新動向のシミュレーション言語—SLAM (1)

森戸　晋

1. はじめに

OR技法の利用度を考えた場合，数理計画法（主として線形計画法）とならんでシミュレーションが他の諸技法（ここで統計的方法は確立した一分野としOR技法からはずして考える）を大きく引き離していることは各種調査の一致した結果と言えよう．ORワーカーとしてわれわれが考えるシミュレーションは

A．離散型(discrete)
B．連続型(continuous)

に大別できる．離散型，より正確には離散事象型(discrete event)のシミュレーションのために開発されたシミュレーション言語は，さらにモデル化の考え方により

A 1）「もの」中心—モデル化しようとするシステム内を「もの」（以後，構成要素あるいは単に要素と呼ぶ；entity)がどのように動くかを記述

A 2）「事象」中心—システム内の状態に何らかの変化をもたらす出来事（事象；event)が発生した時に，システムの状態がいかに変化するかを記述

と分類できる(この他，「活動」中心のものもあるがその数は少ない)．GPSSに代表される「もの」中心のモデル化は一般にわかりやすく説明も容易でモデル化に要する時間短縮につながることが多いが，問題によっては柔軟性に欠けるという欠点がある．一方，SIMSCRIPTに代表される「事象」中心のモデル化は柔軟性に富み，複雑な問題への対応が容易で玄人好みと言えるが，一般に言語の理解により時間がかかる．本稿では，これら2つのアプローチによる離散型モデル化機能をあわせもつばかりでなく，DYNAMOのような連続型モデル化の機能を備え，しかもそれら3機能の組合せを許すという新しいシミュレーション言語SLAM(Simulation Language for Alternative Modeling；開発 Pritsker & Associates, Inc., West Lafayette, Indiana)を紹介することにしよう．本稿は2回に分けて，今回はSLAMの離散型モデル化機能を中心に話を進め，次回は連続型および混合型モデル化機能を紹介し，例を示すことにしよう．

2. SLAMソフトウェア

SLAMは，Pritsker教授(Purdue University)開発による一連のFORTRANにもとづくシミュレーション言語中最新のものである．Pritskerはこれまでに，事象中心の離散型言語GASP-II(1969)，これに連続型

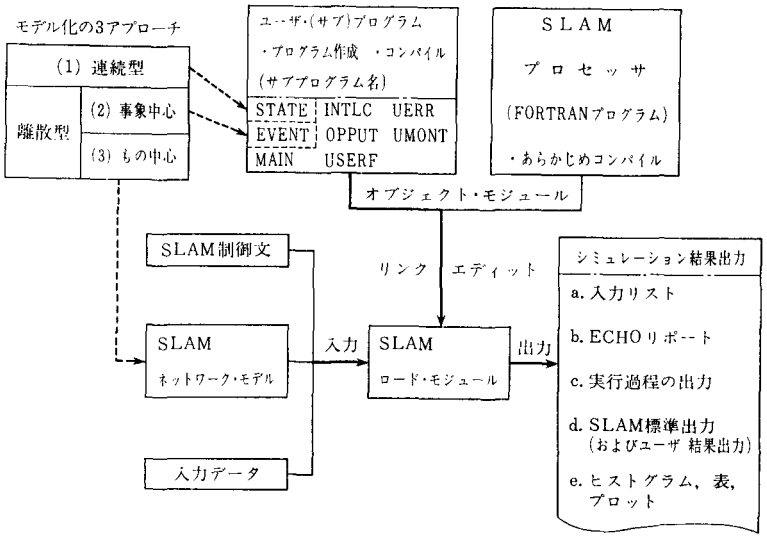


図 1　SLAMの構成

もりと　すすむ　筑波大学　社会工学系

モデル化の機能を加えたGASP-IV(1973,[2]),GERT (Graphical Evaluation and Review Technique)の考え方を待ち行列型問題に応用したQ-GERT (1977, [3]) 等，多数のシミュレーション・ソフトウェアを開発し市場に送り出しているが，ここでとりあげるSLAMは，GASP-IVとQ-GERTを統合・改良したものと考えることができる．SLAMは，GPSS, SIMSCRIPT, DYNAMOとまではいかないまでも，1979年の発表以来急成長をとげ米国を中心に現在およそ200(会社，官公庁，大学等)のユーザがあるとのことである．なお，1981年4月にやや機能拡張されたSLAM-IIが発表されたが基本的な機能に関してはSLAMと同じである．

