

ペトリネットを用いたロボットシステムの制御

Control of Robot System Using Petri Net

○船見洋祐*、工藤亨**、渡部慶二*

○Y.Funami*、S.Kudo**、K.Watanabe*

*山形大学、**富士通株式会社

*Yamagata University、**Fujitsu Ltd.

キーワード： ユーザーインターフェイス (User Interface)、
カラーペトリネット (Colored Petri Net)、
階層化 (Hierarchy)、アーキテクチャ (Architecture)

連絡先： 〒 992 山形県米沢市城南 4-3-16 山形大学工学部 電子情報工学科 渡部研究室
船見洋祐、Tel.: (0238)24-3325、Fax.: (0238)24-2752、E-mail: funami@ewata.yz.yamagata-u.ac.jp

1. まえがき

今日、ロボットは、様々なフィールドで効果的に適用されている。大まかに二分すると、多機能ロボットと単機能ロボットのどちらかに属する。前者は、汎用的な作業をすることができるが、開発コストがかかる、作業速度が遅い、作業精度が上がらないなどといった問題がある。一方後者は、そのような問題はないが、一基の単機能ロボットのみでは汎用性に欠ける。汎用性をもたせるためには、目的に応じた単機能ロボットを複数組み合わせることで解決される。さらに、これらシステムのオートメーション化を行うことを考える。

この場合、システムが大きくなることが懸念される。また、各ロボットの作業順序、非同期的並列動作に伴う協調制御が不可欠である。複数の単機能ロボットの協調制御をするためには、膨大な把握し難いシステムの管理が要求される。

ユーザーとロボット間の意思伝達は、コンピュータを介したプログラミング言語によって行われるのが一般的である。運営するシステムが複雑であれば、プログラミングは相当な熟練者でなければ不可能であり、また、熟練者であっても大きな負担となる。さらに、プログラミングでは非同期的な並列動作の記述は困難なものである。もしプログラミング

言語が、並列動作を記述でき、理解しやすい記号や図形であれば、システムの保守運営は非常に扱いやすくなるであろう。簡単にシステムの保守運営を行えるシステムツールであれば、その問題が解消される。

このようなマン・ロボットインターフェイスは、FMS(フレキシブル生産システム)のような非同期的かつ並列的なシステムの振る舞いに対して、その中の情報の流れや制御を記述し、解析するのに有能なグラフィック言語であればよい。そこで、そのような性質を持つグラフィカルな言語であるペトリネットを導入する。

本研究では、大規模システムを扱う際に、各ユニットやシステム全体の状態の把握、システムの変更などを安易に行えることを目的とする。それには、情報をもったトークンを導入し、ネットの階層化をはかることでシステム表現の簡略化を行う。以上のものより、ペトリネットを用いたロボットシステム制御のアーキテクチャを提案する。

2. ペトリネット

離散的な事象の発生で特徴づけられるシステムを離散事象システムという。

2.1 離散事象システムとペトリネット

連続事象システムのモデル化は、その特性の入出力関係に注目している。伝達関数や状態方程式などがその例である。一方、離散事象システムでは入出力関係のかわりに因果関係に注目し、システムのモデル化をはかる。

このような観点から見た離散事象システムの基本モデルを Fig.1 に示す。これは、入力条件

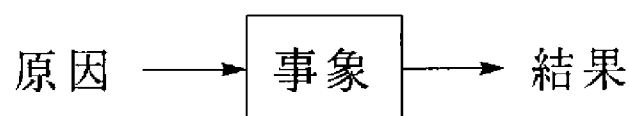


Fig. 1 基本離散事象モデル

(原因) が成立すると事象が生起し、その結果、新たに出力条件(結果)が成立するというものである。

この基本離散事象モデルに対するペトリネット表現を Fig.2 に示す。ペトリネットでは、

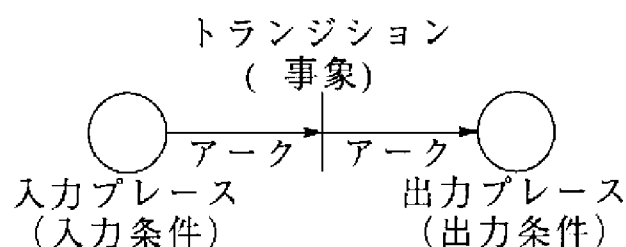


Fig. 2 ペトリネットモデル

入出力条件をプレースと呼ばれる丸印“○”で表し、入力側を入力プレース、出力側を出力プレースと呼ぶ。事象をトランジションと呼ばれる棒“|”で表す。これらプレースとトランジションの接点はアークと呼ばれる矢印“→”で結び、これは、有向性をもつ。

