Analysis and Supervisory Control of a Pressurizer using Coloured Petri Nets

János Rudan * **Katalin Hangos** ** **Gábor Szederkényi** **

* *Faculty of Information Technology, Pázmány Péter Catholic University, Budapest, Hungary. Email: rudan.janos@itk.ppke.hu*
** *Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest, Hungary. Email: {hangos, szeder}@sztaki.hu*

Abstract: An approach for constructing a discrete event supervisory controller is proposed in this paper that is based on the qualitative dynamic model of the controlled system described by coloured Petri nets (CPNs). The structure of the state space is explored and partitioned by generating the reachability graph of the controlled system. The supervisor is then constructed to avoid the system to reach states that require emergency cooling or heating based on the partitioned reachability graph.
The proposed approach is illustrated by using a simple, but industrially important example of a pressurizer in a pressurized water nuclear power plant.

Keywords: Petri net, reachability analysis, supervisory control, pressurizer

---

1. INTRODUCTION

Discrete event system (DES) approaches that emerge from computer science give us the opportunity to discover important qualitative properties of a given system. DES is a well-known approach in theoretical computer science where the system is described as a state machine (i.e. with states and transitions) and the current state of the system is obtained as a state of the automaton.

The main advantage of the DES-based description is the ability to extract functional and qualitative information from models and to get know the structure of the state space (see e.g. (Fanni and Giua, 1998)). Based on these properties a special way of controller design can be achieved which is often used for supervisory or hierarchical controller design (Giua, 1992).

1.1 Background and motivations

Results emerging from graph theory based deterministic system analysis were applied to to DES description. (Aveyard, 1974) uses a Boolean matrix equation model to represent the state changes occurring in the system in a deterministic way. This enables the examination of basic properties of a discrete event system, for example, if it has any deadlocks or cyclic behaviour.

Petri nets (PNs) were introduced by Carl Adam Petri (1966) as special types of DES, originally to describe chemical processes. In his work PNs are introduced as graphical tools for the description and analysis of concurrent processes which arise in systems with many components. The structure of Petri nets is a biparite graph, its nodes can only be places (denoting substances, states or conditions) and transitions (denoting reactions, state transformations) connected into a bipartite directed graph. Tokens are introduced to represent the occurrence of a logical value at a given place. Later these type of directed graphs were extended with additional parameters: adding time parameters to the transitions leads to timed PNs, while extending the pool of available tokens to an unordered set leads us to coloured PNs.

All types of Petri nets can be effectively used in several fields of research and applications. Comprehensive surveys of the properties and usage of Petri nets are written by Murata (1989) and Jensen (1997). The main application areas of Petri nets are the following: task scheduling and parallel processing (Hanzalek, 2003), investigating liveness problem of event based systems (Su et al., 2005), supervisory control (Giua, 1992) and urban traffic control (Dotoli and Fanti, 2004). Several theoretical and practical results are published about the usage of Petri nets in hybrid systems, namely where the described system contains continuous-time and discrete components, too (Vázquez et al., 2010).

The dynamic analysis of Petri nets is an important research area in itself. As it is shown in several papers, PN is a proper tool to examine properties of a given system such as deadlock-free operation (liveness), discover the places in concurrency or in conflict, analyse bottle-neck nodes and resource allocation (Iordache and Antsaklis, 2005).

Most of the dynamic analysis is built upon the reachability analysis, that is to determine the reachable states, of this type of discrete event systems (Wang et al., 2000). In most cases, reachability analysis is done by search algorithms exploring the state space (Hangos et al., 2001). Our approach to construct a discrete event supervisory controller is also based on reachability analysis.

$\star$ This work is supported by the Hungarian Research Fund though grant K83440.

1.2 Aims

Set theory based techniques are popular tools to examine the reachability properties of a nonlinear dynamic system (Blanchini and Miani, 2008). Using the classical, differential equation-based model of the investigated system the complete analysis of the reachability sets could be a complex and computationally expensive problem (see e.g. (Luspay, 2011)). By using an automaton-based, discrete-event system model and qualitative analysis, the reachability sets can be constructed in an easier way.