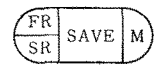
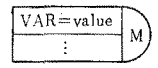
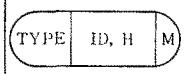
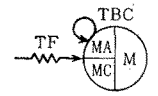
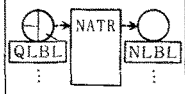
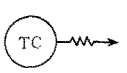
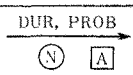
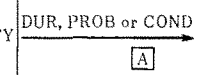
SLAMの「もの」中心の離散型モデル化は，構成要素が動いてゆくプロセスをノードとアークからなるネットワーク(GPSSのブロック・ダイアグラムに相当)として表現することによってなされ，他方，「事象」中心の離散型ならびに連続型モデル化では，SLAMプロセッサと呼ばれるFORTRANサブプログラムをユーザが書く．一般にユーザからのインプット（計算機のジョブ制御文を除く）は，図1に示すように，①SLAM制御文，②SLAMネットワーク・モデル，③ユーザ・プログラム，④入力データ，の4種に大別でき，①以外はオプショナルな入力である．なお，MAINを除くユーザ・プログラムを書かない場合には，結合・編集時のエラーを防ぐため，ダミー・ルーチンを加えておかなければならない．

3. 「もの」中心の離散型モデル化 ——SLAMネットワーク・モデル

SLAMネットワーク・モデルは，表1，表3に示すおよそ20の特定の機能をもつノードとアークからなるネットワークとして表現される．GPSSのブロック・ダイアグラム同様，ノード，アークにはいくつかのパラメータがあり，これらに値を与えることにより各ノード，アークが具体的に規定される．GPSSとの主な違いはSLAMではアークが活動(activity)を表わしたり(GPSSのADVANCEブロックに相当)，要素のシステム内での動きを規定(GPSSのTRANSFERブロックに相当)することにある．

SLAMにおける活動は，さらに(a)(基本的に)待ち行列(QUEUEノード)直後に位置する「サービス」(service)活動，と(b)その他の「一般」(regular)活動とに分類される．サービス活動に対してサーバー（サービスを提供する人またはもの；server）の数を適当な整数値に限定することができるが，一般活動に対しては無限のサーバーが想定され能力に限度を設けられない．一般活

表1 SLAMネットワークの基本ノード/アーク一覧

ノード/アーク名*	記号	機能
ACCUMULATE		SR個の要素をまとめてひとつの要素とする．出てくる要素の属性はSAVE規則にもとづき決まる．
ASSIGN		要素の属性またはSLAMシステム変数VARに値（value）を付与する．
COLCT		TYPEに指定されたSLAM変数等に関する（観察にもとづく）統計情報を収集し，ヒストグラムHを作る．
CREATE		指定された到着間隔パターンTBC（初期到着時刻TF）にしたがって最大MC個の要素を生成する．
GOON		FORTRANのCONTINUE文に相当し，何もしない（go on）ノードで，活動を直列にならべたいとき等に使う．
MATCT		MATCHノードに先行する複数個のQUEUEノードにNATR番目の属性値が一致する要素が1つずつたまるまで要素を先行するQUEUEノードにとどめておく．
QUEUE		QUEUEノードに続くサーバーが空くまで要素をファイルIFLに待たせておく．(MATCHノードとの組合せで使われることもある．)行列の初期長さICや保有能力QCを指定でき，能力一杯の場合は要素の行先（BALKing)や，先行するサービス活動の停止(BLOCKing)も指定できる．
SELECT		複数の待ち行列（QUEUE）から行列選択規則QSRによりひとつを選択したり，数種のサーバーからサーバー選択規則SSRに基づきサーバーを決定する．
TERMINATE		要素をモデルから消し去り，TC個の要素がこのノードに到着した場合にはシミュレーションを終了する．
ACTIVITY サービス (Service)		指定された長さDUR（確率分布からのランダム・サンプルの場合もある）の活動をN人の同一機能をもつ並列サーバーを想定し行なう．Aは活動番号を示す．
ACTIVITY 一般 (Regular)		指定された長さDUR（同上）の活動（活動番号はA）を行なう．条件CONDが指定されている場合は条件が満足される場合に限ってのみ活動を行ない，確率PROBが指定されているときは，確率にしたがって採用する活動を決定する．

