

シミュレーション利用状況調査と シミュレーション言語に関する最近の話題

森戸 晋, 矢崎 義行

1. はじめに

シミュレーションによる分析を必要とする複雑なシステムの増加によるシミュレーションの需要とシミュレーションの作業を容易にするシミュレーション言語等の道具の供給とがうまくマッチして、米国では離散系のシミュレーションが再びブームを巻き起こしており、わ国でも同様な傾向が窺われる。本稿では、アンケート調査の結果得られたわが国におけるシミュレーションの最近の利用状況を報告するとともに、シミュレーション言語とそれに付随するさまざまな道具の普及・充実により新たに生まれてくるシミュレーション実施上の重要な検討証題について考えることにする。

2. シミュレーション利用状況のアンケート調査の結果

筆者らの研究室で行なった調査の主目的は、

- 1)シミュレーション技法の一般的利用状況, 2)シミュレーション・プロジェクトの現状と問題点,
- 3)プロジェクト実施に当って使用される道具, 4)モデルやプログラムの検証の方法, の4点を明らかにすることである。本調査はランダムに広く各機関を対象とするのではなく、むしろシミュレ-

もりと すずむ, やざき よしゆき
早稲田大学 理工学部 工業経営学科

〒160 新宿区大久保 3-4-1

表 1 アンケート回答会社の業種

業 種	度数	業 種	度数
鉄鋼・金属	7	諸 工 業	4
電気機械	16	電力・エネルギー	11
精密機械	5	ソフトウェア・サービス	13
輸送用機器	5	通 信	2
建 設	10	金融・保険	4
プラント	4	そ の 他	10
化学・薬品・食品	10	合 計	101

(発送総数 231)

ション(ことに離散型)を実際に使っていると考えられる機関を中心に行なった(表1)。またアンケートは1企業1回答というわけではなく、企業によっては3~4の複数回答もある。以下に、結果に関する若干の補足、コメントを記す。

2.1 シミュレーションの一般利用状況(表1-表4)

1)OR技法の利用頻度(表2):シミュレシヨ

表 2 各企業で使っているOR関連技法

技 法	度数	技 法	度数
シミュレーション	71	ネットワーク計画	21
線形計画法	56	動的計画法	17
人工知能	46	整数計画法	16
PERT/CPM	42	マルコフ過程	13
待ち行列	37	情報理論	10
在庫理論	29	A H P	10
非線形計画法	28	ゲーム理論	9
信頼性理論	24	探索理論	9
スケジューリング理論	24	取り替え理論	6

(全101社, 複数回答)

表 3 シミュレーションの用途

生産システム	度数	社会システム	度数	ネットワーク	度数	その他	度数
工場設計	31	経営・経済	10	計算機	17	人工知能	27
スケジューリング	26	流通システム	9	通信	13	電力系統	13
在庫	23	人事・組織	7	交通	9	構造・材料	12
工程設計	22	資源・環境	6	その他	2	化学反応・分子運動	11
その他	3	その他	4			機械制御	10
		地域開発	4			回路素子	6

表 4 使用シミュレーション技法

技法	度数
離散型	48
連続型	45
離散/連続混合型	12

ン・ユーザを狙って調査を実施したため若干シミュレーションの頻度が高めに出現している可能性があるが、使われているOR技法は従来の調査結果とあまり変わりはなくシミュレーション、LP、PERTの順で頻度が高い。ただ注目すべきは人工知能の頻度が非常に高い点である。

2)適用分野(表3)：予想どおりに生産システム的设计, 運用によく使われているが, ここでも人工知能の回答が多い。

3)シミュレーション技法(表4)：現状では, 離散/連続混合型を利用している機関は全体の10%程度にすぎず, またその利用率も低い(5~20%), 混合型モデル化を許すシミュレーション言語の歴史の浅さを考えると混合型の利用増加が予想される。

2.2 プロジェクト実施の現状と問題点(表5, 図1, 図2)

