# Linear programming techniques for analysis and control of batches Petri nets

Isabel Demongodin,
LSIS, Univ. of Aix-Marseille, France (isabel.demongodin@lsis.org)

Alessandro Giua
DIEE, Univ. of Cagliari, Italy (giua@diee.unica.it)

**Abstract**

In this paper we consider Generalised Batches Petri nets (GBPN) and develop new linear algebraic techniques for the analysis of this model. Two main contributions are presented. The first contribution lies in the fact that although we consider the same GBPN model that has already be presented in the literature, we associate to this model a different semantics considering that the instantaneous firing flow of continuous and batch transitions are control variables that can take an arbitrary value provided they satisfy given constraints. The second contribution consists in the analysis of the steady state behavior of GBPN. We show that under the assumption that no discrete transition fires, a steady state can be characterized by solving a linear programming problem that takes into account the net structure and the initial marking.

Published as: I. Demongodin, A. Giua, "Linear programming techniques for analysis and control of batches Petri nets," *WODES10: 10th Int. Work. on Discrete Event Systems* (Berlin, Germany), Aug-Sep 2010.

This work has been partially supported by the European CommunityŠs Seventh Framework Programme under project DISC (Grant Agreement n. INFSO-ICT-224498).

# 1 Introduction

Petri nets have originally been defined as discrete event models. In the last years, however, the Petri net formalism has been extended to also encompass continuous and hybrid models [9, 1, 2], thus offering formal techniques for expressing both fundamental discrete event and continuous time behaviors. However, in order to represent the delay due to continuous transfer elements, basic hybrid Petri nets are inadequate. This was the main drive that led to the definition of batches Petri nets [3], that extend the hybrid Petri nets class by defining the concept of *batch*, i.e., a group of entities moving through a transfer zone at a certain speed and the corresponding notion of *batch node*. These Petri nets allow by their hybrid dynamic formalization to represent in a very detailed manner transfer elements with the possibility of accumulation of entities and thus generating variable delays on continuous flows.

In particular in this paper we consider Generalised Batches Petri nets as defined by [3] and develop new linear algebraic techniques for the analysis of this model. Two main contributions are presented.

The first contribution lies in the fact that although we consider the same GBPN model that has already be presented in [3], we associate to this model a different semantics inspired by FOHPN [1]. In fact, we assume that the instantaneous firing flow of continuous and batch transitions are control variables that can take an arbitrary value provided appropriate constraints are satisfied. These constraints may be structural, e.g., a transition flow cannot exceed the maximal firing flow, or behavioral, e.g., the total flow exiting an empty place cannot be greater than the input flow. This has three important consequences. On one hand we generalize previous semantics that consider a single possible autonomous evolution: this was the assumption in [2, 3] where it is assumed that a transition should always fire at its maximal admissible flow. Secondly, this allows us to consider problems of conflict that could not well be handled in the framework of autonomous evolutions. Thirdly, in our framework a simple linear algebraic approach can be used to select a vector of instantaneous firing flows.

The second contribution consists in the analysis of the steady state behavior of GBPN, i.e., a pair $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$ where $\boldsymbol{m}^s$ is a constant marking and $\boldsymbol{\varphi}^s$ is a constant vector of instantaneous firing flows. We show that under the assumption that no discrete transition fires, a steady state can be characterized by solving a linear programming problem that takes into account the net structure and the initial marking. Related work on the steady state analysis of continuous nets can be found in [5, 7, 8, 6].

The paper is structured as follows. In Section 2, the basic definitions of Generalized Batches Petri nets are presented, including the enabling and firing rules for transitions and the description of the hybrid dynamics of batches. Section 3 proposes a linear programming problem to compute the instantaneous firing flow vector where continuous and batch transitions are supposed to be controlled. Finally, Section 4 is dedicated to the computation of steady states in polynomial time using LPP.

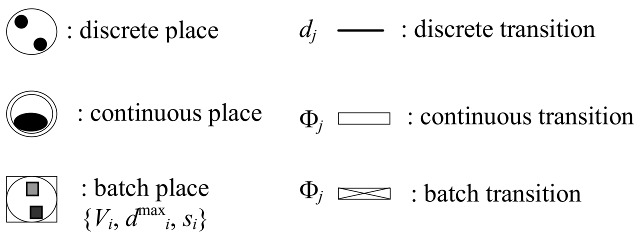


Figure 1: Nodes of GBPN.

## 2 Background on batches Petri nets

### 2.1 Basic definitions

The following definition, rewritten with a slightly simplified notation, is taken from [3].

**Definition 2.1** *A* Generalized Batches Petri net *(GBPN) is a 6-tuple* $N = (P, T, Pre, Post, \gamma, Time)$ *where:*

- $P = P^D \cup P^C \cup P^B$ *is finite set of places partitioned into the three classes of* discrete, continuous *and* batch *places.*
- $T = T^D \cup T^C \cup T^B$ *is finite set of transitions partitioned into the three classes of* discrete, continuous *and* batch *transitions.*
- $Pre, Post : (P^D \times T \to \mathbb{N}) \cup ((P^C \cup P^B) \times T \to \mathbb{R}_{\geq 0})$ *are, respectively, the pre-incidence and post-incidence matrixes, denoting the weigth of the arcs from places to transitions and transitions to places.*
- $\gamma : P^B \to \mathbb{R}^3_{>0}$ *is the* batch place function. *It associates to each batch place* $p_i \in P^B$ *the triple* $\gamma(p_i) = (V_i, d_i^{\max}, s_i)$ *that represents, respectively,* speed*,* maximum density *and* length *of* $p_i$*.*
- $Time : T \to \mathbb{R}_{\geq 0}$ *associates a non negative number to every transition:*
  - *if* $t_j \in T^D$*, then* $Time(t_j) = d_j$ *denotes the firing delay associated to the discrete transition;*
  - *if* $t_j \in T^C \cup T^B$*, then* $Time(t_j) = \Phi_j$ *denotes the maximal firing flow associated to the continuous or batch transition.* ■

To every continuous and batch transition, $t_j \in T^C \cup T^B$, is associated an instantaneous firing flow (IFF), noted $\varphi_j(\tau)$, representing the quantity of markings by time unit that fires transition $t_j$. Section 3 will discuss on the computation of this vector.

