

教育用連続型シミュレーション・システム

BASIC で DYNAMO 的記述を

林 亜夫

1. はじめに

本システムは連続型シミュレーションの教育、あるいは簡易な動的システムの実験を、DYNAMO 言語に準拠した記法を用いてマイクロ・コンピュータで行なうプログラム・パッケージである。

本システムは適切な指導のもとで、マイコン利用の未経験者にも利用できるものを意図して開発した。プログラム経験があれば教育向きの簡易なモデルを BASIC 言語で実験することはそれほど困難なことではないし、そのような試みも小玉[5]により為されている。しかし未経験者に対して限られた時間内に教育効果を上げるためには、ぜひとも簡易なモデル記述言語が必要である。そこで目をつけたのが DYNAMO 言語の構文¹⁾で、その多くが BASIC の代入文に似ており、簡単な規則のもとで読み替えが可能であることである。

システムの基本的着想はきわめて単純で、利用者は DYNAMO の言語に似せた BASIC の代入文のみでモデルを記述し、システムはそれを部分プログラムとして BASIC プログラムを構成して実行するだけである。それゆえ本システムはマイコンの BASIC モード上で稼働するが、このままでは各種の BASIC コマンドを状況に応じて投入しなければならないことになる。これでは初心者にとってモデルの記述はともかく、コマンドを使い

こなすに時間を要し非効率である。それゆえ簡単に操作できるオペレーティングシステムも不可欠になる。

ここで紹介するのは N88—BASIC 上で稼働するシステムで、初心者でも容易にシミュレーションが行なえるように、モデルの編集・実行・管理等を簡単に操作できる強力なオペレーティング・システムを組み込んだものである²⁾。

2. モデルの記述例

図1はデモ用として作成した、生物個体数の成長モデル³⁾のフロー・ダイアグラムであり、リスト1は DYNAMO、リスト2は本システム上でこれを記述したものである。両者同じ機能をもつ文が対応するように並べてある。両者が大きく異なる点は、文の種別識別の方法と変数名の時間添字の省略である。

DYNAMO ではレイト方程式、レベル方程式、補助方程式、テーブル文、初期値設定方程式等の区別を文頭の文字 R, L, A, T, N 等で行なっているが、本システムでは文番号で行なう。すなわち文番号8000番台はレベル方程式、7000番台はレイト、補助方程式、9000番台は初期値設定、定数方程式およびテーブル文として識別される。またシミュレーションのスペックを決める各種システム変数、そして出力変数とそのスケール処理を決めるプロット文は4000番台を用いる。

また DYNAMO 独特の時間添字 J, K, L は、

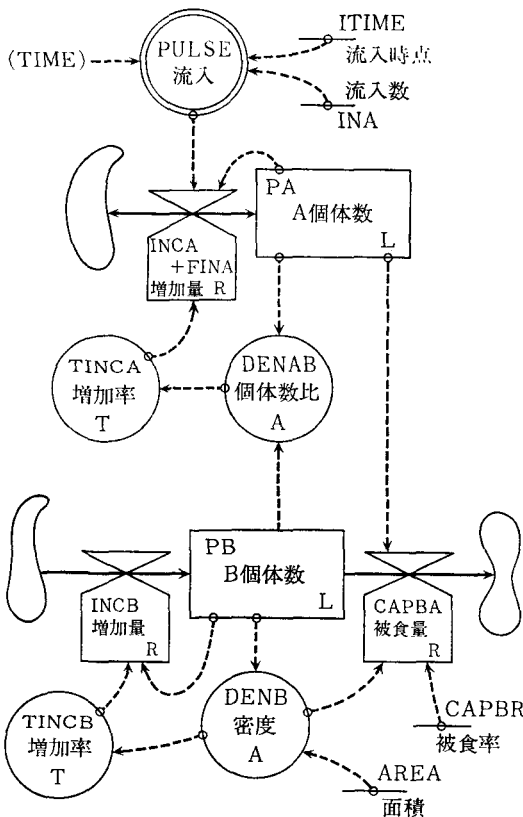


図 1 生物個体数の成長モデルのフロー・ダイアグラム

リスト 2 では一切書いてない。DYNAMO では方程式の種類ごと時間添字のつけかたが一意に決められているので、暗に時間添字があると想定す

```
C NTIME=-5
N TIME=NTIME
C DT=.1
C LENGTH=110
C PLTPER=2.5
PLOT PA=A,PB=B(0,10000)/INCA=R,FINA=I(0,*)
C PA0=0
N PA=PA0
C PB0=10000
N PB=PB0
C ITIME=0
C INA=100
C AREA=1000
C CAPR=.004
T TINCA=.1/.03/0/-.02/-.04
T TINCB=.07/.06/.04/0
A DENAB.K=PA.K/PB.K
R INCA.KL=PA.K*TABHL(TINCA,DENAB.K,0,2,5)
R FINA.KL=PULSE(INA,ITIME,500)
A DENB.K=PB.K/AREA
R INCB.KL=PB.K*TABHL(TINCB,DENB.K,0,10,2.5)
R CAPBA.KL=CAPR*DENB.K*PA.K
L PA.K=PA.J+(DT)*(FINA.JK+INCA.JK)
L PB.K=PB.J+(DT)*(INCB.JK-CAPBA.JK)
```