1)人数/期間(図1, 図2)：プロジェクト遂行

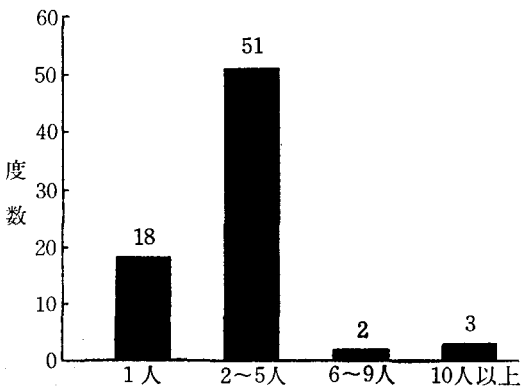


図 1 シミュレーションプロジェクトの人数

に相当のマンパワーがかけられていることがわかる。なお, モデル作成, プログラミング/デバッグ, 結果解析に投入している時間は平均的にはそれぞれ29%, 35%, 36%ずつであるが, 個々の機関では, かなりのばらつきが見られる。

2)実施上の問題点(表5)：苦勞している点は, 「モデルの作成」, 「結果の提示・説明の仕方」, 「結果を意思決定に反映させる」等で, 逆に問題とならない点は, 「シミュレーション言語の選択」, 「乱数の信用性」である。

2.3 シミュレーション実行に利用する道具(表6~表8, 図3~図5)

1)言語(図3~図5, 表6)：71%の機関が連続型を含み何らかの形でシミュレーション言語を使用している。と同時に, 依然として一般言語がかなり使用あるいは併用されているが, その理由が表6に示されている。融通性の欠如では, 具体的にモデル化機能の不足(離散/連続混合システム

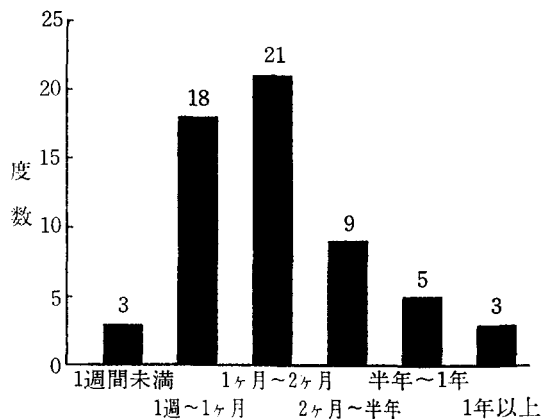


図 2 シミュレーションプロジェクトの期間

表 5 シミュレーションプロジェクト実施に当たっての問題点
(表中の数字は度数)

項 目	←問題有					平均
	5	4	3	問題無→ 2	1	
シミュレーション言語の選択	9	5	12	18	21	2.4
モデルの作成	29	19	19	5	3	3.9
プログラミング	8	12	22	15	11	2.9
デバッグ	12	14	22	12	9	3.1
乱数の信頼性	2	7	19	14	16	2.4
入力データの収集	20	18	18	7	3	3.7
入力分布形の決定	6	13	23	7	7	3.1
入力パラメータの決定	7	13	19	8	8	3.1
実験計画の立てかた	7	8	21	9	9	2.9
初期条件の決めかた	3	14	21	13	12	2.7
ランの長さ/回数決定	6	11	22	13	9	2.8
結果の分析方法	12	18	26	7	6	3.3
結果の提示・説明の仕方	19	15	20	10	6	3.4
結果を意思決定に反映させる	14	14	18	11	2	3.5

への対応性や、詳細レベルのシステム・モデル化への対応性等), 他のツール (CAD等) との連動の悪さ, 配列データの演算への対応の悪さ, アウトプットの加工性の悪さ, 等があげられている。

一般言語ではFORTRANの利用が多い。BASICやCはアニメーションを行なうために用いられたケースが多い。(図4)

離散型シミュレーション言語では、予想どおりGPSSの利用が圧倒的に多い(図5)。なお、本調査の対象には昭和61年10月のSLAMのシンポジウムにおける参加者が含まれているために、SLAMの利用頻度が若干高めにでている可能性がある。また、連続型言語ではCSMP, DYNAMO

