

# ペトリネットのシミュレーションへの応用

## (V)

——カラーペトリネットによるモデル化とその応用——

椎塚 久雄

今回は、ペトリネットによる生産システムの簡単なモデル化の例について紹介した。今回は、FMSなどのような複雑な離散事象システムを簡潔な形でモデル化するために、カラーペトリネットを導入し、その基本的モデリング技法と関連する性質について示す。

### 10.5 ペトリネットの不利な点

ペトリネットの主な不利な立場は、FMSなどのように非常に複雑な離散事象システムをモデル化して得られるネットのサイズから生じてくる。つまり大規模モデルに対する対処の仕方をどう乗り切るかということである。たとえば、フローショップシステムの場合では、機械や部品数が3~4以上になると、ネットの規模が大きくなって大きくなるのが懸念される。AGVシステムの場合でも、入力(ロード)ステーションと出力(アンロード)ステーションの数が増加すると同様の問題が生じ、ネットの接続構造(topology)を修正することが必要になる。

一方、10.2および10.3で示した2つの例からもわかるように、システムの機能的な変更がネット構造に関して多大な変更を余儀なくされる。たとえば、フローショップのスケジューリングの変更、あるいはAGVシステムの運行経路の変更などがある場合は、ネットモデル上では多大な修正・変更が生じることになる。

このようなペトリネットの不利な点は、カラーペトリネット(colored Petri net)を導入することによって解消することができる。カラーネットはこれまで述べた通常のペトリネットの多くの利点をそのまま保持している。

## 11. カラーペトリネット

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

ペトリネットのトークンに色をつけ何らかの属性を持たせることにより、FMSなどのような複雑なシステムのモデルが単純化される。つまり、カラートークンを導入し対応する発火規則を定めることで、ペトリネットにある種の階層化をほどこすことができ、より簡潔なモデルの構築が可能となる。

ここでは、前回[IV]で紹介した生産システムのペトリネットモデルをカラーペトリネットによってモデル化し、両者を比較することでカラーネットの理解を容易にしている。

### 11.1 直観的表現

フローショップの例について考えてみると、たとえば機械の集合あるいは生産される部品の集合のように、同じふるまいをするいくつかの要素があることに気づく。同一のネット図面を用いて、このような類似動作を表わすことを考えてみよう。そうすることによってモデルはより簡潔になる。モデルにおける類似要素は属性あるいはカラーを用いて区別される。すなわち、カラーは各要素の識別のために使われるのである。

一例として次のようなカラーを導入して、フローショップのモデル化について考えてみよう：

$MCH = \{mch_1, mch_2\}$ : 各機械に対応するカラーの集合  
 $PARTS = \{part_1, part_2\}$ : 各部品に対応するカラー集合

一般的には要素は2つ以上の属性で特徴づけられる。したがって、合成カラーあるいはカラーの組によって属性を表わすことが適当である。たとえば、カラー  $\langle part_i, mch_j \rangle \in PARTS \times MCH$  によって、機械  $\langle mch_j \rangle$  に関するある部品  $\langle part_i \rangle$  の状態を表わすことができる。

カラーペトリネットはプレースとトランジションより成っている。通常のペトリネットと同じように、プレースはシステムの状態(条件)を表わし、トランジションはその状態の可能な変化(事象の生起)に対応している。プレースはカラーの部分集合(何色かのカラーの集合)に

よってマーキングされる。カラーベトリネットのプレースのトークンは、カラートークン (colored token) あるいは、単に、カラーと呼ばれる。プレースのマーキングは、形式的にはカラーの和で表わされる。たとえば、プレース  $p$  におけるトークンの個数が、カラー  $\langle \text{part}_1, \text{mch}_1 \rangle$  が2個、カラー  $\langle \text{part}_2, \text{mch}_1 \rangle$  が3個ある場合は、 $p$  のマーキングは

$$M(p) = 2 \langle \text{part}_1, \text{mch}_1 \rangle + 3 \langle \text{part}_2, \text{mch}_1 \rangle$$