We denote the number of places and transitions, resp., $m = |P|$ and $n = |T|$. The *preset* and *postset* of transition $t_j$ are: ${}^\bullet t_j = \{p_i \in P \mid Pre(p_i, t_j) > 0\}$ and $t_j^\bullet = \{p_i \in P \mid Post(p_i, t_j) > 0\}$.

Similar notations may be used for pre and post transition sets of places and its restriction to discrete, continuous or batch transitions is denoted as ${}^{(d)}p_i = {}^\bullet p_i \cap T^D$, ${}^{(c)}p_i = {}^\bullet p_i \cap T^C$, and ${}^{(b)}p_i = {}^\bullet p_i \cap T^B$.

In this paper we will only consider well-formed nets, introduced in the next definition.

**Definition 2.2** *A GBPN is said to be (*well-formed*) if the following conditions hold:*

- *discrete places can be connected to continuous and batch transitions only by self-loops, i.e., for all $p_i \in P^D$ and for all $t_j \in T^C \cup T^B$ it holds $Pre(p_i, t_j) = Post(p_i, t_j)$.*
- *the pre and post sets of batch places contain only batch transitions, i.e., for all $p_i \in P^B$ it holds ${}^\bullet p_i \cup p_i^\bullet \subset T^B$.* ■

The first condition, that is also commonplace in the framework of hybrid nets, is required to ensure the marking of discrete place is not changed by the firing of continuous and batch transitions. The second condition is due to the rules concerning the creation and destruction of batches.

The *incidence matrix* of a net is $\mathbf{C} = Post - Pre$ and for a well-formed GBPN can be partitioned as follows.

$$\begin{array}{cccccc} & & T^D & T^C & T^B & \\ \boldsymbol{C} = & \left[ \begin{array}{c} \\ \\ \\ \end{array}\right. & \begin{array}{c} \boldsymbol{C}^{DD} \\ \boldsymbol{C}^{CD} \\ \mathbf{0} \end{array} & \begin{array}{c} \mathbf{0} \\ \boldsymbol{C}^{CC} \\ \mathbf{0} \end{array} & \begin{array}{c} \mathbf{0} \\ \boldsymbol{C}^{CB} \\ \boldsymbol{C}^{BB} \end{array} & \left. \begin{array}{c} \\ \\ \\ \end{array}\right] \begin{array}{c} P^D \\ P^C \\ P^B \end{array} \end{array}$$

The main extension of GBPN with respect to Hybrid Petri Nets [2] is related to the notions of batch, i.e., a group of discrete entities characterized by three continuous variables.

**Definition 2.3** *A* batch *$\beta_r$ at time $\tau$, is defined by a triple, $\beta_r(\tau) = (l_r(\tau), d_r(\tau), x_r(\tau))$, where $l_r(\tau) \in \mathbb{R}_{\geq 0}$ is the* length*, $d_r(\tau) \in \mathbb{R}_{\geq 0}$ is the* density *and $x_r(\tau) \in \mathbb{R}_{\geq 0}$ is the* head position*.* ■

A batch place contains a series of batches, ordered by their head positions and moving forward at the same speed $V_i$.

The state of a GBPN is represented by its marking.

**Definition 2.4** *The* marking *of a GBPN at time $\tau$ is defined as $\boldsymbol{m}(\tau) = [m_1(\tau)...m_i(\tau)...m_n(\tau)]^T$, where:*

- *if $p_i \in P^D$ then $m_i \in \mathbb{N}$, i.e., the marking of a discrete place is a non negative integer.*
- *if $p_i \in P^C$ then $m_i \in \mathbb{R}_{\geq 0}$, i.e., the marking of a continuous place is a non negative real.*
- *if $p_i \in P^B$ then $m_i = \{\beta_h, ..., \beta_r\}$, i.e., the marking of a batch place is a series of batches.*

*The* quantity of marks *contained in batch place $p_i \in P^B$, with $m_i = \{\beta_h, ..., \beta_r\}$, is defined by:*

$$q_i(\tau) = \sum_{\beta_j \in m_i} l_j(\tau) \cdot d_j(\tau),$$

*and represents the sum of the quantities of the batches contained in the place.* ■

We denote $\boldsymbol{m}_0 = \boldsymbol{m}(\tau_0)$ the initial marking. When time can be omitted, we denote the marking as $\boldsymbol{m}$.

**Definition 2.5** *Let $\beta_r(\tau) = (l_r(\tau), d_r(\tau), x_r(\tau)) \in m_i(\tau)$ be a batch in place $p_i \in P^B$, with $\gamma(p_i) = (V_i, d_i^{\max}, s_i)$.*
*$\beta_r$ is called an* output batch *if its head position is equal to the length associated to the batch place, i.e., $x_r(\tau) = s_i$.*
*A batch is said to be* dense *if its density is equal to the maximal density of batch place $p_i$, $d_r(\tau) = d_i^{\max}$. The* output density *$d_i^{\text{out}}$ of batch place $p_i$ is defined as follows. If at time $\tau$, place $p_i$ has an output batch $\beta_r(\tau)$, then $d_i^{\text{out}}(\tau) = d_r(\tau)$, else $d_i^{\text{out}}(\tau) = 0$.* ■

Note that a place in GBPN can have at most one output batch. Due to the bounded characteristics of a batch place, some constraints on batches characteristics have to be respected: $0 \le l_r \le x_r \le s_i$ (position and length constraints) and $0 \le d_r \le d_i^{\max}$ (density constraint).

