HOKKAIDO UNIVERSITY

Title	E-NET手法による連続-離散混合系システムの記述
Author(s)	牧野, 圭二; 栃内, 香次; 永田, 邦一
Citation	北海道大學工學部研究報告, 87, 123-134
Issue Date	1978-06-05
Doc URL	https://hdl.handle.net/2115/41471
Type	departmental bulletin paper
File Information	87_123-134.pdf

Hokkaido University Collection of Scholarly and Academic Papers : HUSCAP

北海道大学工学部研究報告
第87号（昭和53年）

Bulletin of the Faculty of Engineering,
Hokkaido University No. 87 (1978)

E-NET 手法による連続—離散混合系システムの記述

牧野圭二* 栃内香次** 永田邦一**

（昭和52年9月30日受理）

A Continuous-Discrete Hybrid System Description by the E-net Modeling Method

Keiji MAKINO Koji TOCHINAI Kuniichi NAGATA

(Received September 30, 1977)

Abstract

The E-net modeling method, which has been developed for describing a computer system and evaluating its performance, is a general modeling method for a discrete system. The method can represent an asynchronous parallel process, specify a spending time with an action, describe a data flow, a control flow, and their interaction and the change of an attribute value with an action, and so on.

In remarking on the above features and the resemblance between an E-net structure and a block diagram for analog computer programs, we show that the E-net modeling method can describe a continuous system, and more generally, a continuous-discrete hydrid system. The method by which it is shown, is by means of the correspoding relation of the block diagram element to its E-net representation, and then by means of an E-net description for the interface between a continuous subsystem and a discrete subsystem. And an example is exhibited concerning a mass-spring damper system.

As a result, it was clearly shown that the E-net modeling method is an excellent unified description for a continuous-discrete hybrid system.

1. はじめに

E-netは，本来計算機システムや計算過程の非同期並列動作の記述を目的として開発されたモデル化手法である[1]。もともとE-netの母体はPetri-netであり，E-netはそこで開発された図的表現法をもとにその解釈や記法に改良を加えながら発展してきた手法であるが，E-netに到って形式的，代数的表記法が整備された[1]。その結果，単に視覚的，直観的にモデル化対象システムの論理的相互作用が記述，把握できるということにとどまらず，対象システムの動作時特性量を表すデータの表現及びその定量的取り扱いが可能となり，シミュレーション・モデル構成手法として広く計算機システムの性能評価に使用されるようになった[1)~9)]。

E-netによる記述の特徴としては，決定性モデル化手法であること，モデルはわずか5種類

* 工学部 汎用シミュレータ施設（全学共同利用施設）
** 工学部電子工学科 電子機器工学講座

の“基本遷移図式”により記述されるにもかかわらず実際のシステム構成に即した形で表現できること，データの流れと制御の流れあるいは仕事と資源との相互作用が容易に表現できること，代数的表記法が汎用プログラミング言語による記述を考慮した形式となっていること，などがあげられる。

E-net 手法は，従来，並列動作の記述に重点を置いていたため，条件の成立，事象の生起という面から対象システムを捉える事象型のモデル化法として発展してきたが，徴標に属性値としてデータを保持できるようにしたことにより，属性徴標の動きに重きを置いてモデル化を考えるならば物流型のモデル化法としても用いることができる。このように，E-net 手法は，計算機システム等に対するモデル化法にとどまらず汎用性にとんだ離散系システムのシミュレーション・モデル構成手法と言うことができる。

我々は，E-net の持つ，非同期並列動作が記述できること，データの流れが表現できること，E-net 構造とアナログ計算機のプログラムであるブロック図との間に類似性があること，などの特徴に着目し，E-net が，単に離散系システムだけでなく，連続系システムさらに連続—離散混合系システムをも記述できる能力を有すると考え，それらを記述することを試みた。

以下，第2章において E-net の記述法の概略を述べた後，第3章において，ブロック図表現と E-net 表現との対応関係を示すことにより，実際に連続系システムが E-net で表現できることを示す。さらに第章において，連続系システムと離散系システムとの同期やデータの受け渡しを行うインタフェース部分が E-net で表現できることを示すことにより，連続—離散混合系システムが E-net で統一的に記述できることを示す。

2. E-net シミュレーション・モデル構成手法の概略[1)~3)]

2.1 基本遷移図式

E-net 手法における基本的構成概念は，事象を表す“遷移 (transition)”，事象が生起する部分的条件を表す“居処 (location)”，及び，その条件が成立していることを表すための“徴標 (token)”であり，図的表現においては，遷移は縦棒で，居処は円あるいは六角形で，それらの順序関係は矢線で記述され，徴標は居処内に置かれる黒丸で示される。従って，対象システムの構造は遷移と居処との組み合せにより，又，対象システムの動作の進行は居処から居処への徴標の移動により示される。

E-net 手法の骨格をなす遷移と居処との組み合せを行うにおいて，その結合は交互に行われるが，結合の仕方にはその他にもいくつかの制限がある。まず，一つの居処に対しては1個又は2個の遷移とだけ結合が許され，一つは居処に入る矢線によって，一つは居処から出る矢線によって結ばれる[註1]。一方，一つの遷移に対する居処の結合の仕方は Fig. 1 で基本遷移図式 (primitive transition schema) として示される5種類に制限される。これら基本遷移図式は各基本遷移型 (primitive transition type) の図的表現であって次のように解釈される。すなわち，X 遷移型，Y 遷移型は，それぞれ，ある動作系列の開始及び終了の選択[註2]を表し，F 遷移型，J 遷移型は，それぞれ，並列動作系列の開始 (fork) 及び終了 (join) を表し，T 遷移型は単にある動作が起きる

