

## 離散事象システム研究の動向と課題

こ 児      だ ま      し ん      そ う  
見      玉      慎      三\*

### 1. 離散事象システムの背景

離散事象システム (Discrete Event System 以下 DES) が制御対象として認知され、制御分野の重要課題の1つとして「市民権」を得るに至ったのは最近のことである<sup>1), 2)</sup>。DES は組立・加工・貯蔵・搬送ラインを含む生産システム、計算機および通信ネットワークのプロトコルシステム、交通管制システム、バッチプロセス、離散レベルのロボット動作、計算機オペレーティングシステムなどのシステムクラスの総称である。DES においては、システム内に有限個の離散的な事象が存在し、それらが不規則な時間間隔で発生する。システムの状態 (状況) も離散的であり、その変化は図1のように、ある間隔保持されたのち事象の発生によってつぎの状態へ移る、という区分的に一定の軌道を示す。たとえば、ロボットが部品を場所Aでつかみ場所Bに置くというシステムを DES として見ると、部品がAにある ( $p_1$ )、ロボットが部品をつかんでいる ( $p_2$ )、Bにある ( $p_3$ ) という3つの状態と、ロボットがAで部品をつかむ ( $t_1$ )、Bに置く ( $t_2$ )、という2つの事象がある。状態  $p_1$  が成立しているとき、事象  $t_1$  が発生すると状態  $p_2$  に移り、さらに  $t_2$  によ

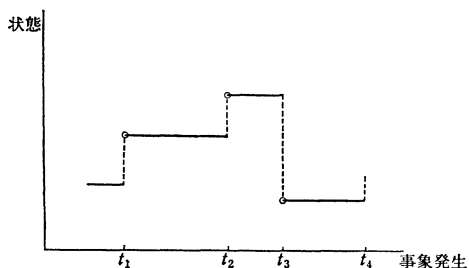


図1 離散事象システムの状態軌道

\* 大阪大学工学部 吹田市山田丘 2-1

キーワード: モデリング (modelling), ペトリネット (petri-net), 制御理論 (control theory), min-max 代数 (min-max algebra), スーパーバイザ制御 (supervisor control).

り  $p_3$  に至る。場合によってはロボットが空いている ( $p_4$ ) という状態も考えることがある。

計算機の能力の発達とともに規模の大きい DES が開発され、これに対応して情報伝送・変換を適切に管理し、事象の生起を制御する方法論を確立する要求が生じるのは当然のなりゆきである。

制御の方法論が体系化されている連続系の場合と同様に、DES の制御技法を構築するには、まずシステムモデリングから考えなければならない。ここではまず頭にうかぶのは、もともと制御技術の一分野としてシーケンス制御が存在し、そこで対象としてきたのは上述の DES に属するシステムということである。しかし、従来シーケンス制御で扱った対象は機械レベル (下位レベル) の小規模なシステムであり、このため動作表現として論理回路やラダー回路など論理式の図的表示を用いることが多かった。論理式に基礎をおくモデリングはシステムの状態が明示されないため、ダイナミカルシステムとして動作を追跡するのに不便であり、階層構造、分散構造などを含む大規模 DES を制御対象として表現するのに適当でない<sup>3)</sup>。

そこで新しいモデル表現を探す努力が、制御工学はもちろん隣接する計算機科学や OR を含む範囲で行われた。しながらこの努力は現在のところきわめて成功しているとはいえない状況にある。連続系における微分方程式 (あるいはその変形) のような汎用で強力なモデルは存在せず、DES の種類により、またどのような問題を取り上げるかに応じて、いろいろなモデルが提案されているのが現状である。汎用モデルとそれに基づく統合的な制御技術が存在しない以上、DES の解析には離散型シミュレーションが現在でも重要な技法である事情は変わらない<sup>4)</sup>。

とはいえ、全体として眺めると制御対象として明示性・有用性の高いモデルを導入し、「DES の制御理論」を構築しようとするところみはすでにいくつかの興味ある成果を生み、大きな研究の流れを作り出しつつある<sup>2), 5)</sup>。いずれ取捨選択されて、あるものは定着

し、あるものは消えるであろうが、この分野に不可逆な流れが存在することは確かである。以下では 2. で DES のモデリング問題を取り上げ、3. においては理論的な興味に焦点を当て、DES においてもある意味で連続系と平行した制御理論が展開できることを説明し、4. で今後の課題にふれる。