リストー1 DYNAMO による記述

れば DYNAMO 言語として容易に読みかえ可能である。

その他テーブル文やプロット文が文字代入文になっている点を除けば、ほぼ DYNAMO と同等の記述と言えよう。特にテーブル関数やステップ関数の引用も同等に扱えることに注目されたい。ただし、後に述べるように文の記述順序は本家の DYNAMO のように自由でない。

このように代入文しか用いないので BASIC プログラミングに精通していなくても、SD の基本を理解していれば容易にモデルの記述が可能とな

```
4000 REM -----SYSTEM/CONTROL VARIABLES.
4010 MODEL$ = "DEMO01: MODEL OF LOGISTIC CURVE"
4020 TIME = -5
4025 DT = .1
4030 LENGTH = 110
4040 PLTPER = 2.5
4050 PLOT$ = "PA=A,PB=B(0,10000)/INCA=R,FINA=I(0,*)"
6000 REM -----INITIAL VALUES.
6010 PA = 0
6020 PB = 10000
6030 ITIME = 0
6035 INA = 100
6040 AREA = 1000
6045 CAPR = .004
6050 TINCA$ = "0.10/0.03/0.00/-.02/-.04"
6060 TINCB$ = "0.07/0.06/0.04/0.00"
7000 REM -----RATE/AUXILIARY EQUATIONS.
7100 DENAB = PA/PB
7110 INCA = PA*FNTABL(TINCA$,DENAB,0!,2!,.5)
7120 FINA = FNPULS(INA,ITIME,500)
7200 DENB = PB/AREA
7210 INCB = PB*FNTABL(TINCB$,DENB,0!,10!,2.5)
7220 CAPBA = CAPR*DENB*PA
8000 REM -----LEVEL EQUATIONS.
8100 PA = PA + DT*(FINA+INCA)
8200 PB = PB + DT*(INCB-CAPBA)
```

リストー2 本システム上での記述

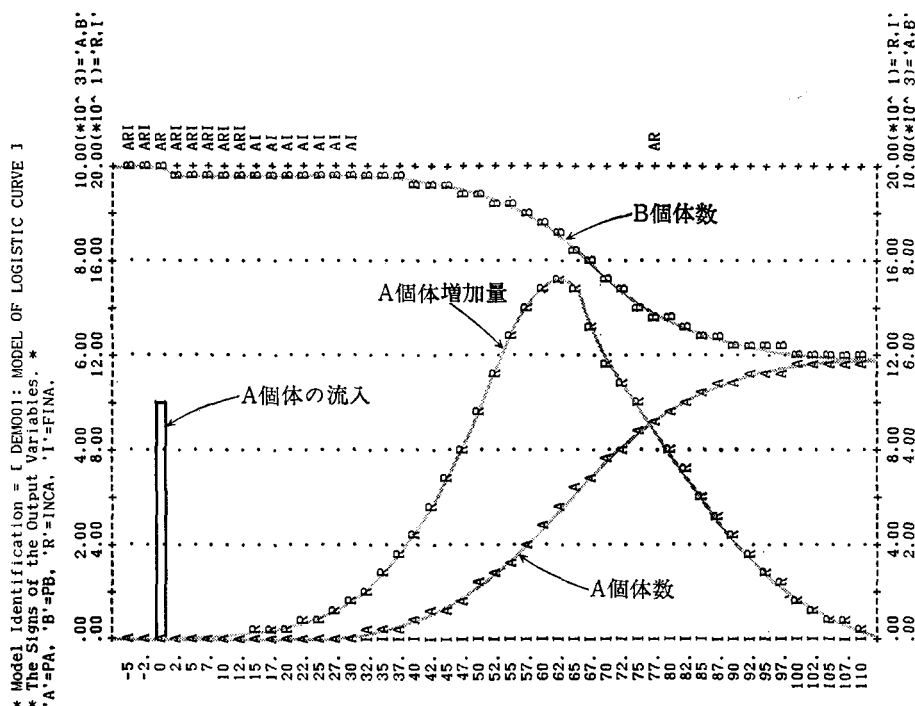


図2 図1のモデルの本システムによる出力 (見やすいように、計算機のアウトプットに線を補足した)

る。

生物個体数の成長モデルの本システムによる出力例を図2に示す。

3. システムの概要と操作法

本システムは大まかに4つの機能から構成されている。第1は利用者モデルの入力・編集・登録・構文チェック、そして実行管理を行なうオペレーティング・システム、第2は利用者モデルを駆動する実行システム、第3は結果の表示・出力を行なうシステム、第4は各種維持・管理を行なうシステムである。