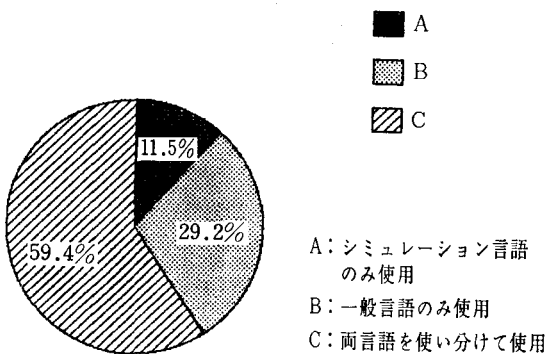


図 3 使用する言語

表 6 一般言語を使用する理由

理 由	度数
A: シミュレーション言語の融通性の欠如	26
B: 手元に見えるシミュレーション言語がない	16
C: シミュレーション言語に使い慣れていない	13
D: シミュレーション言語の実行速度の遅さ	5

の利用頻度が高かった。

2) 統計的側面(表7): 離散型のユーザの多い中で、組平均法, 独立繰り返し法などの標準的方法の回答が異常なまでに少ないのは、シミュレーションの統計的側面に対する基礎的知識が不足していることを窺わせる。シミュレーションに関

連して使われる手法として、時系列分析, 多変量解析などの回答頻度が高いが、必ずしもシミュレーションに直結して使われているわけではないのではないかという疑問が残る。

3) アニメーション(表8): アニメーションの利用目的は、トップへのPR, エンドユーザへの説明ツール(プレゼンテーション), 正常稼働の確認, 等があげられている。アニメーションのユーザ(19件)の8割(15件)が独自開発であるということを見ると、シミュレーション言語に直結したアニメーション機能が普及することによってアニメーションが急速に普及する可能性がある。

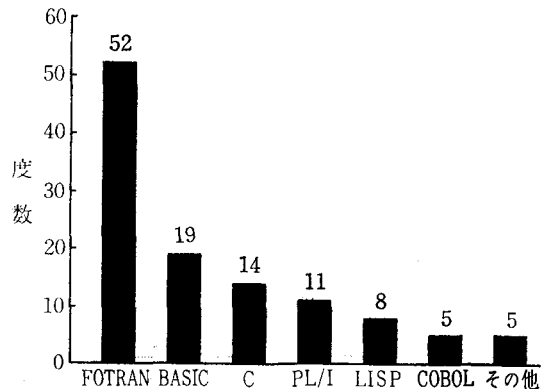


図 4 使用している一般言語

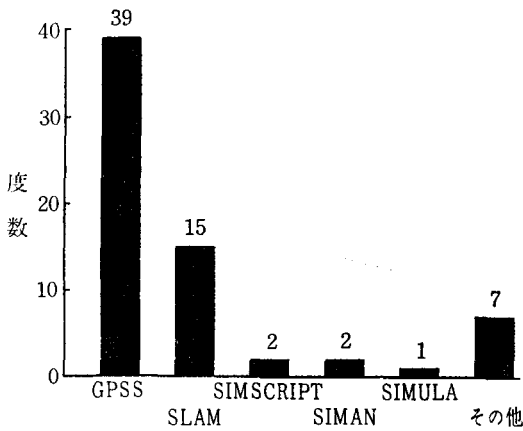


図 5 使用シミュレーション言語 (離散型)

2.4 妥当性の検証 (表9)

1)プログラムの正当性(表9:A, B):「プログラムが仕様どおりに動いていないのではないか」という疑いをもっているユーザが多い。仕様どおりに動いていないシミュレーションをもとにシステム評価が行なわれてしまうという事象の推定発生確率は、5~15%という回答が多いが、なかには50%や80%という回答もあった。