2.2 ペトリネットの動的性質

フローチャートがコンピュータプログラムの静的な性質をモデル化しているように、ペトリネットをグラフィックに表現することはシス

テムの静的性質を表現している。Fig.2はペトリネットの静的構造を表している。これに対して、ペトリネットは、それを実行して得られる動的な性質も持っている。ペトリネットの実行は、プレースに置かれたトークンと呼ばれる黒丸“●”の位置とその動きによって制御される。プレースにトークンを割り当てることをマーキングといい、システムの初期状態には、初期マーキングが割り当てられている。トークンの動きは発火規則に従い、トークンがペトリネット内を動きまわる様子、すなわち状態遷移は次のような規則に従っている。

- トークンは、ペトリネットのトランジションを発火させることによりネット内を移動する。
- トランジションを発火させるためには、トランジションが発火可能でなければならない。
- トランジションのすべての入力プレースにトークンがあるとき、トランジションは発火可能である。
- トランジションが発火すると、その入力プレースからトークンを取り除き、新しいトークンを生成してそれを出力プレースに置く。

この様子を Fig.3に示す。Fig.3(a) は初期マーキングである。このとき、トランジション t_1 は、入力プレースにトークンがあるので発火可能であるが、トランジション t_2 は、発火可能ではない。Fig.3(b) はトランジション

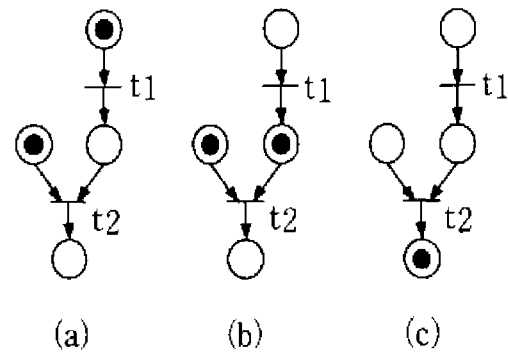


Fig. 3 ペトリネットの状態遷移

t_1 の発火後である。入力プレースのすべてにトークンが存在したので、トランジション t_2 は発火可能であり、Fig.3(c)はその発火後である。

2.3 例1

ロボットアーム R_1 がM地点にある物体を掴んで、L地点でロボットアーム R_2 に渡し、ロボットアーム R_2 がM'地点に物体を置くという作業を考える。この時のペトリネット表現と各トランジションの作業内容は Fig.4となる。

ここで注意すべきことは、トランジション t_2C は、バッファプレースにより、 t_1L が発火しなければ発火することができないことである。つまり、ロボットアーム R_1 がL地点に行く前にロボットアーム R_2 がグリップを閉じることを避けている。同様に、物体の手渡しでは、ロボットアーム R_2 がグリップを閉じてからロボットアーム R_1 がグリップを開き、その後始めて、ロボットアーム R_2 がM'地点に移動する。このような、作業の順序付けを簡