これらはすべてBASIC言語で記述されており、20個のシステム・ファイルとしてフロッピーに格納されている。さらに同じフロッピーには利用者の作成したモデルおよび結果ファイルが出力されるほか、作業ファイルおよび利用者モデルのバックアップ・ファイルが一時的ファイルとして存在する。システム・ファイルはそれほど大きくない

ので、教育用に利用する場合は1枚のフロッピーで十分処理可能である。

図3は操作手順を示しており、円で囲まれた状態をモードと呼び、そこで利用者がファンクション・キーにより各種の処理を選択する。利用者のモデルは各モードでシステムと結合、分離、改編され処理されていく。

たとえば利用者が最初にモデルを構築するとき、メニューでファンクション・キー1を押し、編集モードに移行させる。このモードはBASICの編集モードに他ならず、文番号による行編集やスクリーン編集機能がそのまま使用できる。まったくの初心者はこちらでキー・ボードに慣れる必要があるが、カーソル移動による編集ができるため比較的容易に扱える。入力後構文エラーやその他のエラーをキー3を押すことでチェックする。エラーがなければキー4で実行させると、シミュレーション終了後実行モードから出力モードに移行する。出力モードでは結果をディスプレイ上の図

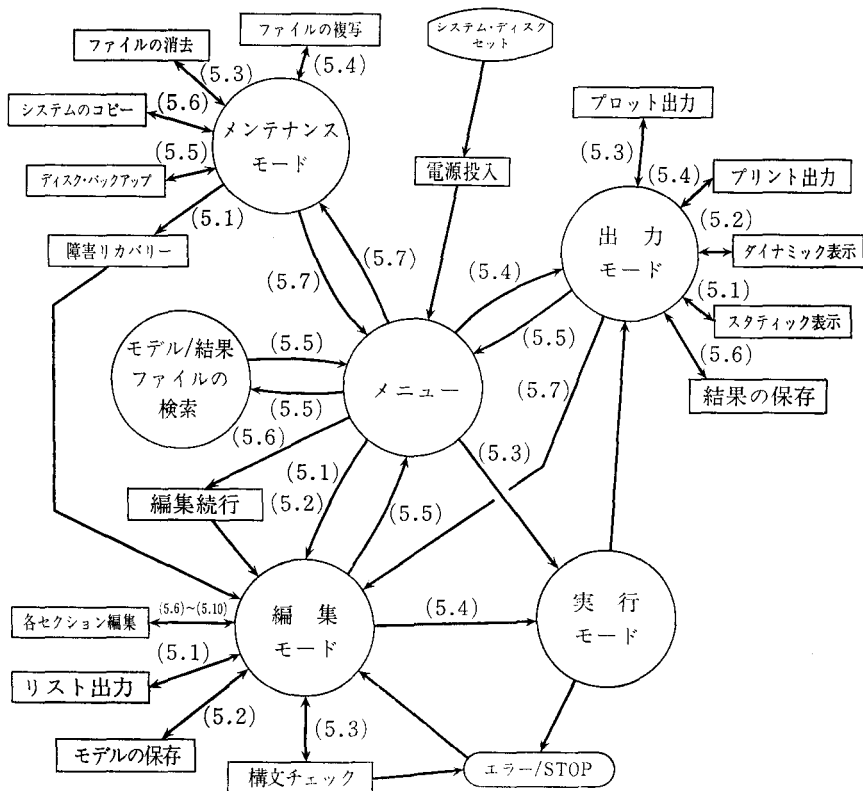


図 3 本システムの状態推移図、図中(f.1)～(f.10)はファンクション・キーを示す。

で見たり、プリントしたりして検討し、必要に応じて結果をファイルにセーブし、編集モードにもどってモデルを改訂していく。編集モードではモデルをセーブし、後の編集にそなえることができる。

図3に示した各処理は複数の BASIC コマンドにより制御される。たとえば利用者のモデルをファイルにセーブするときは、図4のような一連のプロセスが必要になる。このような複雑な操作はもちろん初心者には無理だし、そもそも実用的でない。このため本システムではこれら一連のコマンドをコマンド・プロシージャ化し、ファンクション・キーを押すことにより、一連の BASIC コマンドがあたかもキー・ボードから投入されたように次々実行されるようになっている。

1つのコマンドだけをファンクション・キーあるいは文番号をつけてプログラムに組み込むのでは、連続した処理ができないので、本システムで

はコマンド群をキー・バッファに直接送出するようにしている。CHAIN コマンドを用いる方法も考えられるが、処理効率が悪いので本システムではこのコマンドは一切採用してない。この方法によりかなり複雑な処理も短時間で処理でき、ファイル・スペースおよびファイル・アクセス時間の節約が可能となっている。

4. 文の記述方法の概要

4.1 変数名、算術式、式の記述順序

利用者の用いる変数名は BASIC で許される範囲で自由につけることができる。ただし Z で始まる変数名を用いることは許されない⁴⁾。

算術式は BASIC の算術式そのものであるから、BASIC の演算子と演算順序を理解すればよい。それゆえ DYNAMO の算術式のうち、() () による掛け算の記述は許されない。

