Optimizing the Controller of Production Flows

CALIN CIUFUDEAN, CONSTANTIN FILOTE
Electrical Engineering and Computer Science Department
University of Suceava
Universitatii 13, RO-720229, Suceava
ROMANIA
{calin / filote}@ eed.usv.ro

Abstract: In this paper, we consider the discrete event systems (DES) modeled via a state – space representation. The control objective is the avoidance of a given set of states. We achieve this goal by verifying that certain predicates, specified in terms of states, are always false. We model the DES involved in physical systems with a class of controlled state machines, respectively the formalism of the controlled Petri nets. A model of a railway shunting system illustrates the theoretical approach.

Key-Words: Controlled Petri nets, discrete event systems, controller, production flow, railway shunting systems.

1 Introduction

In this paper, we consider discrete event systems (DES) modeled via a state space representation. The model objective is the avoidance of a given set of states, or equivalently the fact that certain predicates, specified in terms of states are always false. This problem was addressed in [1], [2] for DES modeled as cyclic marked graphs, and in [3] for DES, which can be modeled as controlled state machines, another class of controlled Petri nets. We address the state space controlled Petri nets and a technique to reduce the complexity of these nets, by taking into account the fact that the complexity of the considered nets depends mainly on the representation of the control design, respectively on the forbidden sets of places. Our goal is to describe in an efficient way the forbidden sets of the modeled system, e.g. to describe in an efficient way the DES modeled with controlled Petri nets. An example from the railway systems illustrates this approach: the Petri net model of railway station incompatible tracts. We mention that every railway station has such a table, given to the general traffic rules allocated to a certain station.

Petri nets graphically describe the system behavior in terms of the successive states realized after the occurrence of events. Events correspond to a change from one state to the next state. The number of tokens in each place represents the state of the Petri net. When a transition fires, tokens are moved from one place to another place. Firing transitions correspond to an event. In a controlled Petri net, the transitions fire only when all their input control places are marked. The control objective is to prevent a set of forbidden states from being reached, while at the same time enabling a maximal set of achievable state sequences. The Petri nets property to allow a graphical and analytical representation, closely related to the technological process being modeled, makes them a very useful tool for enabling the application of the theory.

2 Controlled Petri Nets

2.1 Basic properties

A controlled Petri net is defined [1] as a five-tuple $G=(P,T,F,C,B)$, where $P$ and $T$ are finite sets of state places and transitions, respectively; $F \subset (P \times T) \cup (T \times P)$ is a set of directed arcs connecting state places and transitions; $C$ is a finite set of control places; $B \subset (C \times T)$ is a set of directed arcs associating control places with transitions. A state place $p \in P$ is said to be an input to (respectively output from) transition $t \in T$ if $(p, t) \in F$ (respectively $(t, p) \in F$). Denote by $(p)_t$, respectively $t(p)$ the sets of these input, respectively output places. The sets $(t)_p, p(t), c(t),$ and $(c)_t$ are defined similarly.

A transition, in a controlled Petri net, is enabled if it is both state and control enabled, that means that under marking $m$ must have: $m(p) \geq 1, \forall p \in (p)_t$, if we assume that there is at most one arc in $B$ and $F$ between any place and any transition. When an enabled transition $t_i$ fires, a token can be removed from each state place, which has an arc to $t_i$. Then, the token will be added to any place, which has an arc from $t_i$. We assume that several transitions can fire simultaneously provided they are always enabled, and the new state is described by:
\( m'(p) = m(p) - \sum_{t \in T} I(p' \in (p,t)) + \sum_{t \in T} I(p \in i(p)) \)  

where \( v, w, k_i \) are positive integers.

One may say [4] that a marking \( m \) can be reached uncontrollably from an initial marking \( m_0 \) if, even if \( u_{zero} \) is used and thus all controllable transitions are disabled, the initial marking \( m_0 \) can be transformed via enabled transitions into the final marking \( m \). An influence path \( \text{inf}_p \) of a transition \( t \) is a sequence of places \( \text{inf}_p = (p_1, p_2, \ldots, p_n) \subset P \) such that [5]: \( t^{(0)} = (p_1) \); for \( j = 2,3, \ldots, n \), there exists an uncontrollable transition \( t_j \in T \) leading from \( p_{j-1} \) to \( p_j \); all transition following \( p_n \) are either controllable or lead to a place \( p_j \) already encountered earlier in the influence path \( \text{inf}_p \).

Once a token has entered in an influencing path \( \text{inf}_p \), it can uncontrollably reach \( p \). The complexity of the control synthesis algorithms depends strongly on the number of places in the influencing zones for the different simple forbidden sets. It is thus useful to try to reduce the size of the network, before starting the control design. We can replace several parallel transitions between a pair of places, by one single transition. This transition should be uncontrollable since one of the parallel transitions is uncontrollable. If in a controlled Petri net, in certain condition, exist influence paths, where a sequence of uncontrollable transitions moves the tokens from any upstream place in \( \text{inf}_p \) to \( p_1 \). This implies that places of an uncontrollable path are indistinguishable from a control viewpoint. Thus, from a control point of view, it is allowed to replace all the places in \( \text{inf}_p \), and the transitions connecting these places, by one single place. The reduced net is characterized by a new finite marking set \( M' \). We call this equivalence between functional properties the functional abstraction equivalence (f.a.e.) [4, 5]. An example will facilitate the understanding of this concept.