となる。モデルの状態は、カラーベトリネットのすべてのプレースのマーキングによって定まる。プレースのカラーの部分集合は、対応するトランジションに関係づける。後の章で示すようにフローショップをモデル化したカラーベトリネットにおいて、すべてのプレースとトランジションは、カラー  $\text{PARTS} \times \text{MCH} = \{ \langle \text{part}_i, \text{mch}_j \rangle \}$  の部分集合に対応している。システムの状態は、与えられたカラーに関するトランジションの発火によって変化する。発火可能トランジションに対する条件と発火規則は、ネットのアーキをラベルづけする線形関数 (linear function) によっている。この関数は、トランジションを発火可能にするためにどのカラーがプレースにマーキングされなければならないのか、また発火のためにどのカラーをプレースにつけ加えたり、プレースから取り去ったりしなければならないかを表わしている。

## 11.2 カラーベトリネットの形式的定義[21]

**定義3** カラーベトリネット (CPN) は6項組  $\text{CPN} = (P, T, C, I_-, I_+, M_0)$  である。ただし、

- (1)  $P = \{ p_1, p_2, \dots, p_n \}$  はプレースの有限集合。
- (2)  $T = \{ t_1, t_2, \dots, t_m \}$  はトランジションの有限集合。
- (3)  $P \cap T = \emptyset$  および  $P \cup T \neq \emptyset$ 。
- (4)  $C$  は  $P \cup T$  から非ゼロ集合の中へ定義されるカラー関数である。すなわち、各プレースに対しては可能なトークンカラーの集合を対応づけ、各トランジションに対しては可能な発火カラーを対応づける関数である。
- (5)  $I_-$  と  $I_+$  は、それぞれ、すべての  $(p, t) \in P \times T$  に対して、 $I_-(p, t), I_+(p, t) \in [C(t)_{ms} \rightarrow C(p)_{ms}]_L$  であるような  $P \times T$  上で定義される、負と正の接続関数 (incidence function) である。ただし、右下添字  $ms$  は多重集合 (multi-set あるいは bag) を、 $L$  は線形関数であることを表わす。また、 $I_-$  と  $I_+$  はそれぞれ入力関数および出力関数とも呼ばれる。
- (6) 初期マーキング  $M_0$  は、すべての  $p \in P$  に対して  $M_0(p) \in C(p)_{ms}$  であるような  $P$  について定義される関数である。

**定義4** カラーベトリネットのマーキングは

$$M(p) : C(p)_{ms} \rightarrow N$$

であるような  $p$  について定義される関数  $M$  である。ただし、 $N$  は非-負整数の集合を表わす。

カラーベトリネットも通常のベトリネットと同じように、プレースが丸で、トランジションがバーである有向グラフによって表わされる。関数  $I_-(p, t)$  は、プレースからトランジションに連結するアーキの重みを与える (そのアーキがなければ  $I_-(p, t)$  はゼロ関数に等しい)。関数  $I_+(p, t)$  はトランジションからプレースに連結するアーキの重みを与える (そのアーキがなければ  $I_+(p, t)$  はゼロ関数に等しい)。フロー行列 (flow matrix)  $W$  を

$$W = I_+(p, t) - I_-(p, t)$$

と定義する。  $W$  は  $n \times m$  行列の線形関数である。

トランジション  $t$  はカラー  $c$  に関して、マーキング  $M$  が  $M(p)(c) \geq I_-(p, t)(c)$ ,  $\forall p \in P$  であるならば、発火可能である。トランジションは発火可能のとき、発火できる (発火カラー対応)。発火は次の2つのステップで実行される:

- (1) 入力プレースからカラートークン  $I_-(p, t)(c)$  を取り去る (入力カラー対応)。
- (2) 出力プレースにカラートークン  $I_+(p, t)(c)$  を加える (出力カラー対応)。