式の記述順序が問題になるのは 7000 番台の補

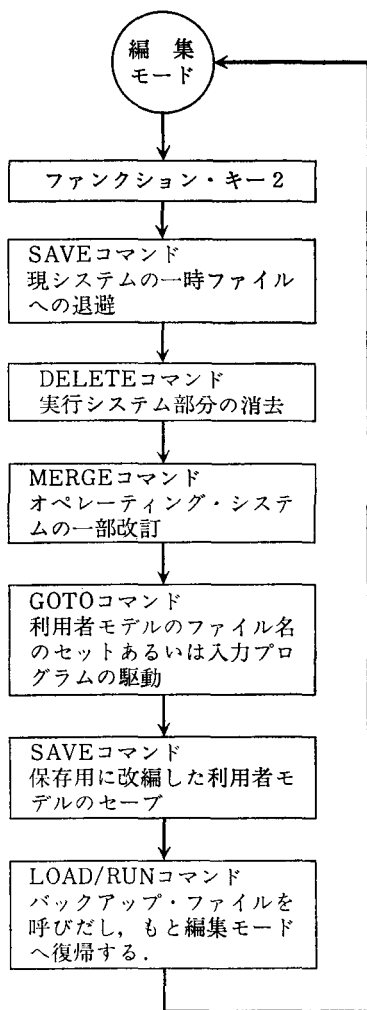


図4 コマンド・プロシーチャーの例

助、レイト方程式を記述する時である。プログラムを一度でも書いたことのある者であれば困難なことではないが、最終的に各レイトを決定するように当該レイトに関係する補助方程式を文番号順に記述する必要がある。この点が本システムの一番の難点と思えるが、以下のように指導することでこの難点を克服できる。

通常、モデルを記述する前に、図1のようなフロー・ダイアグラムを書かせるが、この中の点線とその矢印が記述順序を決めている。SDモデルの構築法にしたがえば、すべてのレイトは点線の終着点になっており、その点線をさかのぼると途

中合流・分岐することがあっても、最終的には始点となっているレベル変数か定数、あるいはシステム変数か外生変数に到達する。そうでなければ間違ったモデル構築を行なっていると言える。そこでまずフロー・ダイアグラム上で、1つの始点から終点までの補助変数に文番号を昇順に記入していく。次に番号のついた補助変数のうち、他の補助変数から矢印が流入しているものについては流入方向へさかのぼり、出会う補助変数に降順に番号をつけていく。そして始点あるいはすでに番号がついている補助変数に出会うまでつけてゆくが、後者の場合、最後につける番号はすでについている番号より大きな番号であるかをチェックする。これをくりかえすことによりすべての点線矢印をはさむ変数の文番号は、矢印方向に大きくなるようになり、各補助変数の文の記述順序が決まる。

プログラムの未経験者にとっては最初はむずかしいように聞こえるが、教育用のモデルはそれほど複雑ではないので、筆者の経験では思ったほど混乱は生じない。プログラム経験者には、最終的に各レイトを決定するように記述せよというだけで十分であるが、フロー・ダイアグラムを吟味する意味でもこのプロセスを踏ませた方がよい。

4.2 システム変数

システム変数である TIME, LENGTH, DT, PLTPER は DYNAMO とまったく同じ意味で、利用方法も同じである。ただしこれらの初期値はすべて4000番台の文番号をもつ代入文として記述されなければならない。もし記述されていない場合は各デフォルト値が設定され、記述されていても相互に不整合があるときは自動的に再調整される。これらの規則も DYNAMO と同等である。ただし数値出力間隔を与える PRTPER は、本システムでは PLTPER と同じに設定される。

リスト2ではシミュレーション開始時点が、5、終了時が110、計算の時間間隔が0.1、出力間隔は2.5に設定されている。この設定のもとでは

PC-98VM2のクロック10Mzで実行した場合、計算に約3分強かかる。もちろん誤差が若干小さくなるが、DTを0.5程度にすれば40秒程度で結果が得られる。

図形出力する変数とそのスケールを与えるのがプロット文であり、これも文字代入文であることを除けば、記述方法はDYNAMOと同じである。リスト2のプロット文では変数PA, PB, INCA, FINAをそれぞれ文字A, B, R, Iで、スケールはPA, PBが下限0, 上限10000, INCA, FINAが下限0, 上限は自動スケーリングでプロットすることを指定している。プロット文字を省略すれば変数を左から数えた順位が出力文字となる。ただし一度に出力できる変数の数は5つまでである。

