# Control and Error Recovery of Petri Net Models with Event Observers *

Alessandra Fanni, Alessandro Giua, Nicola Sanna

Dip. di Ing. Elettrica ed Elettronica, Università di Cagliari, Piazza d'Armi, 09123 Cagliari, Italy

email:giua@diee.unica.it, tel: +39 (70) 675-5892, fax: +39 (70) 675-5900

March 1, 1997

## 1 Introduction

This paper discusses the problem of controlling a Petri net whose marking cannot be measured. Observers are used to estimate the actual marking of the plant based on event observations. A software tool for simulation and analysis of controlled plant with observers is described.

When the structure and the inital marking of a net is known, the knowledge of the transition firings is sufficient to reconstruct the marking that each new firing yields. In this work we assume that only the net structure is known and consider the cases in which the initial marking is know to belong to a "macromarking", i.e., we know the token contents of subsets of places but not the exact token distribution.

In [4] it was shown how it is possible to estimate the actual marking of the net based on the oservation of a word of events (i.e., transition firings) and an algorithm was given for computing the estimate, given in terms of estimate marking $\hat{M}_w$ and bound $B_w$. The estimate is always a lower bound of the actual marking.

The system that computes the estimate is called an observer. In this paper we will show

*Extended abstract submitted to the 2nd Int. Work. on Manufacturing and Petri Nets. *Contact author is: Alessandro Giua.*

that a special net structure, called *observer net*, can be used to describe the observer: such a net has a set of $m$ places (corresponding to the places of the plant) whose marking is at each step $\hat{M}_w$ plus $r$ places (one for each bound in the 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 be easily 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 will also descrive a software tool, that is an extention of [3], that we have developed using Smalltalk (a well known object-oriented language). The tool can be used for the simulation and the construction of the reachability tree of a Petri net controlled in a closed loop. The tool can make use of the actual marking of net (if it can be measured) or of the estimate computed by an observer (if the actual marking cannot be directly measured).

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.

Control and state estimation under partial observation has been discussed in the discrete event control literature [6, 7, 8, 1, 2, 5].

## 2 Observers with macromarking

In [4] an algorithm was given for estimating the marking of a net system $\langle N, M_0 \rangle$ whose marking cannot be directly observed 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 $r+1$* disjoint *subsets: $P = P_0 \cup P_1 \cup \cdots \cup P_r$. 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 $M_0$ belongs to the macromarking $(V, \vec{b})$, i.e., it satisfies 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*

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

Given an evolution of the net $M_0[t_1\rangle M_1[t_2\rangle \cdots$, we use the following algorithm to compute the estimate $\hat{M}_{w_i}$ of each actual marking $w_i(M_0)$ based on the observation of the word of events $t_1, t_2, \cdots, t_i$, and of the knowledge of the initial macromarking $(V, \vec{b})$.

**Algorithm 3.** *Marking Estimation with Event Observation and Initial Macromarking (MEEOIM)*

1. Let the initial estimate be $\hat{M}_{w_0}$, with $\hat{M}_{w_i}(p) = \min M(p)$ such that $V^T \cdot M = \vec{b}$.

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

3. Let $i = 1$.

4. Wait until $t_i$ fires.

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

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

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

8. Let $i = i + 1$.

9. Goto 4.

A note about the computation of the initial estimate. Let $p \in P_j$. If $j = 0$ or if $P_j$ is not a singleton set, then clearly $\hat{M}_{w_0}(p) = 0$. We consider, however, in all generality the case that $P_j = \{p\}$ (with $j > 0$); in this case the macromarking implies that $\hat{M}_{w_0}(p) = M_0(p) = \vec{b}_j$.

In [4] it was shown that the set of consistent markings can be characterized in terms of a set of linear inequalities as a function of $\hat{M}_w$ and $B_w$.

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

$$\mathcal{M}(w \mid V, \vec{b}) = \{M \in \mathbb{N}^P \mid V^T \cdot M = V^T \cdot \hat{M}_w + B_w, M \geq \hat{M}_w\}.$$

Intuitively, any marking $M$ in $\mathcal{M}(w \mid V, \vec{b})$ can be constructed as follows: (a) Let $M' = \hat{M}_w$. (b) Assign arbitrarily $B_w(j)$ tokens to the set of places $P_j$ constructing the marking $M''$. (c) Add an arbitrary number of tokens to all places in $P_0$ to obtain $M$.

# 3 Net structure of observers

Given a net $N = (P, T, Pre, Post)$ and an initial macromarking $(V, \vec{k})$, the corresponding observer net can be constructed as follows.

The observer net will have:

- A set of place $P_{ob} = P \cup \mathcal{B}$. Each bounding place $b_j$ in $\mathcal{B}$ is associated to a constraint $\vec{v}_j$ of the initial macromamarking.
- A set of transitions $T_{ob} = T$ with pre and post arc as in the original net.
- A *link* from each place $p \in P_j$ to the corresponding bounding place $b_j$.

The initial marking of the set of places $P$ is $\hat{M}_{w_0}$, while the initial marking of the set of places $\mathcal{B}$ is $B_{w_0}$.

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

- max $\{\hat{M}_w(p), Pre(\cdot, t)(p)\}$ tokens are taken from each place $p \in P$ and if $\hat{M}_w(p) < Pre(\cdot, t)(p)$ then the extra $Pre(\cdot, t)(p) - \hat{M}_w(p)$ tokens are taken from the bounding place $b$ connected to $p$ (if it exists).
- $Post(\cdot, t)(p)$ tokens are added to each place $p \in P$ to obtain $[\hat{M}_{wt} B_{wt}]$.

