

シミュレーション技法からみた システム・ダイナミックス

椎塚 久雄

1. はじめに

システム・ダイナミックスの基本概念は、多くの変数があり、かつその変数間に遅れフィードバック (lagged feed back) のあるシステムをストック (stock) とフロー (flow) の観点から簡明にモデル化しようとする方法論である。過渡の状態を含むこのようなシステムを数学的に解くことは、実質上不可能であるから、普通はシミュレーションによる方法が用いられている。このシミュレーション技法が、システム・ダイナミックスの特徴を浮き彫りにしていると言っても過言ではない。

システム・ダイナミックスは、本来コンピュータ言語を選ばない方法論であるが、最近ではパソコン機種の使いやすいソフトウェアが入手できるようになり、システム・ダイナミックス・モデルのシミュレーション・ツールに関しては、かなり改良されてきていることも見逃せない。

ここでは、システム・ダイナミックスがふるまう姿をシミュレーション技法上の観点から眺めてみよう。

2. シミュレーション言語の特徴とモデルの分類

シミュレーション言語は通常、離散システム (discrete system) シミュレーション言語と連続システム (continuous system) シミュレーション言語に大別される。連続-離散の両系を扱うシミュレーション言語もあるが[3]連続系、離散系の両システムは思想的にかなり異なるモデルの構成概念にもとづいているのでシミュレーション言語もそれぞれ独立に発展して今日に至っている。

シミュレーションの時刻法: 離散システムと連続システムの基本的な

しいづか ひさお
工学院大学 電子工学科

〒160 新宿区西新宿1-24-2

相違は、シミュレーションの時刻法にある。それらの顕著な特徴を比較して表1に示す。離散システムシミュレーションでは、事象 (event) 発生概念があり、これが重要な役割を果たしている。この事象の発生時刻は必ずしも等間隔ではなく、事象が発生する毎に時間が進行すると考えられる[2]。

これに対して、連続システムシミュレーションでは、ある一定の微少時間増分で時間が進行しシミュレーションが行なわれ、そこには事象の発生概念はない。システム・ダイナミックス・シミュレーションはこの連続システムに属している。

離散システムシミュレーション言語: 1960年代の初め頃から開発され、その後実用化が進んだ代表的なものとして、GPSS, SIMSCRIPT, SIMULAなどが知られている。最近では両系にもある程度適用されるSLAMIIやSIMANなども注目されている。

連続システムシミュレーション言語: 1955年Selfridgeによって開発されたのが最初である。現在IBM系のCSMP, CDCやヨーロッパ系のCSSLが代表的言語とされている。連続システムシミュレーション言語では、対象となるシステムを普通は微分方程式で記述するが、システム・ダイナミックスで用いられているDYNAMOでは、ある時期から次の時期までの変化の量に着目することから、対象システムを差分方程式で記述している。

シミュレーション・モデルの分類: 一般的に、シミュレーション・モデルは、かなり広い範囲をカバーしている。シミュレーション・モデルの関係を図式化したもの

表1 シミュレーションの時刻法の比較

離散型システム	連続型システム
事象 (イベント) の生起は等間隔ではなく、時間の増分をあらかじめ規定しないでモデルの構成要素の活動にもとづいて、各時間ステップごとに個々に決定され時間が刻まれる。事象は活動を開始したり終了したりする要素の順に発生する。	事象の生起に関係なく微少等間隔で時間が刻まれる。モデルの時間進行は一定量で増加する時間のジャンプ列でそのつど段階的に時間が進む。たとえば、映画フィルムの動きはこのような連続システムの典型的な例である。

