

# Invariant state progress and relation modelling of DEDS

*Gabriel Juhás and Miroslav Kocian*

*Institute of Control Theory and Robotics, Slovak Academy of Sciences, Dúbravská cesta 9, 842 37 Bratislava, Slovak Republic.*

*Tel: 42-7-378 3216. Fax: 42-7-376045.*

*e-mail: utrrjuhi@savba.sk*

## Abstract

The states are used as the fundamental concept (instead of events) and the discrete state change systems (DSCSs) are defined. The possibility of DSCSs modelling, control and description of behaviour specifications using the presented state relation approach is investigated. The generalized T-invariants of Petri nets are defined as a special case of the presented approach.

## Keywords

Discrete event system, relation, Petri nets, T-invariants

## 1 INTRODUCTION

From a historical general system theory point of view, the DEDSs follow the state/transition systems formally defined by a state set and binary relation on this set (Klir, 1972). In the paper it is tried to use the notion of (binary) relation (between states) in the DEDS modelling and control in similar way like predicate calculus is used (Kumar et al., 1993). The simple set structure given in (Klir, 1972) is used and related calculus is developed. The state space and state changes (instead of events) are used as fundamental concepts, consequently a discrete state change system (DSCS) is defined instead of DEDS. An example of the approach is shown on the cat and mouse maze problem (Ramadge and Wonham, 1989; Capkovic, 1993).

Generalized T-invariants of Petri nets can be considered as the special case of the presented theory, if the system is modeled by a Petri net. By means of generalized T-invariants it is possible to describe the relevant kind of discrete processes, where the cycle is understood to repeatedly reach states that are in the given invariance relation. The behaviour according to this relation determines the state progress. The presented definition of T- invariants is more global than the standard definition that can be included in it as the special case.

## 2 RELATION MODELLING OF DEDS

### 2.1 Discrete state change systems

Let  $D$  is the state set of a given system  $D$ . The notion direct discrete state change (or shortly direct state change) of a state  $d \in D$  to a state  $d' \in D$  expresses the fact, that during this state change the system is not found in any state  $d'' \in D$ .

A discrete state change system (DSCS) is a system, where each state change of the system is arisen from a sequence of direct state changes. If the time is ignored and only the order of states is considered, then each possible state trajectory of DSCS can be reduced to a sequence of states that for each pair of neighbours in this sequence holds that direct state change from the first element of this pair to the second element is possible. As follows, these sequences are called the logical state trajectories of the DSCS. A (logical) state trajectory that there is considered to run the time to infinity, is called total (logical) state trajectory. The notion (logical) behaviour means the set of all possible (logical) state trajectories of the given DSCS.

Similarly to (Ramadge and Wonham, 1989), in control of DSCS it is premised that certain direct discrete state changes of the system can be disabled when it is desired.

### 2.2 Relation state change generator

Let  $S$  is an arbitrary set and  $L$  is an arbitrary binary relation on  $S \times S$ .

Let  $L(s, \cdot)$  is a relation given so that  $L(s, \cdot) = L \cap (\{s\} \times S)$  and let  $L(\cdot, s) = L \cap (S \times \{s\})$ .

The support of a binary relation  $L$  is given by the union of the first and the second domain of it and denoted by  $L_{\text{supp}}$  ( $L_{\text{supp}} = \{s; \exists s': [s, s'] \in L \vee [s', s] \in L\}$ ).

The image of an element  $s$  in a relation  $L$  is denoted by  $L(s)$  ( $L(s) = \{s'; [s, s'] \in L\}$ ).

Binary relation  $L^k = \{ [s_i, s_{k+1}]; \exists \{s_i\}_{i=1}^{k+1}: [s_i, s_{i+1}] \in L \text{ for } \forall i \in \{1, \dots, k\} \}$  ( $k$  is a positive integer), is called  $k$ -power of  $L$ . The symbol  $L^{\otimes}$  denotes the binary relation that is the union of all powers of the relation  $L$  ( $L^{\otimes} = \bigcup_{k \in Z^+} L^k$ , where  $Z^+$  is denoting the set of positive integers).