2)モデルの妥当性(表9:C, D):表5でモデルの作成を問題点としてあげている機関が多いことから明らかのように、現実のシステムとモデ

表 9 シミュレーションプログラムについて質問と回答

質 問	回 答	度数
A: モデルが作成者の意図どおりに動いているか否かをどのように確認しますか?	手計算との比較	30
	トレース	25
	現実のシステムとの比較	23
	解析解との比較	11
	アニメーション グラフ・プロット	9 5
B: プログラムが作成者の意図どおりに動いていないにもかかわらずシステム評価が行なわれると思いますか?	まったく考えられない	19
	たまにはあり得る	29
	ときどきある	11
	かなりの確率であり得る	6
C: シミュレーション結果と実際のシステムとを比較しますか?	する	53
	しない	10
D: シミュレーション結果と実際のシステムとの差はどの程度ですか?	ほとんど違わないことが多い	9
	違うこともときどきある	40
	違うことが多くて困る	5

表 7 シミュレーションに関して使用している統計的手法

手 法	度数	手 法	度数
回帰分析	36	独立くりかえし法	5
多変量解析	31	RESPONCE SURFACE METHOD	3
統計的推定・検定	30	組平均法	2
時系列分析	29	分散減少法	2
実験計画法	15	REGENERATIVE METHOD	2

全 101 社、複数回答

表 8 アニメーションについての質問と回答

質 問	回 答	度数
アニメーションを使っていますか?	使っている	19
	使っていない	41
どのようにアニメーションを作成しますか?	独自に開発する	15
	アニメーション専用の市販ツールを用いる	2
	シミュレーション言語に組み込まれた機能を利用	1

ルとの違いで苦勞しているユーザが多い。

2.5 シミュレーションの将来

シミュレーションが将来ますます使われるようになるかと答えた人は全体の8割であった。逆に使われなくなると答えた人は1人もいなかった。また将来どのような分野で使われるようになるかという質問については、人工知能技術と結びついて、生産システムや情報システムの分野で大きく発展するであろうという回答が多かった。

3. シミュレーション言語に関する最近の話題

調査からも明らかのようにシミュレーションに対する必要性が高まりつつあるなかで、シミュレーションの作業を容易にする道具を提供するシミュレーション言語の最近の普及・機能拡張には目を見張るものがある。以下では、これ

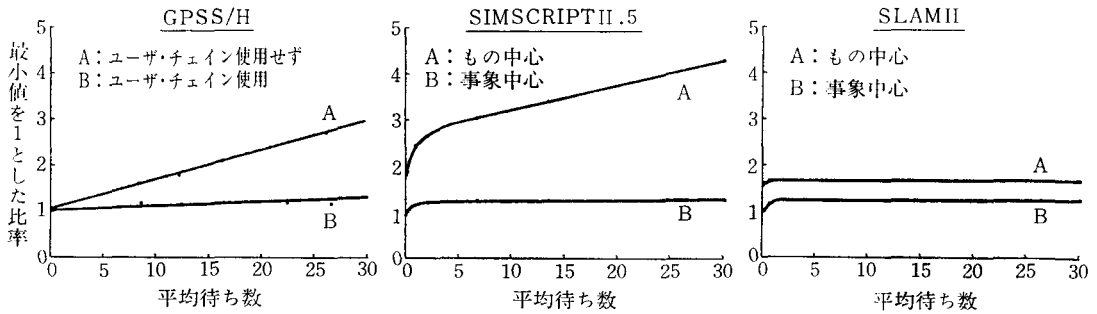


図 6 M/M/1 待ち行列モデルの処理効率

らの言語の普及・発展にもなつて話題となっているトピックのうち、各言語の特性分析、プログラミングの戦略、プログラムの正常稼働の検証、アニメーションについて解説する。

3.1 離散型シミュレーション言語の最近の動向

1970年代中頃を境として、シミュレーション・ソフトウェアは計算機メーカー主導から、シミュレーション専門とするソフトウェア・ハウス主導に変わりつつある。パソコンの普及と相まって、この傾向は1980年代に入ってことにいちじるしく、離散型言語の代名詞と言える GPSS, SIMSCRIPT に加えて、SIMULA, SLAM, SIMAN, SIMPLE 1, SEE WHY 等の言語がつぎつぎと開発され、Simulation 誌のサーベイ[S1]によると米国ではパソコン用を含めて50種を越えるソフトがしのぎを削っている。

