ANALYSIS OF DISCRETE SYSTEMS WITH UNCERTAINTY BY GENERALIZED PETRI NET

Vladimír OLEJ, Josef STRELEC, Department of Computer Science, Technical University, 031 19 Liptovský Mikuláš, Czechoslovakia

Abstract

The paper deals with generalized Petri net as flexible formal means for analysis of discrete systems. On the basis of generalized Petri net we are able to define any deterministic and non-deterministic (discrete stochastic and fuzzy) Petri nets. The research in this article is the continuation of that in[1].

1. Introduction

Deterministic Petri nets [2,3,4,5] and non-deterministic Petri nets [6,7,8,9] are formal graph means for property description and analysis of discrete systems. They can be generalized and represented by one Petri net.

2. Generalized Petri Net

Generalized Petri net (GPN) is a bichromatic oriented graph defined as a 5-tuple [10,11,12]:

$$GPN = \langle P, T, X, Y, Z \rangle \qquad (1)$$

where :- P is a finite set of vertices called places, represented by circles , $P = \{p_1, p_2, \dots, p_k\}$,

- T is a finite set of vertices called transitions, represented by lines, $T = \{t_1, t_2, \dots, t_r\}$, $P \cap T = 0$,
- X is an ordered 4-tuple that defines the qualities of k places P,
- Y is an ordered 5-tuple that defines the qualities of r transitions T,
- Z is an ordered 3-tuple that defines the qualities of edges which are given by forward and backward inci-

dence function [2,3,4,5].

An ordered 4-tuple that defines the qualities of k places P can be defined as follows:

$$X = < C, IC, M_0C, UP > \qquad (2)$$

where:- C is a finite set of the used colors (types of tokens), $C = \{c_1, c_2, \ldots, c_h\}$,

- IC: PxT-->NxC, (N= 0,1, ...), $IC((n,c)_m, i, j)$, where $m \in <1,h>$, $i \in <1,k>$, $j \in <1,r>$, is the forward incidence function. It states m ordered 2-tuple (n,c), $n \in N$, $c \in C$, for each edge from place $p_i \in P$ to transition $t_j \in T$. Through the edge from place $p_i \in P$ to transition $t_j \in T$ n_m tokens of color c_m, can pass,
- M_0C: PxN-->C is an initial marking, $M_0C((n,c)_m, i)$. It states for each place $p_i \in P$, m ordered 2-tuples (n,c). It represents how many tokens of the given color occurs in place $p_i \in P$,
- $UP = \{up_1, up_2, \ldots, up_k\}$ is a finite set of qualities of tokens in the places $p_i \in P$, which can be deterministic, stochastic or fuzzy.

An ordered 5-tuple that defines the qualities of r transitions T can be defined as follows:

$$Y = < QC, \tau, PR, D, UT > \qquad (3)$$

where:- QC: TxP-->NxC, $QC((n,c)_m, i, j)$, is a backward incidence function. It states m ordered 2-tuple (n,c) for each edge from transition $t_j \in T$ to place $p_i \in P$. Through the edge from transition $t_j \in T$ to place $p_i \in P$ n_m tokens of color c_m, can pass,

- $\tau = \{\tau_1, \tau_2, \ldots, \tau_r\}$ is a finite set of times of firings of r transitions T,
- $PR = \{pr_1, pr_2, \ldots, pr_q\}$ is a finite set of predicates, where for each $i \in <1,q>$, then $pr_i \in \{TRUE, FALSE\}$ holds true. Each predicate $pr_i \in PR$ can be connected with arbitrary transition $t_j \in T$ by normal or inhibit connection. This connection is given by incidence function D,
- D: TxPR -->{1,-1,0} is an incidence function. Its mea-

ning is as follows:

a) $D(t_j, pr_i) = 1$, when there exists connection between transition $t_j \in T$ and predicate pr_i (transition $t_j \in T$ is enabled only if the value of predicate $pr_i \in PR$ is equal TRUE),

b) $D(t_j, pr_i) = -1$, when there exists inhibit connection between transition $t_j \in T$ and predicate $pr_i \in PR$ (transition $t_j \in T$ is enabled only if the value of predicate $pr_i \in PR$ is equal FALSE),

c) $D(t_j, pr_i) = 0$, when no connection exists between transition $t_j \in T$ and predicate $pr_i \in PR$ (firing of transition $t_j \in T$ is not determined by the value of predicate pr_i),

- $UT = \{ut_1, ut_2, \ldots, ut_r\}$ is a finite set of qualities of transitions $t_j \in T$, which can be deterministic, stochastic or fuzzy.