*Example 5.* Consider the net system in Figure 1. It represents a pool of three machines. Each token represents a machine that may be in any of three states: working (token in place $p_1$), idle (token in place $p_2$), loading (token in place $p_3$).

The initial macromarking $M(p_1) + M(p_2) + M(p_3) = 3$ captures our knowledge that there are three machines in the pool. Their initial state is, however, unknown. Here $P_1 = P$.

The bound place $b_1$ is shown as a thick circle. The links from $b_1$ to the places in $P_1$ are shown as thick dotted arcs. The pre arcs of the transitions are shown as thick arrows, to denote that they do not impose preconditions on the firing of the transitions.

# 4 Control using observers

In the final paper we will 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. A manufacturing example will be fully discussed.

We make several assumptions that are briefly discussed here.

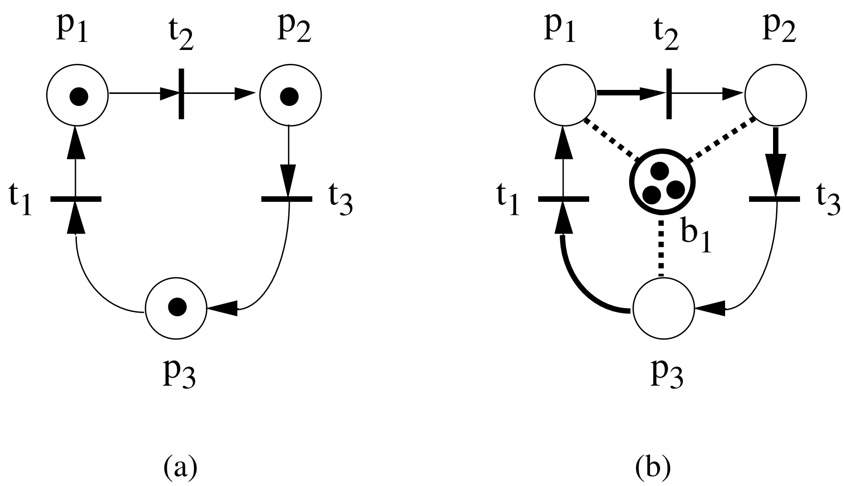


Figure 1: Net system (a) and corresponding observer (b) in Example 5.

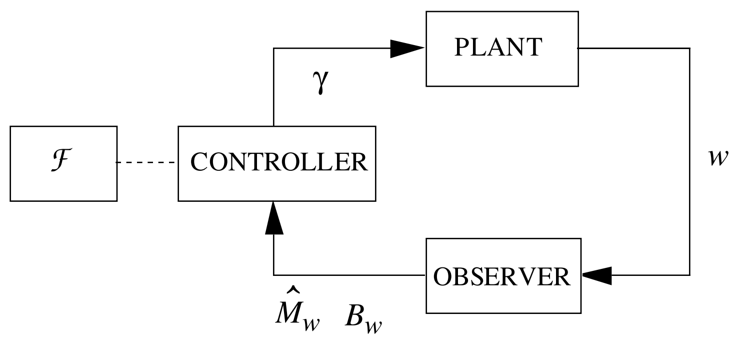


Figure 2: State feedback control loop with observer.

- The specification is given as a set of forbidden markings $\mathcal{F}$. The set of legal marking is $\mathcal{L} = \mathbb{N}^P - \mathcal{F}$.
- The controller may disable transitions to prevent the plant from entering a forbidden marking. From the knowledge of $\hat{M}_w$ 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 2.

Under the assumption that the inital marking $M_0 \in \mathcal{L}$, the following algorithm may be used by the controller at each step to ensure that markings in $\mathcal{F}$ are not reached.

**Algorithm 6.** *Let $w$ be the observed wors, and $\mathcal{M}(w \mid V, \vec{b}) = \{M \in \mathbb{N}^P \mid V^T \cdot M = V^T \cdot \hat{M}_w + B_w, M \geq \hat{M}_w\}$, where $\hat{M}_w$ and $B_w$ are computed by the observer.*

**for** *all* $t \in T$

**begin**

$\gamma(t) := 1$;

**if** $\exists M \in \mathcal{M}(w \mid V, \vec{b}) \cap \mathcal{L}$ *such that* $M[t > M'$, $M' \in \mathcal{F}$

**then** $\gamma(t) := 0$;

**end**.

Clearly this algorithm prevents all transition firings that lead from $\mathcal{L}$ to $\mathcal{F}$. The condition under which this controller is optimal (i.e., it prevents *only* transition firings that lead from $\mathcal{L}$ to $\mathcal{F}$) has been discussed in [6, 7, 8].

# 5 Simulation tool

We have developed a software tool for simulation and analysis, called SmallPet, implemented in Smalltalk. The tool is based on the software described in [3] with several extensions.

- A new class `ObserverNet` is used to represent an observer net. This required creating the new classes `BoundingPlace` and `Link`.
- A new class `Controller` is used to represent a controller. Different objects of this class are used to implement different control policies. It is possible to use observer-based controllers (when the marking of the plant is not measurable) or full-feedback controllers (when the actual marking of the plant is known).
- The `Simulator` can perform simulation of open-loop or controlled plants, with or without observers. The reachability graph of the plant-observer system under control can also be automatically constructed.

# 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, M. Marchesi, N. Sanna, "A Petri Net Simulator Based on Object-Oriented Programming," *Proc. Annual Conf. Operational Research Society of Italy AIRO* (Perugia, Italy), pp. 557–560, September, 1996.

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

[5] 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.

[6] 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.

[7] 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.

[8] 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.