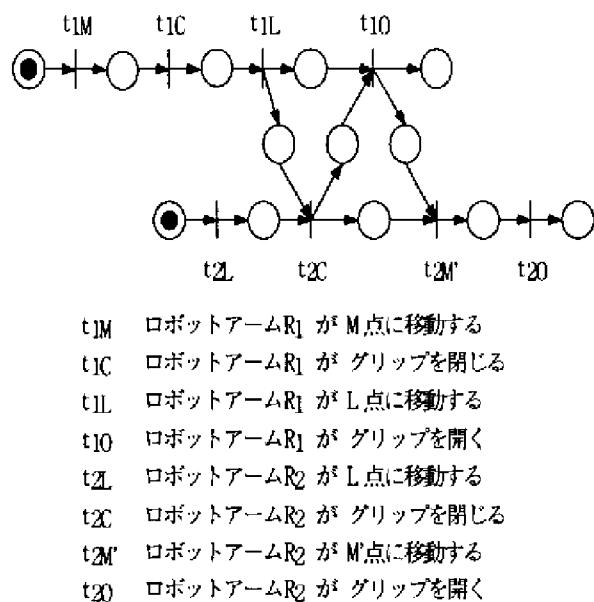


Fig. 4 例1のペトリネット表現

単に行えることが、ペトリネットの利点の一つである。

3. システム制御ツール

本研究では、マン・マシンインターフェイスあるいはマン・ロボットインターフェイスとして、ペトリネットを用いてロボットシステムを制御する。この概念図を Fig.5に示す。

そのシステム制御ツールとして、パソコン用ペトリネットシミュレーションツール SANET¹⁾をもとに改良し開発したものをしている。その特徴は、

- ほぼすべての作業がマウス操作で行うことができる
- システムの動作はネットの実行にあわせて制御される

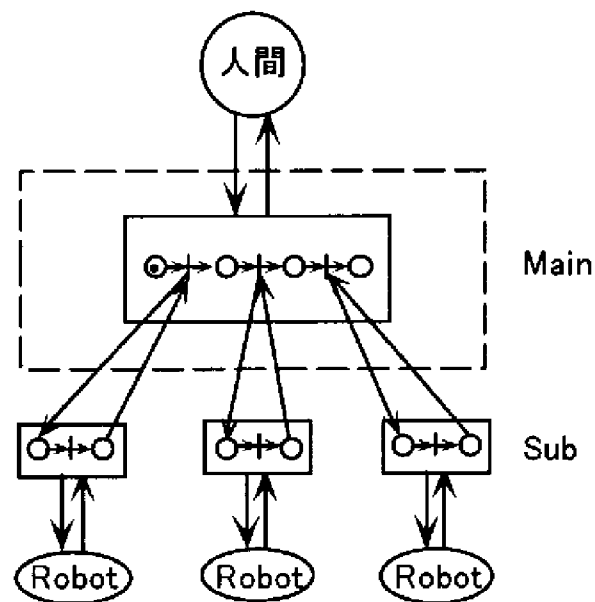


Fig. 5 ペトリネットによるシステム制御

- 紙面に書くのと同じような感覚でネットの変更、修正が可能
- シミュレーションにより正常な動作を確認してからシステムを動かすことができる

というものである。このツールの画面構成を Fig.6に示す。この画面は、

- ペトリネットグラフ表示ウィンドウ
- マウスポインティングカーソル
- コマンド選択用メニューウィンドウ
- 補助表示ウィンドウ
- メッセージウィンドウ

で構成されるものであり、マウスベースでシステム制御のネットを作成することができる。

このシステムツールは改良の余地がある。それは、ある一つのトランジションの発火に

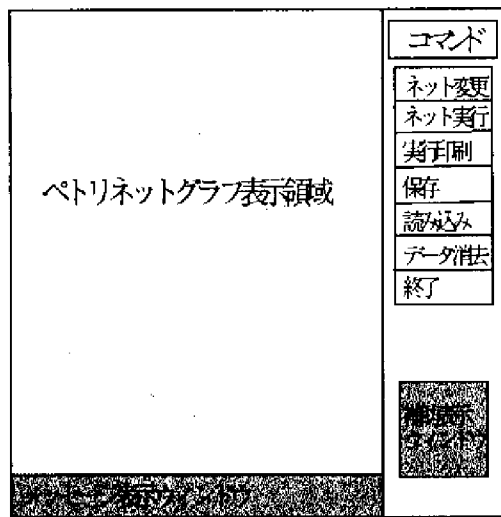


Fig. 6 画面構成

より、あらかじめ決められたある一つの作業しか行えないことである。また、ネットが膨大であると、システム状態の把握が困難なことである。そこで、これらの改良のために、前者には情報をもったトークンを導入し、後者にはネットの階層化をはかる。

4. カラーペトリネット

4.1 カラートークン

前節で述べたトークンの情報は、本研究では、ロボットアーム R_1 、 R_2 のどちらで物体を取りに行ったかとその取りに行った位置が必要である。これらをそれぞれ変数 R 、 M とする。そこで、 R と M という情報を持ったトークンとしてカラートークンを導入する。カラートークンとは、ノーマルネットにおけるトークンにある属性をもたせ、それに対応する発火規則を新たに定めることで、より簡潔なモデルの構築ができるものである。本研

究では、カラートークンを以下のように定義する。

$$CT \equiv (R, M)$$

例えば、 $R = 1$ のとき赤であり $R = 2$ のとき青などといったものである。

4.2 トランジション

カラートークンにより発火するトランジションを次のように定義する。

$$T_R X_k (k = 1, 2, \dots)$$

ここで、 R はユニットナンバー、 X_k は作業内容表し、その詳細をそれぞれ Table.1、Table.2 に示す。

Table 1 ユニットナンバー R

R	ユニット
0	CCD カメラ
1	ロボットアーム R_1
2	ロボットアーム R_2

Table 2 作業内容 X_k

X_k	作業内容
D	物体の位置情報取得 カラートークンの生成
M_k	移動 (M, M', L)
G_k	グリップの開閉 (O, C)
N_k	なにもしない

また、 X_k の詳細を Table.3 に示す。

さらに、カラートークンにより発火するトランジションの発火規則とそのネットの構造的特性を次のように定義する。

Table 3 X_k の詳細

M	物体の初期位置
M'	手渡し後に物体の置かれる位置
O	グリップのオープン
C	グリップのクローズ

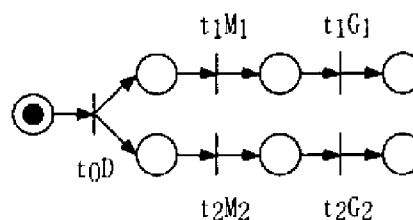
- ユニットナンバー R によりトランジションの発火に対応する作業内容が異なる。
- その作業内容はあらかじめそれぞれのトランジションがもつ。
- その他の発火規則はノーマルネットでの発火規則にしたがう。
- ネットの構造的特性はノーマルネットで定義されているよう同様に扱う。

4.3 例 2

カメラによる位置情報により適切なロボットアーム R_1 または R_2 で M 地点にある物体を掴む作業を考える。この時のペトリネット表現とカラートークンによる各トランジションの作業内容を Fig.7 に示す。ここでは、トランジションの発火に対応する作業 (M , C) を一方のロボットアームが行えば、もう一方のロボットアームのトランジションの発火に対応する作業内容は、なにもしないことである。

4.4 例 3

カメラにより物体の位置 M 点を計算し、適切なロボットアーム R_1 または R_2 で物体を掴み、 L 地点で手渡しを行い、手渡された口



```

IF R = 1 THEN M1 = M
              M2 = N
              G1 = C
              G2 = N

IF R = 2 THEN M1 = N
              M2 = M
              G1 = N
              G2 = C
    
```

Fig. 7 例 2 のペトリネット表現

ロボットアームが M' 地点に物体を置くという作業を考える。この時のペトリネット表現とカラートークンによる各トランジションの作業内容を Fig.8 に示す。これにより、2基のロボットアームによる物体の手渡しという作業をカラートークンを用いたペトリネットで制御できる。

5. ネットの階層化

これまで図形や記号を用いてシステムの保守運営を扱いやすくしてきたが、前節の Fig.8 のようにネットが大きくなるとシステム状態の把握が困難になる。よって、ネットを縮小化をはかり、システム状態の把握を安易にする必要がある。

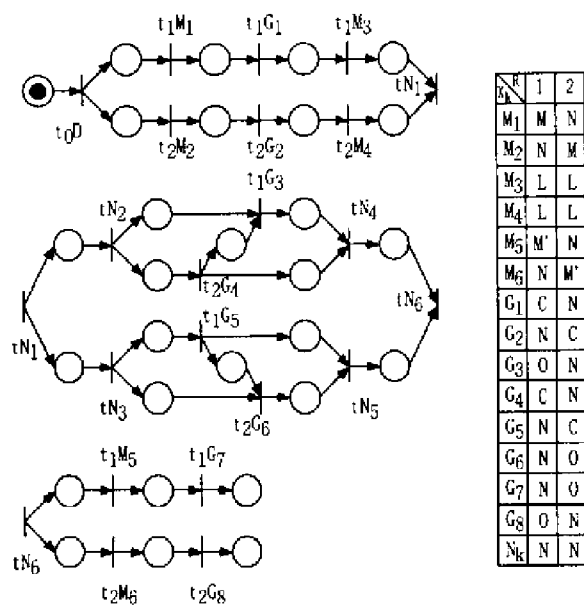


Fig. 8 例2のペトリネット表現

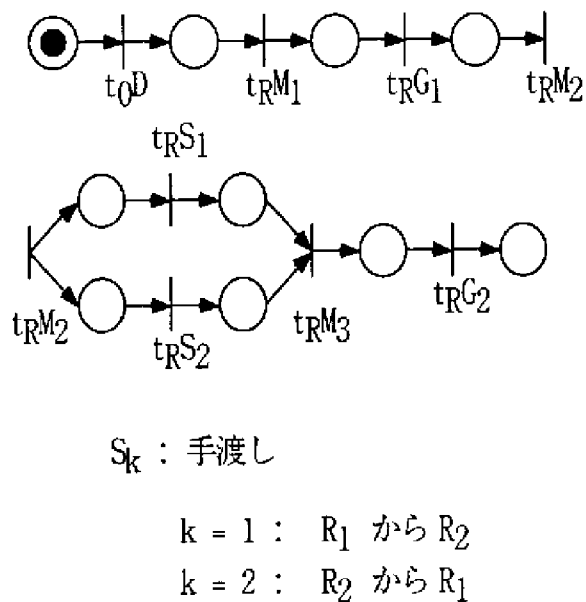


Fig. 9 Fig.7の簡略化

5.1 ネットの簡略化

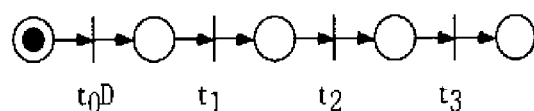
まず、前節の Fig.8で用いたユニットナンバーを変数のままで扱いネットを表現すると、Fig.9のように簡略化される。ただし、ここで新たな作業 S_k を導入する。作業 S_k は、物体の手渡しを行う作業であり、 k が 1(2) のときは、ロボットアーム $R_1(R_2)$ からロボットアーム $R_2(R_1)$ への手渡しを意味する。

5.2 ネットのボトムアップ表現

さらに簡略化をはかるために Fig.9をコンポーネントへ分割する。この分割は、ある程度まとまった作業に分割するのが好ましいので、ここでは、

- 物体を掴みに行く
- 物体の手渡しを行う
- 物体を置きに行く

という作業内容のもとでそれぞれをコンポーネントとする。この結果、最終的には Fig.8 が Fig.10のような大変シンプルなネットで表現できることになる。



- t_1 ロボットアーム R_1 または R_2 が物体を掴みに行く
- t_2 ロボットアーム R_1 と R_2 による物体の手渡し
- t_3 ロボットアーム R_1 または R_2 が物体を置きに行く

Fig. 10 ネットの上位レベルでの記述

6. あとがき

本研究では、使用者の負担を減らし、誰もが簡単にシステムの保守運営を行えることを目的とし、事象と情報の分離およびネット

の階層化を行ったカラーペトリネットを用いたロボットシステムの制御アーキテクチャを提案した。これにより、詳細ネットにより階層化されたコンポーネントがそれぞれロボットの作業を行い、ユーザーはシンプルな上位ネットのみを扱うことで、大規模なシステムの保守運営を簡単に行うことができる。

今後の研究では、そのシステムを扱うためのツールを開発し、実際にシステムを構築し、実験を行う。また、どのようなシステムに対しても適応できるより一般化したカラートークンの定義を考察する。さらに、その実験結果をもとに、本研究のカラーペトリネットのシステムに対する妥当性、逆に、本研究のカラーペトリネットに最も有効なシステムは、いかなるものであるかなどを議論していく。

本研究を進めていく上で、御指導、御鞭撻をいただいた山田功助手に深く感謝いたします。

参考文献

- 1) 椎塚久雄：実例ペトリネット、コロナ社 (1995)
- 2) 熊谷貞俊、薦田憲久：ペトリネットによる離散事象システム論 (1995)
- 3) 工藤亨：ペトリネットによる複数ロボットの協調制御、山形大学大学院工学研究科修士論文 (1997)