6th IEEE Mediterranean Conference on Control and Systems

MANUFACTURING SYSTEMS CONTROL WITH PETRI NET OBSERVERS

Alessandro Giua, Carla Seatzu
Dip. di Ingegneria Elettrica ed Elettronica, Università di Cagliari Piazza d'Armi — 09123 Cagliari, Italy
Email: {giua,seatzu}@diee.unica.it.

This paper discusses the problem of controlling a Petri net whose marking cannot be measured. An observer is used to estimate the actual marking of the plant based on partial information of the initial marking and on event observation. This estimate is used to design a state feedback controller, that ensures that the plant in closed-loop evolves through a set of legal states. We present an efficient algorithm, that generalizes previous results, to design a safe observer-controller for legal states defined by linear constraint sets.

1. Introduction

In the classical approach of Ramadge and Wonham [12] to the supervisory control of discrete event systems, the *event-feedback* control scheme shown in Figure 1.a is adopted. Here the plant spontaneously generates events. The supervisor observes the word of events w generated and, given a set of legal words $\mathcal{K}$, computes at each step a suitable control pattern γ to ensure that no illegal word be generated.

Other authors have used a different *state-feedback* control scheme, shown in Figure 1.b. Here the supervisor observes the actual plant state M and, given a set of legal states $\mathcal{L}$, computes at each step a control pattern to ensure that no illegal state be reached. This scheme is particularly appealing when dealing with Petri net models of the plant [6], since the state of a net is given by an integer vector called *marking* and linear algebraic techniques may be used to solve the control problem.

A slightly different scheme is shown in Figure 1.c. Here the controller observes the word of events generated and, by means of an observer, it reconstructs the actual plant state M. The observer simply duplicates the plant model, and is driven by the observed events. If the structure (that is assumed to be deterministic) and the inital state M_0 of the plant are known, the knowledge of the word generated is sufficient to reconstruct the new state that each new firing yields.

In [4] a different scheme, shown in Figure 1.d, was considered and used in the context of Petri nets. In this scheme, the initial marking (state) M_0 is not completely specified, but is known to belong to a "macromarking", i.e., we know the token contents of subsets of places but not the exact token distribution. An algorithm was given for computing a marking estimate μ_w and error bound B_w. The estimate is always a lower bound of the actual marking.

In [4] a particular structure of the initial macromarking has been considered, i.e., it was assumed that the partition induced by the initial macromaking is disjoint. In this paper we exend the previous results to a more general case in which the initial macromarking is not completely disjoint: the presence of a single subset, whose intersection with all the others may be different from the empty set, is taken into account.

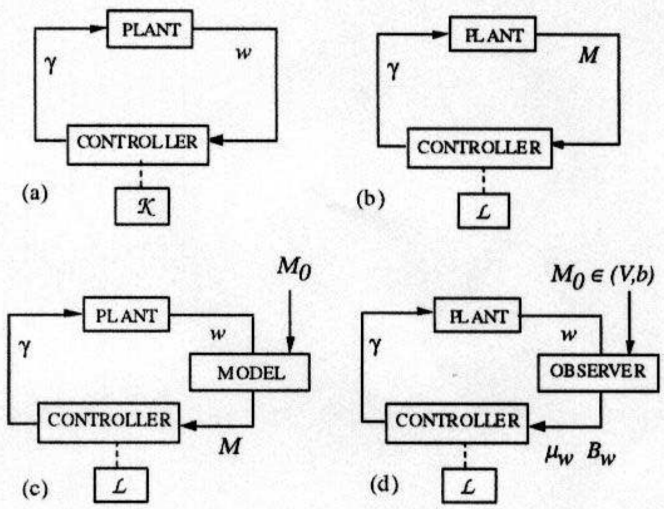

Figure 1: Different control schemes. (a) Event-feedback. (b) State-feedback. (c) State-feedback with event observer and initial marking. (d) State-feedback with event observer and initial macromarking.

The system that computes the estimate is called an observer. It has been shown in [3] that a special net structure, called *observer net*, can be used to describe the observer: such a net has a set of places corresponding to the places of the plant whose marking is at each step μ_w, plus a set of bounding places (one for each place subset in the partition induced by inital macromarking) whose marking is B_w.

The special structure of Petri nets allows us to use a simple linear algebraic formalism for estimate and error computation. In particular, the set of markings consistent with an observed word, i.e., the set of marking in which the system may actually be given the observed word, can easily be characterized in terms of the observer marking.