## 2. モデリング

適切かつ有用なモデルを導入することは DES においても重要であることに変わりない。しかし連続系における微分方程式のような汎用の表現手段はなく、多くのモデルとそれをふまえた多様なアプローチの方法が存在する状況である。これはなぜであろうか。一口でいうと DES の特質に起因しているのであるが、もう少し詳しく考えてみよう。

- (i) DES のモデルはシステムの状態、事象、あるいはその両方の変化を記述できなければならない。ここで DES は「人工システム」であり、連続系で対象となる「物理システム」(自然システム)と異なり、状態変化すなわち事象発生は、システム内で所定の論理関係が成立したとき生じ、物理法則に従うのでなくいわばソフトウェアに従う性質のものである。すなわち物理法則をモデリングの基礎とするわけにはいかない。
- (ii) 連続系のダイナミクスの記述は近似表現に基礎をおいている。すなわち微分方程式はダイナミクスの近似表現である。一方、DES では事象や状態は時間的空間的に離散的で近似は成立しない。
- (iii) 連続系の制御仕様が統一的であるのに比較し、DES の制御の問題と仕様は個別的である。
- (iv) DES のスループットの時間的経過などの評価を行うには、事象や状態の時間的変化の情報を必要とする。このためには DES モデルに時間の要素

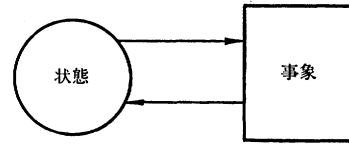


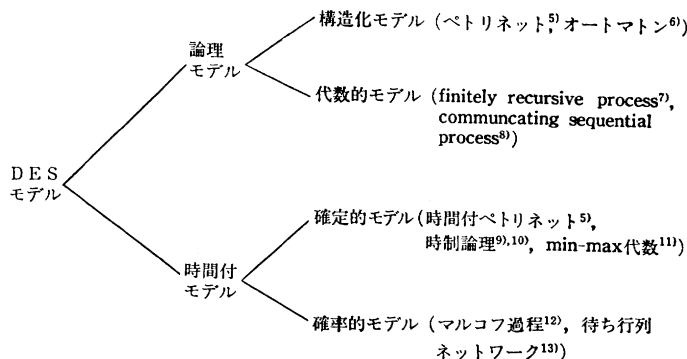
図 2 事象駆動型システム

を導入しなければならないが、微分方程式のように空間と時間情報が融合した表現は DES (の確定的モデル) ではむずかしい。DES では、事象や状態の生起順序にのみ着目し、生起時刻を無視したモデル化を行う場合 (このクラスを論理モデル *logical model* とよぶことにする) と、論理モデルに時間要素を導入したより高度のモデルを考える場合 (これを時間付モデル *timed model* とよぶ) に大別される。連続系の微分方程式にはこの両者の区別はない。

さて、DES に限らず、一般にモデルを設定するには、どのような視点から対象をとらえるかという立場を定めなければならない。さいわいこの点 DES については研究者の意見の一致がみられる。制御対象として DES を考えるとき、事象駆動型 (*event driven*) システムとしての視点からモデル化するのが、制御仕様の表現の容易さ、理解しやすさ、自然さなどにおいてもっともよいとされている。事象駆動型システムを DES の定義するむきもあるくらいである<sup>2)</sup>。事象駆動型はシステムのふるまいを状態と事象の因果性から眺める。すなわち、ある状態に対応して発生可能な事象が定まり、事象が発生するとそれに駆動され新しい状態に移り、そこでまたつぎの事象の発生がうながされる……、という過程のくり返しでシステムの動作が進行するとみる (図 2 参照)。

一般にモデルの性能は、どのような機能を記述できるか (表現能力) と取り扱いの容易さ (解析能力、操

表 1 離散事象システムモデルの分類



作性)により評価される。DES を事象駆動型としてモデル化するという立場は一貫するとしても、その具体的な表現になると、すでに述べたような事情により多様なモデルを生じうる。Ho<sup>2)</sup>を参考にモデルの分類を示しておこう。まず全体は論理モデルと時間付モデルに大別され、論理モデルはさらに状態遷移と事象発生が構造的に規定されるグループと、代数的に規定されるグループに分かれる。時間付モデルのクラスは確定的グループと確率的グループ(性能評価モデルともいう)に分類できる(表1)。