## 12. カラーベトリネットによるモデル化

11.1で述べたように、カラーネットを構成するためには、カラーの集合と関数の集合の定義が必要である。一般に、カラーベトリネットでモデル化を行なうと、大抵のモデルは小数の関数で構成することができる。このような関数の集合を関数カタログ (catalog of functions) と呼ぶことがある。関数カタログの有効利用はモデルの構築と実行を容易にする。関数カタログとは、たとえば、恒等関数 (identity function)、脱色関数 (decoloration function)、後者関数 (successor function)、射影関数 (projection function) などについては、次のように定義される:

- 恒等関数:  $\text{id} \langle c_i \rangle = \langle c_i \rangle$
- 脱色関数:  $\text{dec} \langle c_i \rangle = \langle \cdot \rangle$  (ただし、 $\langle \cdot \rangle$  は中性カラーを表わす)
- 後者関数:  $\text{next} \langle c_i \rangle = \langle c_{i+1} \rangle$
- $i$  番目の射影関数:  $\text{proj}_i \langle c_1, c_2, \dots, c_i, \dots, c_n \rangle = \langle c_i \rangle$

同じような理由で、FIFO (First In First Out) や LIFO (Last In First Out) などのような待ち行列の初

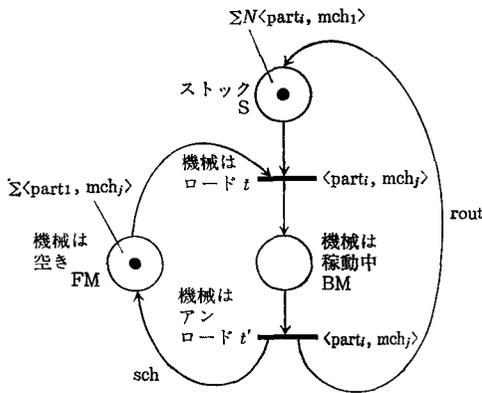


図12.1 フローショップ生産システム(図10.1)のカラーペトリネットモデル

等的構造のいくつかはあらかじめ定義しておくことができる。

### 12.1 フローショップのカラーペトリネットモデル

ここでは、10.2で紹介したフローショップシステムをカラーペトリネットによってモデル化してみよう。このフローショップで重要なことは、図12.1に示すように、3つのプレースに対応するカラー集合  $\langle \text{part}_i, \text{mch}_i \rangle$  を考えることである。まず、機械の中を部品が流れる順序(スケジューリング(scheduling))と各部品がめぐる機械の順序(ラウティング(routing))を定めることが必要である。普通、スケジューリングはヒューリスティックな計算によってオフラインで実行される。この例でのスケジューリング関数は

$$\text{sch} \langle \text{part}_i, \text{mch}_j \rangle = \langle \text{part}_{i+1}, \text{mch}_j \rangle$$

のように簡単な形で記述することができる。また、一般に、ラウティングは、フローショップシステムではきわめて簡単に、

$$\text{route} \langle \text{part}_i, \text{mch}_j \rangle = \langle \text{part}_i, \text{mch}_{j+1} \rangle$$

として与えられる。

図10.1のフローショップシステムをカラーペトリネットによってモデル化したものを図12.1に示す。ラベルの付いていないアークは恒等関数が対応している。

プレースS(ストック)の初期マーキング、すなわち  $M_0(S) = \sum N \langle \text{part}_i, \text{mch}_i \rangle$  は、すべての部品が入庫ストックで機械MCH1による処理を待っていることを表わしている。プレースFM(機械は空き)の初期マーキング、すなわち  $M_0(FM) = \sum \langle \text{part}_i, \text{mch}_j \rangle$  は、すべての機械が空いてPART1の型の部品の到着を待っていることを意味している。