The general aim of this work is to improve the performance of a controlled dynamic system - namely a pressurizer tank of a nuclear power plant - using a supervisory controller. The investigation of the control problem is completed by using a coloured Petri-net based model. The examined properties of the system are the temperature dynamics and the effect of the pressure controller. The qualitative analysis of the controlled system leads us to the usage discrete event system description techniques to design a supervisory controller.

2. COLOURED PETRI NETS FOR DISCRETE EVENT SYSTEM DESCRIPTION

In this paper, first the notion of CPN is described, thereafter the algorithms used for CPN analysis - especially for reachability analysis - are introduced, where we follow Jensen (1997) in the presentation.

2.1 Coloured Petri nets

A coloured Petri net is biparite multigraph described by a 6-tuple

$$< P, T, C, I^{-}, I^{+}, M_0 >$$

where P is a set of places, T is a set of transitions and P and T are disjoint. C is a color function defined from P into a finite and non-empty sets. I^{-} and I^{+} are the backward and forward incidence functions defined on $(P \times T)$ that $I^{-}(p,t), I^{+}(p,t) \in [C(t) \rightarrow C(p)], \forall (p,t) \in (P \times T)$. A marking is a distribution of coloured tokens on the places of the CPN described by a marking function M defined on P. An initial marking $M_0(p) \in C(p), \forall p \in P$ is given for each CPN. Arcs representing I^{-} and I^{+} are often called input arcs and output arcs, respectively. These have a weight, called transcription describing the actual function they represent.

The operation of a CPN is as follows: firing a transition t in a marking M consumes $I^{-}(p,t)$ tokens from each of its input places $p \in P$, and produces $I^{+}(p',t)$ tokens in each of its output places $p' \in P$. A transition is enabled (it may fire) in M if there are enough tokens in its input places (noted by p) for the consumptions to be possible, i.e. iff $\forall p : M(p) > I^{-}(p,t)$. Note that enabled transitions may fire in arbitrary order.

2.2 Reachability and the reachability analysis algorithms

Reachability is a fundamental property of a dynamic system. Considering a known system dynamics and given constraints, a state S_1 is reachable, if it is possible to reach the final state S_1 starting the system from a given S_0 initial state. This definition is similar to the concept of controllability or reachability in control theory.

The concept of reachability in a CPN deals with the following problem: given a CPN called N, an initial marking M_0 and a marking M_1, is it possible that $M_N(M_0) = M_1$, where M_N is the effect of the CPN on the marking after N steps. This means that starting the CPN from the initial state, using its transition rules the final state M_1 can be reached.

Because of a standard CPN has a finite number of different states, the reachability set of a given initial state can be calculated using a search algorithm exploring all descendant states. The search algorithm could be either a depth-first or a breath-first one with a loop-rejecting capability. Starting the algorithm for each and every possible states the advance of the states in the state space can be reconstructed as a set of trees.

In several cases the control aim of a system is to avoid a given operation range. In these cases the state-space partitioning can be a proper technique. This approach groups the possible states to a desired and an undesired set, and the controller tries to keep the system in the desired set, or drive it back from the undesired set if necessary. Forming these sets from the result of a complete reachability analysis the control aim can be completed, as it is shown in this paper.

2.3 Discrete event supervisory controllers

In a distributed control system, i.e. when the system is jointly controlled by different controllers, a coordination of their operation is usually needed. One possible solution of it is to use a supervisory controller, that takes into account global control aims and global system information, and influences some of the controllers that operate independently according to their local control aims. In most of the practical industrial cases, the supervisory controller is implemented in the system in a later stage, and it is required that it should only affect the operation if the original independently operating controllers fail to satisfy the global control or performance requirements. In other words, a "loose" supervisory control operation is desired.

