RESOLUTION OF ACTUAL CONFLICTS IN CONSTANT SPEED CONTINUOUS PETRI NETS

Zdenek Hanzalek

Department of Control Engineering
Czech Technical University in Prague, Karlovo nam. 13
121 35 Prague 2, Czech Republic, hanzalek@fel.cvut.cz

Abstract: A computation of an instantaneous firing speed in an invariant behavior state of a Constant speed Continuous Petri Net is presented in this article. The instantaneous firing speeds, constrained by a system of inequalities, are represented by polytopes in order to clarify a nondeterminism issued by an actual conflict. The instantaneous firing speeds are determined by a subset of the polytope vertices or, when the actual conflicts are resolved by global priorities, they are found by one formulation of a linear programming problem per each priority level. An example of water production shows consequencies of the fact that an infinitely small quantity of the marking in a conservative component enables to run the water production system at its maximum speed.

Keywords: Continuous Petri Nets, actual conflicts, convex optimisation

1. INTRODUCTION

This article is based on a Constant speed Continuous Petri Net (CCPN) model presented by R. David and H. Alla (H. Alla, 1998). These authors obtained a continuous model by a fluidification of a discrete Petri Net. This approach can be particularly useful when a Continuous Petri Net is used to approximate a discrete Petri Net, since the evolution graph can represent the net behavior in a very dense form. For a given CCPN, a set of enabled transitions is identified for each Invariant Behavior state (IB-state) when an *evolution graph* is constructed. The *Instantaneous Firing speed* (IF-speed) of each enabled transition is constrained by the transition's maximal speed. Both, the IF-speed and the marking derivative, are constant in a given IB-state. Consequently, a marking of CCPN is a continuous function of time, but the IF-speed is not a continuous function of time. This article is motivated by the fact that the iterative algorithm finding IF-speed given in (H. Alla, 1998) may not be used when there is an actual conflict (informally we say that there is the actual conflict between two transitions when possible increase in IF-speed of one transition must be compensated by the decrease in IF-speed of the other transition).

B. Gaujal and A. Giua (Gaujal and Giua, 2002) considered conflicts in Cohen's model where a release delay is associated to each place. Consequently, neither the marking nor a *firing rate* is a continuous function of time in this model. The authors use a stationary routing (Alpan and Gaujal, 2000) to solve a conflict via transformation to a conflict free net.

In the model presented by L. Recalde, J. Julvez and M. Silva (Recalde *et al.*, 2002), where the speed parameters are associated to the transitions, the decrease rate of the marking depends of its size. Therefore evolution of this model is given by a set of differential equations with the *min* operator (when the net is join-free then it is given by an ordinary set of differential equations without *min* operator). Consequently, the marking and a *flow through the transition* are continuous functions of time in this model. The authors study so called Equal Conflict nets, where the ratio of the flows through two conflicting transitions is equal to the ratio of the speed parameters of these transitions.

First-Order Hybrid Petri Nets (Balduzzi *et al.*, 2000) and (Balduzzi *et al.*, 2001) use linear programming to simulate behavior of the hybrid systems. The First-Order Hybrid Petri Nets (FOHPN) model is very close to CCPN model, but it differs from the one presented by R. David and H. Alla in two aspects. First, in FOHPN model a weakly enabled transition does not require an upstream strongly enabled transition. Consequently, transitions forming a conservative component with zero marking are enabled in FOHPN. Second, a minimum firing speed is defined for FOHPN model.

Autonomous Continuous Petri Nets (Recalde *et al.*, 1999) and other models like DAE (Differential Algebraic Equations) Petri Nets (Champagnat *et al.*, 2001), Batches Petri Nets (Demongodin, 2001) have been studied intensively since this research area presents an important bridge to hybrid systems (a bibliography on hybrid Petri Nets can be found at http://bode.diee.unica.it/~hpn). CCPNs constitute a part of Hybrid Petri Nets (R. David, 2001). Further, Extended Hybrid Petri nets, defined in (R. David, 2000), enable to model delays on continuous flows.

The rest of this article is organized as follows: Section 2 shows a motivation example and it surveys basic terms of CCPNs. Section 3 presents a free speed model where the speed maximization is not assumed. It shows how the space of IF-speeds in the free speed model can be determined by a polytope. Section 4 presents a *maximum speed area* to which the IF-speed has to belong, when the maximum speed model is assumed. Examples of actual conflicts show that this area is not always convex (e.g. when it consists of several polytope faces). Section 5 presents a resolution of actual conflicts by priorities, it defines a *priority-determined speed* and it proposes two algorithmic solutions, one based on polytopes and the otherone based on linear programming.