こうした言語開発の動きの主な特徴は次のようにまとめることができる。

- A) 汎用機版とともにパソコン版が提供され、開発されたプログラムがどちらでも動く。
- B) 離散型機能に加えて連続型の機能を加える。また、離散型については、「もの中心」のモデル化機能を提供する。
- C) モデル化機能の柔軟性を高めるために FORTRAN 等の一般言語とのインターフェイスを容易にする。
- D) デバッグ効率を高めるためにデバッガを提供する。

E) アニメーション機能を提供する。

F) データベースとの対応を容易にする機能を提供する。

G) SIMSCRIPT に対して SIMFACTORY, SLAM に対して MAP/1 というように汎用離散系言語に対応する生産専用シミュレーション言語あるいはシミュレータを提供する。主要な言語に関する解説は、たとえば、中西 [N2] を参照されたい。

3.2 シミュレーション言語の特性分析

シミュレーション言語にはそれぞれ癖があり、得意不得意があるが、その実態、特に不得意な側面は各言語のテキストにもマニュアルにも明らかにされていない場合が多い。この事実は古い歴史をもつ GPSS にもあてはまり、言語の処理特性やプログラミング・ノウハウ的な情報が収められた文献はほとんど見当たらない(中西 [N1] はこの意味で貴重である)。言語を有効に使いこなすためには言語の特性を十分に把握しておくことが重要である。

特性分析の一例として、図 6 は簡単な M/M/1 待ち行列モデルを GPSS, SIMSCRIPT, SLAM で実行させた場合の実行時間の伸びを示したものである。この図は、システムを通過する要素数を一定数(10000)に固定し、利用率を変えることによりシステムの混雑度を変化させたときの処理時間の変化を見たものである。各言語とも2種類の等価なモデルを作成し、処理時間を比較している。すなわち、GPSS ではユーザ・チェーン(UC)

を使用する場合と使用しない場合、SIMSCRIPTとSLAMでは「もの中心」と「事象中心」のモデルを作成・実行し、各言語とも処理時間の最小値を1としたときの比率を縦軸にとっている。これより次のことがわかる。

A) GPSSでは、その特異な事象処理ロジックが故に、ユーザ・チェーンを使用しない限りシステム中の滞留要素数が多くなるにしたがって処理時間が長くなる。このことは比較的よく知られている[N1]が、調査の結果ユーザ・チェーンを常時使うと回答したのは23回答中7件(30%)に留まり、5件がときどき使う、11件がほとんどあるいはまったく使わないと答えている。

B) SIMSCRIPTに付加された「もの中心」のモデル化機能にも前項AのGPSSと同じ傾向が見られる。このことは筆者の知るかぎり指摘されていない。

C) SLAMでは「もの中心」の場合も「事象中心」の場合も処理時間が滞留要素数に依存しない。

このような分析は言語の比較においてのみでなく特定の言語の効率的利用においても重要な情報を提供するが、ほとんど実施されていないのが実状であり、言語を使いこなすために今後ますます必要となる。

3.3 プログラミングのアプローチ

システムの動きを規定するルールや仕様が完全に定まり、しかもこの動きを離散型シミュレーション・モデルで表現することに決めても、与えられた言語(たとえばGPSS)でこの動きを表現する代替的方法がいくつも考えられるのが普通である。分析したいシステムの動きという観点からは完全に等価な複数のシミュレーション・プログラム(シミュレーション・モデルと言うことも多い)が、処理効率やモデルの作りやすさ、わかりやすさ、修正のしやすさ等の観点からは大幅に異なる場合も少なくない。

「もの中心」の離散型シミュレーション言語は分析したいシステムを簡単に表現できる柔軟なモ

デル化機能を提供するように設計されているが、30種類前後のシンボルであらゆるシステムを表現することには無理があり、モデル化しづらい状況がどうしても出てくる。このような状況をいかに乗り切るかがモデル作成者の「腕」の見せどころとなる。