In order to have a distributed control system with a uniform representation, the supervisory controller is usually designed in the form compatible with the other controllers. Therefore, there is an established research direction that deals with designing a supervisory controller to a discrete event system described by PNs that is also in PN form (see e.g. (Giua and Seatzu, 2001) or (Giua, 1992)).

3. PRESSURIZER TANK - A CASE STUDY

A simplified dynamic model of a pressurizer tank in the primary circuit of a pressurized water nuclear power plant (Fazekas et al., 2007) was used in this work. The model introduced below was used as a case study to show the capabilities of the CPN-based system analysis.

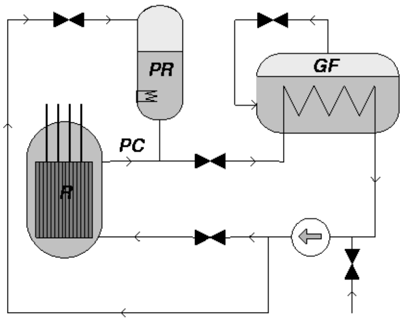

Fig. 1. Schematic diagram of the primary circuit of a VVER type pressurized water nuclear power plant. The pipes of the primary circuit (PC) connects the reactor (R), the pressurizer (PR) and the steam generator (GF) to each other.

3.1 *The role of the pressurizer tank in a nuclear power plant*

A simplified schematic diagram of a primary circuit in a pressurised water nuclear power plant can be seen in Fig. 1. This simplified scheme includes the following parts. The reactor vessel containing the fuel- and control rods filled with water which is circulated in the circuit by the coolant pump. The hot water goes through the steam generator while heating up the secondary circuit.

The water in the primary circuit is kept on high pressure (up to 124 bar) with the help of the pressurizer to increase the boiling temperature of the water (around 327 Celsius degree). Evidently in this closed system the water temperature and the pressure is strongly connected to each other thus temperature control of the circuit means control of the pressure in the circuit.

There are two important controllers that are considered in this work. The *pressure controller* keeps the pressure of the primary circuit on a constant level by adjusting the heaters in the pressurizer tank. The reactor's power is controlled independently of the other subsystems of the primary circuit by the *power controller.*

3.2 *Dynamic model of the pressurizer tank*

The flowsheet of the pressurizer is shown in Fig. 2. The pressurizer controls the primary circuit pressure with active heating, passive cooling and active emergency cooling according to the current pressure and the operation limits. The tank contains a constant volume of water. Due to the natural heat loss, the tank is cooling constantly, while extra cooling is provided by a valve which allows to let cold water into the tank (at the same time the same amount of hot water is released from it to keep the water level constant). The heating is provided by four electric heaters divided into two groups which are controlled separately.

The temperature of the primary circuit is mainly determined by the reactor power. From the viewpoint of the pressurizer tank, the changes in the reactor power can be considered as *disturbances* in the primary circuit's water temperature. Note that this temperature can be considered as the *output of the reactor power controller* and has an effect on the pressurizer dynamics.

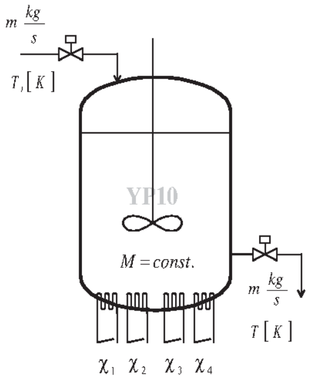

Fig. 2. Simplified scheme of the pressurizer tank. The four heater elements appear at the bottom, the cooling water injection valve is at the top of the tank.

A simple nonlinear dynamic model of the pressurizer is developed earlier (Szabó et al., 2010) that was used to design the pressure controller (see later). Based on this model the following discrete-time model of the pressurizer tank is proposed:

$$x(k+1) = x(k) + \kappa_h(k) * \lambda_h + \lambda_l + \delta(k) \qquad (1)$$

where x denotes the temperature in a given time step, κ_h is a time-dependent Boolean value representing the state of the active heating, λ_h and λ_l are proportional tags weighting the heating and heat loss effect, respectively. The variable δ represents the time-dependent disturbance in the model, that is related to the time-derivative of the primary circuit temperature controlled by the power controller.