The finite set of qualities of edges, which is given by forward and backward incidence function, can be defined as follows:

$$Z = \langle I, E, L \rangle \qquad (4)$$

where: -$I = \{i_1, i_2, \ldots, i_x\}$ is a finite set of inhibit edges (x),

-$E = \{e_1, e_2, \ldots, e_y\}$ is a finite set of empty edges (y),

-$L = \{l_1, l_2, \ldots, l_z\}$ is a finite set of logical edges (z).

Further it holds $I \cap E = I \cap L = E \cap L = 0$ where each edge is a member only of one set from sets I or E or L.

Generalized Petri net is called a deterministic Petri net, if all members of UP and UT are deterministic and non-deterministic Petri nets, if UP and UT are finite sets of probabilities or values of membership function.

On the basis GPN we are able to define deterministic Petri nets (some of which are illustrated in table 1) and non-deterministic Petri nets (examples are introduced in table 2). The type of Petri nets, which is derived from GPN, depends only on the type of the system, which is modeled by the given modification of GPN.

Tab.1

Kind of Petri nets	Qualities of place, transition and edge		
place-predicate 1	X=(IC,MoC)	Y=(QC,PR)	Z=(L)
place-predicate 2	X=(IC,MoC)	Y=(QC,τ,PR)	Z=(L)
place-predicate 3	X=(C,IC,MoC)	Y=(QC,τ,PR)	Z=(L)
place-color 1	X=(C,IC,MoC)	Y=(QC)	Z=(L)
place-color 2	X=(C,IC,MoC)	Y=(QC,τ)	Z=(L)
timed 1	X=(IC,MoC)	Y=(QC,τ)	Z=(L)
timed 2	X=(C,IC,MoC)	Y=(QC,τ)	Z=(L)
place-transition	X=(IC,MoC)	Y=(QC)	Z=(L)
transition-action 1	X=(IC,MoC)	Y=(QC,AC)	Z=(L)
transition-action 2	X=(IC,MoC)	Y=(QC,τ,AC)	Z=(L)
transition-action 3	X=(C,IC,MoC)	Y=(QC,τ,AC)	Z=(L)

Tab.2

Kind of Petri nets	Qualities of place, transition and edge		
non-deterministic 1	X=(IC,MoC)	Y=(QC,UT)	Z=(L)
non-deterministic 2	X=(IC,MoC)	Y=(QC,τ,UT)	Z=(L)
non-deterministic 3	X=(IC,MoC,UP)	Y=(QC,UT)	Z=(L)
non-deterministic 4	X=(IC,MoC,UP)	Y=(QC,τ,UT)	Z=(L)
non-deterministic 5	X=(C,IC,MoC,UP)	Y=(QC,UT)	Z=(L)
non-deterministic 6	X=(C,IC,MoC,UP)	Y=(QC,τ,UT)	Z=(L)
non-deterministic 7	X=(C,IC,MoC,UP)	Y=(QC,PR,UT)	Z=(L)
non-deterministic 8	X=(C,IC,MoC,UP)	Y=(QC,τ,PR,UT)	Z=(L)

3. Quantity characteristic

The place $p_i \in P$ is an input place of transition $t_j \in T$, if $m \geq 1$ is in $IC((n,c)_m,i,j)$. Then $p_i \in IN(t_j)$, where $IN(t_j)$ is a set of input places of transition $t_j \in T$.

The place $p_i \in P$ is an output place of transition $t_j \in T$, if $m \geq 1$ is in $QC((n,c)_m,i,j)$. Then $p_i \in OUT(t_j)$, where $OUT(t_j)$ is a set of output places of transition $t_j \in T$.

Each edge is described by an ordered 2-tuple (p_i,t_j) (input edges), or (t_j,p_i) (output edges). The set of input edges $INH(t_j)$ of transition t_j are those edges (p_i,t_j), for which $p_i \in IN(t_j)$ holds.

