Osaka University Knowledge Archive

Title	ワークフロー記述向きの時間付きカラーペトリネット
Author(s)	岡野, 浩三; 山口, 昭; 谷口, 健一
Citation	電子情報通信学会技術研究報告. CST, コンカレント工学. 2002, 102(96), p. 13-16
Version Type	VoR
URL	https://hdl.handle.net/11094/27428
rights	Copyright © 2002 IEICE
Note	

Osaka University Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

Osaka University

Institute of Electronics, Information, and Communication Engineers

社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.
CST2002-4 (2002-05)

ワークフロー記述向きの時間付きカラーペトリネット

岡野浩三† 山口昭†,†† 谷口健一†

† 大阪大学大学院情報科学研究科 〒560–8531 豊中市待兼山町 1–3
†† 現在 三菱電機株式会社
E-mail: †{okano,taniguchi}@ist.osaka-u.ac.jp

あらまし 本研究ではワークフローを記述するための，時間制約付きカラーペトリネット (CPN-t) を提案する．CPN-t を用いて，作業の中断再開およびそれにともなう経過累積作業時間を表現でき，時間制約を持つグループワークを自然に表現することができる．また，CPN-t で記述されたワークフローから各作業者のスケジュールを線形制約式を用いて導出する方法を提案する．

キーワード カラーペトリネット, 時間制約, ワークフロー, 線形計画法

Coloured Petri-nets with time constraints for workflow descriptions

Kozo OKANO†, Akira YAMAGUCHI†,††, and Kenichi TANIGUCHI†

† Graduate School of Information Science and Technology, Osaka University
1–3 Machikaneyama, Toyonaka-shi, 560-8531 JAPAN
†† currently, Mitsubishi Electric Corp.
E-mail: †{okano,taniguchi}@ist.osaka-u.ac.jp

Abstract This paper presents semantics of coloured Petri-nets with time constraints (CPN-t). In this model, we can naturally describe a workflow of group-work with time constraints, such as suspend/resume of work. Also this paper presents a method to derive a schedule for each worker from a given description of workflow in CPN-t. This method uses linear programming to derive schedules from a given description.

Key words coloured Petri-net, time constraint, workflow, linear programming

1. はじめに

ペトリネットは Workflow Management の表現手段として有用である [1]．グループワークをより適切に表現するためには，ペトリネットに時間の概念を導入しなければならない．これまで様々な時間ペトリネットが提案されている [3], [8]．また，ワークフロー記述への適用例もいくつか報告されいている [4], [5], [6], [7]．実際のグループワークでは，作業できない時間帯があったり，ある作業中に他の作業 (協調作業など) を優先させることがある．しかしながら一般の時間ペトリネットを用いたワークフローでは，このようなグループワークの動作を表現すると煩雑になり，自然な記述にはならない．なぜなら，トランジションの発火をアトミックに扱っており，発火の途中で中断することができず,「作業の中断再開およびそれにともなう各作業毎の経過累積作業時間の自然な表現ができない」からである．著者らはこの点をふまえて時間制約付きカラーペトリネット (CPN-t) を [9] で提案した．CPN-t では，トランジションの発火途中で発火を停止し，時間の経過にともない，発火を再開したり，他のトランジションを発火したりすることができる．それをワークフローに適用すると，作業の中断，再開，他の作業の割込みなどに自然に対応できる．

本稿では，文献 [9] で提案した，ワークフローから作業者各自の作業スケジュールの導出法とは異る導出方法について新たに提案する．この導出方法は線形計画法に基づいており，スケジュール導出目的に応じたパラメータ調整が容易にできるという利点がある．

以後，2. では CPN-t について述べる．3. では CPN-t を用いたワークフローの記述例を挙げる．4. ではスケジュール導出について述べ，5. でまとめる．

2. CPN-t

ペトリネットの拡張の一つであるカラーペトリネット (CPN) [3] をもとに CPN-t を定義する．

[定義 2.1] [構文]

CPN は次の 9 項組 $\langle \Sigma, P, T, A, N, C, G, E, I \rangle$ である．Σ はカラー集合と呼ばれる有限集合であり，型 (すなわちトークン)

NII-Electronic Library Service

Institute of Electronics, Information, and Communication Engineers

の集合である. P,T,A は通常のペトリネットと同様に, プレース集合, トランジション集合, アークの集合を表す. N は個々のアークに対し, 始点と終点を与えるノード関数の集合である. C(カラー関数) は, プレースに対し, そこに存在できるトークンを表す関数である. G はトランジションに付与されるガード関数である. 関数の引数は T の要素である. E は個々のアークに付与されるアーク表現式である. 関数の引数は A の要素であり, 値域はアークの始点または終点ノードであるプレースが保持可能なトークンのマルチ集合である. I は初期マーキングを与える. □

CPN の動作の意味定義は以下のように与えられる.