In order to obtain a discrete event system model, the range space of the variables in the above equation has been discretized as follows.

Temperatures The temperature range U_T of the variable x was divided into five intervals:

$$U_T = \{E^-, L, N, H, E^+\}$$

representing *emergency low*, *low*, *normal*, *high* and *emergency high* temperature intervals. The value of the disturbance δ was also represented by this set.

Switching variables The effect of the pressure controller is described by the time-dependent Boolean variable κ_h with a natural discrete range space

$$U_B = \{0, 1\}$$

The multiplication between the elements of U_T and U_B interpreted as follows:

$$\kappa * \lambda = \begin{cases} \emptyset & if\, \kappa = 0 \\ \lambda & if\, \kappa = 1 \end{cases}$$

where $\lambda \in U_T$ and $\kappa \in U_B$.

The heating power λ_h and the natural heat loss λ_l were modelled as discrete valued constants from the range space U_T, with $\lambda_h = H$, $\lambda_l = L$.

Computing the solution The solution of the discrete dynamic equation (1) is computed using the following qualitative addition table:

$[a] + [b]$	E^-	L	N	H	E^+
E^-	E^-	E^-	E^-	L	N
L	E^-	E^-	L	N	H
N	E^-	L	N	H	E^+
H	L	N	H	E^+	E^+
E^+	N	H	E^+	E^+	E^+

In order to keep the model simple the qualitative range set U_T was mapped to the set of integers and the usual integer arithmetic was used.

The model solution table In order to facilitate the use of the discrete event model (1) in a CPN modelling and simulation framework, the solution of the model is represented by a so called *solution table*. This contains the qualitative values of the time dependent signals $x(k+1), x(k), \kappa_h(k), \delta(k)$ in its columns, where we give all possible combinations of the qualitative values appearing on the right hand side of Eq. (1) and give the resulting value of $x(k+1)$.

Table 1. Evaluation table of Eq. (1). Each variable is time dependent.

x (k)	κ_h	δ	x $(k+1)$	x (k)	κ_h	δ	x $(k+1)$
E^-	0	$*$	E^-	N	1	E^-	L
E^-	0	E^+	L	N	1	L	N
E^-	1	E^-	E^-	N	1	N	H
E^-	1	L	E^-	N	1	H	E^+
E^-	1	N	L	N	1	E^+	E^+
E^-	1	H	N	H	0	E^-	E^-
E^-	1	E^+	H	H	0	L	L
L	0	$*$	E^-	H	0	N	N
L	0	H	L	H	0	H	H
L	0	E^+	N	H	0	E^+	E^+
L	1	E^-	E^-	H	1	E^-	N
L	1	L	L	H	1	L	H
L	1	N	N	H	1	$*$	E^+
L	1	H	H	E^+	0	E^-	L
L	1	E^+	E^+	E^+	0	L	N
N	0	E^-	E^-	E^+	0	N	H
N	0	L	E^-	E^+	0	H	E^+
N	0	N	L	E^+	0	E^+	E^+
N	0	H	N	E^+	1	E^-	H
N	0	E^+	H	E^+	1	$*$	E^+

3.3 The pressure controller

The goal of the pressure controller is to keep the pressure of the pressurizer at a constant normal level (depicted as interval N) by manipulating the heaters. This is achieved by keeping the temperature of the pressurizer constant. The pressure controller is modelled as a hysteresis type controller with discrete actuation, i.e. it can only switch on and off the heating in normal operation mode.