The notion of batch place function associated to a batch place implicitly assumes that the place capacity is finite. This is formalised in the following definition.

**Definition 2.6** *The* maximum capacity *of batch place $p_i \in P^B$, with $\gamma(p_i) = (V_i, d_i^{\max}, s_i)$, is $Q_i = s_i \cdot d_i^{\max}$. A place such that $q_i(\tau) = Q_i$ is called a* full batch place. ■

## 2.2 Enabling and firing conditions

The enabling and firing conditions of discrete transitions are those of classical transition-timed discrete Petri nets.

**Condition 2.7** *A discrete transition $t_j \in T^D$ is enabled at $\boldsymbol{m}$ if for all $p_i \in {}^{\bullet}t_j$, $m_i \ge Pre(p_i, t_j)$. The firing of discrete transition $t_j \in T^D$ that is enabled at $\mathbf{m}$, yields a new marking $\mathbf{m}' = \mathbf{m} + \boldsymbol{C}(\cdot, t_j)$.*

The enabling conditions of continuous transitions are those of First Order Hybrid Petri nets [1] i.e., one distinguishes weakly and strongly enabled transitions. We define similar conditions for batch transitions.

**Condition 2.8** *A continuous transition $t_j \in T^C$ is enabled at $\boldsymbol{m}$ if for all $p_i \in {}^{(d)}t_j$, $m_i \ge Pre(p_i, t_j)$. We say that the continuous transition is:*

- strongly enabled *if $\forall p_k \in {}^{(c)}t_j$, $m_k > 0$.*
- weakly enabled *if $\exists p_r \in {}^{(c)}t_j$, $m_r = 0$.*

**Condition 2.9** *A batch transition $t_j \in T^B$ is enabled at $\boldsymbol{m}$ if:*

- *$\forall p_i \in {}^{(d)}t_j$, $m_i \ge Pre(p_i, t_j)$.*

- $\forall p_s \in {}^{(b)}t_j$, $d_s^{\text{out}} > 0$.

*We say that the batch transition is:*

- strongly enabled *if* $\forall p_k \in {}^{(c)}t_j$, $m_k > 0$.
- weakly enabled *if* $\exists p_r \in {}^{(c)}t_j$, $m_r = 0$.

The computation of the IFF of enabled continuous and batch transitions will be described in Section 3 and here we simply discuss the net evolution assuming that the IFF vector, $\boldsymbol{\varphi}(\tau)$, is given at time $\tau$.

The evolution in time of the marking of a continuous place, $p_i \in P^C$ is described by:

$$\dot{m}_i(\tau) = \left[ \begin{array}{cc} \boldsymbol{C}^{CC}(p_i, \cdot) & \boldsymbol{C}^{CB}(p_i, \cdot) \end{array} \right] \boldsymbol{\varphi}(\tau).$$

## 2.3 Dynamics of batch places

To resume the hybrid dynamics of batches, let us first introduce some concepts necessary to the understanding of the evolution.

**Definition 2.10** *The input flow and the output flow of batch place $p_i$ at time $\tau$ are defined by:*

- $\phi_i^{\text{in}}(\tau) = \sum_{t_j \in {}^\bullet p_i} Post(p_i, t_j) \cdot \varphi_j(\tau)$.
- $\phi_i^{\text{out}}(\tau) = \sum_{t_j \in p_i^\bullet} Pre(p_i, t_j) \cdot \varphi_j(\tau)$. ■

**Definition 2.11** *At time $\tau$, various static functions can be applied on batches composing the marking of place $p_i$:*

- Creation. *If the input flow of $p_i$ is not null, i.e., $\phi_i^{\text{in}}(\tau) \neq 0$, a batch $\beta_r(\tau) = (0, d_r(\tau), 0)$ with $d_r(\tau) = \phi_i^{\text{in}}(\tau)/V_i$, is created and added to the marking of $p_i$, i.e., $m_i(\tau) = m_i(\tau) \cup \{\beta_r(\tau)\}$.*
- Destruction. *If the length of a batch, $\beta_k(\tau)$, is null, $l_k = 0$, and if it is not a created batch, $x_k \neq 0$, batch $\beta_k(\tau)$ is destroyed, noted $\beta_k(\tau) = \mathbf{0}$, and removed from the marking of $p_i$, i.e., $m_i(\tau) = m_i(\tau) \setminus \{\beta_k(\tau)\}$.*
- Merging. *If two batches with the same density are in contact, they can be merged. Let batches $\beta_r(\tau) = (l_r(\tau), d_r(\tau), x_r(\tau))$ and $\beta_k(\tau) = (l_k(\tau), d_k(\tau), x_k(\tau))$ in $m_i(\tau)$, such that $x_r(\tau) = x_k(\tau) + l_r(\tau)$ and $d_r(\tau) = d_k(\tau)$. In this case, batch $\beta_r(\tau)$ becomes $\beta_r(\tau) = (l_r(\tau) + l_k(\tau), d_r(\tau), x_r(\tau))$, batch $\beta_k(\tau)$ is destroyed, $\beta_k(\tau) = \mathbf{0}$, and $m_i(\tau) = m_i(\tau) \setminus \{\beta_k(\tau)\}$.*
- Splitting. *It is always possible to split a batch into two batches in contact with the same density.* ■

Batch places describe the transfer of batches according to a switching dynamics between two behaviors: the free behavior and the accumulation behavior. Both dynamics of a batch place are governed by the state of the batches composing it and various equations govern the evolution of batches: inputting, moving and exiting.

