

ペトリネットのシミュレーションへの応用

【II】

——モデル化からシミュレーションへ——

椎塚 久雄

前回は、離散事象システムの基本的なふるまいからペトリネットによるモデル化への入口まで示した。今回は、ペトリネットモデルに親しんでいただくために、まず簡単なネットモデルの作り方を示し、シミュレーションの具体的方法へと進む。

4-3. ペトリネットによるモデル化

すでに述べたように、ペトリネットは、システムにおいて、特に事象と条件という2つの側面とそれらの間の相互関係をモデル化している。この見地からシステムを観察すると、任意の時間にはある条件が成立している。この条件が成立している結果、ある事象が生起する。これらの事象が生起すれば、システムの状態が変わり、その結果、前の条件が成立しなくなり、他の条件が成り立つようになる。事象と条件はシステムをペトリネット的に見る見方の根本であり、モデル化するときの重要なポイントである。

たとえば、図4.3は生産者—コンテナ—消費者システムをペトリネットでモデル化したものである。このシステムの条件と事象は次のようである。

条件

- a. 生産者は仕事を待機中。
- b. コンテナは空。
- c. 生産者がコンテナに詰め替え中。
- d. コンテナは満杯。

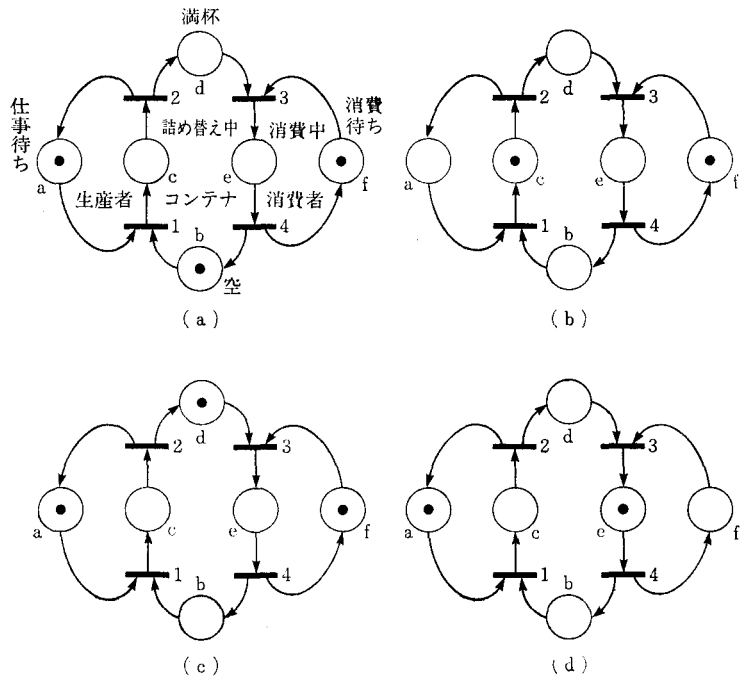


図4.3 生産者—コンテナ—消費者モデル

- e. 消費者が消費中。
- f. 消費者が消費を待っている。

事象

- 1. 生産者がコンテナに詰め替えを開始する。
- 2. 生産者が詰め替えを終了する。
- 3. 消費者が消費を開始する。
- 4. 消費者が消費を終了する。

事象1（生産者がコンテナに詰め替えを開始する）の前提条件（preconditions）は、a（生産者は仕事を待機中）、およびb（コンテナは空）であることは明らかである。事象1の後提条件（postconditions）は、c（生産者がコンテナに詰め替え中）である。同様にして、各事