4010番の文は代入された文字列が出力時のモデル識別名になる機能を有し、また文字列の最初6文字がモデルまたは結果をセーブするときのディフォルト名となる。

表1 本システムの組込関数

DYNAMO と同等の仕様	同等でない関数
パルス関数	指数平滑関数
ステップ関数	3次情報遅れ関数
最大値, 最小値関数	1次指数遅れ関数
クリップ関数	3次指数遅れ関数
スイッチ関数	パイプ遅れ関数
ノイズ関数, 正規乱数関数	サンプル関数
テーブル関数	ランプ関数

4.3 関数

DYNAMO において組込関数は重要な働きを担うが、本システムでも基本的な関数はすべてサポートしている。表1は本システムの組込関数で左側の関数はDYNAMOと引数の対応、利用方法が同等のものであり、右側は利用方法が異なる関数である。図5とリスト3はテーブル関数、ステップ関数、パルス関数および指数平滑（1次情報遅れ）関数値の出力例と記述例である。

6110番はテーブル値を設定するテーブル文で、

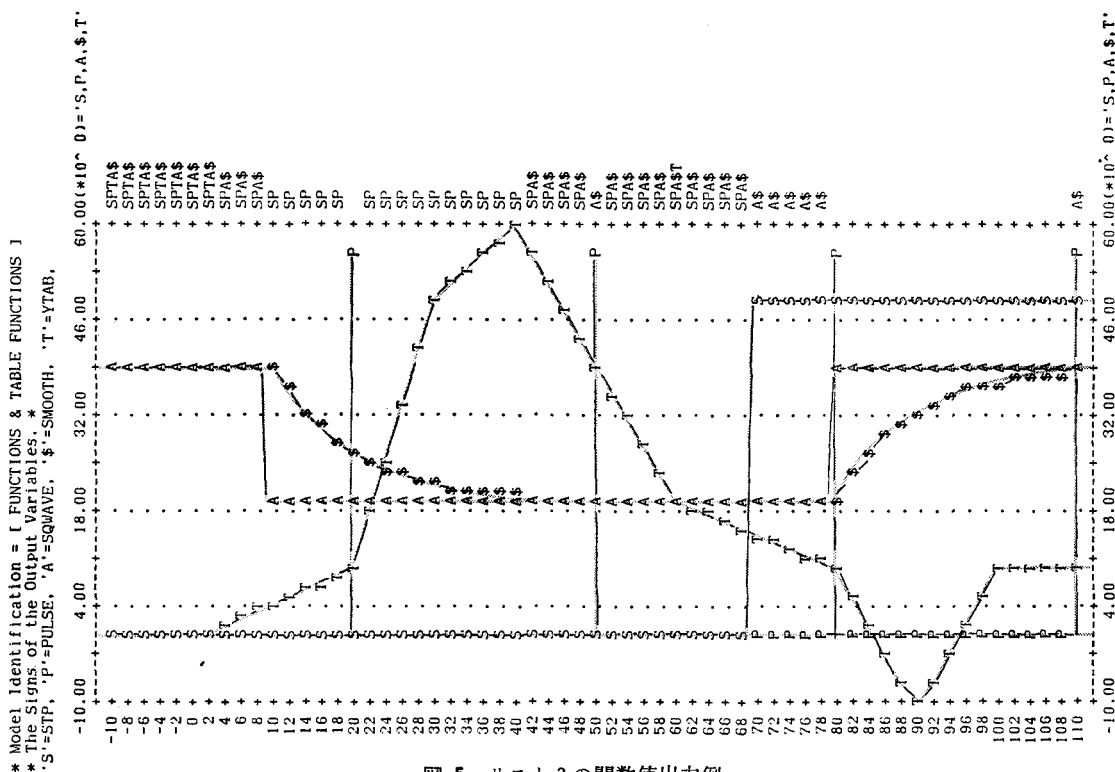


図5 リスト3の関数値出力例
(見やすいように、計算機のアウトプットに線を補足した)

```

4000 REM -----SYSTEM/CONTROL VARIABLES.
4010 MODEL$ = "FUNCTIONS & TABLE FUNCTIONS"
4020 PLTPER = 2
4030 TIME = -10
4040 LENGTH = 110
4050 DT = 1
4060 PLOT$ = "STP=S,PULSE=P,SQWAVE=A,SMOOTH=$,YTAB=T"
6000 REM -----INITIAL VALUES.
6110 TABT$ = "0.0 / 5.0/ 10./ 50./ 60./ 40./ 20./ 15./ 10/ -10/ 10."
6120 SQWAVE = 40
6130 SMOOTH$ = FNSMTH$(SQWAVE)
6140 TCNST = 10
7000 REM -----RATE/AUXILIARY EQUATIONS.
7010 YTAB = FNTABL(TABT$,TIME,0,100,10)
7020 STP = FNSTP(50,70)
7030 PULSE = FNPULS(56,20,30)
7040 SQWAVE = ( FNSTP(1,80) - FNSTP(1,10) + 2 ) * 20
7050 SMOOTH = FNSMTH$(SMOOTH$)
7060 SMOOTH$ = FNSMTH$(SQWAVE,TCNST,SMOOTH$)
8000 REM -----LEVEL EQUATIONS.