The basic operation flow of the pressurizer is as follows. When the water temperature is low, the first heater group is switched on, while in case of an extra-low temperature an emergency actuation mechanism activates both group of heaters. If such a high temperature is reached that the natural heat loss is not enough to keep the temperature in the normal range, the active cooling is switched on. This event is also considered as an emergency actuation.

Formally, the operation of the pressure controller can be described by the following operation table (see Table 2), that has the same structure as the model solution table above (see Table 1). The output of the controller ($\kappa_h(k)$) is depends only on the current state of the tank ($x(k)$) and the sign of the value $d(k) = x(k-1) - x(k)$. The occurrence of the emergency operation is noted in column EM.

Table 2. Operation table of the pressure controller.

$x(k)$	$d(k)$	$\kappa_h(k)$	EM
$= N$	$*$	0	0
$< N$	$*$	1	0
$> N$	$*$	0	0
$= E^+$	$-$	0	0
$= E^+$	$+$	0	1
$= E^-$	$-$	1	1
$= E^-$	$+$	1	0

The emergency actuation mechanisms are considered as undesired events. This encourages us to propose more advanced control techniques to avoid the emergency operation of the pressurizer tank.

3.4 Coloured Petri net model of the controlled pressurizer

The coloured Petri net (CPN) model of the pressurizer tank can be seen in Fig. 3. Places are represented with yellow circles (named p_i) while the blue rectangles represent the transitions (named t_i). Tokens are represented with coloured dots in the places.

The model represents the pressurizer together with the pressure controller that performs temperature control of the tank using the active heating and considers the temperature of the primary circuit as a disturbance.

The above described discrete event models of the pressurizer tank and the pressure controller were used and mapped to the CPN terminology as follows. The transitions in the CPN model correspond to the model equations in such a way that they realize the computing described in their model solution table. Places correspond to signals, where token types either represent different temperature intervals or the state of the active heating.

Therefore, the CPN realization has the following structure: p_1 and p_2 are the places representing the temperature of the pressurizer in the current and in the next time step, respectively. Tokens at place p_4 symbolize the primary circuit temperature which is handled as the disturbance. p_3 stands for the state of the switch of the active heating.

Transition t_1 realizes the dynamics of the pressurizer and computes the solution of the model equation Eq. (1). The output of the transition t_1 is the new tank temperature. The generated token appears at p_2. Transition t_2 represents the heating controller algorithm, the operation of which is described in sub-section 3.3. The resulting temperature control actuation is represented by two types of tokens (heating off, heating on). The two additive token types symbolizing emergency cooling and emergency heating operations have the same effect in the state equations. The state of the active heating represented by the token at p_3.

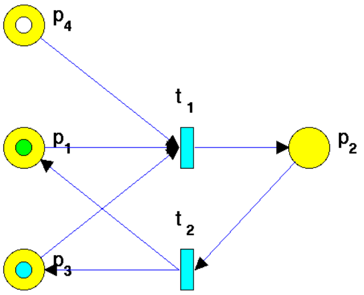

Fig. 3. CPN model of the pressurizer.

3.5 Simulation results

In order to validate the behaviour of the proposed model, we completed several simulations.

For the simulation we used the PetriSimM (2010) MATLAB Toolbox. The toolbox has been largely improved and extended to fit our purposes. Altogether nine colours were defined in the CPN model: five colours for temperature interval description and four for the temperature control actuation.

To test the correct operation of the heating control, we used random disturbance values with uniform distribution in each step. As it can be seen in Fig. 4, the controller can react and actuate correctly and forces the system towards the desired N state, although it can be noticed that the pressurizer temperature changes heavily during the operation. The simulation was run through 50 steps (equals to 50 minutes of simulation time). On the first subplot the pressurizer temperature, on the second the disturbance values, while on the third subplot the heating control values can be seen.