この初期マーキングは、カラー  $\langle \text{part}_1, \text{mch}_1 \rangle$  に関してトランジションtを発火可能にしている。カラー  $\langle \text{part}_1, \text{mch}_1 \rangle$  に関する発火トランジションtによって、PART1が機械MCH1にロードされる(プレースBMにはカラー  $\langle \text{part}_1, \text{mch}_1 \rangle$  が入り機械が稼働中であることを表わしている)。この機械はカラー  $\langle \text{part}_1, \text{mch}_1 \rangle$  に関するトランジションt'が発火することでアンロードされる。カラー  $\text{sch} \langle \text{part}_1, \text{mch}_1 \rangle = \langle \text{part}_2, \text{mch}_1 \rangle$  がプレースFMに付け加えられ(機械MCH1はPART2の型の部品を受け取る用意ができています)、別のカラー  $\text{route} \langle \text{part}_1, \text{mch}_1 \rangle = \langle \text{part}_1, \text{mch}_2 \rangle$  はプレースSに付け加えられる(PART1が機械MCH2に入る用意ができています)。このように、トランジションt'とtの発火によって、フローショップ処理が正しく実行されていることが確認できる。

ところで、このカラーペトリネットモデルでは機械と部品の個数については、まだ何も指定していない。フローショップシステムにおいて、 $n_m$  台の機械と  $n_p$  個の型の部品それぞれ指標  $i \in [1, n_m]$  と  $j \in [1, n_p]$  によって区別することにする。この方法にしたがえば、スケジューリング関数とラウティング関数は、それぞれ

$$\text{sch} \langle \text{part}_{n_p}, \text{mch}_j \rangle = \langle \text{part}_i, \text{mch}_j \rangle$$

$$\text{route} \langle \text{part}_i, \text{mch}_{n_m} \rangle = \langle \text{part}_i, \text{mch}_1 \rangle$$

のように定義することができる。この規則によって、システムはサイクリックにふるまうことをモデル化することができる。その一例として、図12.1のカラーネットモデルにおいて、部品が  $\text{part}_1$  と  $\text{part}_2$  の2個だけの場合のふるまいを図12.2に示す。この場合のカラートークンは○, ●, ⊙, ⊗の4種類である。これによって、カラーネットの意味が容易に把握できよう。

### 12.2 AGVシステムのカラーペトリネットモデル

こんどは、10.3で示したAGVによる輸送システムをカラーペトリネットによってモデル化してみよう。まず、図12.3にカラーペトリネットによってモデル化したものを示す。このモデルでは次の2つの型のカラーを考えている：空きセクションを指示するための  $\langle s_i \rangle$ 、およびセクション  $s_j$  を占領している走行車両のアンロードステーション  $o_i$  を表示するための  $\langle o_i, s_j \rangle$ 。図12.3ではラウティング関数  $r$  を定義しているが、これは車両が占領する次のセクションの“カラー”を伝えるためのものである。車両を進めるためには、次のセクションが空いていることが必要である。これは射影関数でテストすることができる。