```

リスト-3 関数利用の例 1

形式は DYNAMO のそれと同等であるが、テーブル値は 4 カラムずつ記述しなければならない。7010 番はこのテーブル文を用いて、時間軸でテーブル関数を引用した例で、その値は図中文字 T で示されている。

7030 番はパルス関数の引用で、この例では高さ 56、周期 30、出力開始時点 20 と設定されている。

7020、7040 番はステップ関数の引用で、引数はそれぞれステップの高さ、立ち上がりの時点を与え、特に 7040 番では方形波を出力している。

7050 番は方形波に指数平滑をほどこす文であるが、この関数の利用法は残念ながら DYNAMO と引用方法が異なっている。表 1 の右側の関数はすべて関数値を引き出す文の他に 6130 番のような初

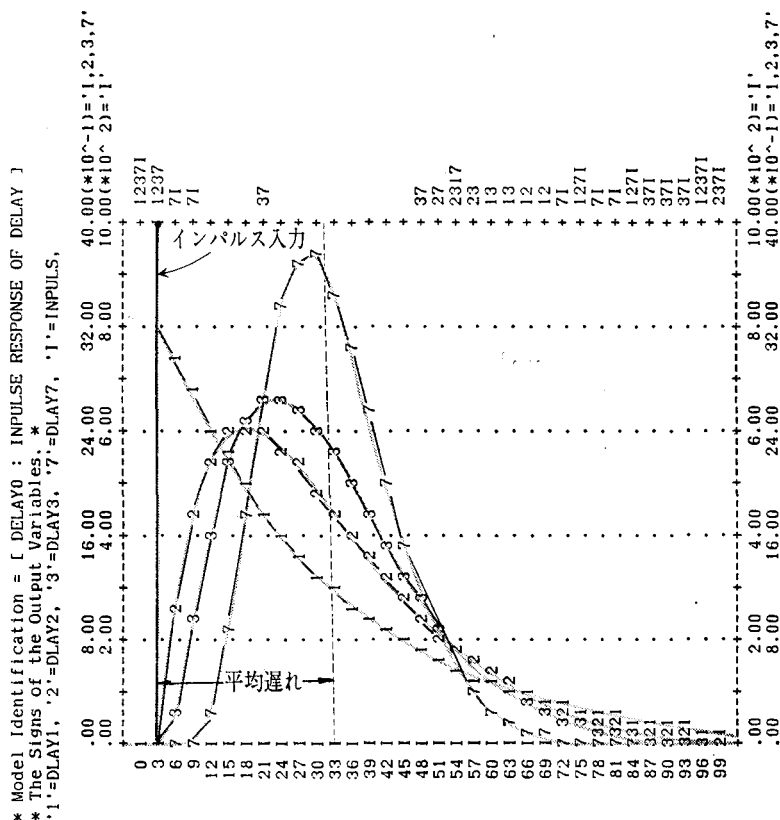


図 5 リスト 4/5 の指数遅れ関数の出力例

```

4000 REM -----SYSTEM/CONTROL VARIABLES.
4001 MODEL$ = "DELAY1 : IMPULSE RESPONSE OF DELAY"
4005 LENGTH = 100
4010 DT = .1
4020 PLTPER = 3
4060 PLOT$ = "DLAY1,DLAY2,DLAY3,DLAY7=7/INPULS=1"
6000 REM-----INITIAL VALUES.
6005 DEL = 30
6010 DL1$ = FNDLY1$(0,DEL)
6020 DL3$ = FNDLY3$(0,DEL)
6030 DL21$ = FNDLY1$(0,DEL)
6040 DL22$ = FNDLY1$(0,DEL)
6050 DL77$ = FNDLY1$(0,DEL/7)
6060 DL72$ = FNDLY3$(0,DEL/7*3)
6070 DL71$ = FNDLY3$(0,DEL/7*3)
7000 REM -----RATE/AUXILIARY EQUATIONS.
7020 INPULS = FNPULS(100/DT,3,1000)
7030 DLAY1 = FNDLY1$(DL1$,DEL)
7040 DL1$ = FNDLY1$(INPULS,DEL,DL1$)
7050 DLAY3 = FNDLY3$(DL3$,DEL)
7060 DL3$ = FNDLY3$(INPULS,DEL,DL3$)
7070 DLAY2 = FNDLY1$(DL22$,DEL/2)
7100 DL22$ = FNDLY1$( FNDLY1$(DL21$,DEL/2),DEL/2,DL22$)
7110 DL21$ = FNDLY1$(INPULS,DEL/2,DL21$)
7120 DLAY7 = FNDLY1$(DL77$,DEL/7)
7130 DL77$ = FNDLY1$( FNDLY3$(DL72$,DEL/7*3),DEL/7 ,DL77$)
7140 DL72$ = FNDLY3$( FNDLY3$(DL71$,DEL/7*3),DEL/7*3,DL72$)
7150 DL71$ = FNDLY3$( INPULS,DEL/7*3,DL71$)
8000 REM-----LEVEL EQUATIONS.