The transition $t_j \in T$ is enabled ($FIRE(t_j)$=TRUE) from marking M_aC ($a \in N$), if it holds:

$$FIRE(t_j)=\left[\prod_{i=1}^{k} (FIRE_1(j,i) \text{ AND } FIRE_2(j,i) \text{ AND } (p_i,t_j)\in L) \text{ AND } \right.$$

$$\text{AND } FIRE_3(j) \text{ AND } FIRE_4(j)] \text{ OR } \left[\prod_{i=1}^{k} (NOT((FIRE_1(j,i) \text{ AND } \right.$$

$$\text{AND } FIRE_2(j,i)) \text{ AND } (p_i,t_j)\in I) \text{ AND } FIRE_3(j) \text{ AND } FIRE_4(j)]$$

where: $FIRE_1(j,i) = [(p_i,t_j)\in INH(t_j)]$

$$FIRE_2(j,i) = \prod_{l=1}^{m} [M_aC((n,c)_{l,i}) \geq IC((n,c)_{l,i,j})]$$

$FIRE_3(j)$ = expired time for firing of transition t_j

$$FIRE_4(j) = \prod_{l=1}^{q} PR(j,l)$$

$$PR(j,l)=[D(t_j,pr_l)=1 \text{ AND } pr_l] \text{ OR } [D(t_j,pr_l)=-1 \text{ AND }$$

$$\text{AND NOT } pr_l] \text{ OR } D(t_j,pr_l)=0 \qquad (5)$$

Firing a transition $t_j\in T$ is defined by the transformation marking M_aC into a new marking $M_{a+1}C$, when $FIRE(t_j)$ = TRUE holds:

$$M_{a+1}C((n,c)_{m,i})=M_aC((n,c)_{m,i})+QC((n,c)_{m,i,j})-IC((n,c)_{m,i,j}) \qquad (6)$$

4. Quality characteristic

Fundamental qualities of GPN are given on the basis of analysis of the forward marking class. The forward marking class is a set of markings reachable from the initial marking M_0C. New marking $M_{a+1}C$ is reached from marking M_aC by solving equation (6). A graphic representation of the set of reachable markings is a marked tree $[M_0C]$. The marked tree gives the set of possible control state of the system behaviour with respect to the initial control state, which is represented by marking M_0C.

The generation of $[M_0C]$ is ended, when no other transitions are enabled, i.e. for any transition it holds

$$FIRE\ (t_j) = FALSE \tag{7}$$

and any transition is not in the firing state (in the time sense) and reached markings are already members of $[M_0C]$.
Generalized Petri net is a live, if and only if for the arbitrary marking $M_aC \in [M_0C]$ and for the arbitrary transition $t_j \in T$ a sequence of firings exists, which begins in M_aC and transition t_j is fired. When GPN is live, the associated system is deadlock-free. Generalized Petri net is live, if and only if from each vertice of $[M_0C]$ an output edge exists. The algorithm for the live analysis is the following:

```
begin
  if were fired all transitions tj∈T then
    if all vertices of the [M0C] have output edge
      then  GPN is live
      else  GPN is not live
end
```

Generalized Petri net is bounded, when vector of positive integer number $V = (v_1, v_2, \ldots, v_h)$ exists (dimension of vector V is the same as the number of member of the finite set of used color C) that for all $M_aC \in [M_0C]$ and for all $p_i \in P$ $(1 \le i \le k)$ and for all $c_m \in C$ $(1 \le m \le h)$ holds:

$$M_aC\ ((n,c)_{m,i}) \le v_m \tag{8}$$

where v_m is a positive integer number. That means that in any places p_i and in any state there is not more v_m tokens of the color c_m.
The GPN is safe, when GPN is bounded (formula (8) holds) and all members of vector V are equal or less than 1, i.e. for all m $(1 \le m \le h)$ holds:

$$v_m \le 1 \tag{9}$$

The algorithm for the bound analysis of GPN is the following (KP is a number of vertices of $[M_0C]$):

```
begin  loop for processing vertices of the [M0C]
  for a=1 to KP do
    begin loop for processing places P
```

```
      for i=1 to k do
        for m=1 to h do
          if number of the tokens of color cm in the
             MaC((n,c)m,i) in the place pi is Ma((n,c)m,i)≤
             ≤vm, where MaC((n,c)m,i)≥0 and vm>0 are in-
             teger
          then GPN is bounded, i.e. for all MaC∈[M0C]
               formula (8) holds
          else GPN is not bounded
      end
  end
```