The notion  $L$ -sequence denotes a sequence of elements from  $S$   $\{s_i\}_{i=1}^N$ , where  $N \in Z^+ \cup \{\infty\}$  that next two conditions are satisfied:

1. If  $N > 1$  then  $\forall i \in \{2, \dots, N\}: [s_{i-1}, s_i] \in L$ . 2. If  $N \neq \infty$  then  $L(s_N, \cdot)$  is the empty set.

The sequence for which holds only the first condition is called part of  $L$ -sequence.

Let  $s = \{s_i\}_{i=1}^N$  is a sequence of elements from  $S$  so that there exists some  $L$ -sequence  $q = \{s_k\}_{k=1}^M$ ,  $M \in Z^+ \cup \{\infty\}$ ,  $q$  is selected from  $s$  and next three conditions hold:

1.  $k_1 = 1$ . 2. If  $N = \infty$  then  $M = \infty$ . 3. If  $N \neq \infty$  then  $k_M = N$ .

Then  $s$  is denoted by symbol  $(L)$ -sequence. It is said that  $s$  is containing  $q$  and on the contrary  $q$  is contained in  $s$ . If  $q$  is a part of  $L$ -sequence, then  $s$  is called part of  $(L)$ -sequence.

A binary relation  ${}^sL = L(s, \cdot) \cup \{ [s', s'']; [s', s''] \in L \wedge [s, s'] \in L^{\otimes} \}$  is represented in the graph of the binary relation  $L$  by the tree with the root labelled by an element  $s \in S$ .

Definition: Relation state change generator is a two tuple  $S = (S, F)$  where  $S$  is a set of states and  $F$  is a direct discrete state change relation on  $S \times S$ . Elements of the relation  $F$  are called possible direct discrete state changes of  $S$ .  $F$ -sequences are called total state trajectories

of the generator  $S$  and parts of  $F$ -sequences are called state trajectories the generator  $S$ . By the notion the behaviour of generator  $S$  is understood the set of its all state trajectories.

A state  $s'$  is reachable from a state  $s$  in  $S$  if and only if  $[s, s'] \in F^{\otimes}$  (i.e. iff  $\exists k \in Z^+$ :  $[s, s'] \in F^k$ ). The binary relation  $F^{\otimes}$  is called the reachability relation.

**Definition:** Let  $S = (S, F)$  is a generator. A state  $s' \in S$  is live according to a state  $s \in S$  in generator  $S$  iff for each  $F$ -sequence  $\{s_i\}_{i=1}^N$ , where  $N \in Z^+ \cup \{\infty\}$ , holds the statement: If  $\exists k \leq N$ :  $s_k = s \Rightarrow \exists l, k < l \leq N$ :  $s_l = s'$ . If  $s'$  is live according to each state from  $S$ , then  $s'$  is live in  $S$  and if each state of  $S$  is live in  $S$ , then it is said that the generator  $S$  is live.

**Definition:** Let  $P = E \times E^*$ , where  $E$  is an arbitrary set and  $E^*$  is the set of all sequences of elements from  $E$ . Let  $G$  is a relation on  $P \times P$  so, that for each  $[[e, q], [e', q']] \in G$  holds that if  $q = \{e_i\}_{i=1}^N$  and  $N \in Z^+$ , then  $q' = \{e'_i\}_{i=1}^{N+1}$ , where  $e_{N+1} = e'$ . Then the relation state change generator  $P = (P, G)$  is called relation state change generator with memory, the set  $E$  is called pure state set of  $P$  and the set  $E^*$  is called memory state set of  $P$ . A state trajectory of the generator  $P$  with first element  $[e, e] \in E \times E$  is called the state trajectory starting from the state  $e$  with the reseted memory.

### 2.3 Modelling and control of DSCS via relation generators

To model the behaviour of a DSCS it is used a generator, where states of the given system are represented by states of the generator, and the possible direct changes of the modelled system are represented by elements of direct discrete state change relation of the generator (direct modelling). If the possibility of direct changes of an actual state depends on a state history of the system, then it is used as the model a generator with memory, where the state history of the modelled system is represented by a state of memory of the generator (memorial modelling). The memorial modelling can be used to model logical behaviour and the control of each DEDS that can be modelled by a formal language.