DES のどの応用分野にせよ、これが決定判というモデルは存在しない。どれもそれぞれ特徴と制約があり、問題によりまた解析法により、異なるモデルが用いられる。

論理モデルは、時間情報を無視して状態や事象の空間的ふるまい、あるいは論理関係を記述する。すなわち DES の定性的な側面に注目する。この簡単化のため、時間付モデルより一般に取り扱いが容易である。このクラスに属するペトリネットは、事象駆動型としてのシステム構造をそのまま自然にモデル化したもので、その意味で基礎的モデルといえる。ペトリネットは、局所的な状態を表わす節点である「プレース」と、事象に対応する節点の「トランジション」、この2種類の節点を因果関係を表わすアークで結合した2部グラフである(図3)。システム機能についての高い表現能力をもち<sup>14)</sup>、またシステム構造を視覚(グラフ)的に把握しやすいが、いまのところ解析能力が一般に高いとはいえない。しかし、これは表現能力と相対的なもので、ペトリネットで記述した DES がそれだけ複雑な動作をする可能性を有しているからでもある。オートマトンモデルは状態遷移図を基礎としているので、表現能力としてはペトリネットのサブクラスに相当する。たとえば事象の並列性などは表現できない。しかしながら、事象系列を言語とみなし、その制御を言語理論の立場から扱うのにこのモデルが用いられることが多い<sup>6), 15)</sup>。代数的モデルは、オートマトンなどが状態を機能表現の基本要素とするのに対し、プロセスと称する事象系列に基礎をおく。システムの結合に

対応する演算が定義され、モデルはそれら演算について閉じた代数系として導入される。

時間付モデルのうち、時間付ペトリネットは通常のペトリネットのプレースやトランジションに時間要素を導入し、それぞれ状態の保留時間や事象発生の所要時間を表わすように拡張したものである。時制論理は時間の概念を導入した論理体系で、DES の表現と性質の検証に利用される。しかし代数的モデルと同様、システムの機能や制約条件の記述は容易であるが、システム構造が直観的に把握し難く、工学モデルとして使用するには工夫を要する<sup>10)</sup>。min-max 代数は、その有用性についてまだ評価は未定であるが、特殊な代数に基礎をおいた興味ある手法である。2つの実数(あるいは行列)間の乗法  $a \cdot b$  は  $\max(a, b)$  (場合によっては  $\min(a, b)$ ) として定義される。この代数系の下で、あるクラスの時間付ペトリネットの解析に線形システム理論が適用できることが示されている<sup>11)</sup>。確率的モデルは DES の時間についての平均的なふるまい、すなわち性能評価に用いられる。ここで表1には示されていないが、モデルに依存しない形で直接的に性能評価を行う手法として、最近盛んに研究されているのに摂動解析(Perturbation analysis)がある<sup>16), 17)</sup>。摂動解析は DES 性能評価関数のパラメータ感度を計算する手法である。これはあるシステムパラメータ値のもと、実データかあるいはシミュレーションにより、1本の事象系列軌道を観測し、これよりパラメータ変動に対応する軌道変動の(ひいては評価関数の変動)を計算するものである。したがってシミュレーションによる解析プログラムを効率化する手法として有望視されている。

### 3. 離散事象システムの「制御理論」

ここで DES の「制御理論」の研究がどのように展開されているか見てみよう。汎用モデルが存在しない以上、モデルに依存する形で理論が展開されることになる。まず論理モデルクラスの基準モデルであるペトリネットに着目する。ペトリネットでは事象の生起(トランジションの発火)に伴う状態変化を表わすため、プレースにおかれたトークンでその時点で成立している局所状態の状況を示す。その状況が保持されていればトークン(たとえば部品3個の到着を示すなら3個のトークン)が存在し、トークンの不在は状況の不成立を表わす。したがってプレースにおけるトークン分布を示すベクトルを  $M$  (第  $i$  成分は  $i$  番目プレースのトークン数)とすれば、 $M$  がペトリネット全体の状況を示す状態ベクトル(マーキングベクトルという)