まず, Var(t) を, (1) $G(t)$ に含まれる変数, (2) t に入次または出次するアークに付与されたアーク表現式に含まれる変数の集合とする. $E(x_1, x_2)$ をプレースまたはトランジション x_1 から x_2 に向かうアークに付与されたアーク表現式すべての集合とする.

[定義 2.2] [トランジション t の束縛]

トランジション t の束縛は Var(t) 上の次の関数 b である.

(1) Var(t) 中のすべての変数 v について $b(v)$ は v の型である.

(2) その束縛 b で $G(t)$ を真にする.

B(t) は t のすべての束縛とする. □

[定義 2.3] [マーキング, ステップ]

トークンエレメントは $p \in P, c \in C(p)$ を満たす 2 項組 (p, c) であり, 束縛エレメントは $t \in T, b \in B(t)$ を満たす 2 項組 (t, b) である.

マーキングはトークンエレメント上のマルチ集合であり, ステップは束縛エレメントの非空マルチ集合で与えられる. また, 初期マーキング M_0 は $(I(p))(c)$ で与えられる. □

CPN-t では CPN にグローバルクロックを導入している. このグローバルクロックは任意のトークンのカラーの一部としてとして参照できるものとする.

具体的に, 各トークンは以下の型をそのカラーの一部として持つこととする.

- gt : グローバルクロック
- wt_i : トランジション t_i の発火時間
- tt : 総発火時間
- WTS : 発火可能な時刻制約の集合

ここで, tt は CPN-t の動作定義には直接は影響しないが, ワークフロー記述への応用を念頭に, 簡単のため導入する.

[定義 2.4] [時間経過, GC 一貫性]

グローバルクロック gc は進捗性 (任意の 2 つの時点 o_1, o_2 に対して o_2 が o_1 よりも未来であれば, 各々の時点の gc の値 $v_1, v2$ に対して $v_1 < v_2$ が成り立つ) と no-Zeno 性 (与えられた定刻に対していつかはその値を gc の値が越える) を満たす. また任意の時点において各トークンの gc は同一の値である. □

[定義 2.5] [発火可能] マーキング M であるトランジション t_i が発火可能であるのは以下の条件が成り立つときである.

$E(p, t_i)\langle b\rangle \leq M(p)$ □

各トランジション t に $G(t)$ を以下のように付与する.

$G(t_i) = P(gc, WTS) \wedge tt < C$ ここで C は与えられた定数であり, P はグローバルクロック gc が WTS 中のすべての制約を満たしているという述語である.

発火可能であるトランジションは発火できる. 発火は通常の CPN と異り, CPN-t では一定の時間幅を持つように定義される.

トランジション t に付与された値を DT_t とする. DT_t はトランジション t の発火に要する時間幅を表す.

[定義 2.6] [発火] トランジョション t_i が発火可能状態であればその状態が保証されている期間は発火できる. 発火中においては wt_i, tt の値は下記のように変化する. もし, wt_i の値が DT_t に付与された値と等しければ $wt_i = 0$. 発火期間を p とする. $wt_i = wt_i + p$, $tt = tt + p$ □

上記の定義において, p の値は gc の変化に従いながら 0 から DT_i までの値を任意に取るものとする.

[定義 2.7] [発火終了] 発火中のトランジョション t_i において wt_i の値が DT_i に等しくなればトランジション t_i の発火は終了する. □

式 $E\langle b\rangle$ で “b の束縛のもとで E を評価する式” を表すものとする.

[定義 2.8] [発火終了にともなうマーキングの変化]

ステップ Y がマーキング M_1 において発火終了になれば, マーキングは次の M_2 に変化する.

$\forall p \in P : M_2(p) = (M_1(p) - \sum_{(t,b)\in Y} E(p,t)\langle b\rangle) + \sum_{(t,b)\in Y} E(t,p)\langle b\rangle$

最初の和項が取り除かれるトークンであり, 2 番目の和項が追加される項である. □

[定義 2.9] [排他性] 競合関係にあるトランジション t_i, t_j において t_i が発火中にあるとき t_j は発火できない. ただし, 発火中のトランジションはいつでも発火可能状態にもどれるものとする. □

[定義 2.10] [一貫性] あるトランジション t_i のソースプレースにある複数のトークンにおいて複数の wt_i の型が存在する場合, この状況を非一貫性があると呼ぶことにする. 任意の発火可能な系列に対して非一貫性が無いことが保証できるネットとそのマーキングについて一貫性があると呼ぶ. □

以降では一貫性のあるネットのみを対象にする.