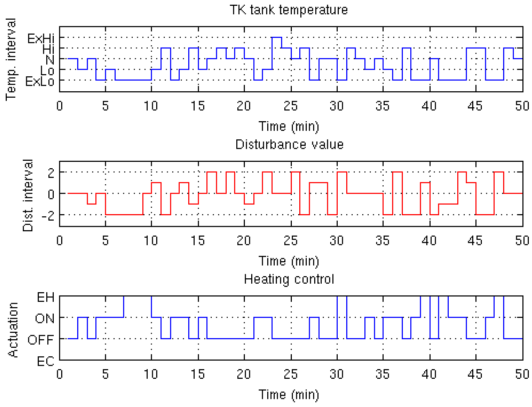

Fig. 4. Simulation results using the CPN model of the pressurizer with random disturbance. The first plot shows the temperature of the tank, the second shows the disturbance value, the third shows the output of the heating controller.

As it is mentioned before, the disturbance in the model is the temperature of the primary circuit, that is the output of the power controller. In order to simulate a more realistic situation, the dynamics of the disturbance is changed in the second test case. The disturbance is set to be constant high value for a given time window, this models an increase in the reactor power. After this phase, it is changed to be a low value for another long time for modelling the reactor power decrease situation. The selected size of the time windows contains enough time step to enable us to disregard the transient phases. As it can be seen on Fig. 5, the controller can not keep the system from the emergency operation in the case of a long disturbance phase that appears with extreme values.

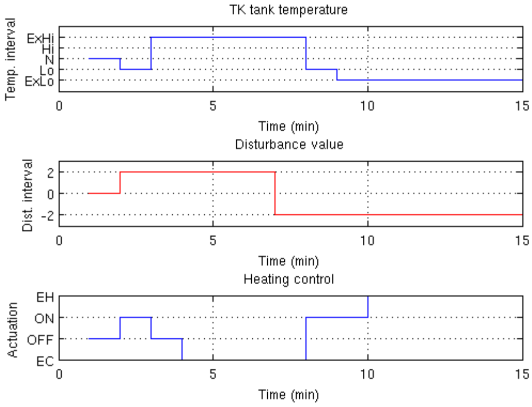

Fig. 5. Simulation results using the CPN model of the pressurizer with disturbance having longer constant-valued blocks. The upper three plots have similar meaning as in Fig. 4, while the last plot shows the output of the heating controller.

3.6 Reachability analysis of the pressurizer model

Using the proposed CPN model, a reachability analysis was completed in order to determine the set of states which can be reached by the system. The algorithm of the analysis was the following: each possible state was considered as an initial state. A simulation was started from all initial states using all possible disturbance values, completing a depth-first exploration of the successive states. If a state is visited by the algorithm, it was marked and never revisited in order to avoid infinite loops. The resulting set of trees was merged into one graph.

In Fig. 6 the reachability graph of the system can be seen. The states are represented by circles and directed arcs represents the allowed state transitions. It is important to note that the "state" of the system includes the qualitative value of the pressurizer temperature $x(k)$ and the temperature control actuation κ_h. The arc colors symbolize the disturbance value which is needed to move between two states. Colour coding at the nodes is designated to show arcs have the same starting and ending node.

3.7 Partitioning the state space based on reachability analysis

In the investigation of the generated graph as the result of the reachability analysis the following was concluded:

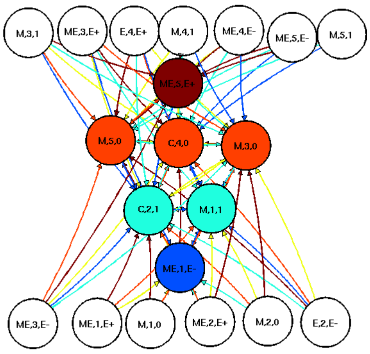

Fig. 6. The reachability graph of the system. Node labelling: $\{set, state, actuation\}$, where $set \in \{M, E, ME, C\}$ stands for margin, emergency, both margin and emergency, core sets, respectively. *state* is the color of the temperature token (depicted by integer values) appearing in the given state of the tank, $actuation \in \{0, 1, E+, E-\}$ depicting heating off, heating on, emergency heating, emergency cooling, respectively.

the state space can be divided into three parts. This partitioning is represented in Fig.6, where node labels depict the class of a node.