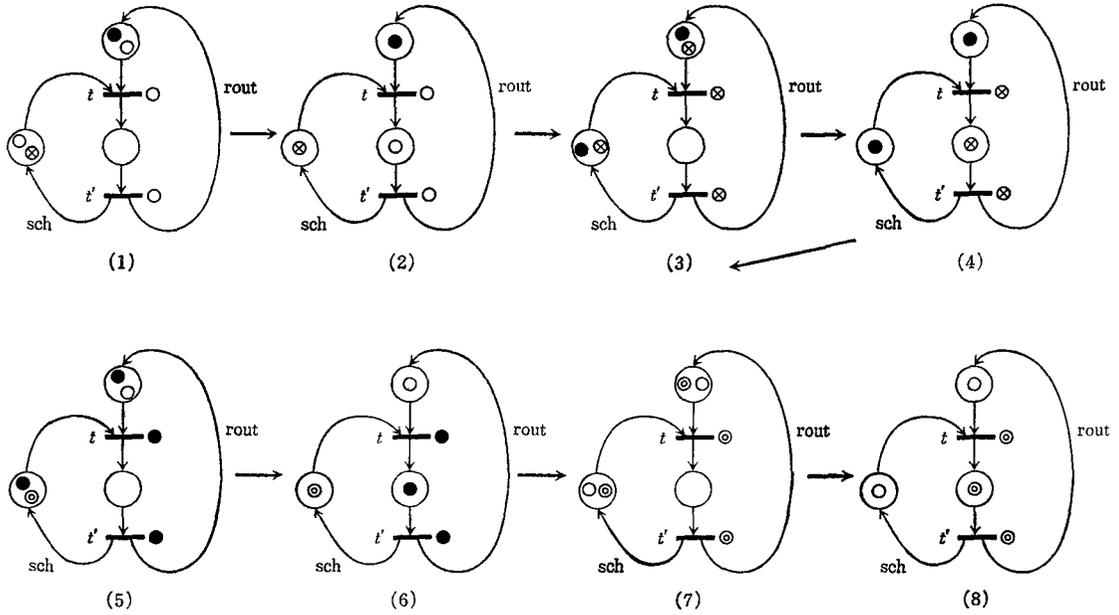
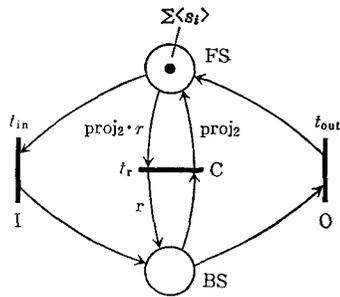


図12.2 図12.1において2個の部品(part<sub>1</sub>, part<sub>2</sub>)の生産処理過程を示す発火系列. ただし, カラー トークンは○=〈part<sub>1</sub>, mch<sub>1</sub>〉, ●=〈part<sub>2</sub>, mch<sub>1</sub>〉, ⊙=〈part<sub>2</sub>, mch<sub>2</sub>〉, ⊗=〈part<sub>1</sub>, mch<sub>2</sub>〉 を意味する.



プレースの意味

FS : セクションが空いている.

BS : セクションが占領されている.

トランジションの意味

t<sub>in</sub> : 車両がロードステーションから出発する.

t<sub>out</sub> : 車両がアンロードステーションに到着する.

t<sub>r</sub> : 車両がその経路で次のセクションに行くことを認める.

カラーの部分集合

$C = \{ \langle o_1, s_1 \rangle, \langle o_1, s_3 \rangle, \langle o_1, s_5 \rangle, \langle o_2, s_1 \rangle, \langle o_2, s_2 \rangle, \langle o_2, s_3 \rangle, \langle o_2, s_4 \rangle, \langle o_2, s_5 \rangle \}$

$I = \{ \langle o_1, s_1 \rangle, \langle o_2, s_1 \rangle, \langle o_2, s_2 \rangle, \langle o_2, s_4 \rangle \}$

$O = \{ \langle o_1, s_6 \rangle, \langle o_1, s_7 \rangle \}$

ラウティング関数 r

$r \langle o_1, s_1 \rangle = \langle o_1, s_3 \rangle, r \langle o_1, s_3 \rangle = \langle o_1, s_5 \rangle, r \langle o_1, s_5 \rangle = \langle o_1, s_6 \rangle, r \langle o_2, s_1 \rangle = \langle o_2, s_2 \rangle,$

$r \langle o_2, s_2 \rangle = \langle o_2, s_3 \rangle, r \langle o_2, s_3 \rangle = \langle o_2, s_4 \rangle, r \langle o_2, s_4 \rangle = \langle o_2, s_5 \rangle, r \langle o_2, s_5 \rangle = \langle o_2, s_7 \rangle$

図12.3 AGVによる輸送システム(図10.3)のカラーペトリネットモデル

$$\text{proj}_2 \langle o_i, s_j \rangle = \langle s_j \rangle$$

のようになる。

プレース F S は空きセクションを表わす。最初の状態では、すべてのセクションが空いているから、プレースの初期マーキングは  $M_0(\text{FS}) = \sum \langle s_j \rangle$  である。