* キーワードは，はじめの3文字であり，以下は省略可能．

表 2　SLAM変数とネットワークの状態を示す関数

	変数／関数名	説　　明
SLAM変数	ATRIB(NATR)	当該要素の第NATR番目の属性
	XX(I)	第I番目のSLAMシステム変数
	SS(I)	第I番目の状態変数
	DD(I)	SS(I)の導関数
	II	整数システム変数（ATRIB(XX(I))のような表現が許されないので，II＝XX(I)とした後ATRIB(II)とするなど間接インデックスとしてしばしば用いられる）
ネットワークの状態を示す関数	NNACT(I)	活動Iを現在実行中の要素数
	NNCNT(I)	活動Iを現在までに完了した要素数
	NNGAT(GLBL)	ゲイトGLBLの現在の状態（0＝開，1＝閉）
	NNRSC(RLBL)	現在利用可能な資源RLBLの量
	NNQ(IFILE)	ファイルIFILE内に現存する要素数

動については後に述べるように条件や確率を定めて要素が複数個の代替的活動のうちどれを採るかを決定したり，要素のシステム内の動きを規定できる．なお，活動の長さDURが0かつ確率・条件等を指定しない場合には，ノード間にいちいちダミーのACTIVITY文を挿入する必要はない．DUR＝0で条件あるいは確率が指定されている場合には，一般活動が要素の動きをコントロールしていると考えればよい．条件CONDの書式は，FORTRANの条件付IF文の条件，たとえば，ATRIB(2)．GT．XX(3)の形をとる．

SLAMモデルで使用される変数は表2上半分に示す5種であるが，このうちSSとDDは連続型モデル用であり，離散型モデルではATRIB，XXがよく用いられる．ATRIB(I)は要素の第I属性を示し，あるノードである要素の属性を変えても他の要素の属性には影響をおよぼさない．これに対してXX(I)のほうは(SS, DD, IIも同じ)グローバルな変数であり，あるノードでXX(I)の値を変えてしまえば，どのノードでXX(I)を調べようがXX(I)には変わった後の値が入っている．なお，ASSIGNノードの右辺や，活動の長さ(時間)，あるいは要素の動きを規定する条件設定に当って，これら5種のSLAM変数のほかに，ネットワークの状態を示すFUNCTION（表2下半分参照；これらは，SLAMソフトウェアではFUNCTIONサブプログラムである）の値や，数多くの分布からのランダム・サンプルを用いることができる．分布としては，指数，一様，ワイブル，三角，正規，対数正規，アーラン，ガンマ，ベータ，ポアソンを考えることができるほか，ランダムサンプルを得るための擬似乱数には10本の流れがある．

要素の動き(分枝)の規定

さて，SLAMネットワーク・モデルのひとつの特徴は要素のバラエティに富んだ動きがネットワークのアーク(正確には一般活動)とノード記号右端(対応するステートメントでは最後のパラメータ)に位置する「M番号」(M-number あるいは "max-take" number)により簡単に表現できる点にある．ここに，M番号とはあるノードからの分枝にあたって最大いくつの分枝が許されるかを示すが，ノードの中にはQUEUE，MATCHのように論理的理由からパラメータにM番号を含まないものもある．ノードから複数のアークが出ている場合，要素は分枝アーク上の条件・確率および先行するノードのM番号の値により具体的な分枝（branching）が規定される．

分枝には無条件分枝・条件付分枝・確率的分枝がありこれらの組合せも可能である．あるノードのM番号＝1とし，このノードから1本の無条件アークが伸びている場合，ノードから出てきた要素は常にこのアークを通って後続のノードに動いてゆく．この場合，$M=\infty$としても要素の動きはまったく変わらない．ちなみに，M番号の既定値(default value)はパラメータMを含むすべてのノードに対して無限大である．

さて図2(a)のようにM＝3とし，3本の無条件アークが出ている場合には，ノードから出てくる要素が3つにデュプリケートされ，各アークに1つずつ要素が送り出される．一方，図2(b)のようにM＝1に対して3本のアークに確率p_1, p_2, p_3($p_1+p_2+p_3=1.0$)が付与されている場合は，確率にもとづいてランダムにどの(ひとつの)アークに分枝するかが決定される．図2(c)では，M＝2で3本のアークに条件c_1, c_2, c_3が定められている．(SLAMプロセッサへの入力は，ネットワーク図そのものではなく，これに対応するネットワーク・プログラムである．条件のチェックにあたりプロセッサは分枝ノード直後に位置する一連のACTIVITY文の順序にしたがって条件の判定を進める．ここでは，これらのACTIVITY文の順序とネットワーク図のアークの上下関係が一致するものと仮定する．)このとき，まず条件c_1が満足されるかを調べ，条件が満たされる場合はこのアークに

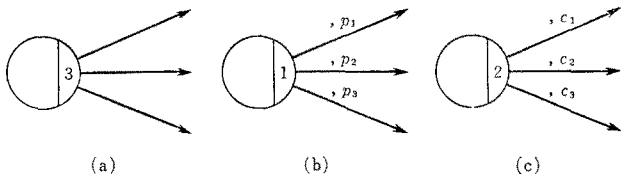


図 2　各種分枝

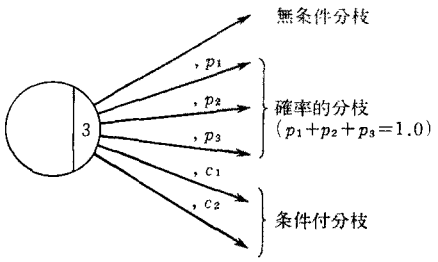


図 3 分枝の組合せ

分枝する．いずれにせよ，$M=2$ なので，次に条件 c_2 を調べ，満足される場合は第2のアークに分枝する．c_1，c_2 双方が満たされた場合には，M 番号に等しい分枝がされたので c_3 のチェックはされない．c_1，c_2 のいずれか一方のみ満たされた場合，c_1，c_2 のどちらも満たされない場合には，c_3 のチェックをし第3のアークに分枝するか否かを決定する．なお，この場合実際に採られる分枝の数は2，1，あるいは0が考えられ，0の場合には当該要素はシステムから消滅する．

前述のように，これら3種の分枝を組み合わせることもできる．図3（アークの上下関係は ACTIVITY 文の順序と一致すると仮定）では，無条件分枝と確率的分枝から各1計2分枝が常に選択され，これに加えて条件付分枝から c_1，c_2 に応じて最大ひとつの分枝がとられる．

資源とゲイト

表1の基本ノードに加えて SLAM では資源（RESOURCE）およびゲイト（GATE）なる概念が考えられている．サービス活動のサーバーはいわばサービス提供のための固定的資源と考えることができるが，これに対して非固定的，流動的資源を考えると都合良いことが多い．たとえば銀行窓口のサーバーにとってサービスを提供するためにソロバンが必要であるとしよう．ところが限られた数のソロバンしかなく，しかも窓口業務以外でも「稀少資源」たるソロバンを共同利用しているとなれば，顧客はサーバーとソロバンがそろってはじめてサービスを受けられる．このような状況のモデル化に当っては，ソロバンを非固定的資源と考えればよい．

同様に，交通信号のように，赤の時は要素が停滞し，いったん青になると流れ出すというメカニズムが便利なことも多い．このためにゲイトが考えられている．表3は資源およびゲイトに関連するノードをまとめたものである．なお，資源とゲイトに関しては FORTRAN の DIMENSION 等のように，あらかじめそれらの使用を「宣言」し，かつ初期状態を指定する必要がある．宣言

表 3 資源・ゲイトに関連するノード一覧

ノード名*	記　号	機　　能
ALTER	RLBL / CC / M	資源RLBLの保有量（capacity）をCC個変化させる．（CCが負の場合は減少）
AWAIT	IFL / RLBL/UR または GLBL / M	利用可能な資源RLBLの現在量が所要資源量URに満たない場合，または，ゲイトGLBLが閉っている場合，要素をファイルIFLに待たせておく．URの資源があるときには，これらが要素に与えられる．
CLOSE**	GLBL / M	ゲイトGLBLを閉める．
FREE	RLBL / UF / M	AWAITまたはPREEMPTノードで付与された資源をUF個返す．
OPEN	GLBL / M	ゲイトGLBLを開ける．
PREEMPT	IFL / PR / RLBL / NATR / M → SNLBL	AWAITないしは優先度（NATR番目の属性および規則PRにもとづき決定）の低い他のPREEMPTノードで資源を与えられた要素から資源RLBLを（1個）横取りする．資源を横取りされた要素は，ラベルSNLBLのついたノードへゆく．なお，横取りが成立しない場合は，要素をファイルIFLに待たせておく．