**Definition 2.12 (Free behavior)** *Batch $\beta_r(\tau)$ of $p_i$ is in a* free behavior *if it moves freely at the speed $V_i$. Three different dynamics can occur.*

- Inputting. *A created batch, $\beta_r(\tau) = (0, d_r(\tau), 0)$ freely enters in place $p_i$ according to:*

$$\dot{l}_r = V_i; \quad \dot{d}_r = 0; \quad \dot{x}_r = V_i$$

- Moving. *A batch, $\beta_r(\tau) = (l_r(\tau), d_r(\tau), x_r(\tau))$ freely moves inside place $p_i$ according to:*

$$\dot{l}_r = 0; \quad \dot{d}_r = 0; \quad \dot{x}_r = V_i$$

- Exiting. *An output batch, $\beta_r(\tau) = (l_r(\tau), d_r(\tau), s_i)$ freely exits from place $p_i$ according to:*

$$\dot{l}_r = -V_i; \quad \dot{d}_r = 0; \quad \dot{x}_r = 0$$

*Batch place $p_i$ is in a* free behavior *if its output batch is in a free behavior, i.e. $\phi_i^{\text{out}}(\tau) = d_i^{\text{out}}(\tau) \cdot V_i$.* ■

**Definition 2.13 (Accumulation behavior)** *Batch $\beta_r(\tau)$ of $p_i$ is in an* accumulation behavior *if it is not moving at the speed of $p_i$. Two situations can cause this behavior.*

- *Let $\beta_r(\tau)$ be an output batch of $p_i$. If the output flow of $p_i$ is lower than the free batch flow $d_r(\tau)/V_i$ then batch $\beta_r(\tau)$ accumulates while it exits the place.*
- *Let $\beta_r(\tau)$ be a batch in contact with a downstream output batch in an accumulation behavior. In this case, batch $\beta_r(\tau)$ cannot move freely at the speed $V_i$, but starts an accumulation that will be merged with the downstream dense output batch.*

*Batch place $p_i$ is in an* accumulation behavior *if its output batch is in an accumulation behavior, i.e. $\phi_i^{\text{out}}(\tau) < d_i^{\text{out}}(\tau) \cdot V_i$.* ■

A complete and general description of the equations that govern this behavior can be found in [4]. Note that in these dynamics, we assume that the density of a batch in an accumulation behavior is equal to the maximal density of the batch place, i.e., it is dense. When a batch starts an accumulation, it is split into two batches in contact where the downstream batch is dense.

## 3 Instantaneous firing flows

The overall behavior algorithm of a GBPN is based on a discrete event approach with linear or constant continuous evolutions between events [3]. Between two events, the state of the net has an invariant behavior state which corresponds to a period of time such that: the marking in discrete places is constant; the instantaneous firing flow of continuous and batch transitions is constant; the output density of batch places is constant.

In this paper, we associate to a GBPN model a semantics inspired by FOHPN [1]. The set of admissible IFF vectors form a convex set described by linear equations.

**Definition 3.1 (admissible IFF vectors)** *Let $\langle N, \boldsymbol{m} \rangle$ be a GBPN with incidence matrix $\boldsymbol{C}$. Let*

- $T_{\mathcal{E}}(\boldsymbol{m}) \subset T^C \cup T^B$ *($T_{\mathcal{N}}(\boldsymbol{m}) \subset T^C \cup T^B$) be the subset of continuous and batch transitions enabled (not enabled) at $\boldsymbol{m}$.*
- $P_{\emptyset}(\boldsymbol{m}) = \{p_i \in P^C \mid m_i = 0\}$ *be the subset of empty continuous places.*
- $P_F(\boldsymbol{m}) = \{p_i \in P^B \mid q_i = Q_i\}$ *be the subset of full bach places.*

*Any* admissible IFF vector $\boldsymbol{\varphi} : T^C \cup T^B \to \mathbb{R}_{\geq 0}$ *, at $\boldsymbol{m}$, is a feasible solution of the following linear set:*

$$\begin{cases} (a) & 0 \leq \varphi_j \leq \Phi_j & \forall t_j \in T_{\mathcal{E}}(\boldsymbol{m}) \\ (b) & \varphi_j = 0 & \forall t_j \in T_{\mathcal{N}}(\boldsymbol{m}) \\ (c) & \sum_{t_j \in T_{\mathcal{E}}} C(p_i, t_j) \cdot \varphi_j \geq 0 & \forall p_i \in P_{\emptyset}(\boldsymbol{m}) \\ (d) & \sum_{t_j \in T_{\mathcal{E}}} C(p_i, t_j) \cdot \varphi_j \leq 0 & \forall p_i \in P_F(\boldsymbol{m}) \\ (e) & \sum_{t_j \in T_{\mathcal{E}}} Post(p_i, t_j) \cdot \varphi_j \leq V_i \cdot d_i^{\max} & \forall p_i \in P^B \\ (f) & \sum_{t_j \in T_{\mathcal{E}}} Pre(p_i, t_j) \cdot \varphi_j \leq V_i \cdot d_i^{\text{out}} & \forall p_i \in P^B \end{cases} \tag{1}$$

*The set of all feasible solutions is denoted $\mathcal{S}(N, \boldsymbol{m})$.* ■

Constraints of the form (a)–(c) are similar to those that describe the set of admissible IFF vectors in FOHPN. In particular, constraint (a) implies that the instantaneous firing flow $\varphi_j$ of transition $t_j$ has a value lower than or equal to its maximum firing flow $\Phi_j$. Constraint (b) implies that a firing flow of a transition that is not enabled is null. Constraint (c) implies that the marking of an empty continuous place cannot decrease (non negativity constraint).