註1] 入る矢線しか持たない居処を出力居処，出る矢線しか持たない居処を入力居処，両者合わせて周縁居処 (peripheral location)，周縁居処でない居処を内部居処 (inner location) と呼ぶ。

註2] 選択を行うためだけに用いられる居処を選択居処 (resolution location) と呼び，図的表現では六角形で表す。その遷移は選択居処の状態 (註3]) に応じ“0”あるいは“1”と名付けられた通路の間の切り換えを行う。

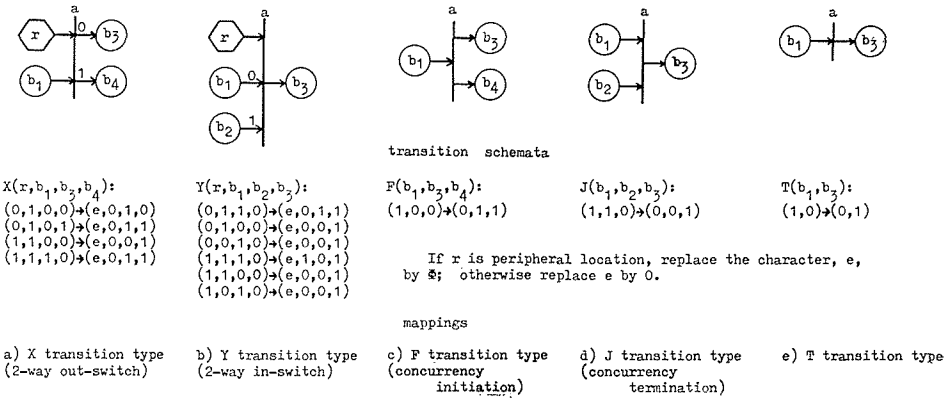


Fig. 1. Primitive Transition Schemata and their Mappings.

ことだけを表す。

対象システムの動作の進行は，上記基本遷移図式により表される各遷移において，入力居処を占有しているある徴標がその遷移を通り抜けて出力居処に移動することによって示される。その際，各遷移に対する具体的な動作生起の条件並びにそれに伴う移動の仕方は写像 (mapping) として定義される (Fig. 1)。すなわち，写像の左辺により動作前の各入出力居処の状態[註3]が，又，写像の右辺により動作後の状態が示され，各入出力居処の状態がその左辺で示される状態に一致した場合にのみ動作が生起しこれを遷移の発火 (transition firing) と言う (付録 A 参照)。なお，写像の定義に示されていない各居処の状態の組み合せでは遷移の発火は起こらず，その状態がそのまま保持される。

一方，徴標は上述のような条件の成立を示す以外に，対象システム内を移動する各種情報をも表すことができる。条件の成立だけを示す徴標を単純徴標 (simple token) と言うのに対し，各種情報を保持できる徴標を属性徴標 (attribute token) と言う[註4]。これと対応して，各居処にも条件の成立を示すためだけのものと属性値を保持できるものとがある[註5]。さらに，居処に対して注意しなければならないのは，各居処は一時点において高々1個の徴標しか保持できないということである。

2.2 E-net 構造

E-net 構造 (E-net structure) は基本遷移図式を居処を介して結合したものである。しかし図的表現のみでは記述できる情報に限度があり，又，記述できたにしても繁雑になり厳密性に欠ける恐れも生ずる。そのため，以下のような形式的，代数的定義を併用する。

$$E = (L, P, R, A)$$

ここで，　L＝居処の有限で空でない集合，

P＝周縁居処[註1]の集合，$P \subseteq L$,

R＝選択居処[註2]の集合，$R \subseteq L$,

A＝遷移の有限で空でない集合 $\{a_i\}$,

$a_i = (s, t(a_i), q)$,

註3] 居処の状態は，居処が徴標を含んでいる (full) とき "1" で表し，含んでいない (empty) とき "0" で表し，どちらかわからない (undefined) とき "Φ" で表す。

註4] それぞれ，K, $K[n]$ で示す。n は保持している属性の個数である。

註5] それぞれ，b, $b[m]$ で示す。m は保持できる属性の個数である。

ここで, s は基本遷移型,
$t(a_i)$ は遷移 a_i の遷移時間 (transition time),
q は遷移手順 (transition procedure)。

遷移時間は, その遷移が表している対象システムの動作時間をモデル化したものであり, X 及び Y 遷移型に対しては各通路それぞれに異なった時間を指定できる。

遷移手順は, 遷移の発火に伴って徴標がその遷移の入力居処から出力居処へ移動する際に起こる属性値の変化を記述するためのものであり, 次のような一般形を持つ。

$$\left[p_1 \rightarrow (e_{11};\ e_{12};\ \cdots;\ e_{1n}) : \cdots : p_k \rightarrow (e_{k1};\ e_{k2};\ \cdots;\ e_{km})\right]$$

ここで, $p_i\ (1 \leqq i \leqq k)$ は関係式を含む論理式であり, e_{ij} は Algol 風算術代入文である[註6]。遷移手順の評価は付録 B のように行われる。又, 形式的記述表現の簡略記法を付録 C に示す。