### 3 Illustrative Example

Considering that the shunting operation is a key activity for increasing the operativity of the railway transport systems, we choose this example: a railway system composed from a railway station, a shunting yard, a shunting board, and two railway engines. We consider the following shunting process: a railway machine takes a wagon convoy \( C \) from an industrial railway and binds it to another convoy \( R \) brought from another industrial railway. This assembly is formed in a shunting controlled railway, together with another convoy \( H \) brought from the kiking horse pass of the shunting yard. This new convoy is triggered (e.g., by the railway engine \( M_2 \)) at a station railway, in order to be sent. The net representing this assembly operation is given in Fig.1. Each task of the plan \( P_1 \) (transitions \( t_p \) and \( t_{q} \) are given in Fig.2, and respectively in Fig.3.)
The PN model control plan

Fig. 1 The representation of a shunting operation

Fig. 2 Refining transition \( t_p \)

Fig. 3 The GoPN for the shunting system
The indices in Fig.2 and in Fig.3, are:

\( C_{OK} (R_{OK}, H_{OK}) = \) the convoy \( C \) (R or H) is ready to be tugged by a railway engine;

\( M_{1OK} (M_{2OK}) = \) the railway engine \( M_1 \) (\( M_2 \)) is available;

\( M_1 C \) (\( M_1 R \), \( M_1 H \)) = the railway engine \( M_1 \) (\( M_2 \)) is attached to the convoy \( C \) (R or H);

\( M_1 M_2 CR \) (\( M_1 M_2 CRH \)) = the railway engine \( M_i \) (\( M_2 \)) is attached to convoy \( CR \) (CRH);

\( CRH_{OK} = \) the convoy CRH is ready to go;

\( M_{1trC} \) (\( M_{1trR} \), \( M_{1trH} \)) = the engine \( M_1 \) (\( M_2 \)) tugs the convoy \( C \), R, or H;

\( CR_{bind} \) (\( CRH_{bind} \)) = the convoys \( C \) and \( R \) are connected (analogous for convoys \( C \), \( R \) and \( H \)).

\( M_{1elCR} \) (\( M_{1elCRH} \)) = the engine \( M_1 \) liberates the convoy \( CR \) (or CRH).

\( exp. = \) the engine \( M_2 \) sends the convoy CRH.

In Fig.2 the refinement of the places corresponding to transition \( t_p \) of the control plan follows these expressions:

\[
m = \{(C, C_{OK}), (R, R_{OK}), (CR, M_{1-M_2 CR})\}
\]

\[
P_{IC} = m(I(t_p)) \cup P_{IC} = P_{IA} \cup P_{IC} = \{C_{OK}, R_{OK}, M_{1OK}, M_{2OK}\}
\]

\[
P_{OC} = m(O(t_p)) \cup P_{OC} = P_{OA} \cup P_{OC} = \{M_{1-M_2 CR}\}
\]

Where \( m \) is an injective function from places in assembly plan and places in control plan, and \( P \) are the corresponding locations for transitions \( t_p \). Analogous, we define the refinement of the places corresponding transition \( t_q \) of the control plan:

\[
m = \{(H, H_{OK}), (CR, M_{1-M_2 CR}), (CRH, CRH_{out})\}
\]

\[
P_{IC}'' = m(I(t_q)) \cup P_{IC} = P_{IA} \cup P_{IC} = \{H_{OK}, M_{1-M_2 CR}\}
\]

\[
P_{OC}'' = m(O(t_q)) \cup P_{OC} = P_{OA} \cup P_{OC} = \{CRH_{out}, M_{1OK}, M_{2OK}\}
\]

The merging operations of the control plans in Fig.3, and Fig.2 results in the assembled control plan shown in Fig.3, where the significance of the indices are the same as in Fig.3, and Fig.3.

### 4 Conclusion

This paper has resumed our model for the forbidden state problem of DES modeled by controlled Petri nets. Our goal is to present, by an adequate control, a given set of forbidden states from being reached. Forbidden markings characterize constraint sets expressing that some places of the Petri net cannot contain more than a certain amount of tokens. In order to do this, a theory that synthesizes Petri nets for modeling controllers of production flows with shared resources has been reviewed [6 -8]. Such research increases the integrability of models with different behaviors. A PN for assembly hierarchy of shunting operations in railway systems was chosen for exemplifying the given method for modeling production flows. We shown that system’ properties in the assembly plan are maintained in the control plan, by simply following the given algorithm for reversibility controller existence in a PN.

This concept allows us to obtain a maximally permissive control law. An example, dealing with a railway system, respectively the railway station incompatible tract of the Petri net model illustrates the theoretical concepts. Future study will extend the results to wider class of timed controlled Petri nets.

### References:


