在庫量シミュレーション結果を示す。

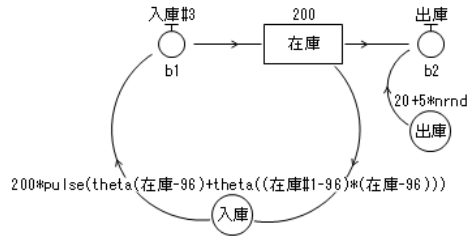


図 2.16 乱数出庫の定量発注方式

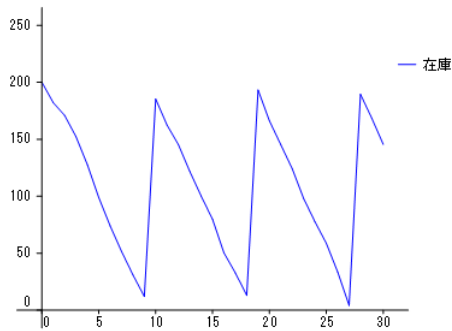


図 2.17 在庫量シミュレーション結果

これまではストックに上限や下限を設けていない例を扱ったが、ストックのデータの書式を変えることによって、これを設定できるようになる。例えば、ストックの値として、100;0;200 のように、セミコロンで区切ると、初期値 100、下限 0、上限 200 となる。また、100;0; や 100;;200 のように、上限や下限を略すとその部分の制約は付かなくなる。この制約は現在のバージョンでは、単独のバルブを介する場合のストック間に有効で、1つのストックに複数のバルブが繋がった場合の制御はまだ完全ではない。図 2.18 と図 2.19 に最も簡単な制約付きストックの例を示す。

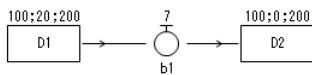


図 2.18 制約付きストック

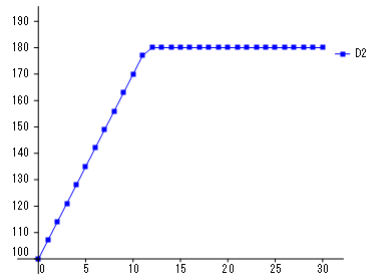


図 2.19 制約付きストックシミュレーション結果

3. 定積分

定積分の問題では、不定積分が初等関数で与えられる場合や性質がよく調べられた非積分関数の場合を除き、数値積分が用いられる。数値積分は積分領域を細く区切ってそれぞれの分割領域の近似的な面積の和として結果を与える。分割領域の近似の方法は、それぞれの領域の縁を結んだ台形で近似する方法や2つの分割域の3つの縁の高さを2次曲線で近似する方法がある。前者を台形則、後者をシンプソン則と呼ぶ³⁾。台形則は区間分割数の2乗、シンプソン則は区間分割数の4乗に逆比例して精度が向上することが知られている（実際は計算機の演算誤差の問題もあり厳密ではない）。このプログラムでは、1変量の積分ではシンプソン則、2変量の積分では台形則を利用している。そのため、2変量の場合は分割数を増やして計算することが望ましい。

メニュー [分析-数学-定積分] を選択すると図 3.1 のような実行メニューが表示される。

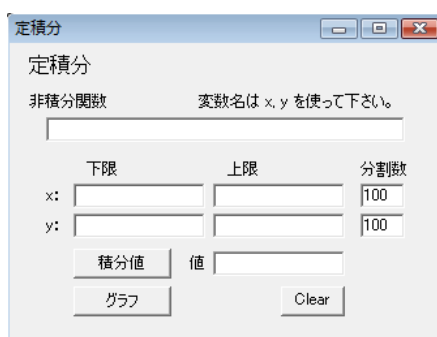


図 3.1 定積分実行メニュー

1 変量関数の場合は、非積分関数 $f(x)$ の式を「非積分関数」テキストボックスに入力し、 x 座標の積分の下限と上限を記入する。ここで、 y 座標の領域も書くと2重積分となり、意味が異なってくるので注意が必要である。例えば、 $f(x) = \sin x$ とし、領域を $0 \leq x \leq \pi$ (π は pi と入力) として、「積分値」ボタンをクリックすると図 3.2 のように「値」テキストボックスに結果が出力される。

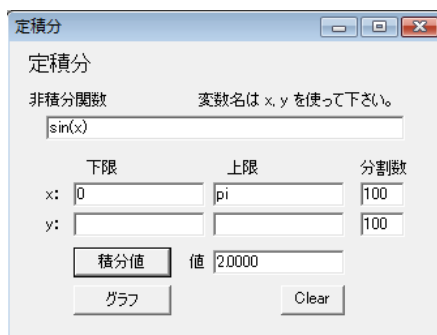


図 3.2 設定と積分結果

ここで、上限と下限には $\pi/2$ のような数式も入力可能である。

定積分の領域をグラフで見たい場合、「グラフ」ボタンをクリックすると、領域と数式が入力された図 3.3 のような 1 変量関数描画メニューが表示され、そのまま「グラフ描画」ボタンをクリックすると、図 3.4 のようなグラフが表示される。

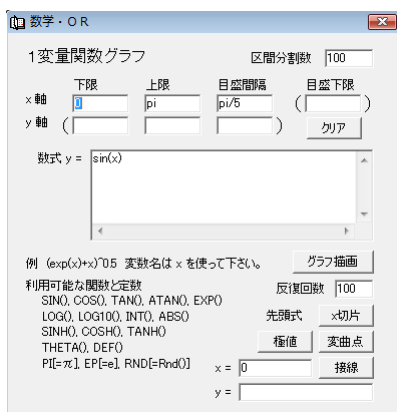


図 3.3 1 変量関数描画メニュー

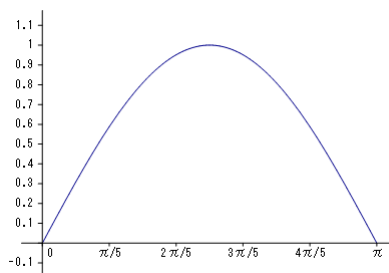


図 3.4 1 変量関数グラフ出力結果

ここでは、上限と下限に式「pi」を使ったので、x 軸の値は π 表示になっているが、一般には数値表示である。

次に 2 変数の重積分の場合を考える。非積分関数として、 $f(x,y) = \sin(x) \cos(y)$ とし、積分領域を、 $0 \leq x \leq \pi/2$ 、 $0 \leq y \leq \pi/2$ とすると、計算結果は図 3.5 のようになる。

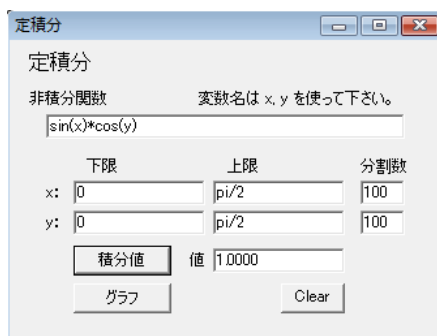


図 3.5 設定と重積分結果

「グラフ」ボタンをクリックすると、重積分の場合は、2 変量関数グラフのメニューが図 3.6 のように表示され、そのまま「グラフ描画」ボタンをクリックすると、図 3.7 のようなグラフが表示される。

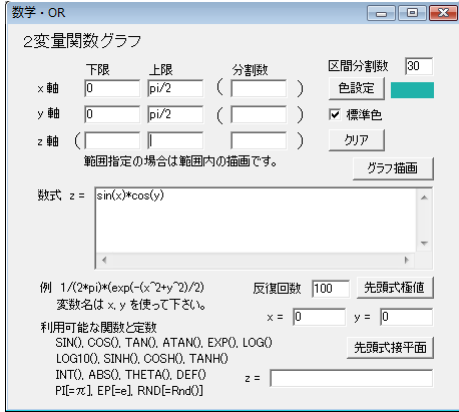


図 3.6 2変量関数描画メニュー

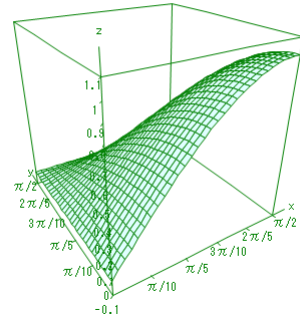


図 3.7 2変量関数グラフ出力結果

ここで、定積分メニューの分割数と2変量関数描画メニューの区間分割数とは、前者が積分の際の分割数で、後者が単なる表示用の分割数であることから、言うまでもなく無関係である。