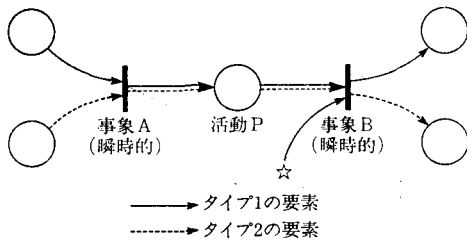


図5.1 ネット形での離散事象システムの表現

象に対する他の前提条件と後提条件をまとめると次のようになる。

事象	前提条件	後提条件
1	a, b	c
2	c	a, d
3	d, f	e
4	e	b, f

初期マーキングは図4.3 (a)のようにトークンが割り当てられている。すなわち、コンテナが空いていて生産者が仕事を待っている状態を表わしている。トランジション1の入力プレース a, b にトークンがあるので、トランジション1は発火可能である。その他のトランジションはいずれも発火可能ではない。トランジション1が発火すれば、トークンが移動して図4.3 (b)に示すマーキングが得られる。この状態ではトランジション2のみが発火可能である。トランジション2の発火後のマーキングは図4.3 (c)となり、ここで新たにトランジション3が発火可能となる。トランジション3を発火させると、マーキングは再び元の状態に戻る(図4.3(a))。

このように、ペトリネットのマーキングはトランジションの発火によって変更され、マーキングが異なれば、発火可能なトランジションも異なってくる。トランジションの発火は、発火可能なトランジションが存在する限り継続して行なわれる。このように、トークンがペトリネット内を動きまわるようすは、システムの動的なふるまいを表わし、これからシミュレーション結果をはじめとして重要な情報が得られるのである。

5. 離散事象システムとペトリネットの間の対応

一般に離散事象システムにおいては、いくつかのタイプの要素が存在している。つまり、それぞれの属性をもった要素がシステム内を流れている。これはペトリネットではカラートークン*を割り当てることに相当し、ト

* カラーペトリネットについては後の章で定義される。

表5.1 ネットの要素と離散事象システムの対応

ペトリネット	離散事象システム
トランジション	⇔ 事象
トランジションの発火	⇔ 事象の生起
プレース	⇔ 活動 (activity)
トークン	⇔ 要素 (entity)
カラートークン	⇔ あるタイプの要素
アーク	⇔ 要素の流れの方向
カラーアーク	⇔ 要素のタイプにしたがう路

ークンは個々にある種の属性(情報)を持つことになる。トークンはトランジションにおいて生成されたり壊されたりする。さらに、トランジションの発火は、合成トークンを形成させるためにトークンの合体、あるいはより小さな個々のトークンへ分裂させる結果をもたらす。また、3.で示したように、離散事象システムにおける事象は瞬間的に発生し、事象と活動を区別した。これらをペトリネットで表現することは容易である。

離散事象システムにおける要素の流れ、合体、事象、活動などをネットの形で示すと図5.1のようになる。この関係をエンゲージメント(engagement)と呼ぶこともある[7]。エンゲージメントとは次の4つの関連した概念：①要素の状態あるいは活動、②状態の到達を示すためのデータ構造、③要素間の合体の形成、④処理の進行段階、を表わすのに用いられる。ペトリネットの要素と離散事象システムの構成要素との間には、表5.1に示すような対応関係がある。一方、Shannon[10]は、離散事象シミュレーションを次のように特徴づけている：

ENTITIES (要素)

having ATTRIBUTES (属性)

interact with ACTIVITIES (活動)

under certain CONDITIONS (条件)

creating EVENTS (事象)

that change the STATE (状態)

これらのことから離散事象シミュレーションとペトリネットはきわめて密接な対応関係にあることがわかる。

6. ペトリネットの一般的な構造とその定義

ペトリネットモデルの基本的概念に関してはすでに4.で述べたが、ここでは、さらに一般的な形でネットの定義をしておこう。

ペトリネットは2つの基本構造, すなわち, プレースの集合 P とトランジションの集合 T から定義されている. 完全に定義するためには, プレースとトランジションの間の関係を定義しなければならない. これは, トランジションとプレースをつなぐ2つの関数, すなわち, 入力関数 I と出力関数 O を規定すればよい. 入力関数 I と出力関数 O は, 各トランジション t_i に対して, それぞれ入力プレースの集合 $I(t_i)$ と出力プレースの集合 $O(t_i)$ を決めるものである.