Finally, we show how the estimate generated by the observer may be used to design a state feedback controller, that ensures that the controlled system never enters a set of forbidden states. We consider a special class of specifications that limit the weighted sum of markings in subset of places. Clearly, the use of marking estimates (as opposed to the exact knowledge of the actual marking of the plant) leads to a worse performance of the closed-loop system in the sense that to rule out the possibility that the plant enter a forbidden marking, the controller may prevent the firing of transitions whose firing is perfectly legal given the actual marking of the plant. It may also be the case that, as a result of this, the controlled system is blocking. In [3] it has been shown how suitable time-out mechanisms may be introduced to recover from a blocking.

Control and state estimation under partial observation has been discussed in the discrete event control literature. Zhang and Holloway [15] used a Controlled Petri Net model for forbidden state avoidance under partial *event* observation with the assumption that the initial marking be known. The use of state-feedback control under partial (state) observation has been discussed by Li and Wonham [8, 9] and by Takai *et al.* [13].

The idea of constructing estimates of the unknown plant state for systems represented as finite automata has also been discussed in literature by Caines *et al.* [1, 2] and by Kumar *et al.* [7]. A comparison of all these approaches can be found in [4].

The present work has several motivations. The assumption that only event occurences may be observed, while the plant state cannot, is common in discrete event control. The assumption that the marking of the plant is not known (or is only partially known) is natural during error recovery. Consider for instance the case of a plant remotely controlled: if the communication fails the state may evolve and when the communication is restablished the state will be at best partially known. In a manufacturing environment, one may consider the case in which resources (i.e., tokens) enter unobserved, or in which we know how many resources have entered the system but not their exact location.

The paper is structured as follows. Section 2 introduces the Petri net formalism used in the following sections. Section 3 discusses the problem of estimating the marking of a net whose initial marking is partially known as a macromarking. Section 4 shows how the observer net can be constructed. Section 5 shows how it is possible to use state feedback control using the estimated marking computed by the observer. A manufacturing example is discussed in Section 6.

2. Background

We recall the Petri net formalism used in this paper. For a more comprehensive introduction to Petri nets see [11]. A *Place/Transition net* (P/T net) is a structure $N = (P, T, Pre, Post)$, where P is a set of m *places*; T is a set of n *transitions*; $Pre : P \times T \to \mathbb{N}$ and $Post : P \times T \to \mathbb{N}$ are the *pre-* and *post-incidence functions* that specify the arcs.

The *incidence matrix* of the net is defined as $C(p,t) = Post(p,t) - Pre(p,t)$.

A *marking* is a vector $M : P \to \mathbb{N}$ that assigns to each place of a P/T net a non-negative integer number of tokens, represented by black dots. A *P/T system* or *net system* $\langle N, M_0 \rangle$ is a net N with an initial marking M_0.

A transition t is enabled at M if $M \geq Pre(\cdot, t)$ and may fire yielding the marking $M' = M + C(\cdot, t)$. We write $M\ [w\rangle\ M'$ to denote that the enabled sequence of transitions w may fire at M yielding M'.

A marking M is *reachable* in $\langle N, M_0 \rangle$ iff there exists a firing sequence w such that $M_0\ [w\rangle\ M$.

We denote $M_{w_0} = M_0$ the initial marking and M_w the marking reachable from M_{w_0} firing the sequence w.

3. Observers and macromarkings

In [4] an algorithm was given for estimating the marking of a net system $\langle N, M_0 \rangle$ whose marking cannot be directly measured when partial information about its initial marking is given in the form of a *macromarking*.

Definition 1. Let us assume that the set of places P be partitioned in $r+1$ subsets: $P = P_0 \cup P_1 \cup \cdots \cup P_r$, with $P_0 \cap P_j = \emptyset$ for all $j > 0$. The number of tokens contained in P_j $(j > 0)$ is known to be b_j, while the number of tokens in P_0 is unknown. For each P_j, let $\vec{v}_j$ be its characteristic vector (i.e., $\vec{v}_j(p) = 1$ if $p \in P_j$, else $\vec{v}_j(p) = 0$). Let $V = [\vec{v}_1\ \vec{v}_2 \cdots \vec{v}_r]$ and $\vec{b} = [b_1 b_2 \cdots b_r]^T$. Then the *macromarking* $(V, \vec{b})$ is defined as the set $\{M \in \mathbb{N}^{|P|} \mid V^T \cdot M = \vec{b}\}$. ∎

We make the following assumptions.