The relation between a generator  $(S, F)$  modelling some system  $D$  and a generator  $(S, C)$  modelling this system  $D$  controlled by some supervisor is briefly formulated as follows: for each  $s \in C_{\text{supp}}$  holds that  $F_u(s) \subseteq C(s) \subseteq F(s)$ , where elements of  $F_u$  represent uncontrollable direct state changes of the system  $D$ .

**Definition:** Let  $S = (S, F)$  is a generator. Let relation  $F_u \subseteq F$ . Let  $L, L \subseteq F$  is a binary relation on  $S \times S$  and let  $s \in L_{\text{supp}}$ . It is said that  $L$  is controllable in the point  $s$  in regard to  $F_u$  if and only if  $F_u(s, \cdot) \subseteq L(s, \cdot)$ . If  $L$  is controllable in each point of its support in regard to  $F_u$ , then it is called controllable in regard to  $F_u$ .

**Definition:** Let  $S = (S, F)$  is a generator. Let relation  $F_u \subseteq F$ . Let  $s, s' \in S$ . If exists a controllable relation  $C$  in regard to  $F_u$  that state  $s' \in S$  is live according to the state  $s$  in generator  $(S, C)$ , then  $s'$  is controllably reachable from  $s$  in regard to  $F_u$  in the generator  $S = (S, F)$ . If, in addition, holds that  $s'$  is the end point of  $C_{\text{supp}}$  ( $C(s')$  is the empty set) then  $s'$  is controllably stabilizable from the state  $s$  in regard to  $F_u$  in the generator  $S$ .

### 2.4 Behavioural specifications and relation calculus for DSCS

The main elements of possible problem settings are properties of the system model, relations

and their properties, relationships between the system model and relations and mutual relationships between relations.

Some relationships between relations can be formulated as follows. Let  $L$  and  $K$  are binary relations on some cartesian product  $S \times S$  of some set  $S$ .

1.  $L \subseteq K^{\otimes}$  (informally it means that  $K$  can go through  $L$ )
  - 1.1.  $L \subseteq K$  ( $K$  can detailly go through  $K$ )
    - 1.1.1.  $L = K$  ( $K$  have to go detailly through  $L$ )
  - 1.2.  $L \subseteq K^{\otimes} \wedge \forall K$ -sequence  $\{s_i\}_{i=1}^N$  so that  $s_1 \in L_{\text{supp}}$  is ( $L$ )-sequence ( $K$  starting in the support of  $L$  have to go through  $L$ )
    - 1.2.1.  $L = K^{\otimes}$  ( $K$  have to go through  $L$ )

The other relevant problems consist in an investigation of relationships between some relations  $L, H$  that both are subsets of any relation  $K^{\otimes}$ .

If above mentioned relation  $K$  is the relation of direct state changes of some system modelling generator, then the relationships between  $L$  and  $K$  represent relationships between relation  $L$  and the system model. The considered system can or have to go (detailly) through the relation  $L$ . The relation  $L$  with some mentioned relationship to the system can characterise a possible and/or required behaviour. From this point of view the behaviour of the system given by relation  $L$  fulfilling the criteria 1.2. can be understood as partial observable.

As follows, let  $L, K$  are arbitrary relations on  $S \times S$ . If for each  $L$ -sequence  $q$  exists  $K$ -sequence  $s$  that  $s$  is containing  $q$ , then  $L \subseteq K^{\otimes}$  ( $K$  can go through  $L$ ). If, in addition,  $K$  is transitive, then  $L \subseteq K$ . Let  $S = (S, F)$  is a generator. If for each  $s \in F_{\text{supp}}$  holds that the cardinal number of  $F(s)$  is equal to one,  $F$  is coherent and  $F^{\otimes}$  is reflexive, then the generator is live.

Logical expressions about properties of a relation  $K$  can be divided into two classes. Let  $V(K)$  denotes a logical expression about a relation  $K$ .