このように, E-net 構造は対象システムの論理的, 静的構造を示したものである。対象システムの状態を示すには, これに各居処の状態を与えればよい。

2.3 E-net

E-net は, 前節で述べた E-net 構造に対象システムの初期状態並びにシミュレーション環境との情報交換手段を与え, 実際にシミュレーションを行えるようにしたものである。E-net は次のように形式的に定義される。

$$\left(E,\ M_0,\ (\xi,\ \psi)\right)$$

ここで, $E = (L, P, R, A)$, すなわち, E-net 構造,
M_0 は初期徴標設定 (initial marking),
ξ は環境変数 (environment variable) の集合,
ψ は選択手順 (resolution procedure) の集合。

初期徴標設定は, E-net 構造に初期状態を設定するためのものであり, 具体的には, 各居処に初期状態を与えることによって, すなわちどのような属性値を持った徴標がどの居処に置かれるかを与えることによって示される。

環境変数は, E-net と外部とが相互作用を行えるようにするインタフェースの役割りをはたすものであり, シミュレーション環境の部分的状態を表す属性徴標としての性格を持つ。環境変数は E-net に対して全域的であり, その使用は遷移手順, 選択手順において直接引用又は定義することによって行われる。

選択手順は, 周縁選択居処を持つ X 又は Y 遷移型の遷移が擬似発火可能段階 (付録 A 参照) に達したとき, 未定義状態にあるその遷移の周縁選択居処を定義する[註7]ための手順を表すものである。その一般形は次のようであって, 評価は遷移手順の場合と同じく付録 B のように行われる。

$$\psi(r) = r : \left[p_1 \rightarrow M(r) := s;\qquad p_2 \rightarrow M(r) := 1 - s\right]^{[註7]}$$

ここで, r は周縁選択居処の名前, p_1, p_2 は関係式を含む論理式, $s \in \{0, 1\}$ である。

従って, シミュレーションの実行は, E-net 構造に初期徴標設定を行った後, 必要に応じて選

註6] 形式的記述において各居処の属性値を含む状態は次のように記述される。m 個の属性値を保持できる居処 b_i の状態を $M(b_i[m])$, 居処 b_j の k 番目の属性の状態を $M(b_j(k))$。

註7] 評価の結果が必ずしも定義状態となるとは限らない。$p_1 \vee p_2$ が T でない場合が許される。

択手順を用いて未定義の周縁選択居処を定義しながら遷移可能状態に達した遷移を次々と発火させ，徴標を移動させると共に遷移手順を実行することにより進められる。このとき，各居処の状態すなわち徴標の配置の状態が対象システムの状態を表しており，又，その時発火している遷移の個数が対象システムのその時の並列実行度に対応していると考えられる。

又，徴標は次に移るべき居処が空になるまで発火状態にならずに待たされることがある。言い換えると，徴標は遷移時間を越えて居処に留まる可能性をいつも持っている[註8]。この，徴標が居処に留まる時間は，対象システムでの待ち時間に関係付けられる重要な特性量であり，占有時間 (dwell time) と名付けられる[註9]。

2.4 マクロ・ネット

E-net においてモデル構成の基礎となる遷移型を基本的かつ単純な5種類に制限していることは，モデル化概念を簡明にし E-net の局所的解析や理解を容易にしているが，逆に，対象システムが複雑大規模になるにつれそのモデル表現としての E-net も複雑になり，モデル全体としては E-net 手法の大きな特徴である直観的，視覚的理解のしやすさが薄れる結果となっている。

マクロ・ネット (macro net) 技法は，本来 E-net モデル中で記述されるべき部分を別個に取り出し前もって基本遷移型による形式的定義を与えておくことによりマクロ化し，単一記号を与え，通常の遷移あるいは居処と同様に取り扱える[註10]ようにする技法である。これにより表現上の便宜をはかると共にモデル全体の直観的把握を助ける役割をはたしている。従って，マクロ化を行う場合，その対象としては機能としてまとまり意味も直観的に捉えやすいものを選ぶことが重要である。

なお，Fig. 2 によく使われるマクロ・ネットの例を示し，Fig 3 にそのいくつかについて定義の例をあげる。

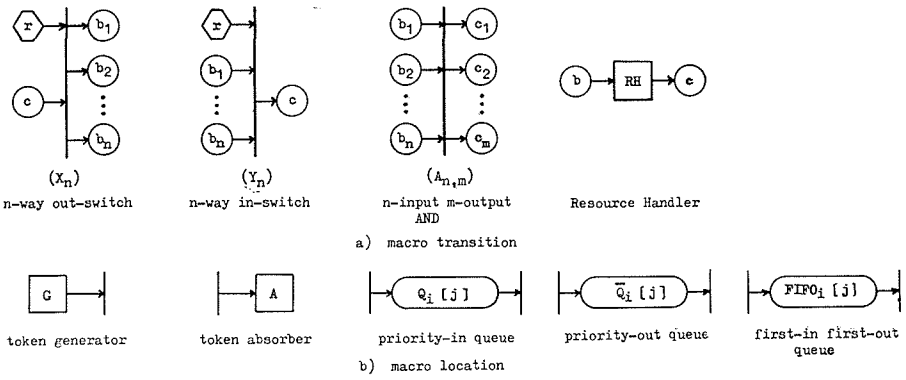


Fig. 2. Examples of Macro Nets.