The first set of nodes belong to the emergency states. This class marked with an extra E character in the node label. If the system reach one of these states, an emergency actuation is initialized to force the system back into a nominal temperature range.

The second set called *margin* contains those nodes from the emergency set are reachable in one step. Nodes belong to this set marked with M in the node label. From the control point of view this set plays a crucial role: if the system steps into one of the states in the margin set, we could be able to determine the proper control input to avoid to reach an emergency state.

The third set is the so called *core* part containing states that are not in neighbourhood of the emergency set. Until the state of the system is in this set the normal operation is performed.

4. SUPERVISORY CONTROL OF THE PRESSURIZER

4.1 Designing a supervisory controller

The supervisory controller is designed to supervise two, originally independent controllers: the pressure controller of the pressurizer tank and the power controller. The main idea behind this is the following. The operation of the power controller (the changes in the water temperature of the primary circuit) is considered as an external disturbance in the pressurizer model. Because of this, when the pressurizer controller itself can not keep the system from the emergency actuation, the supervisory controller can order the power controller lower the power change rate on the primary circuit thus leading the system back to normal operation.

Operation principle The information obtained from the discovering of the structure of the state space was used to design a supervisory controller. The aim is to avoid the emergency states using the predictions based on the reachability graph. As it is shown previously, the state space of the pressurizer can be partitioned into three parts: the set of emergency states, the margin and the core. The supervisory controller is created to drive the original pressure controller and determine the maximal value of supportable disturbance.

The supervisory controller recommends heating control inputs for the pressurizer controller to avoid that the system gets into the *margin* set. This control strategy gives more chance to avoid the emergency states because it has capability to predict the future states using the information extracted from the reachability analysis.

The operation table The supervisory controller was also designed as a discrete event system with its operation specified by an operation (solution) table. Based on the reachability analysis a look-up table was created and used as an operation table containing the states and the disturbance values which takes the system into the *margin* set or into an emergency state.

The supervisory controller has three possible output values (marked as $y(k)$ in the operation table). The ON and OFF values drive the pressure controller to provide the same output. The third possible value (called $NOIDEA$) is presented when there is no need to force the pressure controller to a given output value.

The evaluation table of the supervisory controller can be seen in Table 3. If a possible combination of tank temperature ($x(k)$), pressure controller state (κ_h) and disturbance (δ) is not presented in the table than the corresponding supervisory controller output ($y(k)$) is $NOIDEA$.

Table 3. Evaluation table of the supervisory controller. EM markings in the κ_h field means the emergency operation of the pressure controller.

$x(k), \kappa_h$	$\delta(k)$	$y(k)$	$\delta(k)$	$y(k)$
E^-, ON_{EM}	E^-, L, H	ON	$*$	$NOIDEA$
E^-, ON	E^-, L, H	ON	$*$	$NOIDEA$
L, ON	E^-, N	ON	E^+	OFF
N, OFF	E^-, L, H	ON	$*$	$NOIDEA$
H, OFF	E^-, N	ON	E^+	OFF
E^+, OFF	L	ON	H, E^+	OFF
E^+, OFF_{EM}	L	ON	H, E^+	OFF
E^-, OFF	N, H	ON	$*$	$NOIDEA$
E^-, OFF_{EM}	N, H	ON	$*$	$NOIDEA$
L, ON_{EM}	E^-, N	ON	E^+	OFF
L, OFF	L, N, E^+	ON	$*$	$NOIDEA$
L, OFF_{EM}	L, N, E^+	ON	$*$	$NOIDEA$
N, ON_{EM}	L	ON	H, E^+	OFF
N, ON	L	ON	H, E^+	OFF
N, OFF_{EM}	E^-, L, H	ON	$*$	$NOIDEA$
H, ON_{EM}	E^-	ON	N, E^+	OFF
H, ON	E^-	ON	N, E^+	OFF
H, OFF_{EM}	E^-, N	ON	E^+	OFF
E^+, ON_{EM}	$*$	$NOIDEA$	L, E^+	OFF
E^+, ON	$*$	$NOIDEA$	L, E^+	OFF