3. ワークフロー記述

ここでは, CPN-t を用いたワークフローの記述例について述べる. CPN-t のトランジションはワークフロー上での作業を表す. プレースにはワークフローの進行状況を表すプレースと, 作業者を表すプレースがある. ワークフローの進行状況はマーキングによって表現できる. 作業者を表すプレースと作業を表すトランジションを双方向アークで接続することで, 作業と作業者を関連づける.

ワークフローの図 1 を例として説明する. $t_1, \ldots, t_6$ のトランジションがグループワークの作業を表す. プレース A, B, C は作業者 A, B, C を表し, それぞれ彼らが行う作業と双方向アークで接続されている. t_5 は A, C と関連づけられており,

NII-Electronic Library Service

Institute of Electronics, Information, and Communication Engineers

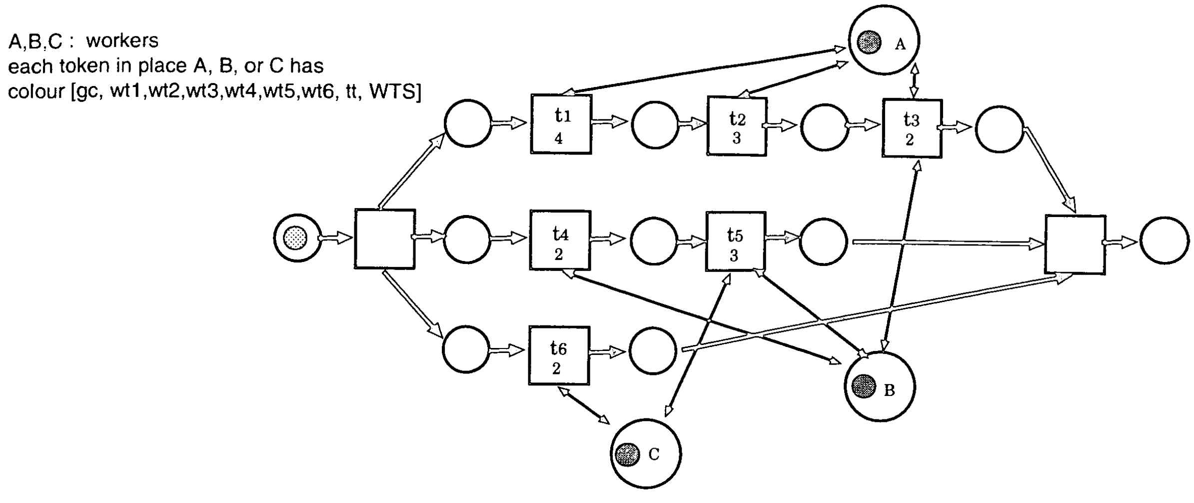

図 1 ワークフローの記述例

Fig. 1 An example of workflow description

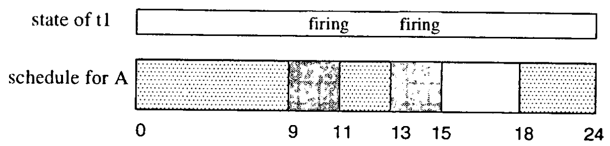

図 2 作業を中断して再開する場合

Fig. 2 Suspend and resume of a task

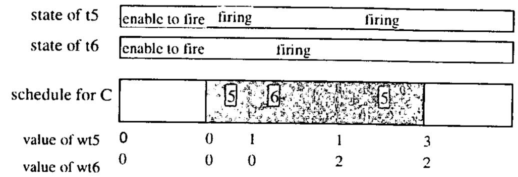

図 3 作業を中断して他の作業を行う場合

Fig. 3 Interruption by other task

t_5 は A, C の協調作業であることを表す.

プレース A のトークンカラー WTS の値が $\{(9\leq gc \leq 11),(13\leq gc \leq 18)\}$ であるとする. このとき作業者 A が t_1 を行う場合, 図 2 のように動作できる. 0 時から 9 時までは作業できない時間帯なので, トランジション t_1 は通常状態にある. 9 時からは作業できる時間帯なので発火状態に移るが, 11 時からは作業できない時間帯なので通常状態に戻る. 13 時からトランジション作業 1 は再開し, 15 時で作業 1 の 4 時間分の作業を終えるので, トランジション作業 1 は発火を完了する. このようにして, 作業者が作業できる時間帯にのみ作業を行い, そうでないときは作業を中断する動作を表現できる.