Constraints of the form (d)–(f) are peculiar to GBPN. Constraint (d) implies that the marking of a full batch place cannot increase. Constraint (e) implies that the total flow entering batch place $p_i$ should not be greater than the maximal flow $V_i \cdot d_i^{\max}$ that the place can accept. Constraint (f) implies that the total flow exiting batch place $p_i$ should not be greater than the output flow $V_i \cdot d_i^{\text{out}}$ generated by the output batch exiting the place.

To structurally bound a batch place $p_i$ with capacity $Q_i$, one may add to the net a complementary continuous place $p_{i'}$ with $Pre(p_{i'}, \cdot) = Post(p_i, \cdot)$, $Post(p_{i'}, \cdot) = Pre(p_i, \cdot)$, and $m_{i'}(0) = Q_i - q_i(0)$. In such a case, constraint (d) can be removed from linear set (1) as it is replaced by the nonnegativity constraint for continuous place $p_{i'}$.

It is important to stress the implicit assumption underlying Definition 3.1: the firing flows of continuous and batch transitions are control inputs whose value can be chosen by the supervisor within the set $\mathcal{S}(N, \boldsymbol{m})$. To choose among the admissible IFF vectors that satisfy (1) the supervisor may use an objective function, or introduce additional constraints, as also discussed in the case of FOHPN in [1].

As a final remark, it is common to assume that the components of the IFF vector are piecewise-constant signals. This means that the selected value of an IFF will be used until the occurrence of an event (such as the firing of a discrete transition that may change the set of enabled transitions, the emptying of a continuous place, the filling of batch place, the change of the output density of a batch place) will require a new computation.

**Example 3.2** *Consider the net in Fig. 2, where* $\Phi_1 = 3$*,* $\Phi_2 = 1$*,* $\Phi_3 = \Phi_4 = 2$*,* $\gamma(p_2) = (V_2, d_2^{\max}, s_2) = (1, 2, 5)$ *and* $\gamma(p_3) = (V_3, d_3^{\max}, s_3) = (1, 2, 5)$*. The initial marking is* $\boldsymbol{m}_0 = [8\ \emptyset\ \{\beta_1(0)\}]^T$ *with* $\beta_1(0) = (5, 2, 5)$*. We can remark that the output batch of place* $p_3$ *is dense. Thus,* $d_3^{\text{out}}(0) = d_1(0) = d_3^{\max} = 2$ *and* $d_2^{\text{out}}(0) = 0$*. Moreover, place* $p_3$ *is initially full as* $q_3(0) = l_1(0) \cdot d_1(0) = 10 = Q_3$ *.*

*At the initial time it holds:* $T_{\mathcal{E}}(\boldsymbol{m}_0) = \{t_1, t_3, t_4\}$*,* $T_{\mathcal{N}}(\boldsymbol{m}_0) = \{t_2\}$*,* $P_{\emptyset}(\boldsymbol{m}_0) = \emptyset$ *and* $P_F(\boldsymbol{m}_0) = \{p_3\}$*.*

*Any IFF vector at the initial time must verify:*

$$\begin{cases} (a) & 0 \leq \varphi_1 \leq \Phi_1 = 3 \\ (a') & 0 \leq \varphi_3 \leq \Phi_3 = 2 \\ (a'') & 0 \leq \varphi_4 \leq \Phi_4 = 2 \\ (b) & \varphi_2 = 0 \\ (d) & -\varphi_3 - \varphi_4 \leq 0 \\ (e) & \varphi_1 \leq V_2 \cdot d_2^{\max} = 2 \\ (e') & 0 \leq V_3 \cdot d_3^{\max} = 2 \\ (f) & \varphi_3 + \varphi_4 \leq V_3 \cdot d_3^{\text{out}} = 2 \end{cases}$$

*If the priority is that of maximizing the output flow of the net* $(\varphi_3 + \varphi_4)$ *while also requiring all other transitions to have a flow as large as possible, one can use as objective function to maximize* $J = \varphi_3 + \varphi_4 + 0.1(\varphi_3 + \varphi_4)$*. In such a case one gets a family of optimal solutions of the form* $\boldsymbol{\varphi} = (2, 0, x, 2 - x)$ *with* $x \in [0, 2]$*.*

*If one also wants to impose a ratio* $2 : 1$ *between the flows of* $t_3$ *and* $t_4$*, it is possible to add a constraint of the form* $\varphi_3 = 2\varphi_4$ *to get solution* $\boldsymbol{\varphi} = (2, 0, 4/3, 2/3)$*.*

# 4 Steady state computation

We now consider the problem of determining a steady state for the considered model. Let us first give some definitions that will be useful in the rest of the paper.

Extending Def. 2.4 (that only applies to batch places), a *marking quantity vector* $\boldsymbol{q} = \mu(\boldsymbol{m}) \in \mathbb{R}^m$

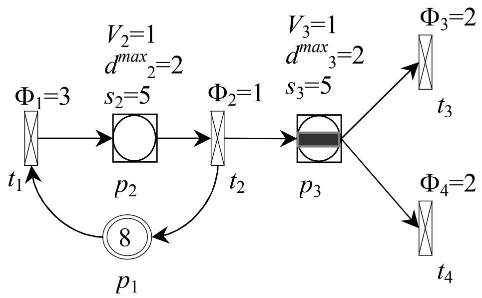


Figure 2: Net in Example 3.2.

can be associated to a marking $\boldsymbol{m}$, such that:

$$q_i = \begin{cases} m_i & \text{if } p_i \in P^D \cup P^C \\ \sum_{\beta_r \in m_i} l_r \cdot d_r & \text{if } p_i \in P^B \end{cases}$$

Note that usually more than one marking $\boldsymbol{m}$ may correspond to a given marking quantity vector $\boldsymbol{q}$, i.e., $\mu^{-1}(\boldsymbol{q})$ is a set of markings.