A1) The structure of the net $N = (P, T, Pre, Post)$ is known, while the initial marking M_0 is not.

A2) The event occurences (i.e., the transition firings) can be observed.

A3) The initial marking M_0 is known to belong to the macromarking$(V, \vec{b})$, i.e., it is known to satisfy the equation $V^T \cdot M_0 = \vec{b}$.

The use of macromarkings comes out quite naturally when describing systems containing a known set of resources (e.g., parts, machines) whose actual conditions (e.g., exact location of parts within the plant, state of a machine) is unknown.

After the word w has been observed we define the set $\mathcal{M}(w \mid V, \vec{b})$ of w consistent markings as the set of all markings in which the system may be given the observed behaviour and the inital macromarking.

Definition 2. Given an observed word w and an initial macromarking $(V, \vec{b})$, the set of w *consistent markings* is

$$\mathcal{M}(w \mid V, \vec{b}) = \{M \mid \exists M' : V^T \cdot M' = \vec{b}, M'[w\rangle M\}.$$

∎

Given an evolution of the net $M_{w_0}[t_1\rangle M_{w_1}[t_2\rangle M_{w_2}[t_3\rangle \cdots$, we use the following algorithm to compute the estimate μ_{w_i} of each actual marking M_{w_i} based on the observation of the word of events $w_i = t_1, t_2, \cdots, t_i$, and of the knowledge of the initial macromarking $(V, \vec{b})$. We denote the empty word as w_0, and the estimate of the initial marking as μ_{w_0}.

Algorithm 3 ([4]). *Marking Estimation with Event Observation and Initial Macromarking*

1. Let the initial estimate be μ_{w_0}, with

$\mu_{w_0}(p) = \min M(p)$ such that $V^T \cdot M = \vec{b}$.

2. Let the initial bound be $B_{w_0} = \vec{b} - V^T \cdot \mu_{w_0}$.

3. Let $i = 1$.

4. Wait until t_i fires.

5. Update the estimate $\mu_{w_{i-1}}$ to $\mu'_{w_{i-1}}$ with $\mu'_{w_{i-1}}(p) = \max\{\mu_{w_{i-1}}(p), Pre(p, t_i)\}$.

6. Let $\mu_{w_i} = \mu'_{w_{i-1}} + C(\cdot, t_i)$.

7. Let $B_{w_i} = B_{w_{i-1}} - V^T \cdot (\mu'_{w_{i-1}} - \mu_{w_{i-1}})$.

8. Let $i = i + 1$.

9. Goto 4.

In [4] a meaningful measure of the estimation error was defined as the token difference between a marking and its estimate and was studied under which conditions an observed word leads to a null estimation error. If place p belongs to a subset P_j $(j > 0)$ of the partition induced by the initial macromarking, it is possible to prove that the place estimation error between a marking M and its estimate μ_w is such that $M(p) - \mu_w(p) \leq B_w(j)$, where $B_w(j)$ is the j–th component of B_w.

Furthermore, the set of consistent markings can be characterized in terms of a set of linear inequalities as a function of μ_w and B_w.

Theorem 4 ([4]). Given an observed word $w \in L(N, M_0)$ with initial macromarking $(V, \vec{b})$, the corresponding estimated marking μ_w and bound B_w computed by Algorithm 3, the set of w *consistent markings* is

$$\mathcal{M}(w \mid V, \vec{b}) = \{M \geq \mu_w \mid V^T \cdot M = V^T \cdot \mu_w + B_w\}.$$

■

A note about the computation of the initial estimate in step 1 of Algorithm 3. This computation requires to solve an integer problem of the form:

$$\begin{aligned} &\max\ \vec{c}^T \cdot M \\ &\text{s.t. } M \in \mathcal{M}(w \mid V, \vec{b}) \end{aligned} \tag{1}$$

A similar integer problem needs also be solved at each step by the controller to compute $\overline{m}$, as described in Algorithm 6.

Given the linear algebraic representation of the set of w consistent markings, one may assume that finding a solution requires integer programming methods, that are notoriously computationally hard.