2. PRELIMINARIES

In order to illustrate the consequences of the fluidification on a CCPN model (e.g., even an infinitely small quantity of the marking in a conservative component enables to run a system at its maximum speed) we show a motivation example.

Example 1: Let us assume a water production system depicted in Fig. 1. Water is produced by two water houses - the first pumping clean water from a drilled well (transition T_1) and the second cleaning river water (transition T_2). Water from both water houses, accumulated in a common water storage tank (place P_3), is used by an oil refinery (transition T_3) producing diesel oil and a chemical factory (transition T_4) producing disinfections. Diesel oil (a storage tank represented by place P_1) is used by the pumping water house (T_1) and the disinfections are used by the cleaning water house (a storage tank represented by place P_2). All storage tanks are empty except the one containing 0.2 units of diesel oil (P_1). Each company produces one unit of the output product from one unit of the input product (it is not realistic and given formalism alows to have real positive numbers associated to arcs, but it keeps the example transparent), therefore the weights of all arcs are equal to 1. Each of the companies has its own upper bound on the production speed. It is given as a quantity of liquid per unit of time (corresponding to a *maximal firing speed* indicated inside of each transition).

None of the companies has a lower bound on the production speed. In addition there are four specific assumptions:

- (a) None of the companies has a *threshold* for the quantity of the input product. Therefore, each company is able to produce 1/k units of the output liquid from 1/k units of input liquid in 1/(k × IF-speed) units of time, even if k tends to infinity. In contrast to the discrete Petri Nets, there is no threshold for the quantity of the marking enabling a transition, i.e. transition T_1 is strongly enabled even if $M_1 < Pre_{1,1}$.
- (b) There is no transport delay in the system, therefore we do not model any *delay* of continuous flows.
- (c) All the companies produce *as much as possible.*
- (d) The chemical factory (transition T_4) takes priority over the oil rafinery (transition T_3).

(a) and (b) are assumed in existing CCPN models without mentioning them explicitly, but they have very important consequences as we will see later in this text. Assumption (c) can also be misunderstood and therefore we give its formal description in Section 4. Assumption (d), related to the conflict resolution described in Section 5, is formalized using global priority. Assumptions (c) and (d) together can be understood as a complex optimization criterion.

All places and transitions of the CCPN model in Fig. 1 are continuous in accordance to the definitions of the CCPN given below.

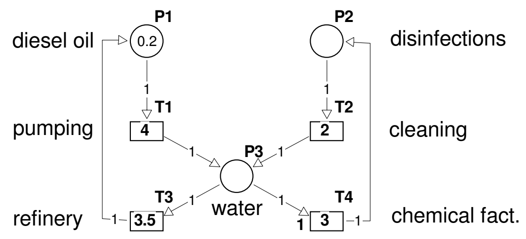

Fig. 1. CCPN with actual conflict

Definition 1. A *Constant speed Continuous Petri Net* is a sextuple $R = [P, T, V, Pre, Post, M(0)]$, where:

- The definitions of $P, T, Pre, Post$ are similar to those of discrete PNs. Examples in this article have only natural valued weights of arcs, however general case, where *real positive numbers* are associated with arcs, can also be considered.
- $M(0)$ is the initial marking of the continuous PN. It is a vector of positive or zero real numbers. $M(t)$ denotes the marking at time t.
- $V : T \longrightarrow R^+$ is a vector of maximal firing speeds; V_j denotes the *maximal firing speed* of transition T_j.

Further $v_j(t)$ denotes the *IF-speed* of transition T_j at time t. The value of $v_j(t)$ is bounded by interval $\langle 0, V_j \rangle$.

Definition 2. Transition T_j is *strongly enabled* at time t if all places P_i of ${}^oT_j$ are marked.

Definition 3. Place P_i is *supplied* at time t if there is at least one transition T_j in ${}^oP_i$, which is enabled (strongly or weakly).

Definition 4. Transition T_j is *weakly enabled* at time t if there is place P_i of ${}^oT_j$, which is not marked and is supplied, and remaining places of ${}^oT_j$ are either marked or supplied.

The recursive definitions of the supplied place and the weakly enabled transition do not allow direct determination of supplied places and weakly enabled transitions. A calculation of the set of enabled transitions for a given marking is given by Algorithm 1 in (H. Alla, 1998). This algorithm, based on an iterative upgrade of vectors A (one bit assigned to each place) and E (one bit assigned to each transition), converges in polynomial time.