このように、①代替的モデル化が可能な状況や②モデル化しづらい状況に直面したときにどのようなアプローチをとることが望ましいかを明らかにするのがプログラミング戦略の問題である。プログラミングの戦略には、使用言語に依存しない一般論と特定言語の枠の中の話とがあるが、いずれも「その道の達人」の頭の中に隠されている「秘伝」にとどまっているのが実状である。

モデルにおける「もの」の捉え方や表現しづらい状況の表わし方等によって定まるプログラミング・アプローチがプログラムの作成時間、処理時間、検証やデバッグの時間におよぼす影響が矢崎等[Y1]によって検討されている。今後シミュレーション・プロジェクトの全般的作業効率向上のためにこの種のシミュレーション・ソフトウェア工学的分析がますます要求される。

3.4 シミュレーション・プログラムの正常稼働の検証

シミュレーション・プログラムが作成者の意図どおりに稼働しているかどうかを確認することを正常稼働の検証(verification)と呼ぶ。

アンケートの結果、ユーザが相当の時間を正常稼働の検証に費やしていることが明らかであると同時に、モデルが正常稼働しているかどうか自信がない場合が少なくないことを示している。われわれの研究室ではシミュレーション・モデリングの研究のために完全に同一の問題状況を複数の代替的プログラムで表現する実験をくりかえしているが、等価であるべきプログラムが異なる結果を出すことがかなりの頻度である。その原因がプログラム作成者の未熟さによるところも少なくないが、にもかかわらず、作成者の意図どおり動いて

いないプログラムが世の中にならりと推測させるのに十分である。

作成者の意図どおりに動いていない場合でも大勢に影響しない程度の誤りならば許容できる場合が多いが、一般に誤りが結果に大きく影響するかどうかはわからない場合が多く、また、事象処理ロジックに関する一見些細な誤りが結果を大きく変えるケースもあるので厄介である。

プログラムの正常稼働を検証するためのアプローチとして以下の方法が考えられる。

A) プログラムをトレースし意図どおり動いていることを確認する。B) 複数の代替的プログラムを作成しそれらの等価性を確認する。C) 実システムと比較する。これにはアニメーションを用いて動きを比較する方法と統計結果を比較する方法とが考えられる。D) 解析解と比較する。

Aは従来からの標準的な検証の方法であり、今後もこの方法が基本となることは変わらない。しかし、進歩いちじるしいアニメーションやデバッグ機能を活用することにより、検証の効率化が進むことが予想される。ただしアニメーションやデバッガーが正常稼働検証の万能薬というわけではなく、アニメーションで表面化しづらい誤りも考えられる。一方、Bを実際規模のモデルでやる時間的余裕は通常ない。実際は、Aを基本として、可能な場合にCやDを併用するという方法が標準的である。

3.5 アニメーションの効果的活用

シミュレーション・プロジェクトを実施するに当たっては、3種類の立場の人々の存在が考えられる。まず第1にシミュレーション技術者（＝モデル設計者）であり、次に、モデル分析の対象となるシステムをよく知る人（＝「現場の人」と呼ぶことにする）、最後にシミュレーションの結果を意思決定に反映させようとするモデル・ユーザ（＝意思決定者）である。シミュレーション・プロジェクトの成否はこれらの人々のコミュニケーションの善し悪しに大きく依存するが、シミュレ

ーションモデルの動きのアニメーションはコミュニケーションの増進の強力な武器となる。より具体的に、シミュレーションにおけるアニメーションの効用は、

- 1) モデル作成時におけるモデル設計者と現場の人やモデル・ユーザとのコミュニケーションの道具としての効用
- 2) モデルの正常稼働検証時の効用
- 3) 結果提示および説明時における効用

という3側面から捉えられるが、以下では1)と3)について考える。

(1) モデル作成時における効用

一般に、モデル作成のプロセスは1回やればおしまいという形ではなく、何回もくりかえし行なわれるものである。モデル作成プロセスがくりかえしをとまなうのは主として以下の2つの理由による。

A) 実際のシステムが時間とともに変化するように、長期にわたって使用されるモデルの場合、モデルも時間とともに変える必要が生ずるのでモデル作成のプロセスもくりかえされる。（Pritsker博士は、この状況を“Models evolve just as systems evolve.”と表現しておられた。）