註 8] 遷移の発火が入力居処だけでなく出力居処にも依存していること，及び，各居処は一時点において高々1個の徴標しか保持できないことによる。

註 9] ある徴標 K が居処 b_i に留まる時間の総量を徴標占有時間 (token dwell time) 又は居処 b_i における徴標 K の占有時間と呼び $d_k(b_i)$ で表し，シミュレーション実行時を通して各徴標が居処 b_i に留る時間の総量を居処 b_i の占有時間と呼び $d(b_i)$ で表す。

註 10] 遷移として定義されているマクロ・ネットをマクロ遷移 (macro transition)，居処として定義されているものをマクロ居処 (macro location) と呼ぶ。

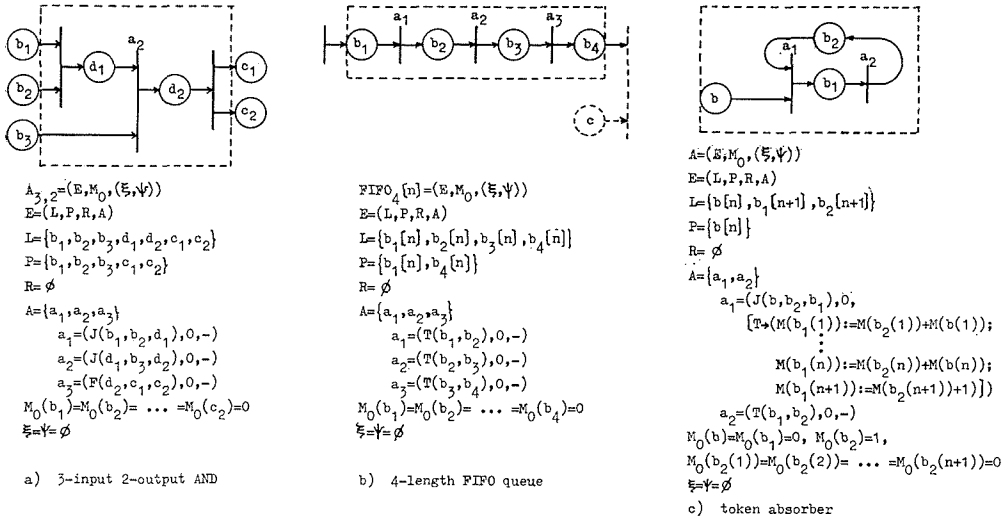


Fig. 3. Examples of Macro Net Difinitions.

3. E-net による連続系システムの記述[10)]

3.1 連続系システム

連続系システムとは，時間の流れの中において各事象が連続的に変化していると捉え抽象化したもので，通常微分方程式という数学モデルの形で表現される。このようにして得られる微分方程式は一般に高次多元であり，解析解を得ることは通常困難でそのふるまいを調べるには数値解によらざるを得ない。

微分方程式の数値解を得る手段としては従来からアナログ計算機がよく使用され，ディジタル計算機の進歩につれこれも非常によく用いられるようになっている。しかし，どの方法をとるにしても多元の偏微分方程式からそのまま数値解を得ることはできず，1独立変数を除いて他の独立変数について差分化を行い連立常微分方程式に変換する必要がある。さらに，高次の常微分方程式は一次の連立常微分方程式に還元することができることを考え合わせると，現在，連続系システムとは，次式の形で与えられる非線型連立一次常微分方程式を骨格として組み立てられ記述されているシステムと考えることができる。

$$\frac{dx_i}{dt} = f_i(x_1, x_2, \cdots, x_n, t) \qquad (i=1, 2, \cdots, n)$$

上記手段を用いて解を得ようとする際，システムの構造すなわち方程式間あるいは変数間の関係等の解析にブロック図表現が多用される。これは，ブロック図表現が連立常微分方程式との対応が良く視覚的にも理解しやすいことに加え，連立常微分方程式の連続的数値解を得ることができるアナログ計算機のプログラムとしてほぼそのまま使用できることによる。

我々は，このブロック図表現と E-net 構造の図的表現とが類似していること，又，E-net は非同期並列動作が記述できデータの流れとデータ値の変更が表現できることに着目し，E-net により連続系システムが記述できることを確認した。しかし，この場合，E-net はあくまでも事象型離散系システム記述として発展してきたこと，E-net の実行はディジタル計算機で行われることが制限となり，連続系のディジタル・シミュレーションと同様に積分はきざみ幅 Δt を持つ数値積分と考えざるを得ない。以下この数値積分を “INT” という名称で用いる。

次節において，ブロック図表現における各要素とそれの E-net 表現との対応関係を示し，ブロック図が E-net で表現できることすなわち前述の意味での連続系システムが E-net で表現できることを示す。

3.2 ブロック図と E-net との対応

ブロック図表現と E-net 表現との対応関係を見るために例として Fig. 4 a) のブロック図を考える。このようなブロック図を構成する機能要素は，データの入出力関係に着目するならば次の 4 種類に分けて考えることができる。

1) 積分要素 (1 入力, 1 出力, 1 初期値)
2) 関数要素 (1 入力, 1 出力) 例えば，係数器，関数，非線型要素 (リミッタ等)
3) 演算要素 (2 入力, 1 出力) 例えば，加・減・乗・除算
4) 分岐要素 (1 入力, 2 出力)