定義1 5項組 $N=(P, T; F, C, W)$ は, 次の場合に限りネットであるという.

- (1) $(P, T; F)$ はネットである. ただし, P の要素はプレースと呼ばれ, T の要素はトランジションと呼ばれる. $F \subseteq (P \times T) \cup (T \times P)$ は2項関係でネット N における流れの関係である.
- (2) $C: P \rightarrow N^+ \cup \{\infty\}$ は容量関数 (capacity function).
- (3) $W: F \rightarrow N^+$ は重み関数 (weight function).

ただし, N^+ は正の自然数の集合を表わす.

ここでは, $|P| < \infty, |T| < \infty$ であるような有限なペトリネットのみについて考える.

このようなネットの発火規則は, 次に示すような定義にしたがって実行される.

定義2 $N=(P, T; F, C, W)$ をネットとする.

- (1) N は, もし $\forall (p, t) \in P \times T: (p, t) \in F \Leftrightarrow (t, p) \notin F$ であるならば, そのときに限り純 (pure) であるという.
- (2) 関数 $M: P \rightarrow N$ は N のマーキングと呼ばれる.
- (3) トランジション $t \in T$ は, もし $\forall p \in I(t): M(p) \geq W(p, t)$ かつ $\forall p \in O(t): M(p) + W(t, p) \leq C(p)$ であるならばそのときに限り N のマーキング M のもとで発火可能であるという.

- (4) トランジション $t \in T$ は N のマーキング M のもとで発火可能であるとする. そのとき, t は次のように与えられる N の新しいマーキング M' を発生しうる:

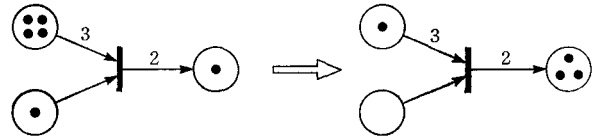
$M'(p) = M(p) - W(p, t)$ (すべての $p \in I(t)$ に対して)

$M'(p) = M(p) + W(t, p)$ (すべての $p \in O(t)$ に対して)

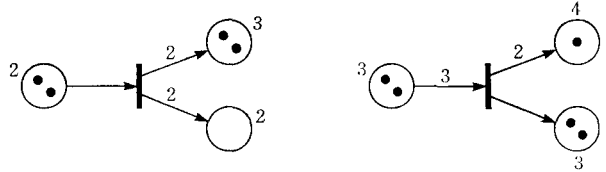
$M'(p) = M(p)$ (すべての $p \notin I(t) \cup O(t)$ に対して)

M' を M の直接次段マーキング (immediate follower marking) と呼ぶ. 一般に M の次段マーキングの集合とは, 直接次段マーキングに続くマーキングの集合を意味する.

以上のような定義にしたがうペトリネットは, 一般に



(a) トランジションの発火



(b) トランジションが発火不可能な場合

図6.1 プレースに容量をつけ, アークに重みを持つ一般的なネットの発火規則の適用例

プレース/トランジションネット (P/T ネット) と呼ばれているが[4], これはアークに重みをつけ, プレースに容量をつけて4-2. で示したペトリネットを一般化したものと考えてもよい. たとえば, 図6.1 (a)は重みをつけたアークと, 容量を持つプレースの発火の状態を示し, 図6.1 (b)は発火不可能な場合の例を表わしている. ただし, アークの重みが1の場合と, プレースの容量が特に指定されていない場合は何もつけない.

7. 簡単なシステムのモデル化とシミュレーション例

ある機械工場で各機械とオペレータの稼働状況を調査することが必要になったことを想定してみよう. そこで, 特に各機械とオペレータの空きの状態, およびオペレータの機械操作の状態を調べることにする.

7-1. モデル化[3]

この機械工場には3種類の機械 M_1, M_2, M_3 が設置され, 2人のオペレータ O_1, O_2 がいる. オペレータ O_1 は機械 M_1 と M_2 , オペレータ O_2 は機械 M_1 と M_3 を操作できる. 注文品は2段の加工を必要とする. 最初の M_1 で加工し, 次に M_2 と M_3 のどちらかで加工する. このシステムは次に示す条件と事象をもつことになる.