Definition 5. Place P_i is *supplying* if there is at least one transition T_j in P_i^o, which is enabled (strongly or weakly).

Definition 6. The *balance* of P_i is:

$$B_i(t) = \sum_{T_j \in {}^oP_i} Post_{i,j} \cdot v_j(t) - \sum_{T_k \in P_i^o} Pre_{i,k} \cdot v_k(t) \qquad (1)$$

The balance of P_i at time t corresponds to the derivative of its marking, i.e., $m_i'(t) = B_i(t)$. If the balance of P_i is positive the marking m_i increases and if the balance is negative the marking m_i decreases.

An algorithm, calculating the IF-speeds of enabled transitions in (H. Alla, 1998), is based on the iterative approach (i.e. $v_j^{r+1}(t)$, the value of IF-speed at iteration step $r + 1$, is dependent on $v_j^r(t)$, the value from the previous step), while using the *min* operator. On the contrary, the approach presented in this article is based on the analytical determination of subspace of IF-speeds, while avoiding the *min* operator.

3. FREE SPEED CCPN

CCPNs are assumed to function at a *maximum speed* (H. Alla, 1998), therefore $v_j(t)$ of strongly enabled transition T_j is equal to V_j and $v_j(t)$ of weakly enabled transition is the maximum possible one (see assumption (c) in Example 1). This classical model will be called a *maximum speed CCPN* in this article.

On the other hand, the systems not functioning at their maximum speed are also a very interesting subject of research (e.g., when studying a marking reachability in Example 1 while omitting assumptions (c) and (d), i.e. assuming that companies are not motivated to produce as much as possible and their priority of access to shared resource is not determined). This model can be used when one describes systems with undetermined but bounded firing speed (in a hypothetic case, when $V_j = \infty$ for all $T_j \in T$, the free speed CCPN is identical to the autonomous CPN).

Therefore, a subspace of IF-speeds for a *free speed CCPN* is constrained by:
Speed limits of enabled transitions.

$$v_j(t) \le V_j \quad \forall j \textit{ such that } T_j \textit{ is enabled} \qquad (2)$$

Non-negative speeds of enabled transitions.

$$v_j(t) \ge 0 \quad \forall j \textit{ such that } T_j \textit{ is enabled} \qquad (3)$$

Non-negative balances of unmarked supplying places (if there is place P_i which is unmarked supplying, then there exists at least one weakly enabled transition in P_i^o):

$$B_i(t) \ge 0 \quad \forall i \textit{ such that } P_i \textit{ is supplying and } m_i(t) = 0 \qquad (4)$$

All variables dependent on time, $v(t), B(t), M(t)$, will be denoted simply v, B, M in the rest of this article, since they are calculated at the begin-ing/end of each IB-state ($v(t), B(t)$ are constant inside the IB-state, and $m_i(t+dt) = m_i(t)+B_i(t)\cdot dt$).

Each B_i can be written as a linear combination of IF-speeds of enabled transitions, due to equation (1). As a consequence the system of inequalities (2), (3), (5) can be written as a linear function of IF-speeds.

Let c be a number of unmarked supplying places, d denotes a number of enabled transitions and k denotes an index of an enabled transition ranging from 1 to d. The subspace Φ of R^d is a *convex polyhedron* (Fukuda, n.d.; Ziegler, 1998) since Φ is the set of solutions to the above mentioned finite system of inequalities (exactly there are $2d + c$ inequalities). Φ is a *polytope*, as it is the convex polyhedron and it is bounded (due to (2) and (3)). One given point in Φ corresponds to one given IF-speed vector v. If T_j is enabled then v_j is equal to the k-th coordinate of this point, otherwise v_j is equal to zero.

A subset F of polyhedron Φ is called a *face* of Φ if inequality $a^T x \leq b$ holds for all $x \in \Phi$ and $F = \Phi \cap \{x; a^T x = b\}$. The faces of dimension 0,1,d-1 are called *vertices, edges* and *facets*, respectively. No vertex can be represented as a convex combinations of two other points in Φ.