B) 実システムの時間にとまなう変化を考えなくてもよい場合でも、多くのシミュレーション・プロジェクトでそうであるようにモデル作成者とシステムを知る現場の人やモデル・ユーザとが別である場合には、モデルの前提条件があらかじめすべて明らかになっていることはまれである。したがって、暫定的なモデルの動きや結果を見ながらモデルを修正するというのが普通である。

Aはモデルの保守・修正とからんで大切な問題であるが、本稿ではふれないこととし、以下ではBの視点から議論を進める。図7のモデル作成手順はアニメーションを別にして考えればことに目新しいものではない。しかしアニメーションを用いることによってこのプロセスの中身が一変することが予想される。アニメーションを用いない場

合にとりうるモデルの前提条件確認の方法には①文章化されたモデルの条件を列挙する，②モデルのブロック図等を見せる，③モデルの動きをトレース等で見せる，④モデルから得られた数値結果を示す，等が用いられてきたが，これらの前提条件確認方法には以下の問題がある。

- 現場の人とモデル作成者とは、使う「言葉」や基本的「考え方」が違うことがあるため、文章化された条件の列挙だけではコミュニケーションが困難なことが多い。誤った条件を発見することは比較的容易であるが、見過ごされた条件を見つけることは非常にむずかしい。

- GPSSのブロック図やSLAMのネットワーク・モデルのように比較的わかりやすいモデルの場合でさえも現場の人や意思決定者にいきなりモデルを見せることには無理がある。

- 数値結果，ことに統計結果だけを見ても条件が正しいか否か判断できない場合が多い。

システムの詳細をよく知る現場の人は、シミュレーションやモデルの知識をもたないことが多いので、少なくとも初期の段階では軸象的な議論には乗ってきにくい。そこで、アニメーションを活用してなるべく具体的な動きや結果を見せることによって、モデルがシミュレートしようとしているシステムを正しく表現しているか否かの確認を効率的に行なうことが可能となる。このように考えると、アニメーションの効用はアニメーションが現場の人や意思決定者とモデル作成者との「共通の土俵」を提供してくれることになる。

(2) 結果提示・説明における効用とリスク

結果提示・説明におけるアニメーションの効用については説明を要さない。しかし、どのようにアニメーションを使ったら一番効果が上がるかは今後の重要な検討課題となる。

結果提示・説明におけるアニメーションの効用は同時にアニメーションのリスクにつながる。すなわち、いわば1枚の写真を見て誤った印象をもってしまいう危険があるように、部分的なアニメー

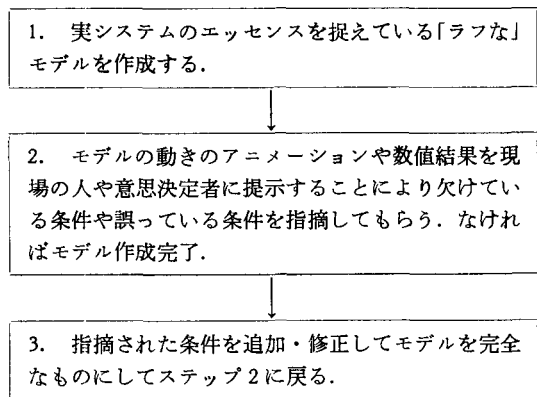


図 7 モデル作成のプロセス

ションだけを見て、統計的な数値にもとづかず誤った印象をもったり誤った判断を下すリスクがあることが指摘されてきた。

しかしながら、アニメーションのリスクを上回る効用が期待できるという見解が支配的である。そこでアニメーションのリスクが表面化しないような使い方を考える必要がある。その一法として、

- 1) ある程度の期間、ユーザにモデルの動きをアニメーションで見ってもらう
- 2) 1)によってユーザにシステムの挙動に関する「感触」を得ってもらう
- 3) シミュレーションのユーザである意思決定者が決定の根拠となる統計の数値をシミュレーション技術者に要求するようしむける
- 4) シミュレーション技術者がシミュレーションから得られた統計的裏づけのある結果をユーザに提供し、これをもとにユーザが最終的な判断を下す。