It has been shown in [4] that a closed form solution of (1) can be given when all partitions are disjopint. In this paper we consider a more general partitioning of the set P (that we formalize in the next assumption A4) and derive a closed form solution of (1) when *the objective function coefficients are nonnegative.*

We consider the following assumption:

A4) The partition $P = P_0 \cup P_1 \cup \cdots \cup P_r$ induced by the initial macromarking is such that:

(a) $P_j \cap P_{j'} = \emptyset$ for all $j \neq j'$, $j, j' > 1$.

(b) Let $\Delta := P_1 \cap (P_2 \cup \cdots \cup P_r)$; then $P_j \Delta \neq \emptyset$ for all $j \geq 1$.

The first part of the assumption states that P_1 may have non empty intersection Δ with the union of all other partitions, while the P_j $(j > 1)$, may intersect P_1 but are mutually disjoint. The second part of the assumption states that at least one place in each partition is not shared with any other partition.

Proposition 5. Consider the integer problem

$$\begin{aligned} &\max\ \vec{c}^T \cdot M \\ &\text{s.t. } V^T \cdot M = V^T \cdot \mu_w + B_w \\ &\quad\ \ M \geq \mu_w \end{aligned}$$

with $\vec{c} \geq \vec{0}$; V, B_w and μ_w given by Algorithm 3; $P = P_0 \cup P_1 \cup \cdots \cup P_r$ given as in definition 1 and satisfying assumption A4.

The optimal solution M^* can be given as follows.

If $P_0 \neq \emptyset$ and $\exists p \in P_0 : c(p) > 0$, M^* is unbounded. Otherwise a finite optimal solution can be computed by means of the following algorithm.

begin
$\Delta := P_1 \cap (P_2 \cup \cdots \cup P_r)$;
$M := \mu_w$; $B := B_w$;
$P_{old} := P_0 \cup \{P_j \mid B(j) = 0\}$;
while $P_{old} \subsetneq P$ **do**
  begin
  let $p^* : c(p^*) = \max\{c(p) \mid p \in P \setminus P_{old}\}$;
  let $j^* = \max\{j \mid p^* \in P_j\}$;
  if $p^* \notin \Delta$ **then begin**
  { *all tokens in* $B(j^*)$ *may be assigned to* p^* }
    $M(p^*) := M(p^*) + B(j^*)$;
    $B(j^*) := 0$;
    $P_{old} := P_{old} \cup P_{j^*}$;
    end
  else begin
  { *in this case* $p^* \in P_1 \cap P_{j^*}$ }
    $p^1 : c(p^1) = \max\{c(p) \mid p \in P_1 \setminus (P_{j^*} \cup P_{old})\}$;
    $p^{j^*} : c(p^{j^*}) = \max\{c(p) \mid p \in P_{j^*} \setminus (P_1 \cup P_{old})\}$;
    if $c(p^*) \geq c(p^1) + c(p^{j^*})$ **then begin**
    { *assign to* p^* *as many tokens as possible* }
      $m := \min\{B(1), B(j^*)\}$;
      $M(p^*) := M(p^*) + m$;
      $\beta(1) := \beta(1) - m$; $B(j^*) := B(j^*) - m$;
      if $B(1) = m$ **then** $P_{old} := P_{old} \cup P_1$
        else $P_{old} := P_{old} \cup P_{j^*}$;
      end
    else begin
    { *assign to* p^{j^*} *as many tokens as possible* }
      $M(p^{j^*}) := M(p^{j^*}) + B(j^*)$; $B(j^*) := 0$;
      $P_{old} := P_{old} \cup P_{j^*}$;
    end;
  end;
$M^* = M$;
end.

□

This algorithm contains, as a particular case, the algorithm given in [4]. The complexity of the algrithm is linear in the number of partitions, since the *while* loop is repeated at most r times.

4. Net structure of observers

Given a net $N = (P, T, Pre, Post)$ and an initial macromarking $(V, \vec{b})$, (that induces a partition $P = P_0 \cup P_1 \cup \cdots \cup P_r$) the corresponding observer net can be constructed as discussed in [3].

The observer net $N_{ob} = (P, T, Pre, Post, \mathcal{B}, L)$ will have the same net structure of N with in addition:
a) a set of r bounding places $\mathcal{B}$: each bounding place $\beta_{ob,j}$ in $\mathcal{B}$ is associated to a subset P_j $(j > 0)$ of the partition induced by the initial macromarking;
b) a set of *links* $L = \bigcup_{j=1}^{r}(P_j \times \{\beta_{ob,j}\})$: a link $(p, \beta_{ob,j})$ connects each place $p \in P_j$ to the corresponding bounding place $\beta_{ob,j}$.