For each IB-state of the evolution graph we are able to derive the system of inequalities (2), (3), (5). The free speed model of Petri Net in Fig. 2 is given by the system of inequalities:

$$
\begin{aligned}
&v_1 \leq 3 \quad facet\ T_1\ in\ Fig.\ 3,\\
&v_2 \leq 2 \quad facet\ T_2\ in\ Fig.\ 3,\\
&v_3 \leq 2 \quad facet\ T_3\ in\ Fig.\ 3,\\
&v_1 \geq 0 \quad redundant\ ineq.\ -\ vertex\ T_{01}\ in\ Fig.\ 3,\\
&v_2 \geq 0 \quad facet\ T_{02}\ not\ visible\ in\ Fig.\ 3,\\
&v_3 \geq 0 \quad facet\ T_{03}\ not\ visible\ in\ Fig.\ 3,\\
&v_1 \geq v_2 + v_3 \quad facet\ P_2\ in\ Fig.\ 3. \qquad (5)
\end{aligned}
$$

This is in fact a (halfspace) *H-representation* of polytope Φ which can also be given by a set of vertices, so called *V-representation.* A transformation of *H-representation* to *V-representation* is known as *vertex enumeration* (Fukuda, n.d.; Ziegler, 1998).

The corresponding polytope in Fig. 3 shows, that any possible IF-speed v_1 can grow up to its upper bound V_1. On the other hand, v_3 can reach its upper limit V_3 (facet T_3) only in the area where $v_1 - v_2 \geq V_3$ holds. In the remaining area ($v_1 - v_2 < V_3$) any possible IF-speed v_3 can grow only up to $v_1 - v_2$ (facet P_2). In other words, the growth of v_3 is not limited only by the preceding transitions but also by v_2, which does not precede T_3. This is caused by conflict place P_2 in Fig. 2.

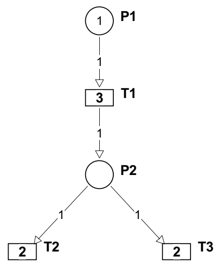

Fig. 2. CCPN with conflict

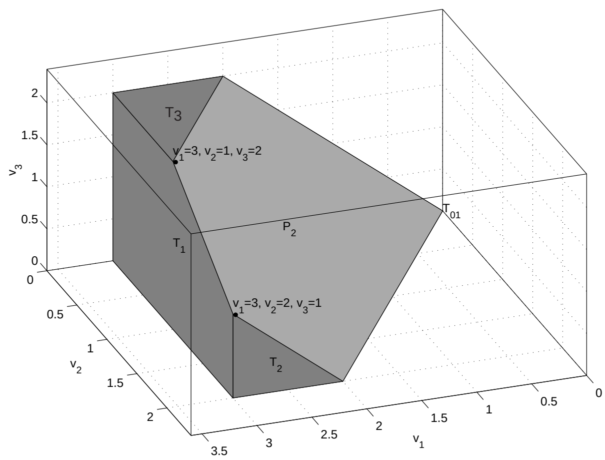

Fig. 3. Polytope Φ representing possible IF-speeds of the free speed CCPN in Fig. 2

The free speed model is not deterministic; therefore it cannot be used to construct the evolution graph, where deterministic marking between two IB-states is needed. But it can be used as an intermediate step to determine the IF-speed as shown in Section 4.

4. MAXIMUM FIRING SPEED

When omitting only assumption (d) in Example 1 one obtains the classical CCPN model (in this article called *maximum speed CCPN*), i.e. the system functioning at maximum speed. Therefore, $v_j = V_j$ for each strongly enabled transition T_j. Consequently new *polytope* Θ of R^w (w =*number of weakly enabled transitions*) can be obtained by a *reduction* of polytope Φ of R^d ($d = w+$*number of strongly enabled transitions*). This reduction is done by changing each inequality (2), which is related to a strongly enabled transition, to the equation. Θ is also convex and bounded as required by the definition of polytope.

Polytope Θ of CCPN in Fig. 2 is shown in Fig. 4. In fact Θ is face of Φ and it is obtained as an intersection of polytope Φ (see Fig. 3)with the plane $v_1 = 3$, corresponding to strongly enabled transition T_1.

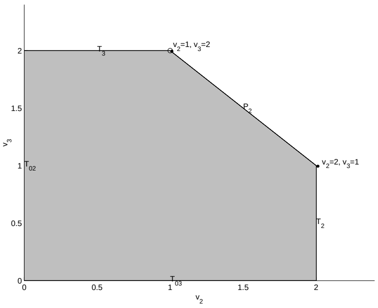

Fig. 4. Polytope Θ used to determine the maximum speeds of weakly enabled transitions in Fig. 2 (Γ consists of edge $([v_2 = 1, v_3 = 2][v_2 = 2, v_3 = 1])$)

Both polytopes Φ and Θ can be used, when studying a maximum speed CCPN. In order to demonstrate various cases in 3-dimensional space, only polytope Θ will be used in the following figures, since $dim(\Theta) \leq dim(\Phi)$. In order to make the definitions simpler, polytope Φ is usually used in the text.

