



HAL
open science

Algorithm for Controlling the Transient Behavior of Controlled Generalized Batches Petri Nets

Ruotian Liu, Rabah Ammour, Leonardo Brenner, Isabel Demongodin

► **To cite this version:**

Ruotian Liu, Rabah Ammour, Leonardo Brenner, Isabel Demongodin. Algorithm for Controlling the Transient Behavior of Controlled Generalized Batches Petri Nets. 12ème Colloque sur la Modélisation des Systèmes Réactifs, Nov 2019, Angers, France. hal-02433029

HAL Id: hal-02433029

<https://amu.hal.science/hal-02433029>

Submitted on 2 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

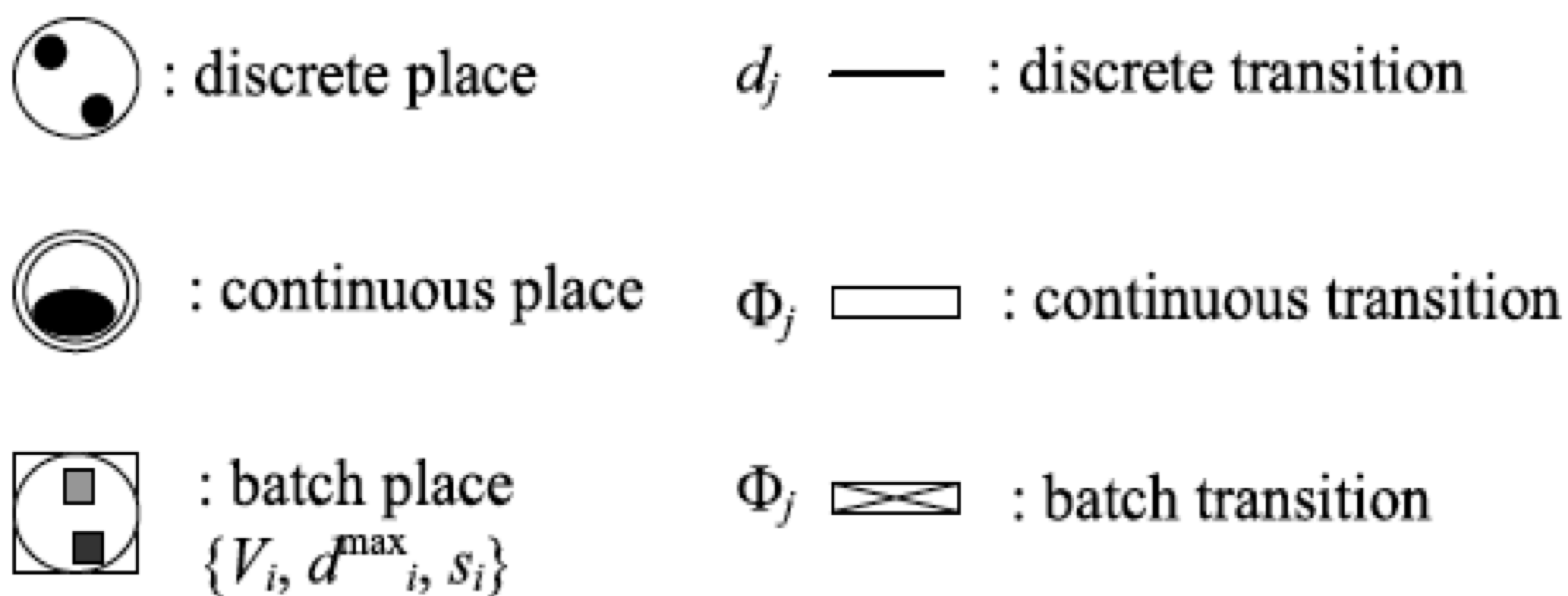
Algorithm for Controlling the Transient Behavior of Controlled Generalized Batches Petri Nets

Ruotian Liu, Rabah Ammour, Leonardo Brenner, Isabel Demongodin

Objective

Compute a **control trajectory** to reach a target **steady state** from a given **initial state** in hybrid / discrete event systems.

Controlled GBPN



➤ A **batch** β_k at time τ is: $\beta_k(\tau) = (l_k(\tau), d_k(\tau), x_k(\tau))$, where $l_k(\tau) \in \mathbb{R}_{\geq 0}$ is the length, $d_k(\tau) \in \mathbb{R}_{\geq 0}$ is the density and $x_k(\tau) \in \mathbb{R}_{\geq 0}$ is the head position.

➤ The **marking** of a batch place p_i is a series of batches: $m_i(\tau) = \{\beta_1(\tau), \dots, \beta_r(\tau)\}$.

➤ The **marking quantity** vector is: $\mathbf{q} = \mu(\mathbf{m})$. For place p_i :

$$q_i(\tau) = \begin{cases} m_i(\tau) & \text{if } p_i \in P^C \\ \sum_{k=1}^r l_k(\tau) \cdot d_k(\tau) & \text{if } p_i \in P^B \end{cases}$$

➤ The **instantaneous firing flow**: $\varphi_j \leq \Phi_j$ represents the firing quantity of transition t_j by time unit.