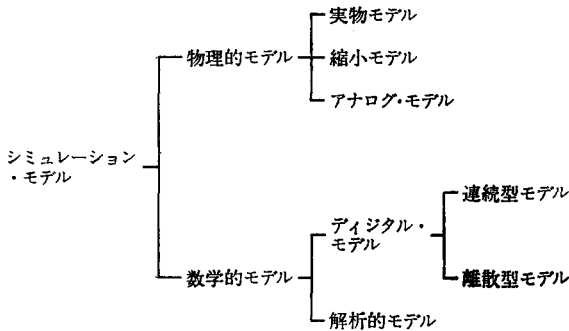


図1 モデルの分類(文献[3]より): システムダイナミックス・モデルは、デジタル・モデルの連続型モデルに属している。

を図1に示す[3]. システムダイナミックス・モデルは図1のデジタル・モデルの中の連続型モデルに属している。

3. 動的システムとシステム・ダイナミックス

動的システム(dynamic system)は1次の微分方程式の集合 $\frac{dL}{dt} = f(L, C, t)$ で記述することができる。ただし、 L は状態変数ベクトル、 C はパラメータベクトル、 t は時間変数をそれぞれ意味する。上式において、 f が L, C および t に関して弱連続性の条件を満たし、かつその解が C と t に関して連続であるならば、これらの方程式の解の存在と一意性は保証される。

システム・ダイナミックスも、上のような微分方程式の集合として表わすことができる。ただし、状態変数はレベル、環境パラメータはシステムの関係を表述するのに必要な遅延時間、テーブル関数などの定数である。

一般にレートベクトル f は C^∞ クラス (滑らかな関数) に属していない、MAX関数、MIN関数、SWITCH関数、およびCLIP関数を含んだモデルは C^∞ クラスではない。TABLE関数あるいはTABHL関数も、それらの微分が最初の点において不連続となるから、 C^∞ クラスからは除外される。結局、システムダイナミックス・モデルの構成要素はシステム状態の滑らかな関数である必要はなく、本質的に複雑なものであるということが出来る。

3. システム・ダイナミックス・シミュレーションの特徴

すでに述べたように、システム・ダイナミックスで

は、対象システムを差分方程式で記述している。

あるレベルに対するインプットフローを X_1 、アウトプットフローを X_2 とすると、レベル Y の変化率は

$$\frac{dY}{dt} = \lim_{\Delta t \rightarrow 0} \frac{Y(t+\Delta t) - Y(t)}{\Delta t} = X_1 - X_2$$

となり X_1 と X_2 の差になる。上式より

$$Y(t+\Delta t) - Y(t) = \Delta t(X_1 - X_2)$$

が得られる。

DYNAMOでは「 dt 」を「 DT 」で表わし、さらに時間表示記号として J ($1DT$ 前)、 K (現在の時間)、 L ($1DT$ 後)、そして JK (J 時点と K 時点の間)、 KL (K 時点と L 時点の間)がある。上の方程式をこの記号を用いて表わすと

$$Y.K = Y.J + DT*(X1.JK - X2.JK)$$

となり、これはレベル方程式と呼ばれている。

モデルを構成する方程式は、一定の順序にしたがって DT 時間ごとに計算される。なかでも、基本となるレベルとレートは次のように計算される。

まず与えられたレベル(レベルには初期値が必要)によってレートの初期状態を計算する。そしてそのレベルを使って K 時点のレベルを計算する。次にすでわかっている JK 期間のレートと、いま計算した K 時点のレベルから次の KL 期間におけるレートの値を計算する。この後、時間を DT だけ進める。すなわちいままでの K 時点新しい J 時点に、 L 時点新しい K 時点として同じステップを繰り返しシミュレーションを進めていく。

このように、システム・ダイナミックス・シミュレーションの基本的な手段は時間を伴った積分概念:

$$\text{現在の量} = \text{その前の量} + \text{経過時間} * \text{変化率}$$

にもとづいているといえる。したがってシステムダイナミックスが他のシミュレーション技法と異なる最も特徴的な点を1つあげるとするならば「時間をとおしてのシステムのふるまいの研究」であるということが出来る。

文 献

- [1] Rolf Clark: "System Dynamics and Modeling", Operations Research Society of America, July 1988.
- [2] 椎塚久雄: "ペトリネットのシミュレーションへの応用 [I]—離散事象システムとペトリネット—", オペレーションズ・リサーチ, Vol.34, No.11, pp.607-611, 1989年11月.
- [3] 森戸晋・相沢りえ子: "SLAM IIによるシステム・シミュレーション入門", 企画センター, 昭和61年.