プレース B S は占領セクションを表わす。プレース B S にあるカラー  $\langle o_i, s_j \rangle$  は、アンロードステーション  $o_i$  に向けて走行中の車両がセクション  $s_j$  にあることを示している。

いまこのネットの操作を説明するために、 $o_2$  に向かっている車両はセクション  $s_5$  にあるものと仮定する。したがって、プレース B S はカラー  $\langle o_2, s_5 \rangle$  でマーキングされる。トランジション  $t_r$  は、もしプレース F S がカラー  $\text{proj}_2 \cdot r \langle o_2, s_5 \rangle = \text{proj}_2 \langle o_2, s_7 \rangle = \langle s_7 \rangle$  を含むときそのときに限り、そのカラーに関して発火可能となる。すなわち、セクション  $s_7$  は空いていなければならない。このような場合、カラー  $\langle o_2, s_5 \rangle$  に関する発火トランジション  $t_r$  は、カラー  $r \langle o_2, s_5 \rangle = \langle o_2, s_7 \rangle$  でプレース B S をマーキングし(車両がセクション  $s_7$  を占領している)、カラー  $\text{proj}_2 \langle o_2, s_5 \rangle$  でプレース F S をマーキングする(セクション  $s_5$  が空いている)。

### 12.3 カラーペトリネットによるモデル化の背景

理論的には、カラーペトリネット(カラーは有限個)は通常のペトリネットと同じ表現力を持っている[22]。つまり、カラーペトリネットとペトリネットの表現能力は等価であることが証明されている。実際、これまでみてきた例からわかるように、図10.2と図10.4のペトリネットは、それぞれ図12.1と図12.3のようにカラーペトリネットを用いて等価的にモデル化される。

カラーペトリネットによるモデル化は、主に次のような背景に基礎を置いている：

- (a) システムが同じふるまいをする要素を含んでいる場合のモデル化。カラーは各要素に対応する。
- (b) システムの進行状況は、トランジションを発火するカラーの変換法を定める線形関数によって記述することができる。したがって、カラーペトリネットモデルは、ペトリネットによって構築されたモデルよりも簡潔である。
- (c) カラーペトリネットは、2つの側面からモデル化される：① ネット構造の定義(すなわちプレース、トランジション、アークによる構成)、② アークをラベルづける関数の定義。
- (d) ネットのアークにある関数は明確な物理的意味を持つ

よう定める(たとえば、スケジューリング関数、ラウティング関数など)。

モデルの機能的な内容を修正するためには、ふつうはカラー集合の拡大・縮小および関数の再定義などを行なうことで十分である。したがって、図12.1のフローショップモデルをジョブショップ(job-shop)に変換したい場合には、ネットの関数(sch と rout)を修正するだけで十分であろう。一方、図10.3の輸送システムの機能を修正したいときに必要なのは、カラー I, C, O とラウティング関数  $r$  の部分集合を変更することである。

カラーペトリネットの主たる不利な点は、モデルの構造や解析、および実現のための手法がまだ十分に確立されていないことにある。これはこのツールが比較的新しいことに起因している。事実、カラーペトリネットに関する仕事のはほとんどは1980年代に行なわれている[27]。しかし、最近では使いやすいソフトウェアツールが、商業ベースで開発され[28]、カラーペトリネットによるモデル化とシミュレーションは現実のものとなりつつある\*

ペトリネットの解析に関して明らかにされた多くの性質は、カラーペトリネットに対しても一般化することができる(たとえば、有界性、活性、可逆性、相互排他など)。しかし、これらの性質を解析するための方法はペトリネット同程度には明らかにされていない。文献[23]では、関数の可能性に関するある性質が満たされれば利用可能なインバリアントの特別な計算方法が紹介されている。カラーペトリネットのインバリアントを計算するための一般的な方法は、文献[24]に示されている。また、Huber 他[25]は、カラーペトリネットの可達解析に関して対称な可達マーキングの集合にもとづいた等価マーキングの概念を用いて可達木のサイズが大幅に減少することを示した。