```

リストー4 関数利用の例 2

期設定文と7060番のような関数利用文が必要となる。初期設定文では入力値の初期値を任意の関数利用名（図5の例ではSMOOTH\$）に設定し、

関数利用文で入力変数、関数利用名、その他パラメータ（図5の例ではTCNSTで平均遅れ/時定数）を与えなければならない。リスト4と図6は1次、

```

4000 REM -----SYSTEM/CONTROL VARIABLES.
4001 MODEL$ = "DELAY0 : IMPULSE RESPONSE OF DELAY"
4005 LENGTH = 100
4010 DT = .1
4020 PLTPER = 3
4060 PLOT$ = "DLAY1,DLAY2,DLAY3,DLAY7=7/INPULS=1"
6000 REM-----INITIAL VALUES.
6005 DEL = 30
6010 LV31 = 0
6020 LV32 = 0
6030 LV33 = 0
6040 LV21 = 0
6050 LV22 = 0
6060 LV11 = 0
6100 FOR I=1 TO 7
6110 LV7(I) = 0
6120 NEXT I
7000 REM -----RATE/AUXILIARY EQUATIONS.
7020 INPULS = FNPULS(100/DT,3,1000)
7030 DR31 = LV31/(DEL/3)
7040 DR32 = LV32/(DEL/3)
7050 DR33 = LV33/(DEL/3)
7060 DR11 = LV11/DEL
7070 DR21 = LV21/(DEL/2)
7080 DR22 = LV22/(DEL/2)
7100 FOR I=1 TO 7
7110 DR7(I) = LV7(I)/(DEL/7)
7120 NEXT I
7200 DLAY1 = DR11
7210 DLAY2 = DR22
7220 DLAY3 = DR33
7230 DLAY7 = DR7(7)
8000 REM-----LEVEL EQUATIONS.
8010 LV31 = LV31 + DT*(INPULS - DR31)
8020 LV32 = LV32 + DT*(DR31 - DR32)
8030 LV33 = LV33 + DT*(DR32 - DR33)
8040 LV21 = LV21 + DT*(INPULS - DR21)
8050 LV22 = LV22 + DT*(DR21 - DR22)
8060 LV11 = LV11 + DT*(INPULS - DR11)
8100 LV7(1) = LV7(1) + DT*(INPULS - DR7(1))
8110 FOR I=2 TO 7
8120 LV7(I) = LV7(I) + DT*(DR7(I-1) - DR7(I))
8130 NEXT I

```

リストー5 関数を用いない指数遅れの記述例

2 次，3 次および 7 次の指数遅れ関数利用例で，同様の扱いが必要となる。

後述の関数利用は初心者にとっては少々煩雑でむずかしい。それゆえ，まったくの初心者には遅れ関数を利用せず，筆者の経験ではリスト 5 のようにそれらの内部構造を記述させた方がよいと思われる。この例では 7 次の指数遅れの計算に FOR/NEXT 文を用いているが，もちろんすべて代入文で記述させる必要がある。内部構造を記述することで遅れの構造，特に物の遅れと情報遅れの相違を完全に理解させることができる。

5. 利用上のその他留意点

筆者は本システムを利用して教育する場合，S D の基本概念であるレベル，レイトの概念，正，負のフィードバック，非線形フィードバックの概念，さらに因果フロー図とフロー・ダイアグラムの書き方をまず集中的に講義する。そしてこれらの簡単な基本モデルを本システムで実験させ，さらに指数遅れの構造や平均遅れ定数の意味や D T の設定の仕方を教えながら応用モデルを扱うようにしている。モデルの記述は初期の段階では DYNAMO 言語で訓練する。

というのは，方程式の区別，時間添字を明示化した方がやはり理解しやすいし，DYNAMO 言語の解釈・利用も可能になるからである。DYNAMO 言語で記述したモデルを本システム用に記述し直すことは初心者でもそれほど困難でない。以下本システムを利用するにさいし間違いを起ししやすい点について，その対処法とともに紹介する。

BASIC 言語の構文上のエラー：初心者が最初に犯しやすいエラーの 1 つである。これらは編集モードでキー 3 を押すことでエラー箇所が指摘される。このキーは初期時点の 1 サイクルのみを実行してエラーを検出するので，初期時点での 0 割り，関数の引数の不整合等はこの機能で検出される。それゆえ実行前に必ずこのキーを押して，初

期時点でのエラーがないことを確認することを勧める。

実行中のエラー：実行してから検出されるエラーのほとんどは，システム変数の設定エラーと計算途中で発生する 0 割りである。前者についてはその内容が表示されるが後者はその発生箇所をつきとめる必要がある。エラー発生までの結果を出力できるので，それを見ることですぐ原因がわかる場合もあるが，そもそもモデルの設計やパラメータの設定による構造的な欠陥による場合もあり，初心者では無理なことも多い。これは本物の DYNAMO でもいえることであり，やはり経験者が見なければならないであろう。