Definition 7. IF-speed $v = [v_1, \dots, v_k \dots, v_d]$ ($v \in \Phi$ in the sense of the polytope given by the system of inequalities (2), (3), (5)) is a *maximum speed* if there does not exist any $u \in \Phi$ such that $u_k \geq v_k$ for all $k = 1 \dots d$.

Definition 8. Subset Γ of polytope Φ is a *maximum speed area* if each $v \in \Gamma$ is the maximum speed.

Due to definition 7 it is obvious that maximum speed area Γ is a subset of the set of all faces of polytope Φ (the set is called *face poset*), since no interior point of any polytope can reach the maximum value of any convex objective function. In order to determine the maximum speed area it is sufficient to check all faces of polytope Φ.

Lemma 1. Let F_q be a q-dimesional face of Φ. F_q belongs to the maximum speed area (i.e. all points $v \in F_q$ belong to the maximum speed area) if and only if all vertices $F_0 \subset F_q$ belong to the maximum speed area.

Proof: All points $v \in F_q$ belong to the maximum speed area if and only if all faces $F_{q-1} \subset F_q$ belong to the maximum speed area. Proof is completed by recursive application of this idea downto vertices of Φ.□

Due to Lemma 1, the maximum speed area Γ is fully determined by the set of vertices belonging to it. First a set of vertices (i.e. V-representation) is obtained by the *vertex enumeration* from the the system of inequalities (2), (3), (5) (i.e. H-representation). Then a given vertex v is compared to all other vertices u following definition 7. Due to definition 7 it is obvious that at least one vertex of Φ belongs to Γ and that $\Gamma \subset \Theta \subset \Phi$.

This procedure is applied to all examples in this article. A vertex belonging to the maximum speed area is indicated by a small dot, labeled by the IF-speed. For example the polytope in Fig. 3 has two vertices of this kind $v = [v_1 = 3, v_2 = 1, v_3 = 2]$ and $v' = [v_1 = 3, v_2 = 2, v_3 = 1]$, therefore Γ is the edge (v, v'). The speed maximization does not specify the deterministic behavior of the maximum speed CCPN given in Fig. 2. If the value of V_1 was raised up to 4, then the facet T_1 would be moved to the left in Fig. 3 and the maximum speed area would consist of only one vertex $[v_1 = 4, v_2 = 2, v_3 = 2]$ and the behavior of the maximum speed CCPN would be deterministic. This is due to the fact, that the existence of a structural conflict is a necessary but not sufficient condition for the existence of an actual conflict.

Definition 9. Let $K = [P_i, \{T_j, T_k\}]$ be a structural conflict. There is an *actual conflict* between transitions T_j and T_k if there are at least two maximum speeds v and v' such that $v_j < v'_j$ and $v_k > v'_k$.

Lemma 2. Maximum speed area Γ is exactly one vertex of Φ if and only if there is no actual conflict.

Proof: see Annex.

If there is no structural conflict then there is no nondeterminism in the maximum speed CCPN. Then the unique vertex belonging to Γ can be found in polynomial time by one call of linear programming (Cook, 1998) aiming at maximization of objective function $J = s^T x$, with arbitrary nonzero positive finite entries of s, i.e. $s_j \in (0, \infty)$ for all $j = 1 \dots w$.

Lemma 3. There exists a face $F_q \in \Gamma$ such that $q \geq 1$ if and only if there exists an actual conflict.

Proof: Due to Lemma 2 we know that there is an actual conflict if and only if there are at least two vertices in Γ. Due to the convexity of Φ and due to definition 7 at least two of these vertices belong to one face, since Lemma 1 holds.□

Example 2: (continuation of Example 1) Maximum speed area Γ of CCPN in Fig. 1 is shown in Fig. 5. Γ is one face of dimension 1, edge $([v_2 = 2, v_3 = 3, v_4 = 3], [v_2 = 2, v_3 = 3.5, v_4 = 2.5])$. There is an actual conflict between T_3 and T_4, i.e. either the chemical factory (T_3) or the oil refinery (T_4) cannot run at its maximal speed.