というアプローチで結果を説明・提示する方法が考えられる。

アニメーションという共通の土俵を介して、モデル設計者・現場の人・意思決定者相互間のコミュニケーションがよくなり、各人の意思決定への参加意識が高まり、一層アイデアや提案が生まれやすい環境が作られる。さらに、シミュレーション・アニメーションの普及はシミュレーションの動きの提示にとどまらず、システムの動きを見

ることにより得られるシステム最適化へのアイデアにつながり、これが他のOR技法の利用・普及・発展につながる可能性を十分秘めている。

4. おわりに

今回実施した調査の結果、シミュレーションが最もよく使われているOR技法の1つであることが再確認された。シミュレーションによる分析を必要とする問題の増加とシミュレーション作業の効率化を可能にする強力なシミュレーション言語の普及・発展によってシミュレーションが今後ますます使われるようになると、言語の特性分析やアニメーションの有効活用法等の今まであまり検討されなかった新しい研究課題が生まれてくる。

〔謝辞〕 アンケート調査にご協力いただいた各

機関の関係者の方、および菅沼陽史君に謝意を表します。本研究は早稲田大学特定課題研究(61B-8)の助成を受けました。

参考文献

- [N1] 中西俊男, 「コンピュータシミュレーション」, 近代科学社, 1977年
- [N1] 中西俊男, シミュレーション言語の新しい動向, システムと制御, 29(11), 706-713, 1985
- [S1] Catlog of simulation software, *Simulation*, 47(4), 152-165, October 1986; 48(2), 69-73, February 1987.
- [V1] 矢崎義行, 他, 「パソコン版ダンブトラック運行シミュレータの開発と走路区間のモデル化」, オペレーションズ・リサーチ, 1987年5月号

●ミニミニ●

●OR●

トラブル・シューティング型の問題解決 (I)

世に行なわれる問題解決法には、いろいろな流儀がある。わがORもその1つと考えるが、もう少しの絞って、トラブルが起こったときに、その原因をつきとめるというタイプのものがある。トラブル・シューティング型の問題解決法である。最近、筆者自身これを体験した。この顛末を述べて参考に供したい。

ワード・プロセッサ、俗にワープロと称する代物、技術的には、まだ到底完成の域には到達していない商品と見るが、その効用はめざましい。筆者の所でも一昨年末、A社製の熱転写式印字装置のついたポータブル・タイプのもを購入した。当時10万円程度の機械である。以来1年半、大いに利用した。

ところが、新しく紙をセットすると、その第1行から第2行にかけて印字が不完全なことがたびたびおこる。それも常というわけではなく、ときどき起こるのである。筆者は最初紙に原因があるものと考えた。紙の上のりなどが付着してそのままかわくと、見た目にはわからなくてもインクがのらない。そんなトラブルだと考えたのである。筆者が使っていた紙はワープロ用として市販されている、大

手製紙会社B社の製品で、白土を塗布したやや厚手のつやのある紙であった。塗布の工程に問題があったのではないかと。

紙を購入した売店を通じて連絡すると、販売元のC社から回答があった。「それは紙のせいではなく、機械のせいだと思う」とのこと。もっとも、この販売元はB製紙会社から紙を購入して、断裁機(ギロチンという)でA4版の大きさに裁断、袋づめて販売している会社である。

とにかく今度はワープロのメーカーに連絡した。営業担当者のいわく「そんなもの、リセット・ボタンを押せば直りますよ」とのこと。

大概のワープロにはメモ機能というものがある。普段よく使う語句、自分の住所氏名等はこれによってワープロに記憶させておけば、ごくわずかのボタン操作によって取り出せるようになっている。筆者のワープロにもこの機能がついているのだが、リセット・ボタンを押すとメモの内容が消されてしまうようになっている。だから、筆者としてはリセット・ボタンは押したくない。しかし、トラブルが防げるのならと思い、止むを得ずボタンを押したが、事態は一向に改善されなかった。(P.280へ)