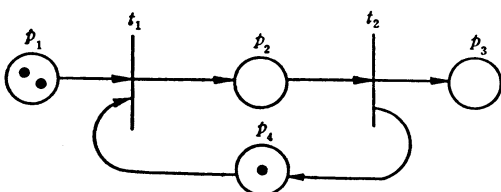


図3 1.のロボット動作のペトリネット表現

である。トランジション発火によるマーキングの変化は

$$M_{k+1} = M_k + A r_k, \quad k=0, 1, 2, \dots \quad (3.1)$$

により与えられることはよく知られている<sup>5)</sup>。  $M_k$  は  $(k-1)$  ステップのトランジション発火後のマーキング、  $r_k$  はどのトランジションを発火させるかを指定する  $(0, 1)$  成分の発火ベクトル、  $A$  はプレース・トランジション間の接続行列である。ペトリネットの状態転移を規定する (3.1) 式は一見連続系線形システムの  $x_{k+1} = Gx_k + Hu_k$  と同形式であり、したがって、線形制御理論がそのまま適用できそうである。しかし事実はそう簡単ではないのであって、(3.1) 式が線形方程式であるのはみかけ上のことであり、本質的には非線形方程式である。それは入力  $r_k$  は自由に指定できるのではなく、(マーキング  $M_k$  に対応する) 発火可能なトランジションのうちから選択しなければならないという制約があるからである。つまり各ステップでそのときの状態により制約を受けた入力しか許容しない。ペトリネットの解析や制御に線形システム理論や制御理論が直接的にもち込めない大きな理由はこの非線形性にある。たとえば DES 制御の基本問題である可達性（与えられた初期マーキング  $M_0$  から目標マーキング  $M$  に到達できるか）を考える。  $M$  が  $M_0$  より可達ならば (3.1) 式よりあるステップ数  $N$  について

$$M - M_0 = A \Sigma \quad (3.2)$$

が成立する（ここで  $\Sigma = r_0 + r_1 + \dots + r_{N-1}$ ）。すなわち  $M$  が  $M_0$  から可達ならば (3.2) 式が非負整数解  $\Sigma$  をもつ。ところが非負整数解  $\Sigma$  が存在することは必要条件であっても、十分条件ではない。十分性には、さらに  $\Sigma$  が実行可能であること、すなわち  $\Sigma$  が許容入力列の和に分解できるということを示さなければならない。これを解析的に解くのは困難で、このためペトリネットの可達性の必要十分条件は一般的には求められておらず、ネット理論の中心課題の1つと考えられている。このような非線形制約にもかかわらず、(3.1) 式はペトリネットのいろいろな性質の解明に有用な手がかりを与えるが、詳しくは文献 5) を参照されたい。

それではペトリネットのある種の動態が、本質的に線形の方程式で記述されることがあるであろうか。Cohen らによると時間付ペトリネットのサブクラスであるイベントグラフ (event graph) については、トランジションの発火時刻は、min-max 代数のもとで入力を入力トランジションの発火時刻とする有限次元定係数線形方程式により表わされることが示される<sup>11)</sup>。

イベントグラフの構造はマークグラフ<sup>5)</sup>であり、事象発火の選択性は表現できないものの、このクラスについては線形システム理論を適用する途が開ける。すなわち可制御性、可観測性、安定性、フィードバック安定化、などを論じることができる。「周波数域での表現」として伝達行列すら定義され、これからシステムの結合の伝達行列による表現が可能となり、またシステムの構成を伝達行列の実現問題として考察することができる。イベントグラフという限定された対象と問題を扱っているものの、線形システム理論と DES を結びつける理論的手法として興味深い。

現在、DES のフィードバック制御を連続系に平行した形で構築するところろみがある程度成功しているのに Wonham らの研究がある<sup>6), 15)</sup>。 Wonham らは DES の制御を、システムにおいて発火可能な事象系列（言語）の集合のうち、ある条件を満たす事象系列（言語）だけが生じるように制御を行うことと定式化している。たとえば資源の有効利用のための排他制御、デッドロックの回避、事象発生回数が事象間にかたよりを生じないための公平性の保証など、この立場から扱うことができる<sup>15)</sup>。対象をオートマトンによりモデル化できるものとして、対象が発生しうる言語を  $L$ 、制御目的を満たす言語を  $K \subseteq L$  とするとき、対象が  $K$  だけを発生するような制御器 (supervisor) を設計する問題をスーパーバイザ制御という。制御は、これまでに対象が発生した事象系列に基づいて、つぎに生起を許容する事象を決定するというフィードバック方式で行う（図 4）。 Wonham らは言語  $K$  が ( $L$  に関し) 可制御であるという概念を定義した。そして連続系の極配置に対応する性質として、図 4 のようなフィードバック制御器が存在する必要十分条件、すなわち制御器を介してフィードバック制御された対象の言語が  $K$  となる必要十分条件は、 $K$  が可制御であることを示した。また制御器を求める計算量は、可制御性の判定と同じく、多項式オーダーであることも示されている。これらはこの方向の研究のもっとも基本的な成果