➤ The **controlled firing flow** vector at time τ : $\mathbf{u}(\tau)$ is the control input that drives the evolution. $0 \leq u_j(\tau) \leq \Phi_j$

Steady state

The cGBPN system $\langle N, \mathbf{m}_0 \rangle$ is in a **steady state** at time τ_s if for $\tau \geq \tau_s$ the marking \mathbf{m}^s and the instantaneous firing flow vector $\boldsymbol{\varphi}^s$ remain constant. Thus a steady state is defined as a pair $(\mathbf{m}^s, \boldsymbol{\varphi}^s)$.

Control of the transient behavior

Given a cGBPN system $\langle N, \mathbf{m}_0 \rangle$, a **control trajectory** is given as $(\mathbf{u}^0, \tau_0), (\mathbf{u}^1, \tau_1), \dots, (\mathbf{u}^i, \tau_i), \dots, (\mathbf{u}^n, \tau_n)$ such that the controlled firing flow vector \mathbf{u}^i is applied at date τ_i until τ_{i+1} .

Control strategy for reaching the steady state

➤ OFF: $u_j(\tau) = 0$ if t_j is not enabled or the marking quantity of one of its input places p_i is lower than its steady marking quantity $q_i(\tau) < q_i^s$.

➤ ON: maximize $\mathbf{u}(\tau)$ for approaching the steady marking quantity vector \mathbf{q}^s .

$$\mathbf{q}^s = \mathbf{q}^0 + C \cdot \left(\int_{\tau_0}^{\tau_1} \mathbf{u}^0(\rho) \cdot d\rho + \dots + \int_{\tau_{n-1}}^{\tau_n} \mathbf{u}^{n-1}(\rho) \cdot d\rho + \int_{\tau_n}^{+\infty} \mathbf{u}^n(\rho) d\rho \right)$$

$$\text{with } C \cdot \int_{\tau_n}^{+\infty} \mathbf{u}^n(\rho) d\rho = 0; \quad \mathbf{u}^n = \boldsymbol{\varphi}^s$$

Assumptions

➤ No discrete nodes ($P^D = T^D = \emptyset$) and no continuous transitions ($T^C = \emptyset$).

➤ The steady firing flow vector is positive ($\boldsymbol{\varphi}^s > 0$).

Algorithm

Algorithm: Computation of control trajectory

Input: A cGBPN system $\langle N, \mathbf{m}_0 \rangle$ and the steady state $(\mathbf{m}^s, \boldsymbol{\varphi}^s)$.

Output: Control trajectory $(\mathbf{u}^0, \tau_0), (\mathbf{u}^1, \tau_1), \dots$

1 Initialize : $\mathbf{q}^0 = \mu(\mathbf{m}_0)$, $\mathbf{u}^0 = \mathbf{0}$, $\tau_0 = 0$, $i = 0$;

2 while $\mathbf{m}^i \neq \mathbf{m}^s$ and $\mathbf{u}^i \neq \boldsymbol{\varphi}^s$ do

3 Determine the sets $T_N(\mathbf{m}^i)$, $T_L(\mathbf{m}^i)$, $S_G(\mathbf{m}^i)$, $S_E(\mathbf{m}^i)$;

4 Solve the following LPP: $\max \mathbf{1}^T \cdot \mathbf{u}^i$ s.t.

$$\begin{cases} \text{(a)} & 0 \leq u_j^i \leq \varphi_j^s & \forall t_j \in T \\ \text{(b)} & u_j^i = 0 & \forall t_j \in T_N(\mathbf{m}^i) \cup T_L(\mathbf{m}^i) \\ \text{(c)} & C(p_k, \cdot) \cdot \mathbf{u}^i \leq 0 & \forall p_k \in S_G(\mathbf{m}^i) \\ \text{(d)} & C(p_k, \cdot) \cdot \mathbf{u}^i = 0 & \forall p_k \in S_E(\mathbf{m}^i) \\ \text{(e)} & Pre(p_k, \cdot) \cdot \mathbf{u}^i \leq V_k \cdot d_k^{out} & \forall p_k \in P^B \end{cases}$$

5 Determine all the next timed events from considered ones, select the nearest at time τ_{i+1} ;

6 Determine the new marking \mathbf{m}^{i+1} and the marking quantity vector \mathbf{q}^{i+1} ;

7 $i = i + 1$;