The state equation that governs the dynamic behavior of a GBPN in terms of marking quantity vector is [3]: $\boldsymbol{q}(\tau) = \boldsymbol{q}(\tau_0) + \boldsymbol{C} \cdot \boldsymbol{z}(\tau)$, where $\boldsymbol{z}(\tau) \in \mathbb{R}^n_{\geq 0}$, called *characteristic vector*, denotes how many times a discrete transition has fired and the quantity fired for continuous and batch transitions during $[\tau_0, \tau]$.

We denote $R(N, \boldsymbol{m}_0)$ the set of reachable markings of a GBPN and define the *reachable marking quantity set* as

$$RQ(N, \boldsymbol{m}_0) = \{\boldsymbol{q} \mid \exists \boldsymbol{m} \in R(N, \boldsymbol{m}_0) : \ \boldsymbol{q} = \mu(\boldsymbol{m})\}.$$

In this section we consider GBPN only composed by continuous and batch nodes. This will allow us to study the stationary behavior of the net. The same results, however, also apply to arbitrary GBPN during a period in which no discrete transition fires. The state equation of a GBPN $\langle N, \boldsymbol{m}_0 \rangle$ with $P^D = T^D = \emptyset$ can be written:

$$\boldsymbol{q}(\tau) = \boldsymbol{q}(\tau_0) + \boldsymbol{C} \cdot \boldsymbol{z}(\tau) = \boldsymbol{q}(\tau_0) + \boldsymbol{C} \cdot \int_{\tau_0}^{\tau} \boldsymbol{\varphi}(\rho) d\rho$$

We define a steady state as follows.

**Definition 4.1 (Steady State)** *Let a GBPN $\langle N, \boldsymbol{m}_0 \rangle$ with $P^D = T^D = \emptyset$. The net is in a* steady state *at time $\tau_s$ if for $\tau \in [\tau_s, +\infty)$ the marking and the instantaneous firing vector are constants. Thus a steady state is defined by the pair $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$.* ■

Note that this definition also implies that the output density of batch places is constant at the steady state.

Firstly, we make the following obvious observation.

**Proposition 4.2** *Assume that a net* $\langle N, \boldsymbol{m}_0 \rangle$ *with* $P^D = T^D = \emptyset$ *is in a steady state* $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$. *Then the marking quantity vector is such that:* $\dot{\boldsymbol{q}}^{\boldsymbol{s}} = \boldsymbol{C} \cdot \boldsymbol{\varphi}^s = \boldsymbol{0}$.

Proof. *The first equality follows from the state equation, and the second one from the fact that a constant marking implies a constant marking quantity in each place.* □

We now present two results that characterize the steady state of batch places. Obviously, for place $p_i$ in a steady state $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$ its input and output flows coincide, i.e.,

$$\phi_i^s = Post(p_i, \cdot) \cdot \boldsymbol{\varphi}^s = Pre(p_i, \cdot) \cdot \boldsymbol{\varphi}^s. \tag{2}$$

Furthermore let us denote $\delta_i^{\min}$ the *minimum delay* of batch place $p_i$ defined by:

$$\delta_i^{\min} = s_i / V_i \qquad \text{if } p_i \in P^B. \tag{3}$$

This represents the time spent in the place by an entity of a batch in free behavior. This transfer time will be greater if the place is in accumulation behavior.

**Proposition 4.3** *Assume that a net* $\langle N, \boldsymbol{m}_0 \rangle$ *with* $P^D = T^D = \emptyset$ *is in a steady state* $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$. *The marking quantity* $q_i^s = \mu(m_i^s)$ *of a batch place* $p_i \in P^B$ *satisfies*

$$Q_i \geq q_i^s \geq \phi_i^s \ \delta_i^{\min}. \tag{4}$$

Proof. *The first inequality trivially follows from the boundedness of the place. The second inequality follows from Little's law for stationary behavior applied to each batch place* $p_i$*, that implies that the average quantity of marking it contains* $q_i$ *is equal to the product of its average input flow and of its average delay* $\delta_i$*. Since in our particular case these quantities are constants at the steady state, we write* $q_i^s = \phi_i^s \ \delta_i \geq \phi_i^s \ \delta_i^{\min}$. □

**Proposition 4.4** *Assume that a net* $\langle N, \boldsymbol{m}_0 \rangle$ *with* $P^D = T^D = \emptyset$ *is in a steady state* $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$. *The marking of a batch place* $p_i \in P^B$ *— with input/output flow* $\phi_i^s$ *and marking quantity* $q_i^s = \mu(m_i^s)$ *— takes the following* regular form*:*

1. If $\phi_i^s = 0$, *marking* $m_i^s = \{\beta_o\}$ *contains a single dense output batch* $\beta_o = (l_o, d_i^{\max}, s_i)$ *with a length* $l_o = q_i^s / d_i^{\max}$.

2. If $q_i^s = \phi_i^s \ \delta_i^{\min} > 0$, *marking* $m_i^s = \{\beta_o\}$ *contains a single output batch* $\beta_o = (s_i, d_o, s_i)$ *with a length equal to the length of the place and with a density* $d_o = \phi_i^s / V_i$.

3. If $Q_i > q_i^s > \phi_i^s \ \delta_i^{\min} > 0$, *marking* $m_i^s = \{\beta_e, \beta_o\}$ *contains a dense output batch* $\beta_o = (l_o, d_i^{\max}, s_i)$ *in contact with one input batch* $\beta_e = (l_e, d_e, l_e)$ *such that* $d_e = \phi_i^s / V_i$ *and*

$$l_e = \frac{s_i d_i^{\max} V_i - q_i^s V_i}{d_i^{\max} V_i - \phi_i^s} \quad \text{and} \quad l_o = s_i - l_e. \tag{5}$$

4. If $Q_i = q_i^s > \phi_i^s \ \delta_i^{\min} > 0$, *marking* $m_i^s = \{\beta_o\}$ *contains a single dense output batch* $\beta_o = (s_i, d_i^{\max}, s_i)$ *in accumulation behavior with a length equal to the length of the place.*

*Thus from $\boldsymbol{q}^s$ and $\boldsymbol{\varphi}^s$ the regular marking $\boldsymbol{m}^s$ can be uniquely reconstructed; we denote this $\boldsymbol{m}^s = \nu(\boldsymbol{q}^s, \boldsymbol{\varphi}^s)$.*

Proof. *In case (1) nothing can enter or leave the place: all the entities will move to the end of the place forming a single dense batch.*

*In case (2) the transfer time is equal to the minimum delay of the place. In such a case there exist an input batch $\beta_e = (l_e, d_e, l_e)$ and an output batch $\beta_o = (l_o, d_o, s_i)$ that are in free behavior (see Def. 2.12), have the same density $d_e = d_o = \phi_i^s/V_i$, and must be in contact (i.e., $l_e + l_o = s_i$) for the marking to remain constant. Hence they can be merged into a single batch $\beta_o = (s_i, d_o, s_i)$.*

*In case (3) the transfer time is greater than the minimum delay of the place. In such a case the output batch $\beta_o = (l_o, d_i^{\max}, s_i)$ must be in accumulation behavior. Consider now the input batch $\beta_e = (l_e, d_e, l_e)$ where $d_e = \phi_i^s/V_i$. The two batches must be in contact for the marking to remain constant. It holds:*

$$\begin{array}{rcl} l_e + l_o & = & s_i \\ l_e d_e + l_o d_i^{\max} & = & q_i^s \end{array} \quad \Longrightarrow \quad \begin{array}{rcl} l_e + l_o & = & s_i \\ l_e \phi_i^s / V_i + l_o d_i^{\max} & = & q_i^s \end{array}$$

*and it is easy to prove that if $s_i d_i^{\max} = Q_i > q_i^s > \phi_i^s\ \delta_i^{\min} = \phi_i^s\ s_i/V_i$ holds this system with unknown $l_e$ and $l_o$ admits a single solution given by* (5)*.*

*In case (4) the place is full, hence it must contain a single dense batch with a length equal to the length of the place. Furthermore, since it is not in a free behavior, it must be in accumulation behavior.* □

We can now state the main result of the paper.

**Proposition 4.5** Given a GBPN $\langle N, \boldsymbol{m}_0 \rangle$ with $P^D = T^D = \emptyset$, consider the following constraint set:

$$\left\{ \begin{array}{ll} (a) & \mathbf{0} \leq \boldsymbol{y} \leq \boldsymbol{\Phi} \\ (b) & Q_i \geq q_i \geq Pre(p_i, \cdot) \cdot \boldsymbol{y} \cdot \delta_i^{\min} \qquad (\forall p_i \in P^B) \\ (c) & Post(p_i, \cdot) \cdot \boldsymbol{y} \leq V_i \cdot d_i^{\max} \qquad (\forall p_i \in P^B) \\ (d) & \boldsymbol{C} \cdot \boldsymbol{y} = \mathbf{0} \\ (e) & \boldsymbol{q} \in RQ(N, \boldsymbol{m}_0) \end{array} \right. \tag{6}$$

where $\boldsymbol{q} \in \mathbb{R}^m$ and $\boldsymbol{y} \in \mathbb{R}^n$ are unknown, and all other parameters, that depend on structure on the net, have previously been defined.

(a) A steady state $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$, reachable from the initial marking, is such that $(\boldsymbol{q}, \boldsymbol{y})$, with $\boldsymbol{q} = \mu(\boldsymbol{m}^s)$ and $\boldsymbol{y} = \boldsymbol{\varphi}^s$, satisfies eq. (6).

(b) Given a solution $(\boldsymbol{q}, \boldsymbol{y})$ of eq. (6), there exists a steady state $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$, reachable from the initial marking, such that $\boldsymbol{m}^s = \nu(\boldsymbol{q}, \boldsymbol{y})$ and $\boldsymbol{\varphi}^s = \boldsymbol{y}$.

*Proof.* We prove separately the two conditions.

(a) Since $\boldsymbol{m}^s$ is reachable, then by definition, constraint (6).e is satisfied. Furthermore, $\boldsymbol{\varphi}^s$ is an admissible firing vector that satisfies eq. (1), hence eq. (6).a and eq. (6).c also hold. The assumption that $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$ is a steady state has two implications. By Proposition 4.4 eq. (6).d holds. Secondly by Proposition 4.3, eq. (6).b holds.

(b) Assume $(\boldsymbol{q}, \boldsymbol{y})$ is a solution of eq. (6). We first claim that if any marking $\boldsymbol{m} \in \mu^{-1}(\boldsymbol{q})$ is reachable, as implied by eq. (6).e, then the regular marking $\boldsymbol{m}^s = \nu(\boldsymbol{q}, \boldsymbol{y})$ is also reachable. We propose to do this in two steps.

In a first step, from $\boldsymbol{m}$ blocking all transitions the marking quantity in each batch place accumulates in a single *output dense* batch reaching a marking $\boldsymbol{m}^{od} \in \mu^{-1}(\boldsymbol{q})$.

In a second step, from $\boldsymbol{m}^{od}$ it is possible to choose an IFF vector $\boldsymbol{\varphi}^s = \boldsymbol{y}$. In fact, constraint (6).a satisfies eq. (1).a; constraint (6).c is equivalent to eq. (1).e; constraint (6).d satisfies eq. (1).c and eq. (1).d. Constraint (1).b is always satisfied as we consider a net without discrete nodes. Finally constraint (6).c can be rewritten as $Pre(p_i, \cdot) \cdot \boldsymbol{y} \leq q_i V_i / s_i \leq V_i d_i^{\text{out}}$. One can easily verify that from marking $\boldsymbol{m}^{od}$ the application of the IFF vector $\boldsymbol{\varphi}^s$ yields the regular marking $\boldsymbol{m}^s$ in a time $\tau \leq \max_{p_i \in P^B} \{\delta_i^{\text{min}}\}$.

Finally, once $\boldsymbol{m}^s$ is reached, the IFF vector $\boldsymbol{\varphi}^s$ can still be applied but will not change the marking thus $(\boldsymbol{m}^s, \boldsymbol{\varphi}^s)$ is a reachable steady state. □

Constraint (6).e, that implies that the marking quantity is reachable from the initial marking, can be often replaced or approximated by the state equation, or by the conservation law given by the place invariants thus leading to a linear algebraic problem.

**Example 4.6** *Consider the net obtained from the one in Fig. 2, removing place $p_3$ and transitions $t_3, t_4$. The initial marking is $\boldsymbol{m}_0 = [8\ \emptyset]^T$ and $\delta_2^{\text{min}} = s_2/V_2 = 5$. It holds*

$$\begin{cases} (a) & 0 \leq y_1 \leq 3 \\ (a') & 0 \leq y_2 \leq 1 \\ (b) & 10 \geq q_2 \geq y_2\ \delta_2^{\text{min}} \\ (c) & y_1 \leq 2 \\ (d) & y_1 = y_2 \\ (e) & q_1 + q_2 = 8 \end{cases}$$

*With $\max\{y_1 + y_2\}$ as objective function, this system has solution $y_1 = y_2 = 1$, $q_1 \in [0, 3]$ and $q_2 = 8 - q_1$ (constraint (6).b implies $q_2 \geq 5$). Thus a family of steady states is given by $\varphi_1^s = \varphi_2^s = 1$ where place $p_2$ contains:*

- *for $m_1 = 3$: an output batch $\beta_1 = (l_1, d_1, x_1) = (5, 1, 5)$;*
- *for $0 \leq m_1 < 3$: a batch $\beta_1 = (2 + m_1, 1, 2 + m_1)$ and a dense output batch $\beta_2 = (3 - m_1, 2, 5)$ in accumulation behavior.*

## 5 Conclusions and future work

In this paper we have developed new linear algebraic techniques for the analysis and control of Generalised Batches Petri nets. Two main contributions have been presented. The first contribution lies in the fact that although we consider the same GBPN model that has already been presented in the literature, we associate to this model a different semantics considering that the instantaneous firing flow of continuous and batch transitions are control variables. The second contribution consists in the analysis of the steady state behavior of GBPN.

Several lines are open for future research. First, we plan to also consider the speeds of batch places as control variables: this extension is particularly useful to describe transportation systems with variable speed limits and to analyze congestion control problems. Furthermore, we believe that the assumption that a steady state be characterized by constant marking and IFF vector is rather strong. We plan to characterize more general stationary evolutions where the marking quantity and the IFF vectors change in time (e.g., periodically) but have a constant average value.

## References


[1] F. Balduzzi, A. Giua, and G. Menga. First–order hybrid Petri nets: a model for optimization and control. *IEEE Trans. on Robotics & Automation*, 16(4):382–399, 2000.

[2] R. David and H. Alla. *Discrete, Continuous and Hybrid Petri Nets*. Springer-Verlag, 2005.

[3] I. Demongodin. Generalised batches Petri net: Hybrid model for high speed systems with variable delays. *Discrete Event Dynamic Systems*, 11(1&2):137–162, 2001.

[4] I. Demongodin. Modeling and analysis of transportation networks using batches Petri nets with controllable batch speed. In *ICATPN 2009, LNCS 5606*, pages 204–222, 2009.

[5] B. Gaujal and A. Giua. Optimal stationary behavior for a class of timed continuous Petri nets. *Automatica*, 40(9):1505–1516, 2004.

[6] A. Giua, C. Mahulea, L. Recalde, C. Seatzu, and M. Silva. Optimal control of timed continuous Petri nets via explicit MPC. In *POSTA 2006, LNCS 341*, pages 383–390, 2006.

[7] J. Júlvez, L. Recalde, and M. Silva. Steady-state performance evaluation of continuous mono-t-semiflow Petri nets. *Automatica*, 41(4):605 – 616, 2005.

[8] Cristian Mahulea, Laura Recalde, and Manuel Silva. Basic server semantics and performance monotonicity of continuous Petri nets. *Discrete Event Dynamic Systems*, 19(2):189–212, 2009.

[9] M. Silva and L. Recalde. On fluidification of Petri net models: from discrete to hybrid and continuous models. *Annual Reviews in Control*, 28(2):253–266, 2004.