The initial marking of the set of places P is μ_{w_0}, while the initial marking of the set of bounding places $\mathcal{B}$ is B_{w_0} (as computed by Algorithm 3).

Let the observer be in a marking $[\mu_w^T \ / \ B_w]^T$. As the plant evolves firing a transition t the corresponding transition is fired on the observer and the new observer marking $[\mu_{wt}^T \ / \ B_{wt}^T]^T$ is computed as described in Algorithm 3. That is:

- max $\{\mu_w(p), Pre(\cdot,t)(p)\}$ tokens are taken from each place $p \in P$.
- If $\mu_w(p) < Pre(\cdot,t)(p)$ then $Pre(\cdot,t)(p) - \mu_w(p)$ tokens are taken from each bounding place linked to p.
- $Post(\cdot,t)(p)$ tokens are added to each place $p \in P$ to obtain $[\mu_{wt}^T \ / \ B_{wt}^T]^T$.

5. Control using observers

In this section, we show how the marking estimate computed by an observer can be used by a control agent to enforce a given specification on the plant behaviour. In general, a specification is given as a set of *forbidden markings* $\mathcal{F}$ that should never be reached by the system under control, or, equivalently, by set of *legal markings* $\mathcal{L} = \mathbb{N}^{|P|} - \mathcal{F}$.

We make several assumptions that are briefly discussed here.

- The set of legal markings is given in the format of linear constraints, i.e., $\mathcal{L} = \{M \in \mathbb{N}^{|P|} \mid S^T \cdot M \leq \vec{k}\}$ where $S = [\vec{s}_1 \cdots \vec{s}_u]$ with $\vec{s}_j \in \mathbb{Z}^{|P|}$ and $\vec{k} = [k_1 \cdots k_u]^T$ with $k_j \in \mathbb{Z}$. This kind of specifications, that we call *generalized mutual exclusion constraints*, have been considered by various authors [5, 10, 14].
- The controller may disable transitions to prevent the plant from entering a forbidden marking. From the knowledge of μ_w and B_w, the controller computes a control pattern $\gamma : T \to \{0, 1\}$. If $\gamma(t) = 0$ then t is disabled by the controller.
- All transitions are controllable, i.e., can be disabled by the controller.

The considered control scheme is shown in Figure 1.d.

Assume that the initial marking M_0 of the plant does not necessarily belong to $\mathcal{L}$ (this is a natural assumption when considering error recovery problems). Then, given a marking M we may want to prevent the firing of transition t such that $M[t\rangle M'$ when both these two conditions are verified:

1. There exists $\vec{s}_j$ with $\vec{s}_j \cdot M' > k_j$, i.e., $M' \in \mathcal{F}$;
2. $\vec{s}_j \cdot M' > \vec{s}_j \cdot M$, i.e., the firing of t either leads to a violation of the constraint (if $M \in \mathcal{L}$) or to a "worse" violation of the constraint (if $M \in \mathcal{F}$).

In this case the following algorithm taken from [4] may be used to compute the control pattern γ at each step.

Algorithm 6. Let w be the observed word and $\mathcal{M}(w \mid V, \vec{b})$ the set of w consistent markings. Let the specification be $\mathcal{L} = \{M \in \mathbb{N}^{|P|} \mid S^T \cdot M \leq \vec{k}\}$ with $S = [\vec{s}_1 \cdots \vec{s}_u]$ and $\vec{k} = [k_1 \cdots k_u]^T$.

for all $t \in T$
    begin
    $\gamma(t) := 1$;
    $j := 1$;
    while $j \leq u$ **and** $\gamma(t) = 1$ **do**
        begin
        $\Delta := \vec{s}_j^T \cdot C(\cdot,t)$;
        if $\Delta > 0$ **then**
            begin
            $\overline{m} := \max\{\vec{s}_j^T \cdot M \mid M \in \mathcal{M}(wt \mid V, \vec{b})\}$;
            if $\overline{m} > k_j$ **then** $\gamma(t) := 0$;
            end;
        $j := j + 1$;
        end;
    end.

■

Thus a transition is disabled at M only if its firing leads to a marking M' such that for at least one constraint j: $\vec{s}_j^T \cdot M' > \vec{s}_j^T \cdot M$ (i.e., $\Delta > 0$) and there exists a consistent marking M'' in $\mathcal{M}(wt \mid V, \vec{b})$ that violates the constraint (i.e., $\vec{s}_j^T \cdot M'' > k_j$).

Note that to compute $\overline{m}$ we need to solve an integer problem as discussed in Section 2. If assumption A4 holds, however, the result of Proposition 5 may be used to efficiently compute $\overline{m}$.

6. A manufacturing example

In this section we consider a manufacturing example composed of two machines, $m1$ and $m2$ connected through a buffer of limited capacity. The net associated to each machine is composed of two places: idle ($p1$ for $m1$, $p3$ for $m2$) and working ($p2$ for $m1$, $p4$ for $m2$).

The parts within the flowshop are processed first by $m1$ and then by $m2$. The net associated to the routing of the parts is composed of three places: $p5$ (parts to be processed), $p6$ (parts in the buffer between the machines), $p7$ (parts processed and ready to be removed). There are 3 pallets in the system, and the firing of transition t_5 corresponds to the removal of a processed part from a pallet and the positioning of a new part to be processed on this pallet.

The net corresponding to this system is shown in Figure 2.a while Figure 2.b shows the observer net for this plant.

Since each machine can be either idle or working and there are 3 pallets in the system, our knowledge of the

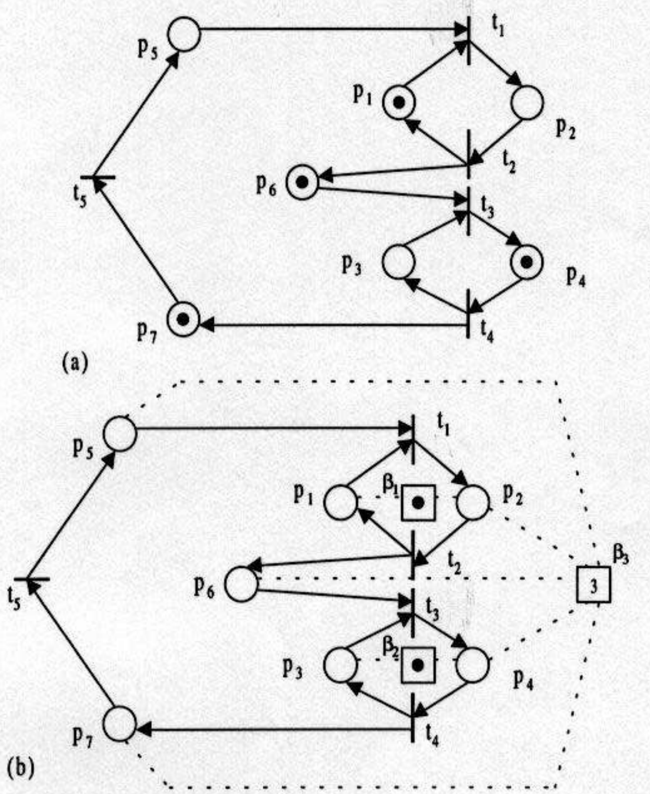

Figure 2: Plant (a) and observer net (b) for a manufacturing example.

initial marking can be expressed by the macromarking: $M(p_1)+M(p_2) = 1$; $M(p_3)+M(p_4) = 1$; $M(p_2)+M(p_4)+M(p_5) + M(p_6) + M(p_7) = 3$.

The controller must enforce three specifications. A first specification requires that at most a machine may be working at any time: this can be represented by the constraint $M(p_2) + M(P_4) \leq 1$. A second specification requires that at most 2 parts may be simultaneously in the buffer: this can be represented by the constraint $M(p_6) \leq 2$. A third specification requires that $M(p_6) + M(p_2) \leq 2$. The last constraint has been introduced to avoid deadlock. In fact, it can be violated only when 2 parts are in the buffer and m_1 is working. The only transitions enabled in this case are t_2 and t_3. Because of the first two constraints these two transitions are disabled by the controller and the closed loop net deadlocks.

In the following we discuss a test case in detail. We assume that the marking of the plant is not measurable, therefore an observer must be used in the control loop. The resulting closed loop behaviour is represented in the reachability graph in Figure 3. Each box is labeled with the marking M_w of the plant (shown between round brackets), and the marking estimate μ_w and bound B_w (shown between square brackets). The initial marking is represented by a round box. A thick box represents a marking reached by a complete word w, i.e., the value of B_w is zero; in this case the observer has perfectly reconstructed the plant state and all future evolutions are the same as if the plant marking can be exactly measured. the future evolution from such a marking is not shown. The thin dashed arrows represent the transitions disabled by the controller. All these transitions would have been disabled even if the actual marking had been known. The thick dashed arrows represent the transitions disabled by the observer, i.e., those transitions that would have been enabled if the actual marking had been known.