このプログラムでは領域を数式で表わすことも可能である。例えば上のグラフで、積分領域を $0 \leq x \leq \pi/2$, $0 \leq y \leq x$ とすると、積分結果は図 3.8 のようになる。

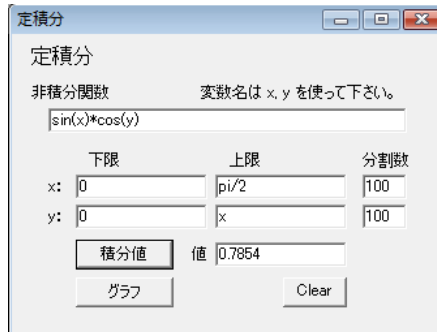


図 3.8 積分領域が関数の場合の結果

このグラフの描画には、図 3.9 のように元の関数に、積分領域でだけ 0 と定義される def() 関数が追加され、描画結果は図 3.10 のようになる。ここで、描画には直接関係がないが、図 3.9 のように y 軸の積分領域は数値で表示されるので、 π で表示するには下限を 0、上限を pi/2 と変更する必要がある。ここで、上に述べた def(x) 関数は、x が 0 以上の時だけ 0 の値を取り、それ以外は未定義となる関数である。数式の中に未定義の量が含まれる場合、グラフはその部分を表示しない。

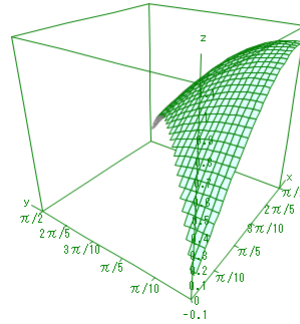
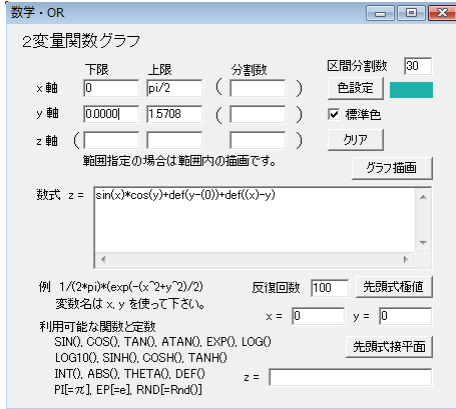


図 3.9 積分領域が関数のグラフ表示メニュー 図 3.10 積分領域が関数の場合のグラフ

4. 常微分方程式

常微分方程式は、時間変化など、1つの変数による関数の変化の様子を式で表わしたもので、自然科学や工学などで広く利用される。特に重要な2階常微分方程式には、初期値問題と呼ばれる、ある点での関数と関数の微分の値を与えて解を求めるものと、境界値問題と呼ばれる、2つの点での関数の値を与えて解を求めるものがある。我々は2階（または1階）の一般的な常微分方程式の初期値問題と2階の線形常微分方程式の境界値問題を扱うプログラムを参考文献3)に習って作成した。これらの解法のアigorithmは多くの応用分野で利用可能であり、今後 College Analysis に物理シミュレーションを加えて行く際には必要不可欠である。