* キーワードは，はじめの3文字．
** 対応するサブルーチン名はCLOSX．

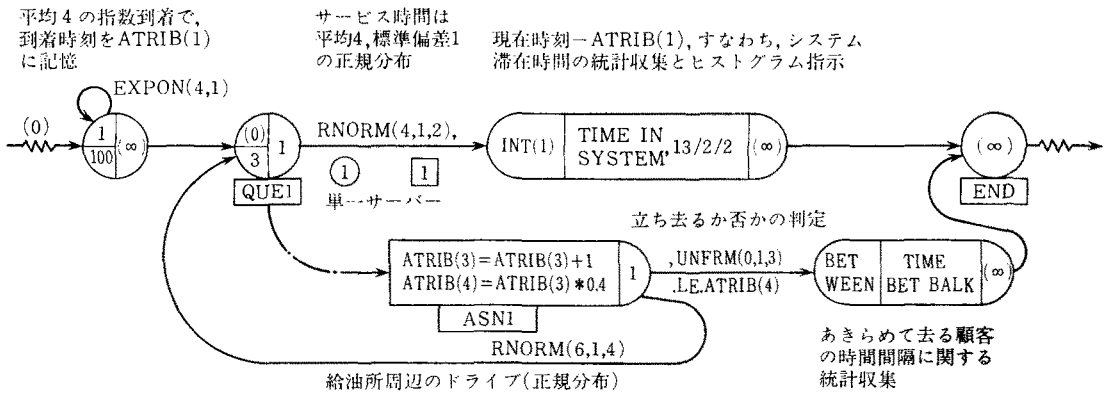


図 4 給油所問題のネットワーク・モデル

表 4　SLAM ファイル操作サブプログラム一覧

	サブプログラム名	機　　能
ファイルの基本操作	FILEM(IFILE, A)	ファイルIFILE（以下単にIFILE）に属性A（ベクトル）をもつエントリを挿入する（実際の挿入位置はSLAM制御文PRIORITYの指定により定まる）
	RMOVE(NRANK, IFILE, A)	IFILEのNRANK番目のエントリをファイルから除去し，属性をAに入れる．
	COPY(NRANK, IFILE, A)	IFILEのNRANK番目のエントリの属性をAに入れる（エントリはそのまま）．
	SCHDL(KEVNT, DTIME, A)	時刻DTIMEに発生予定の事象番号KEVNTの事象を属性Aとともに事象カレンダーに登録する．
	*NFIND(NRANK, IFILE, NATR, MCODE, X, TOL)	IFILEのNRANK番目以降のエントリで，NATR番目の属性値がMCODE, X, TOLで指定される条件を満たす最初のエントリが何番目のエントリかを与える（見つからない場合は0）．
その他の応用操作（一部略）	*MMFE(IFILE)	IFILE内の先頭（最後尾）のエントリのポインタを与える．
	*(MMLE(IFILE))	
	*NSUCR(NTRY)	ポインタNTRYをもつエントリの後続（先行）エントリのポインタを与える．
	*(NPRED(NTRY))	
	*LOCAT(NRANK, IFILE)	IFILEのNRANK番目のエントリのポインタを与える．
	*NNQ(IFILE)	IFILEに存在するエントリ総数を与える．
	ULINK(NRANK, IFILE)	属性を書き写したり消去せずにIFILEのNRANK番目のエントリをファイルから除去し，次に
	LINK(IFILE)	このエントリをIFILEに挿入する．