データの流れにのみ着目するならば，それを徴標の移動に置き換えることにより，積分及び関数要素は T 遷移型で，演算要素は J 遷移型で，又，分岐要素は F 遷移型で，それぞれ表すことができ，積分要素における初期値は初期徴標設定として与えることができる。従って，各機能要素を通過するさいに受けるデータの変更が表現できるならばブロック図の各機能要素は E-net で記述できることになり，それらを結合することによってブロック図に対応する E-net 表現が得られることになる。

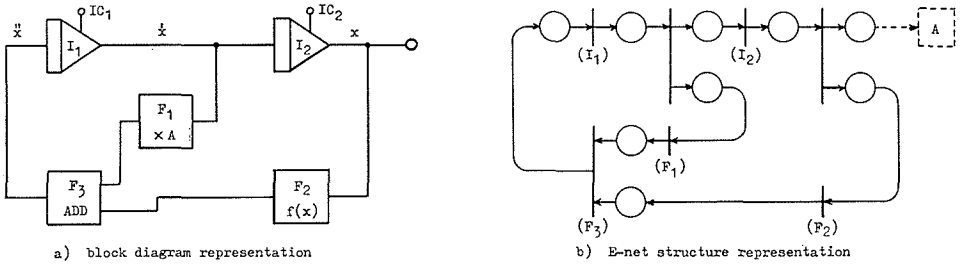


Fig. 4. Example of $\ddot{x}=A\dot{x}+f(x)$.

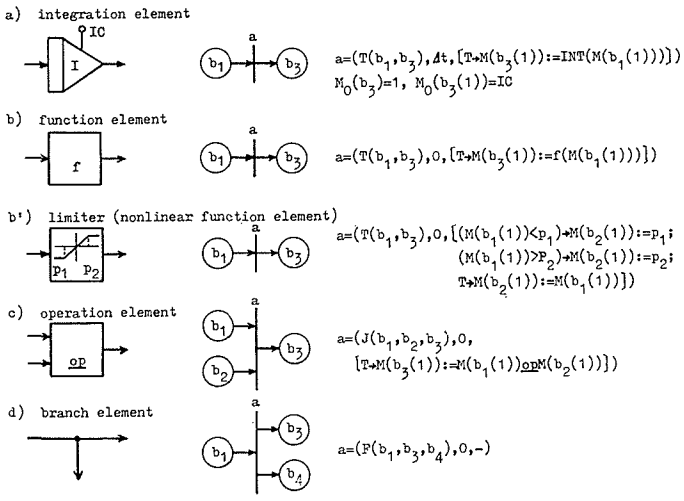
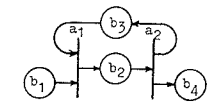


Fig. 5. Correspondence of Block Diagram Element to E-subNet.

各機能要素が行うデータの処理作用は，データの流れを表す徴標にデータ自身を属性値として持たせ（すなわち属性徴標とし）処理内容を属性値に対する数値演算として遷移手順で記述することにより表現できる。各アナログ演算要素では演算時間は零 (0) と考えられるが，E-net では積分演算を数値積分として表現せざるをえないため数値積分におけるきざみ幅 Δt で積分演算器の経過時間を量子化する必要がある。この量子化された時間 Δt は数値積分器の積分演算時間をモデル化したものと考えられ積分演算要素に対応する遷移の遷移時間として表される。これらをまとめると，ブロック図表現の各演算要素とそのE-net 表現との間には Fig. 5 のような対応関係が成り立ち，これらを用いると Fig. 4 a) の例は Fig. 4 b) のように表される。



$a_1=(J(b_1,b_3,b_2),\Delta t,$
$[T\to M(b_2(1)):=M(b_3(1))+\Delta t*M(b_1(1))])$
$a_2=(F(b_2,b_3,b_4),0,-)$
$M_0(b_2)=1,\ M_0(b_2(1))=IC$

Fig. 6. Another E-net Representation of Integration Element.

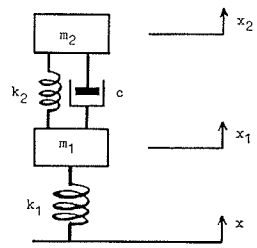


Fig. 7. Model of a Mass-Spring Damper System.

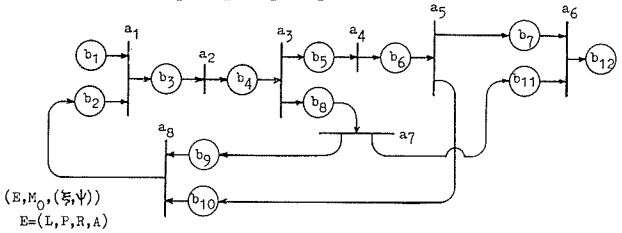
一般のブロック図では多入力加算，加算積分，多変数関数等の要素が用いられることがあるが，これらは上記の機能要素の組み合わせで表現することができ必要に応じてマクロ・ネット化すればよい。又，積分要素は目的に応じ具体的な積分法の構造を示すこともでき (Fig. 6)，その場合 Fig. 5 の積分はマクロ・ネット表現と考えることができる。なお，連続系システムでは遅延要素により明示される以外は制御やデータなどの情報の伝達に遅れは生じないと考えられるので，積分要素の入力居処の徴標占有時間が遷移時間 (Δt) を越えないよう注意する必要がある。