1. If  $V(K) = 1$  (true) and for each  $L \subseteq K$  holds  $V(L) = 1$  then it is said that  $V(K)$  is an expression of the class one.
2. If  $V(K) = 1$  and there exists  $L \subseteq K$  that  $V(L) = 0$  (false) then  $V(K)$  is an expression of the class two.

The following assertion can be use in the modular problem solving.

Let  $H_i = \{1, \dots, n\}$ ,  $n \in \mathbb{Z}^+$  are binary relations on  $S \times S$ . Let  $V_i(H_i)$  are expressions of the class one for all  $i \in \{1, \dots, n\}$ , then  $V(\prod_{i=\{1, \dots, n\}} H_i) = V_1(\prod_{i=\{1, \dots, n\}} H_i) \wedge \dots \wedge V_n(\prod_{i=\{1, \dots, n\}} H_i) = 1$ .

The notion introduced in the paper can be used to derive a proof methodology to examine properties of relations if the other properties (of these relations and/or other relations) are given. The proof methodology is based on the definitions of basic proofs rules so that by the combinations of them more complex proofs can be constructed. Assumptions of proof rules can be given and/or derived by previous applications of proof rules and/or derived from the detail modelling level. The nature of basic proof rules can be shown as follows. Let  $H, K, L$  are some relations on  $S \times S$ , so that  $K \subseteq H^{\otimes}$ . Let holds: if  $K \subseteq H^{\otimes}$ , then  $L \subseteq H^{\otimes}$ . Then  $L \subseteq H^{\otimes}$ .

Let  $L$  is an arbitrary relation on  $S \times S$ . Let  $s_1, s_2, s_3 \in L_{\text{supp}}$  and  $(s_1, s_2) \in L^{\otimes}$ . Let for each  $s \in L_{\text{supp}}$  holds: if  $(s, s_2) \in L^{\otimes}$  then  $(s, s_3) \in L^{\otimes}$ . Then  $(s_1, s_3) \in L^{\otimes}$ .

Let  $H, K, L$  are some relations on  $S \times S$ , so that  $K \subseteq H^{\circ}$  ( $H$  can go through  $K$ ) and  $L \subseteq K^{\circ}$  ( $K$  can go through  $L$ ). Then  $L \subseteq H^{\circ}$ . This assertion can be used to introduce some type of hierarchy of relations on  $S \times S$  and hierarchy of the behaviour description.

## 2.5 The cat and mouse maze example

A cat and mouse are placed in the maze on the Figure 1. Let  $\gamma$  is a function, that maps rooms of the maze onto a set of numbers  $A = \{0,1,2,3,6\}$  in accordance with Figure 1.

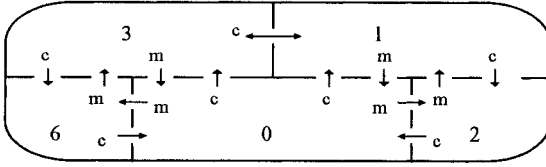


Figure 1 The cat and mouse maze.

Let set  $B = A \cup \{7\}$ . Let set  $S = A \times B$  so, that each pair  $[a,b] \in S$  represents the state, in which the cat is in the room mapped by  $\gamma$  onto  $a$ , and the free mouse is in the room mapped by  $\gamma$  onto  $b$  (if  $b \in A$ ) or the cat has caught the mouse (if  $b = 7$ ).

Let  $F_u, F_c$  and  $F$  are binary relations on  $S \times S$  given as follows.

$$F_u = \{[[a,b], [a',b']]; (((a = 1 \wedge a' = 3) \vee (a = 3 \wedge a' = 1)) \wedge b' = b \wedge b \in B) \vee (b = a \wedge a' = a \wedge b' = 7 \wedge a \in A)\}$$

$$F_c = \{[[a,b], [a',b']]; (([a',b'] = [(a+1)_{\text{mod } 3}, b] \wedge a \in \{0,1,2\} \wedge b \in B) \vee ([a',b'] = [a, (b-1)_{\text{mod } 3}] \wedge a \in A \wedge b \in \{0,1,2\}) \vee ([a',b'] = [(a+3)_{\text{mod } 9}, b] \wedge a \in \{0,3,6\} \wedge b \in B) \vee ([a',b'] = [a, (b-3)_{\text{mod } 9}] \wedge a \in A \wedge b \in \{0,3,6\})\}$$

$$\text{and } F = F_u \cup F_c.$$

Then the possible movement of the animals can be directly modelled by the behaviour of the generator  $S = (S, F)$  starting in the initial state  $[2,6]$ , where elements of  $F_u$  represent direct state changes, that cannot be disabled (the door between rooms 1 and 3 cannot be closed).