＊ は関数サブプログラムを示す．

はネットワークとは直接つながっていないのでブロック（資源ブロックおよびゲイト・ブロック）と呼ばれる．

簡単なネットワーク・モデルの例

図4は給油ポンプ1台かつ車3台が待てる給油所モデルである．顧客が到着して待つスペースがない場合には顧客はある確率であきらめて立ち去ったり，給油所周辺をドライブしてもどってきたりする．このモデルでは，サービスを受けた顧客の（第1回目の）給油所到着からサービス終了までの総所要時間と，あきらめて立ち去る顧客の時間間隔の統計をとることとした．なお図4中カッコ内のパラメータ値は既定値であり，あえて指定しなくてもよい．また任意のノードにラベル（FORTRANの文番号に相当）をつけることができ，これによりアークの飛び先が定められる．

4.「事象」中心の離散型モデル化

SLAMの「事象」中心のモデル化では，ユーザが事象ルーチン(event routine)，すなわち事象発生時にシステムがいかに変化するかを記述するFORTRANサブルーチンを書き，これがSLAMプロセッサに呼ばれながらシミュレーションが進行する．たとえばSIMSCRIPT同様，SLAMプロセッサには事象の発生をコントロールするタイミング・ルーチンが組み込まれており，事象カレンダ(event calendar)と呼ばれるファイルに将来事象を発生時刻順にならべながら次に発生する事象を決定してゆく．事象ルーチン(Subroutine) EVENT(I)の引数Iは事象番号(event code)と呼ばれ，どの事象がおこったかをサブルーチンEVENTに伝える役割をする．簡単は単一サーバー待ち行列問題を考え，事象1を顧客到着事象（ARRIVE），事象2をサービス終了事象(DEPART)とすると，サブルーチンEVENTを：

```
      SUBROUTINE EVENT(I)
      GO TO (10,20), I
   10 CALL ARRIVE
      RETURN
   20 CALL DEPART
      RETURN
      END
```

と書き，サブルーチンARRIVE，DEPARTでそれぞれの場合のシステム変化を記述すればよい．この際，事象ルーチンの中で将来事象をつぎつぎと発生させ，これを事象カレンダに加えたり，要素を待ち行列等に相当するファイルにしまい込む，ファイルからとりだす等，いくつかの基本的なファイル操作が繰り返し必要となる．そこでSLAMプロセッサには表4に示すファイル操作（事象カレンダ・ファイルの操作を含む）サブプログラム（一部は関数サブプログラム）が組み込まれており，ユーザは事象ルーチンの作成に当ってこれらを自由に呼ぶことができる．なお，サブルーチンEVENTの他に，ユーザが書けるサブルーチンには，

- シミュレーション実行前の初期条件設定ルーチンINTLC
- シミュレーション終了時のSLAM標準出力(次回説明)以外のユーザー独自の結果出力ルーチンOTPUT

などがある．

なおネットワーク・モデルにおける資源やゲイトの概念は事象中心のモデル化にも適用可能である．実際，表3のAWAIT, PREEMPTを除く4つのノードはユーザが呼べるサブルーチンである．このほか，30近くのリポート・ジェネレーション・サブルーチンや統計サブプログラム（各種統計量の平均・標準偏差，最大，最小等を与える）をユーザ・プログラム内で呼ぶことができる．これらを有効に利用することにより貴重な情報をきわめて簡単に手に入れる可能性がひらけていることはSLAM

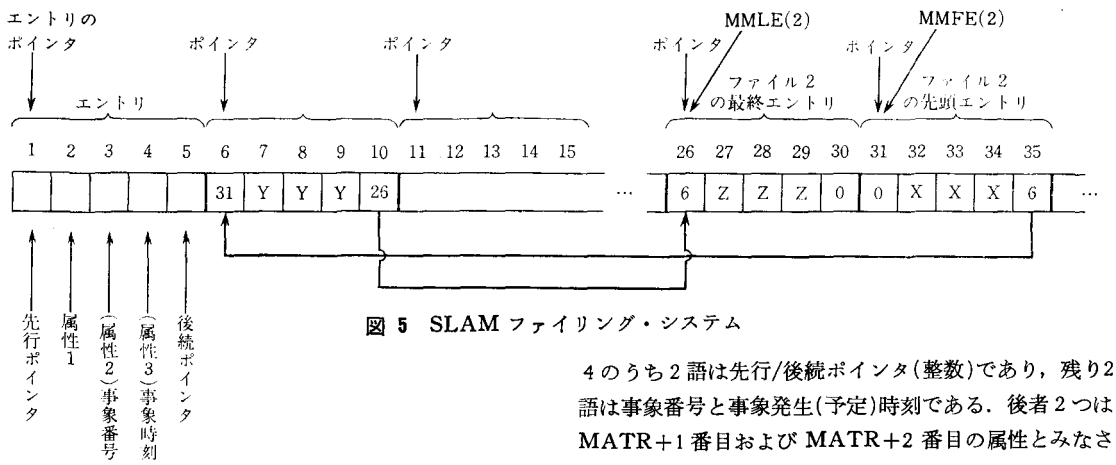


図 5 SLAM ファイリング・システム

の強みであろう.