3.3 E-net による連続系システムの記述例

例として Fig. 7 で示される振動系を考える。この振動系は次の連立常微分方程式で表される。

$$\begin{cases} m_1\ddot{x}_1 - c(\dot{x}_2-\dot{x}_1)+k_1(x_1-x)-k_2(x_2-x_1)=0 \\ m_2\ddot{x}_2 + c(\dot{x}_2-\dot{x}_1)+k_2(x_2-x_1)=0 \end{cases}$$

但し，$t=0$ において　$\dot{x}_1=x_1=\dot{x}_2=x_2=0$



$(E,M_0,(\xi,\psi))$
$E=(L,P,R,A)$
$L=\{b_1[1],b_2[1],\ \cdots\ ,b_{12}[1]\}$
$P=\{b_1[1],b_{12}[1]\}$
$R=\emptyset$
$A=\{a_1,a_2,\ \cdots\ ,a_8\}$

$a_1=(J(b_1,b_2,b_3),0,[T\to M(b_3(1)):=M(b_1(1))+M(b_2(1))])$	(adder)
$a_2=(T(b_3,b_4),\Delta t,[T\to M(b_4(1)):=INT(M(b_3(1)))])$	(integrator)
$a_3=(F(b_4,b_5,b_8),0,-)$	(branching)
$a_4=(T(b_5,b_6),\Delta t,[T\to M(b_6(1)):=INT(M(b_5(1)))])$	(integrator)
$a_5=(F(b_6,b_7,b_{10}),0,-)$	(branching)
$a_6=(J(b_7,b_{11},b_{12}),0,[T\to M(b_{12}(1)):=C*M(b_{12}(1))+D*M(b_7(1))])$	(coefficient adder)
$a_7=(F(b_8,b_9,b_{11}),0,-)$	(branching)
$a_8=(J(b_9,b_{10},b_2),0,[T\to M(b_2(1)):=-A*M(b_9(1))-B*M(b_{10}(1))])$	(coefficient adder)

$M_0(b_4)=M_0(b_6)=1,$ otherwise $0,\quad M_0(b_4(1))=M_0(b_6(1))=0$
$\xi=\Psi=\emptyset$

Fig. 8. E-net Representation of $\ddot{x}=-(A\dot{x}+Bx)+Y$.

両式を次のように変形する。

$$\begin{cases} \ddot{x}_1 = -\left(\dfrac{c}{m_1}\dot{x}_1 + \dfrac{k_1+k_2}{m_1}x_1\right) + \left(\dfrac{c}{m_1}\dot{x}_2 + \dfrac{k_2}{m_1}x_2\right) + \dfrac{k_1}{m_1}x \\ \ddot{x}_2 = -\left(\dfrac{c}{m_2}\dot{x}_2 + \dfrac{k_2}{m_2}x_2\right) + \left(\dfrac{c}{m_2}\dot{x}_1 + \dfrac{k_2}{m_2}x_1\right) \end{cases}$$

この二つの式は次のような共通な構造を持っている。

$$\ddot{X} = -(A\dot{X} + BX) + Y$$

$$\text{出力} \quad C\dot{X} + DX$$

この式の E-net は Fig. 8 のように表され，系全体は Fig. 9 のように表される。

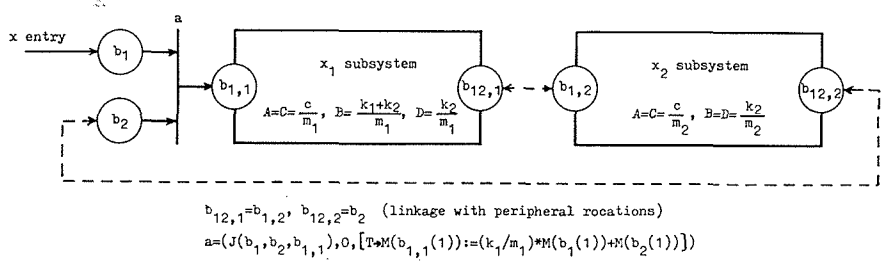


Fig. 9. E-net of a Mass-Spring Damper System.

4. E-net による連続—離散混合系システムの記述[10)]

4.1 連続—離散混合系システム

連続—離散混合系システムというのは，連続系システム及び離散系システムをサブシステムとして含み，それらの間でデータや制御などの情報が受け渡しされ相互作用が行われるシステムのことである。従来，これら連続系及び離散系システムはそれぞれ独自にモデル化手法やシミュレーション言語が考えられ作られてきた(例えば，連続系システムでは各種フローグラフ，CSMP, DYNAMO 等，離散系システムでは GPSS, SIMSCRIPT 等)。しかし，混合系システムに対しては統一的モデル化手法もなく，シミュレーション言語として GASP があるが普及するまでには到っていない。

E-net は離散系システムのモデル化手法として発展してきたが，第3章で述べたように連続系システムの記述に使用することもできる。従って，両システムの間でのデータや制御などの情報の受け渡しが記述できるならば，E-net により統一的に混合系の記述ができることになる。

E-net においては情報は徴標により保持されており徴標の移動により情報の流れが表現できる。この場合，徴標が存在すること自体制御を含め情報が存在することを示し，徴標が保持する属性値がデータを表している。従って，混合系を記述する場合問題となるのは E-net モデル上でのデータ受け渡しの媒体に関してではなく受け渡しの過程の記述そのものに関してである。すなわち，E-net による積分の表現ではきざみ幅 Δt の積分間隔を持っているので，E-net で記述した場合連続系サブシステム部分が離散系サブシステム部分から見て連続系とみなされるためには積分のきざみ幅が離散系事象の生起間隔に比べて十分小さくなければならず，このことと，情報の発生がすなわち徴標の生成であることを考え合わせると，両サブシステムを結合するためにはそれぞれの結合端において徴標到達間隔の整合をとる必要があり (Fig. 10)，特に，連続系の徴標の流れを疎害しないことが重要である。