一般に、2階の常微分方程式は以下の形に書かれるが、

$$\frac{d^2y}{dx^2} = f\left(\frac{dy}{dx}, y, x\right)$$

初期値問題の場合は、初心者への入力も考えて、より一般に以下の形で入力するようにした。

$$f\left(\frac{d^2y}{dx^2}, \frac{dy}{dx}, y, x\right) = 0$$

1階常微分方程式についても同様である。また、連立常微分方程式も同様の形式で、式を改行して並べて入力する。解は Runge-Kutta 法³⁾により求める。

境界値問題の場合は、単独の常微分方程式のみで、以下の線形の形に限定している。

$$\frac{d^2y}{dx^2} + p(x)\frac{dy}{dx} + q(x)y + r(x) = 0$$

解は3重対角行列のLU分解による方法³⁾で求める。

メニュー [分析-数学-常微分方程式] を選択すると、図 4.1 の実行メニューが表示される。

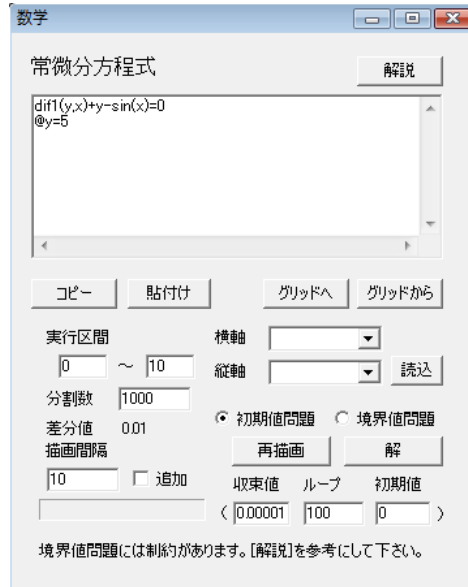


図 4.1 常微分方程式実行メニュー

微分の表記法や入力例などは、「解説」ボタンをクリックすると、図 4.2 のように示される。

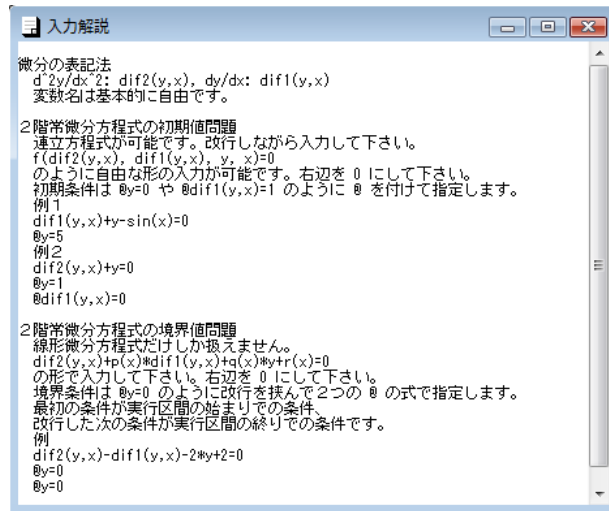


図 4.2 解説画面

数式の中で、微分 dy/dx は $\text{dif1}(y,x)$ 、 d^2y/dx^2 は $\text{dif2}(y,x)$ で表わす。初期値問題の初期値は、 $@y=5$ や $@\text{dif1}(y,x)=0$ のように表すが、これらは実行メニューの「実行区間」の開始点の値として与えられる。また、境界値問題の境界値も図 4.2 の下のように、改行しながら、 $@y=0$ や $@y=0$ のように表すが、これはそれぞれ実行区間の開始点と終了点に相当する。初期値問題と境界値問題の切り替えは、実行メニューの「初期値問題」、「境界値問題」ラジオボタンで行う。以後、サンプルとし

て表示されている以下の微分方程式（初期値問題）を用いて実行メニューの各機能を紹介する。

$$\begin{aligned} \text{dif1}(y, x) + y - \sin(x) &= 0 \\ @y &= 5 \end{aligned}$$

最初にデフォルトの設定で、「読込」ボタンで横軸と縦軸の候補を読み込み、そのまま横軸 X、縦軸 Y として、「解」ボタンをクリックし、微分方程式の解を求める。図 4.3 にその結果を示す。

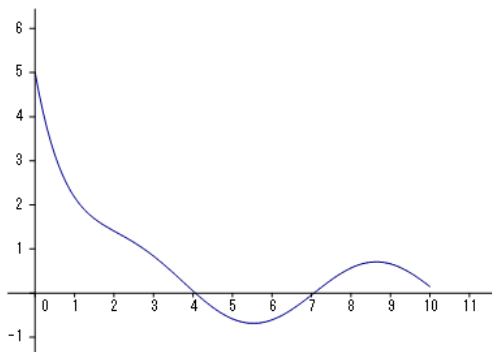


図 4.3 X-Y グラフ

数値的な結果が知りたいときは、グラフメニュー [編集—データ表示] を利用する。

解を求める範囲を変更する場合は、「実行区間」に範囲を入力し、計算の精度を高めたい場合は、「分割数」を多くする。デフォルトでは、初期値問題で「分割数」1000，境界値問題で 100 になっている。「差分値」は実行区間と分割数から、(終了点－開始点) ÷ 分割数で計算され、数値微分の差分値となる。