The required behaviour specification is expressed as follows.

1. The animals initially placed in these rooms (represented by pair of number  $[2,6]$ ) will never occupy the same room simultaneously.
2. It is always possible for the cat and the mouse to return to the initial state.
3. The greatest possible freedom of animal movement is demanded.

The task is to determine, whether the behaviour satisfying these requirements can be reached by a supervision, and if it is possible, then to determine the control properties, that must be fulfilled by the supervisor. To translate this task to modelling formalism notions the following task is obtained. The relation  $C$  on  $S \times S$  satisfying under mentioned requirements is sought.

1.  ${}^{[2,6]}C = C \subseteq F$
2.  $L$  is controllable

3.  $\forall [a,a] \in S: [[2,6], [a,a]] \notin C^\otimes$
4.  $\forall [a,b] \in C^\otimes([2,6]): [[a,b],[2,6]] \in C^\otimes$
5. There exists no relation  $K$  on  $S \times S$  for which all above mentioned requirements are fulfilled and holds:  ${}^{[2,6]}C \subset {}^{[2,6]}K$ .

The relation  $C$  satisfying these all 5 requirements is formally given as follows.

$$C = \{ [[a,b], [a',b']]; ([a',b'] = [(a+1)_{\text{mod } 3}, b] \wedge a \in \{0,1,2\} \wedge b = 6) \vee \\ ([a',b'] = [a, (b-3)_{\text{mod } 9}] \wedge a = 2 \wedge b \in \{0,3,6\}) \vee \\ (a = 0 \wedge a' = 3 \wedge b' = b = 6) \vee \\ (((a = 1 \wedge a' = 3) \vee (a = 3 \wedge a' = 1)) \wedge b' = b = 6) \}$$

Then, by comparison of the relation  $F$  and the relation  $C$  in each point of  $s \in C_{\text{supp}}$  it is obtained the information about direct state changes from the state represented by the  $s$ , which must be enabled by a supervisor (direct state changes represented by elements of  $C(s,.)$ ) and disabled (prohibited) by a supervisor (direct state changes represented by elements of  $(F(s,.) - C(s,.)$ ). Then the behaviour of this supervised system starting in the initial state can be modelled by the behaviour of the generator  $H = (S, C)$  starting in the initial state  $[2,6]$ .

### 3 INVARIANT STATE PROGRESS IN PETRI NETS

The state relation is defined by the state difference function  $\Delta$ , so the states are in this state relation, if the difference between them is equal to given  $\Delta$ . The behaviour according to this relation is examined. Such behaviour (state progress) is understood as to reach the states that are in this relation. There are many issues as the existence of infinite behaviour according to this relation, determination of event sequences assuring this behaviour (state progress).

The problem is investigated by means of Petri nets theory (Reisig, 1985) that enable to use analytic algebraic methods. As follows,  $m \times n$  matrix is denoted by [row 1; ...; row  $m$ ].

Definition: A Petri net is a 4-tuple  $PN = (P, T, I, O)$  such that:  $P = \{p_1, \dots, p_m\}$  is a finite and nonempty set of places,  $T = \{t_1, \dots, t_n\}$  is a finite and nonempty set of transitions  $P \cap T$  is the empty set,  $I(p,t): P \times T \rightarrow N$  is an input function,  $N = Z^+ \cup \{0\}$ ,  $O(t,p): T \times P \rightarrow N$  is an output function ( $I$  and  $O$  can be understood as an integer  $m \times n$  and  $n \times m$  matrices).