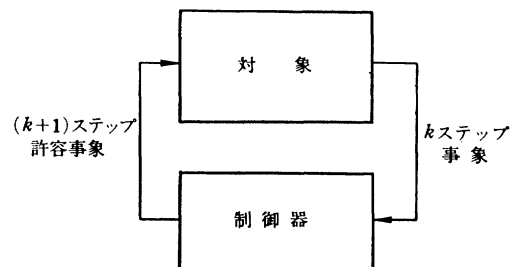


図 4 スーパーバイザ制御

である。このほか、制御目的が部分目的に分割されているとき、部分目的の制御器をまず求め、それを総合して全体の制御器を作るモジュラ設計法、安定化法、オブザーバなどの研究もある（(6), (15)および引用文献を参照されたい）。これらの研究はオートマトンという単純化モデルを基礎にしているが、DES 制御で何が可能で、その達成にはどのような条件が成立していなければならないか、という総括的な視点からの枠を理論的に明らかにした貢献は大きい。

#### 4. いくつかの課題

DES 制御の重要性の認識が高まるにつれ、研究課題として注目を集めるようになったが、まだ連続系に比べて研究の歴史が浅く、種々のモデリングとアプローチの方法が提案されている状況で体系化にほど遠い。この分野では「理論と実際のギャップ」が存在するが、これ自体は当然のことであり、生長期の分野では自然な形といえる。ただし、今後この分野の研究が発展する過程において、実システムからの要求にこたえる努力は持続しなければならない。Ho によると、さもないと DES の研究もかつての理論の花形であったゲーム理論がそうなったように、制御工学の本流に背を向けられることになることになると警告している<sup>18)</sup>。

このような意味で今後の研究課題として2つほど指しておきたい。それは階層化とモデリングの問題である。まず階層化手法であるが、これは直接的には計算量の軽減の対策となるし、またもっと基本的には DES の設計・解析を階層的にすすめる裏付けとなるものである。DES では離散構造のため、ある問題の解が解析的な形で与えられていても、具体的に解を求めたり付帯条件を判定したりする過程の組合せ計算や探索の手間により「非実用的」になる可能性をもっている。DES では、可達な状態空間や事象系列を探索する解法は容易に指数オーダーの計算量となり、避けなければならない。計算量の軽減の直接的な対策はモジュラ制御にみられるように対象と問題の分割であるが、分割をさらに一般化すると階層化となる。階層構造では、全体を1つのレベル（詳細レベル）でとらえるのではなく、いくつかの層に分割し、ある層のコンポーネント（サブシステム）がいくつか結合し上位層のコンポーネントを形成するとみる。階層構造の導入の利点は、システムの構造的性質を明確にするとともに、解析の計算量の軽減を可能としたり、仕様記述を簡単な概念的なレベルからしだいに詳細化していくトップダウン方式、あるいはいくつかの詳細なレベルの記述を積み上げて全体を構成するボトムアップ方式で

行うときの理論的な裏付けを与え、さらには制御ソフトウェアのデバッグ、検証、保守、変更などを容易にする。階層化においては階層（レベル）をどう設定するかという問題、あるレベルでの動作性質が、どのように上位レベルに反映されるかという階層間の性質の継承の問題などが重要である。これらについては、いまのところペトリネットで階層間でライブ性が保存される条件が検討されているが<sup>5), 19)</sup>、本格的な研究はこれからの課題である。

つぎのモデリングは階層化と密接に関連した問題である。DES 制御の研究はまだモデリングをどう設定するかが重要な意味をもつ段階にある。2. でモデルの性能は(i)表現能力、(ii)明示性、(iii)解析能力、(iv)操作性により評価されると述べた。ペトリネットがそのままの形で実システムのモデルとして採用されることが少ないのは、ペトリネットの解析能力が高くないという点よりも、むしろシステムの詳細レベルまで一様にペトリネットで記述すると、表現が複雑・大規模になりやすく、明示性、操作性が失われてしまうところにある。これを対処する1つの方法は、上に述べた階層構造を導入することであるが、もう1つの可能な方法はペトリネットと他のモデルを組み合わせた複合型の（高レベル）モデルを導入することである。一般にペトリネットはシステムの制御の流れ（タスクの流れ）を記述するかぎり明示性は高いが、情報処理やデータ変換などの演算機構まで表示しようとすると複雑化の要因となる。したがって1つの現実的なモデリングとしては、ペトリネットは制御の流れを表現し、演算処理面は他の適切なモデル（あるいはもっと一般的に言語、さらには AI による表現など）で記述し、この両者をペトリネットのプレースやトランジションを介して複合させることが考えられる。事実これまで実用化されているペトリネットを基盤とした DES 制御装置は、このような複合型の形を採用しているものが多い<sup>20), 21)</sup>。これらはいずれも主として図的言語としての表現能力、明示性、操作性（すなわちプログラミングツール）という観点から導入されており、その検証などはシミュレーションに依存していた。これからは複合型モデルの解析能力という側面もあわせて配慮していく必要がある。