「描画間隔」は計算した値のうち、何個飛ばしで、グラフに表示させるかを定めるもので、10 のときは分割点が 0,10,20, … のときに表示し、1 のときは計算した点すべてを表示する。初期値問題ではデフォルトで 10、境界値問題では 1 になっている。図 4.4 に「描画間隔」を 100 にして解を求めた結果を示す。実行区間が 10、分割数が 1000 なので、表示は間隔 1 になる。

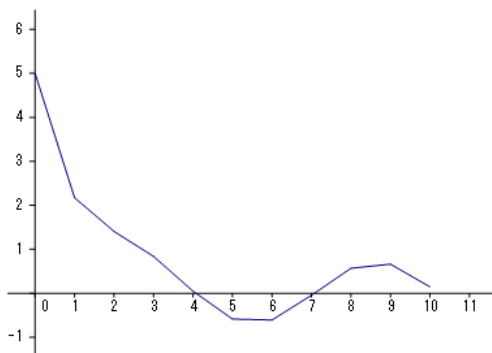


図 4.4 描画間隔を変えた表示

描画間隔を大きくするとグラフは粗くなるが、計算精度には影響がない。

次に、以下のような連立1階微分方程式の例を見る。

```
dif1(y,x)-z=0
dif1(z,x)+y=0
@y=1
@z=0
```

これは、連立1階微分方程式の形をしているが、以下の単振動の方程式と同じである。

$$\frac{d^2y}{dx^2} + y = 0, \quad y|_{x=0} = 1, \quad \left. \frac{dy}{dx} \right|_{x=0} = 0$$

この方程式の場合、関数が2つあるので、軸設定でどちらの関数を表示させるか選ぶことができる。

横軸に X 、縦軸に Y を選択し、「解」ボタンをクリックすると、図 4.5a の結果を得る。

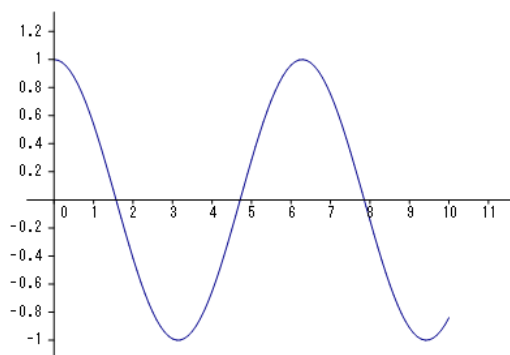


図 4.5a X-Y グラフ

軸設定で、横軸に X 、縦軸に Z を選択した場合の結果を図 4.5b に、横軸に Z、縦軸に Y を選択した結果を図 4.5c に示す。

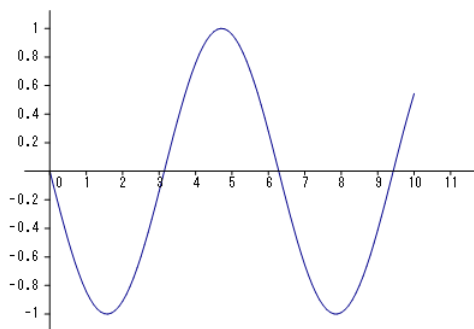


図 4.5b X-Z グラフ

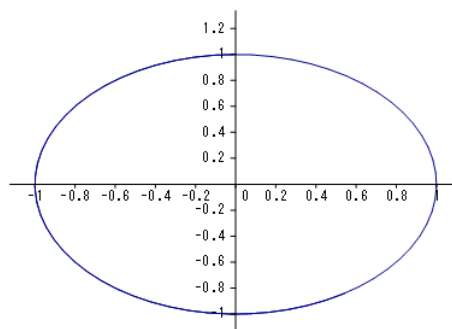


図 4.5c Y-Z グラフ

この連立1階微分方程式と同等な2階微分方程式は、以下のように入力される。

```
dif2(y,x)+y=0
@y=1
```



```
@dif1(y, x)=0
```

これは解を求める初期段階で、連立1階微分方程式に書き直して計算される。この場合も上と同様、軸設定に X, Y, dY/dX（軸選択コンボボックスでは DY_DX で表示される）が選択できる。軸を変えて結果を出力する場合、長く時間のかかる計算など、「再描画」ボタンをクリックすることで、計算を省略することができる。