SLAM 離散型シミュレーション・ファイリング・システム

SLAM ファイリング・システムでは，ファイルを

(A) 事象カレンダ・ファイル

(B) QUEUE, AWAIT, その他，構成要素がシステム中で滞留するときに要素をしまっておくユーザ・ファイル

(C) 未使用ファイル

と3つに大別し，二重リンク・リスト・ポインタ・システム(Double link list pointer system)により(高速)記憶場所の動的割当(dynamic allocation) を行なっている．すなわち上記A～C全体に相当する巨大な整数1次元配列 NSET が存在(配列 NSET は計算機容量により大きくも小さくもできるが，Pritsker から提供されるソフトウェアでは一応5000でディメンジョンをきってある) し，しかもこれが FORTRAN の EQUIVALENCE 文により実数1次元配列 QSET と記憶場所を共有するようにしてある．なお，離散型モデルではシステム内の要素は，イ) 現在事象発生時の処理中，ロ) 事象カレンダ・ファイル内で事象発生時刻が来るのを待機中，あるいは，ハ) ユーザ・ファイルの中に停滞中，のいずれかの状態にあることに注意されたい．

さて一般に各ファイルに入るエントリー (entry；要素に関する情報に相当)の順序は配列 NSET/QSET 内の物理的順序とは一致せず，先行および後続ポインタ(predecessor/successor pointer)によるチェイン構造を利用した論理的順序づけをもとにしている．図5に示すように，ファイリング・システム内の各エントリは，NSET のいくつかの記憶場所を占有する．正確には各要素のユーザ使用の属性の数を MATR とすれば，各要素は(MATR+4)語の記憶場所を占める．ここにプラス4のうち2語は先行/後続ポインタ(整数)であり，残り2語は事象番号と事象発生(予定)時刻である．後者2つは MATR+1 番目および MATR+2 番目の属性とみなされ，事象カレンダ・ファイルで使用される．図5のように各エントリは，先行ポインタ，属性ベクトル(MATR+2語)，後続ポインタの順に配列され，先行(後続)するエントリが存在しない場合は先行(後続)ポインタは0とする．配列 NSET の大きさを NNSET語とすれば，システム内に 同時に存在しうる 最大要素数 MNTRY は

$$MNTRY \leq NNSET/(MATR+4)$$

なる条件を満たさなければならない．なお SLAM プロセッサ，およびユーザ・プログラムでポインタをさすときには整数値配列 NSET を用い，属性をさすときには実数値配列 QSET を使用する．

各エントリポインタを当該エントリが占有するはじめの記憶場所と定義すれば，あるファイル（たとえば待ち行列ファイル）の先頭にいるエントリ(要素に対応)のポインタがわかれば後続ポインタを順次たぐってゆくことでこのファイル中のエントリを列挙でき，逆にファイル最後尾に位置するエントリのポインタがわかれば先行ポインタを使ってエントリを後からたぐることもできる．

未使用ファイルもチェイン構造により論理的に結びつけられる（未使用ファイルも物理的に1カ所にまとまっているとは限らないことに注意）が，この場合は利用可能な最初のポインタと後続ポインタさえ整えておけばよい．なお，事象カレンダ・ファイルやユーザ・ファイルからエントリが脱け出してアキができると，このアキが未使用ファイルの先頭に加えられ，これにともない利用可能な最初の位置が更新される．(類似の話に興味ある方は［1］の第5章を参照されたい)

参 考 文 献

［1］関根，高橋，若山，シミュレーション，日科技連，1976

［2］Pritsker, A. A. B., The GASP IV Simulation Language, John Wiley, 1974

［3］Pritsker, A. A. B., Modeling and Analysis Using Q-GERT Networks, Halsted Press, 1977