論理のエラー：本システムは，BASIC の文法上は正しいが，DYNAMO の記述としては間違っているエラーの検出機能をもっていない。たとえば初期値設定の必要な変数の初期値設定忘れ，変数名の入力ミスによる未定義変数の存在等が主なところである。0 割りが生じたり，あるいは結果が明らかにおかしければよいが，モデルが複雑になるとなかなか発見しにくい。練習段階では定型的なモデルを扱うので問題はないが，自主的なモデルを開発するときはこの危険がつきまとう。それゆえ複雑なモデルを構築するときは，ブロックごとのモデル構築と検証を徹底させる必要がある。これは DYNAMO を使用する時にもいえることで，大規模なモデルを構築する時の定石でもある。

操作上のエラー：マイコン利用の初心者で操作上過ちを犯しやすいのは，編集モードでの CR キーの押し忘れと文番号の入れ間違いである。

本システムの編集機能は BASIC モードのスクリーン・エディターを利用しているために，文の修正後 CR キーを押し忘れる初心者が多い。それゆえ初期の段階ではこの点をよく指導し，リスト・キーを押して修正結果を確認するようにしなければならない。

文番号の入力のミスはそれが 4000～8000 番台で

あれば問題はないが、それ以外の場合モデルに結合されているシステムを壊す可能性がある。この場合ファンクションキーによる処理が実行できなくなるか、エラーで停止してしまう事態が起こりうる。この場合メンテナンス・モードに移行し、リカバリー・キーを押すことで復帰できる。今のところ経験していないが、メンテナンス・モードに移行できない場合も起こりうる。このときは直接 SAVE コマンドを投入して一時ファイルに現システムを退避させ、メンテナンス・モードを駆動することで復帰可能である。SAVE コマンドが投入不可能になることは、本システムがすべて BASIC 上で稼働するため、ほとんどありえないと思われる。

6. おわりに

本システムは教育用に限らず、大規模なモデルでなければ実用的利用も可能である。筆者は大規模住宅団地開発の地方財政への影響を本システムで現在実験しているが、ほぼ満足した結果を得ている。ちなみにこのモデルは約500の変数を用い、30年間のシミュレーション1回に約7分強かかるものである。このような利用にさいしては代入文に限らず他の BASIC の文を用いて効率的なモデル記述が可能で、これは DYNAMO にはない便利な点である。しかしより高度な実用性をもたせるためには、各種のシステム制約の緩和等改良の余地があり、現在その仕様と方法について思案中である。

最後に本システムを当学会賞の候補に推薦していただいた東北大学高橋幸雄先生、および本システムを実際に使用して審査していただいた審査委員諸氏に誌面を借りて厚く感謝申し上げます。また本システム開発の機会を提供してくださった、東京都職員研修所木佐美廣氏、ならびに有益なコメントをいただいた研修生諸氏にも厚くお礼申し上げます。さらに PC-98 の技術的側面で筑波大学荒川治久技官に有益な助言をいただいたことに

感謝いたします。

参考文献・資料

- [1] Alexander L. P. : DYNAMO USERS MANUAL, Fifth edition. The M. I. T. Press, 1976.
- [2] 藤田英時, 幸田敏記: PC-Techknow 9800. システムソフト.
- [3] Goodman, R. M., (蒲生敏輝, 山内昭, 大江秀房訳): システム・ダイナミックス・ノート. マグローヒル好学社. 1981.
- [4] 林亜夫: BASIC-DYNAMO使用手引書, Ver. 3. 1985.
- [5] 小玉陽一: BASICによるシステム・ダイナミックス, 共立出版.
- [6] FACOM OSiv/F4 DYNAMO解説書(99AR-0300-1). 富士通.
- [7] FACOM OSiv/F4 DYNAMO使用手引書(99AR-0300-1). 富士通.

脚注

- 1) 本文内での DYNAMO に関する仕様は、参考資料 [1], [6], [7] にもとづいている。
- 2) 本システムを使用されたい場合は、PC-98用のシステムで5インチ2HDで提供しますので、新品のフロッピー2枚を送付してください。このうち1枚は使用手引書と交換させていただきます。
- 3) ある生物Aは生物Bを捕食し増殖する。時点0で生物Bの生息域に生物Aが流入した後、両生物集団の成長プロセスをシミュレーションするモデルである。テーブル値等のパラメータは仮想的に設定した。
- 4) 本家の DYNAMO では Z Z で始まる変数名は使用できないが、本システムでは Z で始まる変数をシステム自身が使用するので利用者は利用できない。

今回は、“パソコンによるOR”の、いわば続編である。

本稿は、1986年度、本学会事例研究奨励賞（ソフトウェア部門賞）を受賞された林 亜夫氏に、賞の対象となったシミュレーション・システムについてその内容の詳細を紹介していただくことにした。

(古林)