条件 (数字は図7.1(a)のプレース番号に対応):

5. 注文品が到着し, M_1 による加工を待機中.
6. 注文品が M_1 による加工を受け, M_2 または M_3 による加工を待機中.
12. 注文品が完成.

- 3. 機械 M_1 は空き.
- 7. 機械 M_2 は空き.
- 10. 機械 M_3 は空き.
- 1. オペレータ O_1 は空き.
- 11. オペレータ O_2 は空き.
- 2. オペレータ O_1 が機械 M_1 を操作中.
- 4. オペレータ O_2 が機械 M_1 を操作中.
- 8. オペレータ O_1 が機械 M_2 を操作中.
- 9. オペレータ O_2 が機械 M_3 を操作中.

事象 (数字は図7.1(a)のトランジション番号に対応):

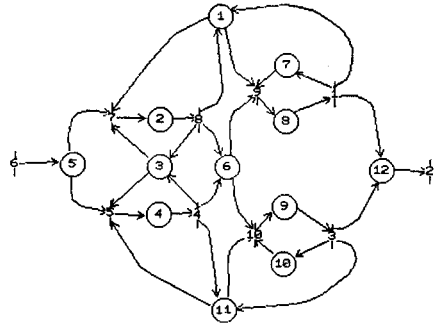
- 6. 注文品が到着する.
- 7. オペレータ O_1 が機械 M_1 を用いて 注文品の加工を開始する.
- 8. オペレータ O_1 が機械 M_1 を用いて 注文品の加工を終了する.
- 5. オペレータ O_2 が機械 M_1 を用いて 注文品の加工を開始する.
- 4. オペレータ O_2 が機械 M_1 を用いて 注文品の加工を終了する.
- 9. オペレータ O_1 が機械 M_2 を用いて 注文品の加工を開始する.
- 1. オペレータ O_1 が機械 M_2 を用いて 注文品の加工を終了する.
- 10. オペレータ O_2 が機械 M_3 を用いて 注文品の加工を開始する.
- 3. オペレータ O_2 が機械 M_3 を用いて 注文品の加工を終了する.
- 2. 注文品を配送に送る.

このような条件と事象を伴うシステムは、図7.1に示すようなペトリネットモデルでモデル化される。図7.1(a)には、各トランジションとプレース番号がつけられている。図7.1(b)はこの場合の初期マーキングを割り当てたペトリネットを示す。

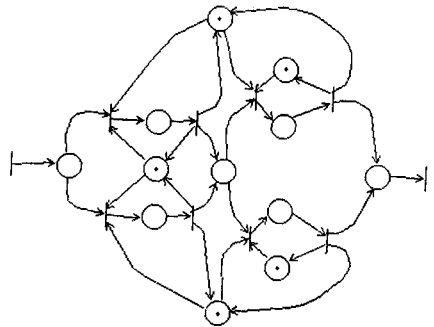
7-2. シミュレーションの実例

図7.1(b)の初期マーキングからスタートする。これは、2人のオペレータが仕事を待っていて、3台の機械が空いている状態である。ペトリネット・シミュレータ[9]を用いて、トランジションの発火順序をランダムに設定し、シミュレーションする。

注文品の総数を200個とする。200個がすべて加工されるためには、このネットモデルのトランジションの発火の回数は800回である(ただし、トランジション2と6は除く)。トランジションの発火によるシステムの動的ふるま



(a) プレースとトランジションの番号づけ



(b) 初期マーキング

図7.1 例題の機械工場のペトリネットモデル (グラフィック・エディタ[9]による作成)