最後に原点からの太陽の重力による惑星の運動モデルを示しておく。但し、万有引力定数×太陽質量は1に設定している。入力例は以下である。

```
dif2(x, t)+x/(x^2+y^2)^1.5=0  
dif2(y, t)+y/(x^2+y^2)^1.5=0  
@x=1  
@y=0  
@dif1(x, t)=0.5  
@dif1(y, t)=0.5
```

この問題は精度が必要なので、「分割数」を 10000 に設定しておく。

横軸に T、縦軸に X を選択し、解を求めると図 4.6a、横軸に T、縦軸に Y を選択すると図 4.6b のようになる。

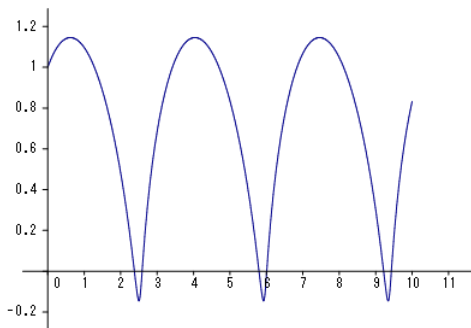


図 4.6a T-X グラフ

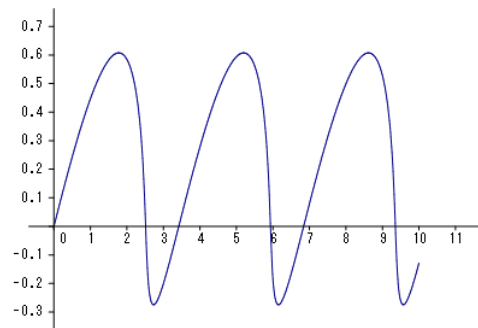


図 4.6b T-Y グラフ

また、軌道を見るために、横軸に X、縦軸に Y を選択すると、図 4.6c のような楕円軌道になる。

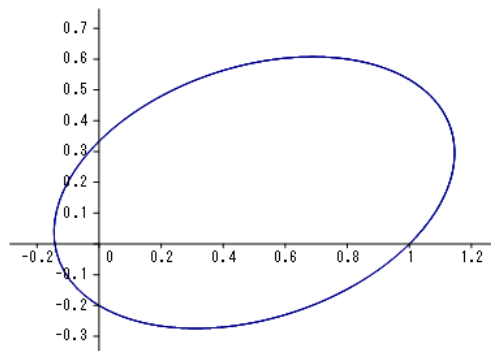


図 4.6c X-Y グラフ

最後に常微分方程式とは直接関係ないが、ケプラーの第3法則（面積速度一定）を見ておくことにする。まず、横軸に X、縦軸に DX_DT を選択して図 4.7a を描き、次に横軸に Y、縦軸に DY_DT を選択して図 4.7b を描く。

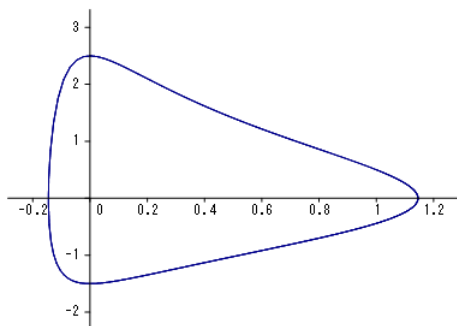


図 4.7a X-DX_DT グラフ

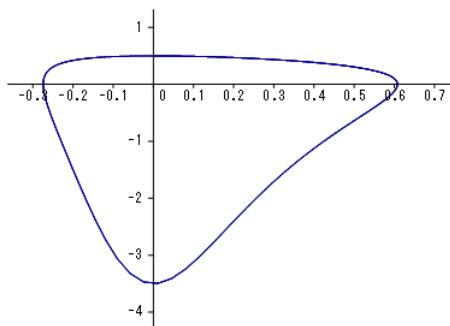


図 4.7b Y-DY_DT グラフ

これらのグラフメニュー [編集→データ出力] で、各時点のデータを出力し、Excel にコピーして（ここでは、時間の部分もコピーしておいた）、面積速度 $(xv_y - yv_x)/2$ を計算する。その結果を図 4.8 に示す。結果は右端の列のように 0.25000 ± 0.00001 になり、面積速度は一定であることが示された。