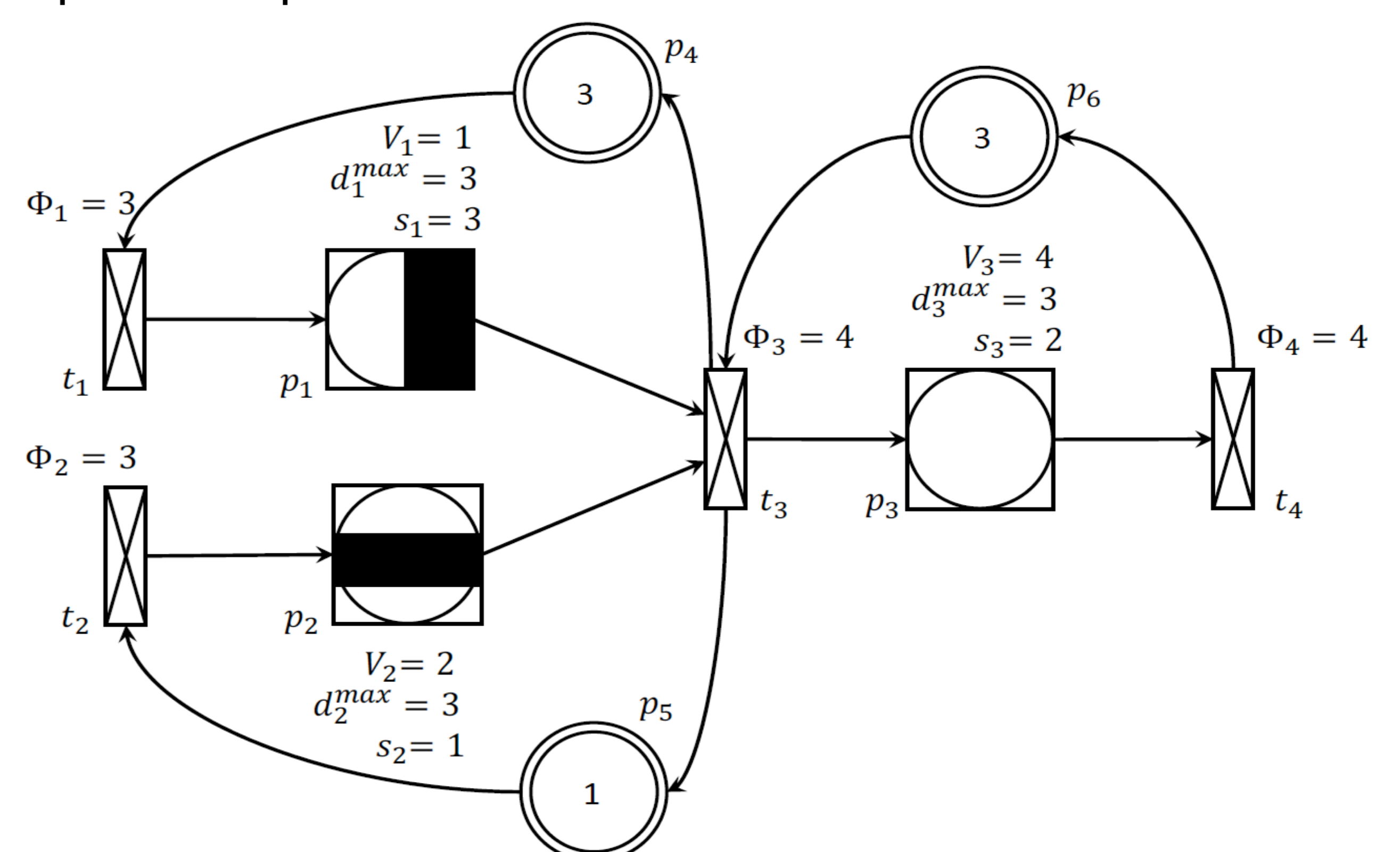
8 Return $(\mathbf{u}^0, \tau_0), (\mathbf{u}^1, \tau_1), \dots$

Example

$$\mathbf{m}_0 = [\{(2, 3, 3)\} \{(1, 2, 1)\} \emptyset \ 3 \ 1 \ 3]^T$$

$$\mathbf{m}^s = [\{(1, 2, 1), (2, 3, 3)\} \{(1, 1, 1)\} \{(2, 0.5, 2)\} \ 1 \ 2 \ 2]^T$$

$$\boldsymbol{\varphi}^s = [2 \ 2 \ 2 \ 2]^T$$



Control trajectory: $(\mathbf{u}^0, 0), (\mathbf{u}^1, 1), (\mathbf{u}^2, 1.5), (\mathbf{u}^3, 2)$ with:

$$\begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \end{pmatrix}_{\mathbf{u}^0} \xrightarrow{\tau_0 = 0} \begin{pmatrix} 2 \\ 0 \\ 2 \\ 0 \end{pmatrix}_{\mathbf{u}^1} \xrightarrow{\tau_1 = 1} \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}_{\mathbf{u}^2} \xrightarrow{\tau_2 = 1.5} \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}_{\mathbf{u}^3} \xrightarrow{\tau_3 = 2} \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}_{\mathbf{u}^3 = \boldsymbol{\varphi}^s}$$

[1] I. Demongodin and A. Giua. Dynamics and steady state analysis of controlled generalized batches Petri nets. NAHS, 2014.

[2] L. Wang, C. Mahulea, J. Julvez and M. Silva. On/off strategy based minimum-time control of continuous Petri nets. NAHS, 2014.