いのデータ、すなわち、プレースからの情報(活動状況)とトランジションからの情報(事象発生状況)のデータ収集を行なう。

表7.1はプレースから得られる情報、すなわち、各機械とオペレータの稼働状況の統計を示す。これに対して、表7.2は各トランジションの発火回数から得られる情報、すなわち、オペレータが各機械を用いて加工の開始と終了に関する統計を示す。

表7.1からオペレータの空きの状態と機械操作との関係については、

オペレータ O_1 に関して:

$$\begin{aligned} & \text{空き}(48.1\%) + M_1 \text{を操作中}(26.3\%) \\ & \quad + M_2 \text{を操作中}(25.6\%) = 100\%, \end{aligned}$$

オペレータ O_2 に関して:

$$\begin{aligned} & \text{空き}(51.6\%) + M_1 \text{を操作中}(12.9\%) \\ & \quad + M_3 \text{を操作中}(35.6\%) = 100\%, \end{aligned}$$

となっていることがわかる。また、各機械の稼働状況については、

表7.1 例題の出力結果 (稼働率)

トランジションの発火総数 = 800回	
状 態	率 (%)
機械 M_1 は空き	60.8
機械 M_2 は空き	74.4
機械 M_3 は空き	64.4
オペレータ O_1 は空き	48.1
オペレータ O_2 は空き	51.6
オペレータ O_1 が機械 M_1 を操作中	26.3
オペレータ O_2 が機械 M_1 を操作中	12.9
オペレータ O_1 が機械 M_2 を操作中	25.6
オペレータ O_2 が機械 M_3 を操作中	35.6

機械 M_1 に関して :

空き(60.8%) + O_1 が操作中(26.3%)
 + O_2 が操作中(12.9%) = 100%,

機械 M_2 に関して :

空き(74.4%) + O_1 が操作中(25.6%) = 100%,

機械 M_3 に関して :

空き(64.4%) + O_2 が操作中(35.6%) = 100%,

となっていることがわかる。

このネットモデルでは、加工に要する時間は考慮していないが、後の章で述べる時間ペトリネットを用いることによって発火条件を満たしてもトランジションの発火を一定時間遅らせることで、ペトリネットに時間の概念

表7.2 例題の出力結果 (事象の発生率)

トランジションの発火総数 = 800回		
事 象	発 生 率 (%)	
オペレータ O_1	機械 M_1 を用いて注文品の加工開始	15.2
	機械 M_1 を用いて注文品の加工終了	15.2
オペレータ O_1	機械 M_2 を用いて注文品の加工開始	7.9
	機械 M_2 を用いて注文品の加工終了	7.9
オペレータ O_2	機械 M_1 を用いて注文品の加工開始	9.8
	機械 M_1 を用いて注文品の加工終了	9.8
	機械 M_3 を用いて注文品の加工開始	17.1
	機械 M_3 を用いて注文品の加工終了	17.1

を導入することができる。

(つづく)

文 献

(前回にあげたものは省く)

[10] R. E. Shannon : "System Simulation : the Art and Science", Prentice-Hall, 1975.

新しいコラム "OR メモランダム" へぜひご投稿を

—みなさん、こんな話、アイディア、経験は、ありませんか？

平成2年1月号から、上記の新しいコラムを設けます。このコラムは、ORにかかわる概念、原理、知識(手法、定理)、それらの図解、よい教材や問題、実学ORの実施経験、そこから得られた知恵やアドバイス、失敗談と教訓、新しい問題提起、新しい観点、視座、フレームワーク、未だ解けていない問題、面白研究テーマなどを、"新鮮に"、しかも、"コンパクト"に表現し、提示していただくものです。

だれでも、自分だけにしまっておくにはもったいな

いユニークなアイディアや概念、フレッシュな見方、発想、他の会員(読者)に伝えて、意見をたたかわせたい問題提起などがあるのではないのでしょうか。どうアプローチしたらよいか分からない研究テーマなどもあるに違いありません。ふるってご投稿ください。

(原稿は、刷り上がり半ページから3ページに納まるようにお書きください。なるべく、コンパクトに！加筆訂正をお願いする場合があります。)