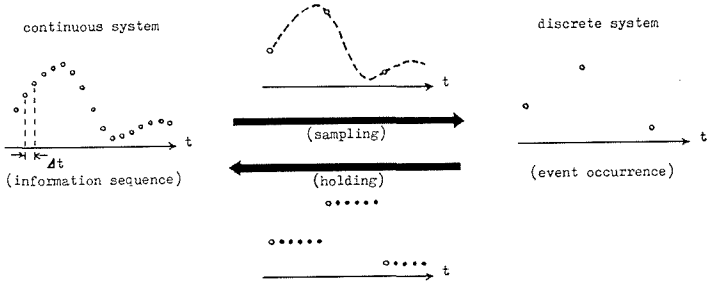


Fig. 10. Communication between Continuous System and Discrete System.

4.2 E-net による連続系—離系インタフェース

連続系サブシステムと離散系サブシステムを結合する際, Fig. 10 に示されるように情報の流れる向きによって整合のとり方が異らねばならない。すなわち, インタフェースは, 連続系から離散系に情報を送る場合には連続系で作られた情報系列から離散系の必要状況に応じて抽出し離散系から連続系に伝達する場合には離散系で作られた情報系列を連続系のきざみ幅に合わせて補間するように構成される必要がある。Fig. 11 と Fig. 12 に E-net で記述されたそれぞれのインタフェースを示す。これらはハイブリッド計算機における A-D, D-A 変換器にも相当するもので, 必要に応じマクロ・ネット化を行うことが有効である。

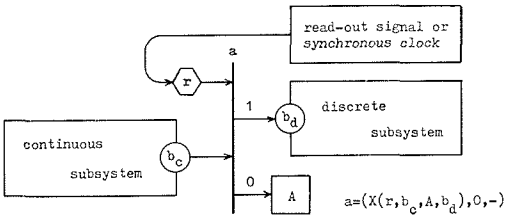


Fig. 11. Interface from Continuous Subsystem to Discrete Subsystem.

4.3 E-net による混合系システムの記述

3.3 節において連続系システムの例として Fig. 7 で示される振動系を考えた。この振動系は車輛の懸架系のモデルと解釈できる。その場合変量 x はこの懸架系に対する地表の形状であると考えられ時間の関数となる。この地表面の変化すなわち障害物の発生を離散系において近似するならば, 離散系から連続系への情報の流れだけを持つ Fig. 12 の型の混合系となる。

次に, シミュレーション対象を懸架系を含めた運転系に広げ走行時を考える。このとき, 車輛振動は地面形状の関数であるが, 地面形状すなわち障害物の発生間隔は走行速度の関数となる。一方, 車輛の居住性を考えるならば, 走行速度は車輛振動をおさえる方向に制御されると考えられ車輛振動の関数となる。従って運転系は, 運転速度→地面形状→車輛振動→運転速度, という

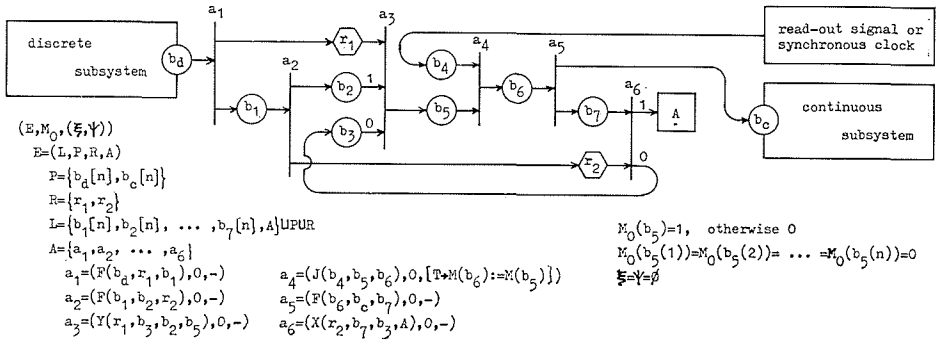


Fig. 12. Interface from Discrete Subsystem to Continuous Subsystem.

フィードバック・ループをなす。このループ中の運転速度の決定及びその操作は人間による判断，意思決定，制御動作の開始として離散系で記述される。このように，運転系では連続系部分と離散系部分とを Fig. 11 及び Fig. 12 で示したインタフェースで結合した型の混合系となる。

5. おわりに

E-net 手法は離散系システム特に計算機システムのモデル化手法として発展してきており，その面では実績も多い[1)~9)]。しかしそれ以外の面における可能性に対しては検討されていなかった。

我々は，E-net は非同期並列動作が記述できデータの流れが表現できること，E-net 構造とアナログ計算機のプログラムであるブロック図との間に類似性があることなどに着目し，実際に E-net により連続系システムが記述できることを示した。さらに，連続系システムと離散系システムとの間でデータや制御などの情報の受け渡しを行うインタフェース部分も E-net で表現できることを示した。従って，E-net は連続—離散混合系システムを記述する能力を有することが示された。