In a Petri net the system states are characterised by markings  $m: P \rightarrow N$ ,  $m = (m(p_1); \dots; m(p_m))$ . The set of all markings is denoted by  $M$ . In the equation  $m' = m + C \cdot y$ , where  $C$  is the structure (incidence) matrix of the given Petri net ( $C = O^t - I$ ), vectors  $y$  determine the firing numbers of each transition that are necessary to reach the state  $m'$  from the state  $m$ . If the difference  $\Delta$  is given, then vectors  $y$ , that are integer solutions of the equation  $C \cdot y = \Delta$ , determine the firing numbers of each transition in sequences of events, that assure the state progress (to reach state  $m'$  from state  $m$  that  $(m, m')$  belongs to the relation given by the difference  $\Delta$ ). Classic  $T$ -invariants are solutions of equation  $C \cdot y = 0$  (Lautenbach, 1986). The solutions of the equation  $C \cdot y = \Delta$  (when  $\Delta$  is understood as linear function) have the similar geometric interpretation as the solution of equation  $C \cdot y = 0$ . That is the reason, that the algorithms for determination of  $T$ -invariants can be simply modified to solve the equation  $C \cdot y = \Delta$ . The properties of solutions of both equations ( $C \cdot y = 0$ ,  $C \cdot y = \Delta$ ) are the same in special cases, where difference  $\Delta$  is a linear function without absolute coefficients. There is the

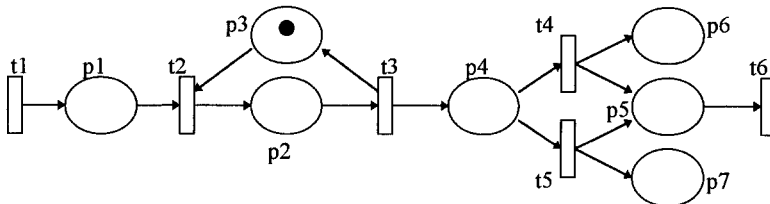
possibility to define generalized T - invariants as solutions of the equation  $C \cdot y = \Delta$ , where  $\Delta$  is understood as invariant function and the states in the relations given by  $\Delta$  are invariant states. The behaviour according to the relation can be formally described by the relation reachability graph, where the neighbouring nodes belong to this relation (given by difference  $\Delta$ ) and the arcs represent realizable solutions of the above mentioned equation ( $C \cdot y = \Delta$ ). The generalized T-invariants can be understood as a special case of the state relation. By means of generalized T-invariants it is possible to describe the relevant and interesting kind of discrete processes i.e. real cyclic processes, where the cycle is understood to repeatedly reach some similar (not exactly the same) state to the initial state (e.g. systems with counters).

Definition (Generalized T-invariants): Let  $PN = (P, T, I, O)$  is a Petri net, let  $M$  is the set of all functions (markings)  $m: P \rightarrow N$ , let  $C = O^t - I$ ,  $m, m' \in M$  and  $y$  is a firing vector. Let  $D = \{[m, m'], [m, m'] \in M \times M\}$  is the invariance relation. Then the integer solutions of the system  $C \cdot y = m' - m$  are called T-invariants in  $m$  in regard to the relation  $D$ .

Let  $\Delta(M, H)$ , where  $H$  is an arbitrary set, is a function  $\Delta: M \times H \rightarrow Z^m$  ( $m$  is the number of places in the Petri net) such that for each  $h \in H \exists [m, m'] \in D$  that  $m' = m + \Delta(m, h)$ . The function  $\Delta$  is called the invariant function and the equation  $C \cdot y = \Delta(M, H)$  is called the invariance relation equation. Evidently, integer solutions of the invariance relation equation in  $m$  are T-invariants in  $m$  in regard to the relation  $D$ . Classic T-invariants represent the special case with the difference function  $\Delta(M, H) = \Delta(H) = 0$ .

The further formal definitions of above mentioned notions, solutions of the equations and the investigation of relation reachability graph properties, are presented in the (Juhás and Kocian, 1994, 1995). A motivation of generalized T-invariant definition is shown on an example of the simple production line.