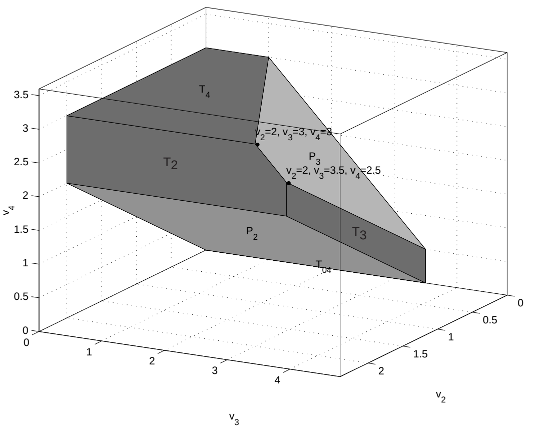

Fig. 5. Polytope Θ used to determine maximum speeds of weakly enabled transitions in Fig. 1 (Γ consists of edge $([2,3,3][2,3.5,2.5])$)

Lemma 4. There exists a maximum speed area Γ which is not convex.

Proof: Fig. 6 illustrates three structural conflicts (be aware of weights of arcs). Γ in Fig. 7 is not convex, since it is given by three faces of dimension 2, i.e. facets P_1, P_2, and P_3, whose are entirely belonging to Γ.□

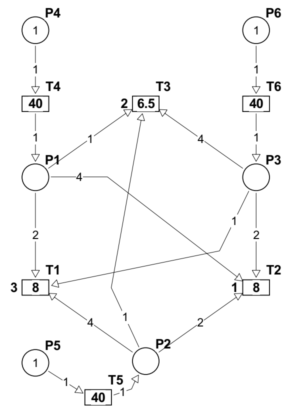

Fig. 6. CCPN with three actual conflicts

5. RESOLUTION OF ACTUAL CONFLICTS BY PRIORITIES

The deterministic behavior of a maximum speed CCPN is not given if actual conflict is present. Therefore we propose a global priority assignment.

Definition 10. Let $R = [P, T, V, Pre, Post, M(0)]$ be a maximum speed CCPN. A *maximum speed CCPN with global priorities* is a seven tuple $R' = [P, T, V, Q, Pre, Post, M(0)]$ where:

- The definitions of $P, T, V, Pre, Post, M(0)$ are the same as those in CCPN (definition 1).
- $Q : T \rightarrow \{0, N^+\}$ is a vector of global priorities; Q_j denotes *global priority* of transition T_j in the sense that T_j has a higher priority than T_l if and only if $Q_j > Q_l$.

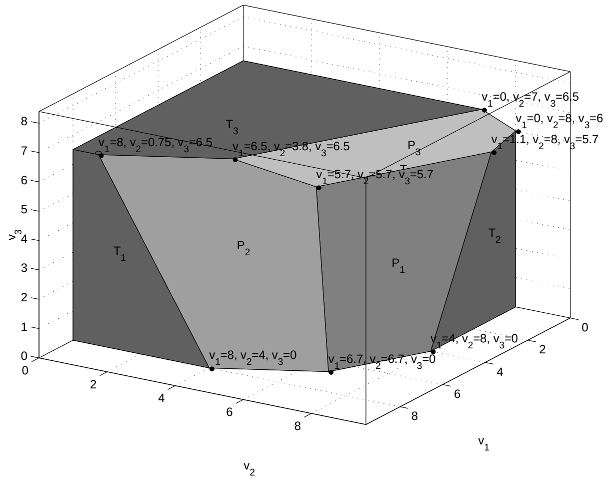

Fig. 7. Polytope Θ used to determine the maximum speeds of weakly enabled transitions in Fig. 6 (Γ consists of facets P_1, P_2, and P_3)

Definition 10 allows two transitions T_j and T_l to have the same priority $Q_j = Q_l$, so that one can use the term *priority level*, to which transitions with equal priorities are associated. The label on the left side of the transition designates its priority (see Fig. 6). Priority is 0, i.e. lowest, if there is no number on the left side of the transition.

The deterministic behavior of a maximum speed CCPN with global priorities R' is given by the choice of one maximum speed in the maximum speed area Γ.

Definition 11. Max. speed $v = [v_1, \ldots, v_j, \ldots, v_d]$, $v \in \Gamma$, is a *priority-determined speed* if for any $u \in \Gamma$ and for any T_j such that $v_j < u_j$ there exists some T_k such that $Q_k \geq Q_j$ and $v_k > u_k$.

Definition 12. Subset Ω of Γ is a *priority-determined area* of Φ if each $v \in \Omega$ is the priority-determined speed.

The priority-determined area Ω can be found by:
Vertex enumeration*. To enumerate vertices of* Θ *(using for example* lrslib *by David Avis), then to select vertices determining maximum speed area* Γ *using definition 8, then to choose the priority-determined speed using definition 11. There does not exist any polynomial bound for this algorithm since the number of vertices of* Θ *is bounded by* 2^w *(w is the number of weakly enabled transitions).*

There are several priority-determined speeds when actual conflicts are not resolved (and we can find Ω, which is not convex). On the contrary, if the system is deterministic (i.e. all actual conflicts are resolved by priorities), the priority-determined area Ω is just one priority-determined speed (one vertex of Θ). In such case the second algorithmic solution can be used:

Linear programming. *To find the maximum speed by calling linear programming for each priority level. Iterations are executed in the order given by the transitions priorities. First, one partial solution* S *is found (for all transitions* $T_j, ..., T_k$ *with the highest priority) by linear programming aiming at maximization of objective function* $J = v_j + ... + v_k$ *subject to* Θ*. Then new equation* $v_j = S_j$ *is added to the system of inequalities (corresponding new polytope* Θ' *is intersection of* Θ *with equation* $v_j = S_j$*). Then algorithm repeats for lower priority level. Further all priority levels are proceeded in the similar way and the final solution* S *determines the maximum speed satisfying priority order.*

It is difficult to construct a graph of reachable markings in CCPN, since the marking is a continuous function of time. Therefore instantaneous firing speed vector v, which remains constant in each IB-state, is used to characterize functioning of CCPN. An evolution graph is a graph, where nodes are associated with IB-states. The changeover from one IB-state to the following one occurs when the marking of a place becomes zero and consequently strongly and weakly enabled transitions have to be detected and new instantaneous firing speed vector v has to be calculated. This is classical procedure described in (H. Alla, 1998).

Example 3: (continuation of Example 2) Fig. 8 shows evolution graph of CCPN in Fig. 1 with priority assignment $Q = [0, 0, 0, 1]$. $v_4 = 3$ and $v_2 = 2$ during the first IB-state, consequently a marking is accumulated in P_2 and P_1 becomes empty. In the second IB-state, T_2 is strongly enabled and CCPN reached its stable marking. It is interesting that $v_1 \geq 0$ and $v_3 \geq 0$ even though $M_1 = M_3 = 0$. This is due to assumptions (a),(b) in Example 1, since in the modeling stage we assumed that the thresholds and the delays are negligible in our system. Therefore, negligible but not zero (T_2 supplies P_3) marking is sufficient to enable T_3 and consequently T_1. Moreover T_1 and T_3 run at the speed 3.5 . Either CCPN model in Fig. 1 is realistic and then Fig. 8 is correct, or it is not realistic and CCPN model has to be extended by threshold and delay (e.g. by making conflict resolution for Extended Hybrid Petri nets, defined in (R. David, 2000)).

6. CONCLUSIONS

This article addresses the problem of the computation of IF-speed for given IB-state of CCPN. The approach shown in this article assumes the speed maximization being prior to priority resolution, since $\Omega \subset \Gamma$.

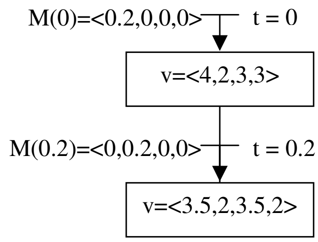

Fig. 8. Evolution graph of CCPN in Fig. 1

Two algorithmic solutions have been implemented in Matlab and constitute a part of Petri Net Matlab toolbox (available from author upon request). The *vertex enumeration* solution is based on the analytical determination of the subspace of IF-speeds. If there is no actual conflict the IF-speed is determined directly, since the maximum speed area Γ is just one vertex (see Lemma 2). The vertex enumeration solution is very illustrative (namely if there are less than 3 weakly enabled transitions), but there is no polynomial bound on the algorithm execution time. While using *linear programming* solution the IF-speed can be found by one formulation of the linear programming problem if there is no actual conflict in given CCPN. Otherwise all actual conflicts have to be resolved by global priorities, and the IF-speed is found by one formulation of the linear programming problem per each priority level.

REFERENCES

Alpan, G. and B. Gaujal (2000). Supervisory control of petri nets using routing functions: starvation avoidance issues. *IEEE Trans. on Systems, Man, and Cybernetics; B: Cybernetics, Vol. 30, No. 5* pp. 684–695.

Balduzzi, F., A. Di Febbraro, A. Giua and C. Seatzu (2001). Decidability results in first-order hybrid petri nets. *Discrete Event Dynamic Systems: Theory and Applications, Vol. 11, Numbers 1/2* pp. 41–58.

Balduzzi, F., A. Giua and G. Menga (2000). First-order hybrid petri nets: A model for optimization and control. *IEEE Transactions on Robotics and Automation, Vol. 16, Number 4* pp. 382–399.

Champagnat, R., R. Valette, J.-C. Hochon and H. Pingaud (2001). Modelling, simulation and analysis of batch production systems. *Discrete Event Dynamic Systems: Theory and Applications, Vol. 11, Numbers 1/2* pp. 118–136.

Cook, W.J. (1998). *Combinatorial Optimization.* John Wiley & Sons.

Demongodin, I. (2001). Generalised batches petri net: Hybrid model for high speed systems with variable delays. *Discrete Event Dynamic Systems: Theory and Applications, Vol. 11, Numbers 1/2* pp. 137–162.

Fukuda, Komei (n.d.). Frequently asked questions in polyhedral computations. Technical report. Swiss Federal Institute of Technology, Lausanne and Zurich, Switzerland.

Gaujal, B. and A. Giua (2002). Timed continuous petri nets and optimization via linear programming. Technical report no 4583. INRIA, France.

H. Alla, R. David (1998). A modeling and analysis tool for discrete events systems: continuous petri net. *Performance Evaluation, Vol. 33, No. 3* pp. 175–199.

R. David, H. Alla (2001). On hybrid petri nets. *Discrete Event Dynamic Systems: Theory and Applications, Vol. 11, Numbers 1/2* pp. 9–40.

R. David, S.I. Caramihai (2000). Modelling of delays on continuous flows thanks to extended hybrid petri nets. In: *Proc. of 4th International Conference Automation of Mixed Processes: Hybrid Dynamical Systems (Dortmund, Germany)*. pp. 343–351.

Recalde, L., J. Julvez and M. Silva (2002). Steady state performance evaluation for some continuous petri nets. In: *Proc. 15th IFAC World Congress, Barcelona*. Springer-Verlag. pp. 479–485.

Recalde, Laura, Enrique Teruel and Manuel Silva (1999). Autonomous continuous P/T systems.. In: *LNCS: ICATPN'99, Williamsburg, Virginia, USA*. Vol. 1630. Springer-Verlag. pp. 107–126.

Ziegler, G.M. (1998). *Lectures on Polytopes, Graduate Texts in Mathematics*. Springer.

7. ANNEX - PROOF OF LEMMA 2

Proof of "if" part: Let vertex V belongs to Γ and let $V \subset F_{d-1}$, where F_{d-1} is a facet of Φ.
(a)The facet F_{d-1} does not correspond to nonnegative speed constraint(3).
(b)The facet F_{d-1} can correspond to speed limit constraint (2), i.e. the facet is subset of $v_j = V_j$. Then V is the only point of F_{d-1} belonging to Γ, since all other points of F_{d-1} have equal coordinate v_j and smaller or equal other coordinates.
(c)The facet F_{d-1} can correspond to nonnegative balance constraint(5) of place P_i which does not constitute a structural conflict, i.e. the facet is a subset of $v_j - v_k = 0$. Again V is the only point of F_{d-1} belonging to Γ, since all other points of F_{d-1} have smaller or equal coordinates.
(d)No facet corresponds to nonnegative balance constraint (5) of place P_i which constitutes a structural conflict, since inequality corresponding to nonnegative balance of P_i is redundant due to the absence of actual conflict, i.e. there do not exist transitions T_j and T_k and two maximum speeds v and v' such that $v_j < v'_j$ and $v_k > v'_k$.
(e) Following (a),(b),(c) and (d) V is the only point belonging to Γ in any of F_{d-1} such that $V \subset F_{d-1}$. Since Φ is convex there no other point of Φ which is in Γ.□
Proof of "only if" part (i.e. if there is actual conflict, then Γ has at least two vertices): assuming existence of actual conflict, there exist two transitions T_j and T_k and two maximum speeds (not necessarily vertices) v and v' such that $v_j < v'_j$ and $v_k > v'_k$
$\Rightarrow \Gamma$ consists of at least two points v and v'
$\Rightarrow$ either both v and v' belong to one face $F_q \in \Gamma$ such that $q \geq 1$ or each of them belongs to a distinct face $F_q \in \Gamma$ such that $k \geq 1$. Therefore Γ has at least two vertices.□