(1991年9月24日受付)

#### 参考文献

- 1) Report of the Workshop Held at the University of Santa Clara: Challenges to Control, A Collective View, IEEE Trans. Automatic Control, **AC-32-4**, 275/285 (1987)

- 2) Y. C. Ho: Scanning the Issue: Special Issue on Discrete Event Systems, Proc. IEEE, **77-1**, 316 (1989)
- 3) 関口, ほか 4 名: シーケンス制御工学—新しい理論と設計法—, 電気学会 (1988)
- 4) ミニ特集: 離散系シミュレーションの今日と明日, 計測と制御, **30-2** (1991)
- 5) T. Murata: Petri Nets: Properties, Analysis and Applications, Proc. IEEE, **77-4** (1984)
- 6) P. J. G. Ramadge and W. M. Wonham: The Control of Discrete Event Systems, Proc. IEEE, **77-1**, 81/98 (1989)
- 7) K. Inan and P. Varaiya: Finitely Recursive Models for Discrete Event Systems, IEEE Trans. Automatic Control, **AB-33-7**, 626/639 (1988)
- 8) C. A. Hoare: Communicating Sequential Processes, Englewood. Cliffs, NJ, Prentice-Hall (1985)
- 9) J. G. Thistle and W. M. Wonham: Control Problems in a Temporal Logic Framework, Int. J. Control, **44-4**, 934/976 (1985)
- 10) 本位田, 内平, 松本: 時制論理とペトリネット, オペレーションズリサーチ, 612/618, 9月号 (1987)
- 11) G. Cohen, D. Dubois, J. P. Quadrat and M. Voigt: A Linear Systems Theoretic View of Discrete-Event Processes and Its Use for Performance Evaluation in Manufacturing, IEEE Trans. Automatic Control, **AC-30**, 210/220 (1985)
- 12) E. B. Dynkin and A. Yushkevich: Controlled Markov Processes, Springer-Verlag, New York (1979)
- 13) L. Kleinroch: Queuing Systems, Wiley, New York (1976)
- 14) J. L. ピータースン: ペトリネット入門, 市川, 小林訳, 共立出版 (1984)
- 15) 潮: 離散事象システムにおける制御問題とスーパーバイザ, システム/制御/情報, **34-9** (1990)
- 16) Y. C. Ho: Performance Evaluation and Perturbation Analysis of Discrete Event Systems, IEEE Trans. Automatic Control, **AC-33**, 563/572 (1988)
- 17) X. R. Cao: A Comparison of the Dynamics of Continuous and Discrete Event Systems, Proc. IEEE, **77-1**, 7/13 (1989)
- 18) Y. C. Ho: Editorial, Basic Research, Manufacturing Automation, and Putting the Cart Before the Horse, IEEE Trans. Automatic Control, **AC32-12**, 1042/1043 (1987)
- 19) 熊谷: ネット理論とその応用-IV, システム/制御/情報, **35-4**, 227/235 (1991)
- 20) T. Murata, N. Komoda, K. Matsumoto and K. Haruna: A Petri Net Based Controller for Flexible and Maintainable Sequence Control and Its Applications in Factory Automation, IEEE Trans. Industrial Electronics, **IE-33-1**, 362/366 (1986)
- 21) N. Nagao et al.: Petri-Net Based Programming System for Flexible Manufacturing System, Proc. Japan-US Symp. on Flexible Automation (1990)

=====

[著 者 紹 介]

と だ ま し ん ぞ う  
児 玉 慎 三 君 (正会員)

昭和7年11月9日生。昭和30年早稲田大学理工学部電気工学科卒業。38年カリフォルニア大学バークレイ大学院博士課程修了, 37年大阪大学工学部講師, 49年同大学教授。非線形システム制御, 医用工学, 離散事象システム制御, およびネット理論の研究に従事。システム制御情報学会, 電子情報通信学会, IEEE などの会員。

=====