Generalized Petri net is conservative when all markings $M_aC \in [M_0C]$ contain a constant number of tokens of the given color, i.e. for all $M_aC \in [M_0C]$ $(0 \le a \le KP)$ and for all colors c_m $(0 \le m \le h)$ holds:

$$\sum_{i=1}^{k} M_aC\left((n,c)_{m,i}\right) = \text{const} \tag{10}$$

The conservatism of GPN secures that realization of the transition does not change a number of tokens of the given color in the GPN. The algorithm for the conservatism analysis is as follows (conservatism is a boolean variable, Help_Value and Value_of_Cons are integer):

```
   begin calculation of conservatism
     Conservatism := TRUE
     for m=1 to h do
       begin loop for processing vertices of the [M0C]
         Value_of_Cons := 0
         for i=1 to k do
           calculation of conservatism for initial marking
           Value_of_Cons := Value_of_Cons + M0C((n,c)m,i)
         for a=1 to KP do
           begin loop for processing places P
             Help_Value := 0
             for i=1 to k do
               Help_Value := Help_Value + MaC((n,c)m,i)
```

```
              Conservatism := Conservatism AND
              AND (Value_of_Cons = Help_Value)
          end
      end
    if Conservatism
      then GPN is conservative
      else GPN is not conservative
  end
```

Generalized Petri net is proper if and only if for each marking $M_a C \in [M_0 C]$ there exists a sequence of firings such that from arbitrary marking $M_a C$ a connection to the initial marking $M_0 C$ exists, i.e. the system returns to its initial state. The algorithm for proper analysis is the following:

```
 begin
   if graph of [M0C] is connected
     then  GPN is proper
     else  GPN is not proper
 end
```

6. Summary

Fuzzy Petri nets enable to analyze discrete systems with non-deterministic properties of uncertainty that are non-descript deterministically and due to the a priori unknowledge of laws of discrete stochastic phenomena distribution neither stochastically. They enable the description and analysis of systems with uncertainty.

On the basis of GPN theory were software tools for the analysis of deterministic and non-deterministic discrete systems created. This program environment enables the input of data, generation of marked tree, realization of analysis and output results. Program is written in the programming language PASCAL and is run on IBM PC compatible computers.

References

[1] STRELEC,J.-OLEJ,V.: Generalized Deterministic, Discrete Stochastic and Fuzzy Petri Nets. BUSEFAL, No.44, 1990, pp.

22-26.

[2] PETERSON,J.L.: Petri Net Theory and Modelling of Systems. Prentice Hall, Inc. 1981.

[3] DIAZ, M.: Modelling and Analysis of Communication and Cooperation Protocols Using Petri Nets. Computer Networks, No.6, 1982, pp. 419-441.

[4] MERLIN,P.M.: A Methodology for the Design and Implementation of Communication Protocols. IEEE Transactions on Communications, Vol.COM-24, No.6, 1976, pp. 614-621.

[5] ETZION,T.-YOELI,M.: Super-Nets and Their Hierarchy. Theoretical Computer Science, Vol.TCS-23, No.3, 1983, pp. 243-272.

[6] MOLLOY,M,K.: Discrete Time Stochastic Petri Nets. IEEE Transactions on Software Engineering, Vol.SE-11, No.4, 1985, pp.417-423.

[7] LIPP,H.P.-GUENTHER,R.: An Application of a Fuzzy Petri Net in Complex Industrial Systems. Mathematical Research. Fuzzy Sets Application, Methodological Approaches, and Results. Akademie-Verlag Berlin, No.30, 1986, pp. 188-196.

[8] CHMÚRNY,J.-OLEJ,V.-MOKRIŠ,I.: Analysis of Discrete Systems by Fuzzy Petri Nets. Elektrotech. Čas., Vol.39, 1988, No.3, pp.226-230.

[9] OLEJ,V.- LEHOTSKÝ,M.- MOKRIŠ,I.: Analysis of Discrete Systems by Non-Deterministic Petri Nets. Proc. of Scienc. Works of VVTŠ, Liptovský Mikuláš, No.1, 1988, pp.69-79.

[10]OLEJ,V. - STRELEC,J.: Generalized Petri Nets. Proc. of 5th Scienc. Conf., EF VŠT, Košice, 1989, pp.140-144.

[11]OLEJ,V.-CHMÚRNY,J.-MOKRIŠ,I.: Analysis of Parallel Discrete Systems by Discrete Stochastic and Fuzzy Petri Nets. Computers and Artificial Intelligence, Vol.10, No.3, 1991 pp.221-237.

[12]OLEJ,V.-STRELEC,J.-CHMÚRNY,J.: Generalization Deterministic and Non-Deterministic Petri Nets. Elektrotech. Čas., Vol.41, No.9, 1990, pp.715-718.