プレース C は t_5 と t_6 に関連づけられており, t_5 と t_6 は並列関係にある. そのため, 作業者 C は図 3 のように動作できる. 作業者 C は t_5 を 3 時間の内の 1 時間だけ作業し, 途中で t_6 を行う. そして t_6 が終了すると t_5 を再開する. t_5 が発火状態にある間プレース C のトークンカラーの wt_5 の値が増加する. そして t_6 の発火中は wt_5 の値は変化せず, t_5 を再開すると wt_5 の値は再び増加する. このようにして, 作業を中断して他の作業を行う場合を表現できる.

「他の作業者との協調作業のために時間を空けることはできるが, 一日に 3 時間しか作業しない」といったような状況を表すために, カラーの一部である tt を利用できる. プレース C のトークンカラーの tt を 24 時間毎にリセットするように指定しておき, また, WTS が $\{(9\leq gc \leq 18), (33\leq gc \leq 42), (tt \leq 3)\}$ であるとする. 作業者 C は例えば, 以下のように作業を行う. はじめに t_6 を行い終了する. 次に t_5 を行うが,「一日に 3 時間しか作業しない」ので t_5 の 3 時間の内の 1 時間を行い. その日の作業は終了する. 二日目になると t_5 の残りを行う.

4. スケジュール導出

文献 [9] では, ペトリネットの可達性解析の手法を用いて各作業者の実行スケジュールを導出する方法提案し, 実際にその導出処理系を作成した. その導出方法では, 状態爆発を避けるために種々のヒューリスティクスを用いており, 最終的には実用的な時間で実用規模の例題に対するスケジュール導出が可能であることを確認している. 今回は, 線形計画法を用いる方法を提案している.

4.1 問題定義

CPN-t によるワークフロー記述と作業全体のデッドライン時刻 T_D が与えられたときに, 各作業者のスケジュールを線形計画法を用いて導出する(注1). この際, 総時間最小など, なるべく最適なものを導出する.

入力となる CPN-t に対する制約は以下のとおり.

- 作業者プレースとトランジションを結ぶ両方向アーク以外は, ループ構造は持たない.
- 作業者プレースとそれに接続する両方向アークを除いたネットは 1 ソース, 1 シンクの安全なマークグラフ.
- カラー属性 tt は考慮にいれない.

ループ構造持つ場合の拡張については後述する.

(注1): 一般性を失うことなく開始時刻を 0 とおく

NII-Electronic Library Service

4.2 制約式

作業中断により同一トランジションが複数回発火するということが生じる．その上限回数は存在しないが，実際のスケジュール導出においては，上限を定めても問題ないし，むしろ，頻繁な作業中断は望ましくないとも考えられる．そこで，トランジション t の発火上限回数を定数 C_t で定めることとする．トランジション t について，作業者 w と w' が共同作業をする関係にあるとき，関係 wR_tw' を満たすとする．整数線形制約式の変数として次のものを導入する．W を作業者の集合とする．

∘ T^s_{wtj} ここで,$w(\in W)$ は作業者を表し，t はトランジションを表す．$j(\leq C_t)$ は，トランジション t の j 回目の発火を意味する．この変数は作業者 w によるトランジション t の j 回目の発火開始時刻を表す．

∘ T^e_{wtj} この変数は作業者 w によるトランジション t の j 回目の発火終了時刻を表す．

∘ T この変数は全作業終了時刻を表すための補助変数である．

ただし，作業者 w がトランジション t に関与しない場合は，T^s_{wtj},T^e_{wtj} の変数は導入しない．また，t の直後に t' が発火するなど，t の実行が t' に常に先行するようなトランジションの組を $t \prec t'$ で表す．制約条件として以下が課せられる．

（1） 各 w,t,j に対して，$0 \leq T^s_{wtj}$,$0 \leq T^e_{wtj}$

（2） 各 w,t,j に対して，$T^e_{wtj} \leq T$

（3） $T \leq T_D$

（4） 各 w,t,j に対して，$T^s_{wtj} \leq T^e_{wtj}$

（5） 各 w,t,j に対して，$T^e_{wtj} \leq T^s_{wtj+1}$

（6） $t \prec t'$ を満たす各 w,t,t' に対して，$T^e_{wtC_t} \leq T^s_{wt'1}$

（7） 各 w,t に対して，$\sum_{0 \leq j \leq C_t}(T^e_{wtj} - T^s_{wtj}) = DT_t$

（8） 関係 wR_tw' を満たす各 w,w',t について，各 j に対して，$T^s_{wtj} = T^s_{wtj}$, $T^e_{wtj} = T^e_{w'tj}$