	F2		fx					=(B2*E2-D2*C2)/2	
	A	B	C	D	E	F			
1	T	X	DX/DT	Y	DY/DT		$(x*v_y - y*v_x)/2$		
2	0	1	0.5	0	0.5	0.25000			
3	0.01	1.00495	0.49005	0.005	0.499975	0.25000			
4	0.02	1.009801	0.480196	0.009999	0.499902	0.25000			
5	0.03	1.014554	0.470438	0.014998	0.499782	0.25000			
6	0.04	1.01921	0.460772	0.019995	0.499615	0.25000			
7	0.05	1.02377	0.451195	0.02499	0.499404	0.25000			
8	0.06	1.028235	0.441706	0.029983	0.49915	0.25000			
9	0.07	1.032605	0.432302	0.034973	0.498854	0.25000			
10	0.08	1.036881	0.42298	0.03996	0.498516	0.25000			
11	0.09	1.041064	0.41374	0.044943	0.498139	0.25000			
12	0.1	1.045156	0.404578	0.049922	0.497722	0.25000			
13	0.11	1.049156	0.395493	0.054897	0.497268	0.25000			
14	0.12	1.053066	0.386482	0.059868	0.496776	0.25000			
15	0.13	1.056886	0.377544	0.064833	0.496248	0.25000			

図 4.8 面積速度一定の検証

このプログラムでは、微分方程式の書き方を、右辺が 0 の形式以外は限定せずに計算しているため、計算時間が長くなっているが、予め書式に制約を付けることによって高速化することは可能である。

5. おわりに

システムダイナミクスについては、基本的枠組みを作成したが、使い易さにはまだ難がある。実際、少人数の授業で試したが、利用者の数式の入力に間違いが見られる。数式の間違いで多いのは、半角と全角の間違い、左括弧と右括弧の不一致、ボックス名の間違いなどである。前者 2 つについては、

プログラムの中で確認するようになっているが、後者については、「関係確認」ボタン以外に間違いを見付ける機能はない。変数名を選択式にすることが最良と思われるが、これは今後の課題である。システムダイナミクスのプログラムの機能を向上させるには、できるだけ多くのモデルを作成して、経験から学ぶ以外にないようと思われる。

定積分は、これまで我々が作ってきた社会システム分析のプログラムの中に出て来なかったが、統計学における密度関数からの平均や分散の計算や今後取り組む物理シミュレーションには必要不可欠である。また常微分方程式の解法は、運動方程式を取り扱うプログラム^{4),5)}の必須知識であり、2階常微分方程式を詳しく見ておくことは特に重要である。運動方程式は形が決まっているので、ここで行ったような文字列の数式を処理する必要はない。そのため処理速度は向上し、例えば10個程度の質点の運動なども取り扱うことができる。この論文で説明した定積分と常微分方程式のプログラムはあまり実用的とは言えないが、シミュレーションプログラムを学ぶ際には避けて通ることができないものである。

参考文献

- 1) シミュレーションによるシステムダイナミクス入門, 土金達男, 東京電機大学出版局, 2005.
- 2) 数理系教育ソフトウェアにおける汎用グラフィックインターフェースの提案, 福井正康, 石丸敬二, 尾崎誠, 日本教育情報学会誌「教育情報研究」, 第27巻, 第4号, (2012) 55-66.
- 3) 理工系の基礎数学8 数値計算, 高橋大輔, 岩波書店, 1996.
- 4) College Analysis による物理シミュレーション1 - 質点系の運動・惑星シミュレーション -, 福井正康・石丸敬二, 福山平成大学経営研究, 8号, (2012).
- 5) College Analysis による物理シミュレーション2 - 電荷と電場・電流と磁場 -, 福井正康, 福山平成大学経営研究, 8号, (2012).
- 6) 社会システム分析のための統合化プログラム15 - 品質管理・在庫管理シミュレーター -, 福井正康, 付鴻鵬, 魏巍, 奥田由紀恵, 細川光浩, 福山平成大学経営研究, 7号, (2011) 129-150.

Multi-purpose Program for Social System Analysis 20

- System Dynamics, Definite Integral, Ordinary Differential Equation -

Masayasu FUKUI, *Keiji ISHIMARU and Makoto OZAKI

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. This time, we created a program on system dynamics which simulates the changes in the social system by the time series. In preparation for physics simulation, we also made basic programs on the definite integral and ordinary differential equation.

Keywords

College Analysis, social system analysis, system dynamics, mathematics, definite integral, ordinary differential equation.

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