From the reachability graph in Figure 3 it can be seen that the actual marking can be reconstructed after quite a small number of transitions.

Furthermore only in few cases transitions are disabled by the observer.

A second test case can be performed assuming $[1\ 0\ 1\ 0\ 1\ 1\ 1]^T$ as initial marking. In this case the use of the observer in the control loop results in a blocking closed loop behaviour. In fact, neither t_1 nor t_2 may fire (even if enabled by the controller) since the observer has not yet reconstructed the state.

This blocking caused by the observer may be avoided introducing suitable time-out mechanisms as discussed in [3].

7. Conclusions

In this paper we have discussed the problem of controlling a Petri net whose marking cannot be measured.

Assuming that only the net structure of the plant is known while the initial marking is known to belong to a macromarking, it is possible to estimate the actual marking of the net based on the observation of a word of events.

In this paper we have extended the results of previous works considering a more general structure of the initial macromarking: we have assumed that a single subset, whose intersection with all the others is different from the empty set, may exist.

The estimate generated by the observer may be used to design a state feedback controller, that ensures that the controlled system never enters a set of forbidden states. We have considered a special class of specifications that limit the weighted sum of markings in subsets of places.

As it has been underlined in the manufacturing example, the use of marking estimates (as opposed to the exact knowledge of the actual marking of the plant) may cause the controlled plant to block. However, such a problem can be overcome by means of suitable time-outs.

References

[1] P.E. Caines, R. Greiner, S. Wang, "Dynamical Logic Observers for Finite Automata," *Proc. 27th Conf. on Decision and Control* (Austin, Texas), pp. 226–233, December, 1988.

[2] P.E. Caines, S. Wang, "Classical and Logic Based Regulator Design and its Complexity for Partially Observed Automata," *Proc. 28th Conf. on Decision and Control* (Tampa, Florida), pp. 132–137, December, 1989.

[3] A. Fanni, A. Giua, N. Sanna, "Control and Error Recovery of Petri Net Models with Event Observers"