従来，連続系システムと離散系システム両者の統一的モデル化手法がなく，それぞれ独立に取り扱われ異なったモデル化が行われてきた。そのため，混合系シミュレーション言語も従来の両システムそれぞれに対するシミュレーション言語を融合した形のもの（例えば GASP）が試みられているにすぎない。E-net の持つ能力として重要なことは，単に連続系システムの記述までできるということではなく，連続系を含めた混合系システム全体を手法間の整合にわずらわされることなく単一の手法で構成できるということにある。

さらに，E-net で連続系システムを表現する技法は，近年試作が行われるようになってきた連続系システム・モデル実行用のディジタル・ハードウェア・システムあるいはもっと一般のマルチマイクロプロセッサ・システム等の動作解析において，その能力を発揮するものと思われる。

いずれにしても E-net の持つ能力にはまだ未知の部分が多く，今後使用分野の拡大や専用言語化のなかで E-net 自体の持つ問題点を見つけ出し，シミュレーション・モデル構成手法として一層洗練されたものとしていくことが必要である。

付　　録

A. 遷移の発火

遷移の発火は次のような各段階からなる。

1) 擬似発火可能段階 (pseudo-enable phase) (周縁選択居処を持つ X 及び Y 遷移型にだけ存在する)： 周縁選択居処が未定義であるのを除いて，他のすべての入出力居処が発火可能状態にある。周縁選択居処の状態を定義するため，選択手順 (2.3 節) の評価を開始する。

2) 発火可能段階 (enabled phase)： すべての居処が発火可能状態にある。この時点で遷移は処理を開始する。

3) 発火動作段階 (active phase)： 遷移動作が進行中であり，遷移時間 (2.2 節) が消費される。この遷移に属する各居処の状態は変えられずに保持されている。

4) 発火終了段階 (terminate phase)： 遷移は処理を完了し，出力居処の状態を写像の右辺の状態に一致させ，遷移手順 (2.2 節) を実行し，入力居処の状態を写像の左辺の状態に一致させる。

B. 遷移手順及び選択手順の評価

$$\left[p_1 \rightarrow (e_{11};\ e_{12};\ \cdots;\ e_{1n}):\ \cdots:\ p_k \rightarrow (e_{k1};\ e_{k2};\ \cdots;\ e_{km})\right]$$

なる一般形に対し次の手順で評価が行われる。

1) p_i を $i=1$ から順次評価し，T (真) となる最初のものを見つける。$i=k$ までに T となる p_i が見つかれば 2) に行き，見つからなければ遷移 (選択) 手順の評価を終了する。

2) p_i が T となる最初のものであるならば $(e_{i1}; e_{i2}; \cdots; e_{il})$ を実行し，遷移 (選択) 手順の評価を終了する。

C. 形式的記述表現における簡略記法

1) 省略遷移手順 (default transition procedure): 遷移の発火に伴う遷移動作が属性値の変化を伴わず遷移の定義 (Fig. 1) で与えられる機能だけを行う場合，遷移手順は "—" によって置き換えられる。但し，J 遷移型では，属性値が変化しなくともその入出力居処に属性居処が含まれるならば，この記法は使えない。又，遷移動作に伴い一部の属性値のみが変更を受ける場合には，その属性値に対する変更だけを遷移手順として記述すればよい。

2) 省略居処記述 (abbreviated location description): 各居処を記述するときに，その居処の属性の個数を明示しなくとも曖昧さが生じない場合には，属性の個数に対する記述部分は省略できる。すなわち，$M(b\,[n])$ のかわりに $M(b)$ と書くことが許される (註 6] 参照)。但し，居処の指定の記述すなわち E-net 構造記述における集合 L の宣言部分においてはこの省略は許されない。

参 考 文 献

1) Nutt, G. J.: The formulation and application of Evaluation nets, Computer Science Group, Univ. Washington, TR-72-07-02 (1972).
2) Nutt, G. J.: Evaluation Nets for Computer system Performance Analysis, FJCC 1972, AFIPS Conf. Proc., Vol. 41, pp. 279-286 (1972).
3) Noe, J. D. & Nutt, G. J.: Macro E-nets for Representation of Parallel Systems, IEEE Trans. on Computer, Vol. C-22, No. 8, pp. 718-727 (1973).
4) 阿部，栃内，永田： E-NET による計算機システムのシミュレーションについて． 昭和 49 年電気四学会北海道支部連合大会 **143** (1974).
5) 橋津，栃内，永田： E-NET 手法を用いたシミュレーションによるマルチプログラミングシステムの評価，昭和 50 年電気四学会北海道支部連合大会 **169** (1975).
6) 橋津，阿部，栃内，永田： E-NET による計算機システムのシミュレーション，北海道大学工学部研究報告，No. 78, pp. 57-67 (1976).
7) 針原，栃内，橋津，永田： E-NET シミュレーションによる計算機システムの評価，昭和 51 年電気四学会北海道支部連合大会 **92** (1976).
8) 栃内，永田： 会計データを用いる計算機システムの評価，電気学会情報処理研究会資料，IP-77-24 (1977).
9) 針原，栃内，永田： E-NET シミュレーションによる計算機システムの評価 (PART II)，昭和 52 年電気四学会北海道支部連合大会 **153** (1977).
10) 牧野，栃内，永田： E-NET シミュレーション・モデル構成手法による，連続—離散・混合系記述の可能性，昭和 52 年電気四学会北海道支部連合大会 **154** (1977).