### 12.4 モデルの利用について

さて、このようなネットツールによって構築した離散並行システムのモデルは、シミュレーションやシステムの制御などに使うことができる。シミュレーションは、FMS の設計においてきわめて有用な情報を提供してくれる[26]。たとえば、① ストックレベル(平均値と最大値)、② 機械の利用時間と利用率、③ 1つの部品の製造に対する平均サイクル時間、④ 輸送時間、⑤ 各輸送要素

\*ペトリネットのソフトウェアツールに関する内外の最新情報については、次回(最終回)に紹介される。

に対する利用率などの情報を得ることができる。

FMSのレイアウト、スケジューリング処理、ストックの大きさなどに関する種々のシミュレーションを行ない、その情報を有効に利用することで、FMSを正しく設計することができるが、そのような結果を得るには、カラーペトリネットに8.で述べた時間パラメータを導入することが必要である。(つづく)

## 文 献

(前回までにあげたものは省く)

- [21] K. Jensen: "Coloured Petri Nets and the Invariant Method", Theoretical Computer Science 14, North-Holland, pp. 317-336, 1981.
- [22] J. L. Peterson: "A Note on Colored Petri Nets" Information Processing Letters, Vol. 11, No. 1, pp. 40-43, 1980
- [23] H. Alla, P. Ladet, J. Martinez and M. Silva: "Modelling and Validation of Complex Systems by Coloured Petri Nets: Application to Flexible Manufacturing Systems", G. Rozenberg (ed.), Advances in Petri Nets 1984, Lecture Notes in Computer Science 188, Springer-Verlag, pp. 15-31, 1985.
- [24] M. Silva, J. Martinez, H. Alla and P. Ladet: "Generalized Inverses and the Calculation of Symbolic Invariants for Coloured Petri Nets", Technique et Science Informatiques, Vol. 4, No. 1, pp. 113-126, 1985.
- [25] P. Huber, A. M. Jensen, L. O. Jepsen and K. Jensen: "Towards Reachability Trees for High-Level Petri Nets", G. Rozenberg (ed.), Advances in Petri Nets 1984, Lecture Notes in Computer Science 188, Springer-Verlag, pp. 215-233, 1985.
- [26] P. Alanche, K. Benzakour, F. Dollé, P. Gillet, P. Rodrigues and R. Valette: "PSI: A Petri Net Based Simulator for Flexible Manufacturing Systems", G. Rozenberg (ed.), Advances in Petri Nets 1984, Lecture Notes in Computer Science 188, Springer-Verlag, pp. 1-14, 1985.
- [27] K. Jensen: "Coloured Petri Nets. A way to describe and analyse real-world systems without drowning in unnecessary details", Proceedings of the Fifth International Conference on Systems Engineering, New York: IEEE, pp. 359-401, Dayton, 1987.
- [28] P. Huber: "design/CPN: An Editor and Simulator for Hierarchical Coloured Petri Nets", The Third International Workshop on Petri Nets and Performance Models, Work-in-Progress, Kyoto, Dec. 10, 1989.

---

## 会 合 記 録

庶務幹事会	1月11日(木)	6名
国際委員会	1月11日(木)	6名
表彰委員会	1月12日(金)	9名
理事会	1月19日(金)	16名
編集委員会 (OR誌)	1月22日(月)	12名
研究普及委員会	1月29日(月)	8名

### 第5回理事会議程

1. 平成元年度第4回理事会議事録の件
2. 入退会の件
3. 昭和63年度・平成元年度会費未納者(除名対象者)

の件

4. 次期会長候補者選考委員会報告の件
5. 各委員会報告
  - 財政問題検討委員会第3次報告の件
  - 第3・四半期収支報告の件
  - 研究部会の新設ならびに継続の件
  - 本部定例講演会開催および収支報告の件
  - 第24回シンポジウム(早稲田)および平成3年度秋季研究発表会の件
  - 広告委員会報告の件
  - 平成2年度事業計画・予算案の件
  - OR事典(事例集)編集委員会の件
  - 会友推薦の件