4.3 WTS に関する制約について

WTS に対する制約は，一般には線形計画法の形では表現できない．線形式の論理式として表現すると以下のようになる．まず，トランジション t_i に関与する作業者の WST からトランジション t_i の実行可能時刻に関する以下の式が得られる．

$$\text{(9)}\quad C^i(t_1,t_2) = \bigvee_k (C^{si}_k \leq t_1 \wedge t_2 \leq C^{ei}_k)$$

，ここで C^{si}_k; C^{ei}_k は時刻の定数である．この式 $C^i(t_1,t_2)$ を満たす時刻 t_1,t_2 間においては，トランジション t_i が発火可能である．従って，「各 w,i,j について $C^i(T^s_{wij}, T^e_{wij})$」が制約となる．(1),...,(9) からなる制約全体は最終的に，複数の線形計画の制約式集合に変形できる．しかしながら，例えば，各作業者の WTS が一致している場合は，作業しない時間帯を無視することにより，制約集合の数を 1 とすることができる．実用上は，各作業者の作業可能時刻帯のかなりの部分が一致しており，有効な制約集合の数はかなり減ると考えられる．

4.4 目的関数

目的関数としては例えば，T が考えられる．この場合は，短時間優先となる．

他にも，新たな変数を導入し，この変数を作業外時間を表す変数と意味付け，これらの変数の線形和を最大化するように目的関数を設定すれば，期間内でなるべく余裕を持ったスケジュール導出を行うことができる．また，協調作業など特定の作業の時間変数を特定の時刻定数で制限することにより，協調作業を特定の時間帯に制限することもできる．さらには，複数の目的関数を設定しておき，それらの線形和を目的関数にすることにより，複数のゴールに対応するスケジュール導出も原理的には可能となる．

4.5 効率化と制約の解消

上記の整数線形計画問題を解く際に，例えば一週間のワークフローを同種出する場合は，単位時間を 30 分や 1 時間とすれば，単位時間を 1 分で解くよりは，現実的でもあり，導出時間の短縮につながる．

なお，CPN-t が選択構造を持つ場合，すぐ直後に合流プレースを持つなど，その選択構造が単純な場合は，暫定的にその選択構造を 1 トランジションと見なし，スケジュールをたてることができる．ただし，選択結果に依存して動的にその後のスケジュールを変更することは難しい．その場合はスケジュールの再導出などが必要となる．

CPN-t がネストが無いない単純なループ構造を持つ場合は，ループ回数をあらかじめ定めておき，異るループ回にごとに異る変数を割当てることにより，対応可能である．

5. あとがき

本稿では，ワークフロー記述に適した時間ペトリネットモデルと，それをもちいた各作業者のスケジュール導出法について述べた．今後の課題の一つは，tt を考慮にいれたスケジュール導出の考案である．

文献

[1] W. M. P van der Aalst: "The Application of Petri Nets to Workflow Management," The Journal of Circuits, Systems and Computers, pp. 21–66, 1998.

[2] V. P. Silva, T. Murata and S. M. Shatz: "Protocol Specification Design using an Object-based Petri net Formalism," International Journal of Software Engineering and Knowledge Engineering, Vol. 9, No. 1, pp. 97–125, 1999.

[3] K. Jensen: "Coloured Petri nets Vol.I, Vol.II, Vol.III," Springer-Verlag, 1992.

[4] 中山直樹，山口真悟，葛崎偉，田中稔: "カラーペトリネットによるワークフローのモデリングおよび評価"，電子情報通信学会技術研究報告，CAS, Vol. 99, No. 416, pp. 41–48, 1999.

[5] 山口真悟，葛崎偉，田中稔: "時間ペトリネットを用いたワークフローの動的変更の性能評価"，電子情報通信学会技術研究報告，CAS, Vol. 99, No. 416, pp. 49–56, 1999.

[6] W.M.P. van der Aalst : "Verification of Workflow Nets," LNCS, 1248, pp.407–426, 1997.

[7] W.M.P. van der Aalst and A.H.M. ter Hofstede : "Verification of Workflow Task Structures: A Petri-net-based Approach," Information Systems, 2000.

[8] P. A. Abdulla and A. Nylen: "Timed Petri Nets and BQOs," LNCS 2075, pp. 53–70, 2001.

[9] 山口昭，岡野浩三，谷口健一: "時間制約付きカラーペトリネットで記述されたワークフローからのスケジュール導出"，電子情報通信学会技術研究報告，SS, Vol. 101, No. 629, pp. 23–30, 2002.

NII-Electronic Library Service