Example: The line (modelled by Petri net in Figure 2) is formed by the input buffer, the numeric control (NC) machine, the output buffer and the counters. In the system there is running repeatedly the cyclic process, where the product is worked up by the NC machine, the quality test is performed and according to the result the related product counter is incremented.



$$C = \begin{bmatrix} 1, -1, 0, 0, 0, 0, 0; & 0, 1, -1, 0, 0, 0, 0; & 0, -1, 1, 0, 0, 0, 0; & 0, 0, 1, -1, -1, 0, 0; \\ 0, 0, 0, 1, 1, -1, -1; & 0, 0, 0, 1, 0, 0, 0; & 0, 0, 0, 0, 1, 0, 0 \end{bmatrix}$$

Figure 2 The Petri net model of a simple product line

Transition t1 - the arrival of the product, place p1 - waiting in the input buffer, t2 - the start of the working up, p2 - the state of working up, t3 - the finish of the working up and the quality test start, p3 - the state of the NC machine (ready or not), p4 - the quality test, t4 and t5 - the

quality test finish and increment of the related counter, p5 - waiting in the output buffer, p6 and p7 - state of the correct and uncorrect product counter, t6 - the departure of the product.

The invariance relation is given as follows:  $D = \{[m, m']; m'(p_i) - m(p_i) = 0 \text{ for } i = 1 \dots 5, (m'(p_6) - m(p_6)) + (m'(p_7) - m(p_7)) = u \wedge u \text{ is an integer}\}$ . This state progress leads to the solving of the system  $A \cdot x = B \cdot u$ , where  $A$  is  $6 \times 6$  matrix and  $A_i = C_i$  for  $i = 1 \dots 5$ ,  $A_6 = C_6 + C_7$ ,  $A_i$  and  $C_i$  are rows of the matrixes  $A$  and  $C$ , and  $B = [0;0;0;0;0;1]$  and  $u$  is an integer parameter. The T-invariants:  $y = k \cdot [1;1;1;1;0;1] + v \cdot [1;1;1;0;1;1]$ ,  $k, v$  are integers. The nonnegative T-invariants indicate numbers of the event executions that are necessary to perform complete pass of the products through the production line.

#### 4 CONCLUSION

DSCSs are defined by using of state changes instead of events. The relation state change generator is used to model DSCS and related calculus is developed. The presented relation approach can be used to describe the possible behaviour as well as desirable behaviour of the system (control specification). The presented approach is illustrated by an example. The generalized T-invariants of Petri nets are defined as a special case of the presented state relation approach. A building of the relation calculus for DSCSs including algorithms for investigation of behavioural specifications represents the main subject of the further research.

#### REFERENCES

- Capkovic, F (1993) Petri Net-based Approach to the Maze Problem Solving. In Balemi, S., Kozák, P. and Smedinga, R. (Ed.) *Discrete Event Systems: Modelling and Control, Proc. of a Joint Workshop held in Prague, 1992*, Birkhäuser, 173-179.
- Juhás, G. and Kocian, M. (1994) Generalized T-Invariants of Petri Nets and Control of DEDS. In *Preprints of 1st IFAC Workshop on New Trends in Design of Control Systems 1994*, Smolenice, Slovakia, 408-413.
- Juhás, G. and Kocian, M. (1995) Invariant Relation Behaviour in Petri Nets. In Snorek, Sujansky, and Verbraeck (Ed.) *Modelling and Simulation 1995, ESM95, Prague, Czech Republic*, A publication of the [SCS], 160-164.
- Klir, G. J. (1972) (Ed.) *Trends in General System Theory*, Wiley - Interscience, New York.
- Kumar, R., Garg, V., and Marcus, S.I. (1993) Predicate and Predicate Transformers for Supervisory Control of Discrete Event Dynamic Systems. *IEEE tr. on Automatic Control*, Vol.38, N. 2, 232-247.
- Lautenbach, K. (1986) Linear Algebraic Techniques for Place/Transitions Nets. In: *Proceedings of an Advanced Course on Petri Nets*, LNCS 254, 142-167.
- Ramadge, P. and Wonham, W.M. (1989). The Control of Discrete Event Dynamic Systems. *Proceedings of the IEEE*, Vol. 77, N. 1, 81-98.
- Reisig, W. (1985) *Petri Nets*. Springer-Verlag, Berlin.