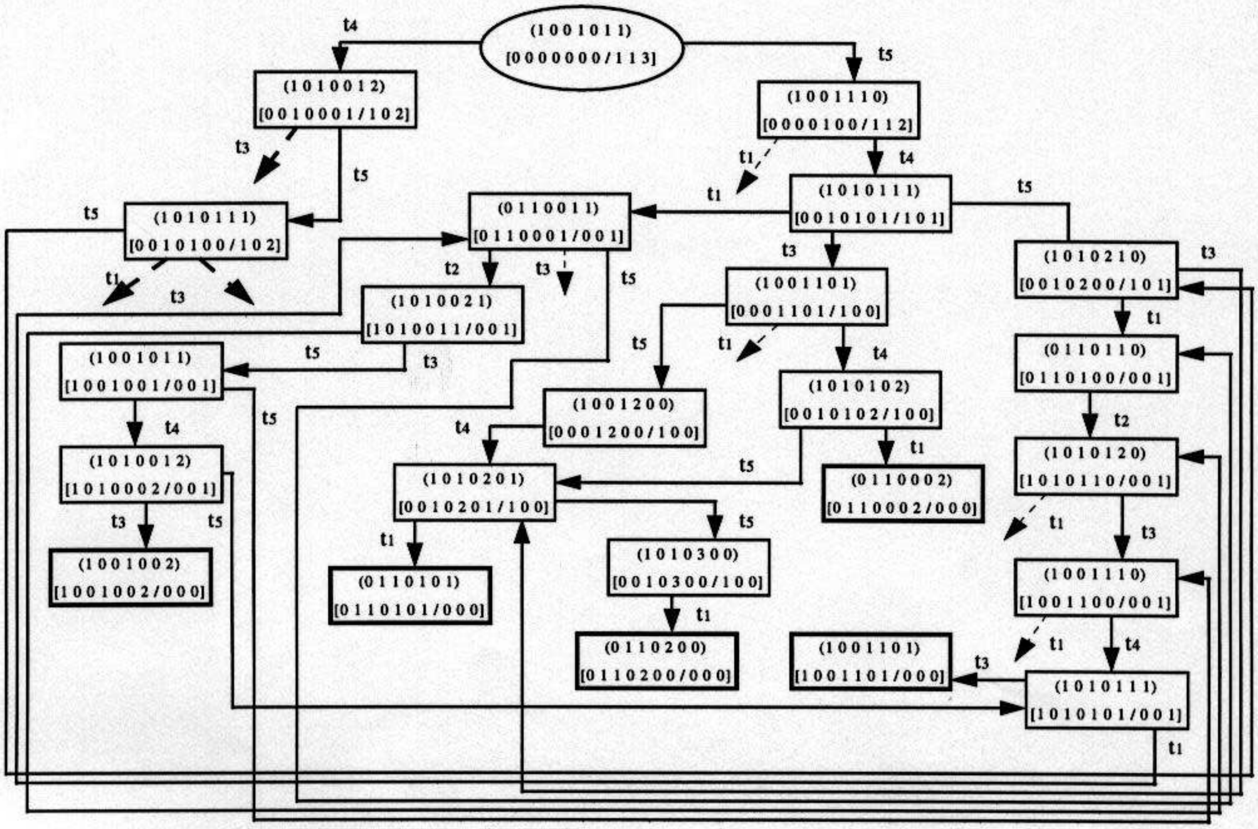

Figure 3: Reachability graph of the net/observer in Figure 2 from initial marking $[1\ 0\ 0\ 1\ 0\ 1\ 1]^T$.

Proc. 1st Int. Work. on Manufacturing and Petri Nets. (Toulouse, France), pp. 53–68, June, 1997.

[4] A. Giua, "Petri Net State Estimators Based on Event Observations," *Proc. 36th Conf. on Decision and Control* (San Diego, California), December, 1997.

[5] A. Giua, F. DiCesare. M. Silva, "Generalized Mutual Exclusion Constraints on Nets with Uncontrollable Transitions," *Proc. 1992 IEEE Int. Conf. on Systems, Man, and Cybernetics* (Chicago, Illinois), pp. 974–979, October, 1992.

[6] L.E. Holloway, B.H. Krogh, A. Giua, "A Survey of Petri Net Methods for Controlled Discrete Event Systems," *J. of Discrete Event Dynamic Systems*, Vol. 7, No. 2, pp. 151–190, April, 1997. .

[7] R. Kumar, V. Garg, S.I. Markus, "Predicates and Predicate Transformers for Supervisory Control of Discrete Event Dynamical Systems," *IEEE Trans. on Automatic Control*, Vol. 38, No. 2, pp. 232–247, February, 1993.

[8] Y. Li, W.M. Wonham, "Controllability and Observability in the State-Feedback Control of Discrete-Event Systems," *Proc. 27th Conf. on Decision and Control* (Austin, Texas), pp. 203–207, December, 1988.

[9] Y. Li, W.M. Wonham, "Control of Vector Discrete-Event Systems — Part I: The Base Model," *IEEE Trans. on Automatic Control*, Vol. 38, No. 8, pp. 1215–1227, August, 1993.

[10] Y. Li, W.M. Wonham, "Control of Vector Discrete-Event Systems — Part II: Controller Synthesis," *IEEE Trans. on Automatic Control*, Vol. 39, No. 3, pp. 512–531, March, 1994.

[11] T. Murata, "Petri Nets: Properties, Analysis and Applications," *Proceedings IEEE*, Vol. PROC-77, No. 4, pp. 541–580, April, 1989.

[12] P.J. Ramadge, W.M. Wonham, "The Control of Discrete Event Systems," *Proceedings IEEE*, Vol. 77, No. 1, pp. 81–98, January, 1989.

[13] S. Takai, T. Ushio, S. Kodama, "Static-State Feedback Control of Discrete-Event Systems Under Partial Observation," *IEEE Trans. Automatic Control*, Vol. 40. No. 11, pp. 1950–1955, November, 1995.

[14] K. Yamalidou, J.O. Moody, M.D. Lemmon, P.J. Antsaklis, "Feedback Control of Petri Nets Based on Place Invariants," *Automatica*, Vol. 32. No. 1, 1996.

[15] L. Zhang, L.E. Holloway, "Forbidden State Avoidance in Controlled Petri Nets Under Partial Observation," *Proc. 33rd Allerton Conf.* (Monticello, Illinois), pp. 146–155, Oct. 1995.