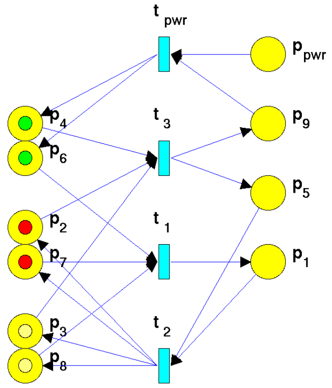

Fig. 7. CPN model of the pressurizer with the supervisory controller. The transition t_{pwr} and place p_{pwr} are unmodelled parts.

CPN implementation During the implementation, an extra transition is introduced into the CPN model depicting the supervisory controller. The resulting CPN model can be seen in Fig. 7 containing the supervisory controller at transition t_3. The transition gets the state of the pressurizer (p_2, p_7), the disturbance (p_4, p_6) and the pressure control values (p_3, p_8), and produces a token describing the recommended heating control value at p_5. This token is processed by the original pressure controller. If it is necessary to reduce the disturbance, the proper information is provided for the power controller (depicted as t_{pwr}) at place p_9. Place p_{pwr} is to indicate the source of the original power command.

4.2 Simulation results using the supervisory controller

Having integrated the supervisory controller into the system, similar simulations were completed as before with the original model.

The result of the first test case with random valued disturbance can be seen in Fig. 8. Investigating the control inputs generated by the controller, it can be noticed that emergency inputs occurred much less frequently compared to the case without the supervisory control.

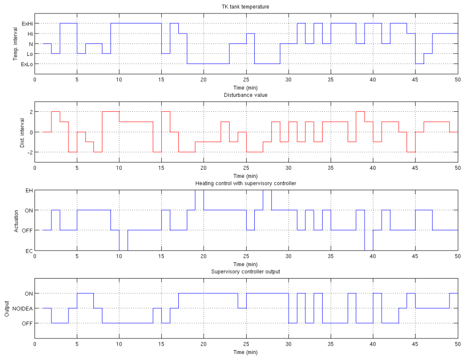

Fig. 8. Simulation results using the CPN model of the pressurizer with supervisory controller using random disturbance. The order of the plots is similar as in Fig. 4.

The second test case, when a disturbance values appear for longer time windows the difference is more significant. As it can be seen in Fig. 9., the appearance of emergency operation is bounded to very few time steps. This is caused by the supervisory controller, which on one hand drives the original pressure controller, on the other hand ordered the power controller to decrease the load change on the primary circuit causing moderated disturbance values.

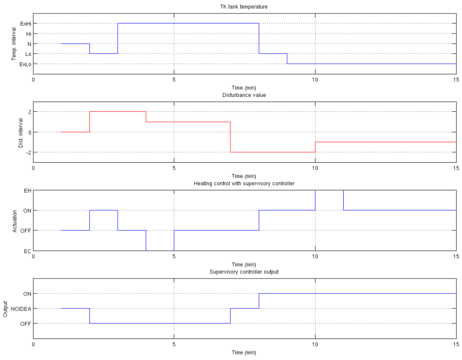

Fig. 9. Simulation results using the CPN model of the pressurizer with supervisory controller using longer constant-valued disturbance blocks. The order of the plots is similar as in Fig. 5. As it can be seen, proper limiting of the disturbance helps to keep the tank temperature in the operation range without emergency actuation.

The performance of the supervisory controller was examined by calculating the sum of time which the pressure controller spent in emergency operation mode. In a simulation modelling 100 minutes of operation without the supervisory controller emergency operation was used 30 times, with the supervisor this value decreased to 10.

In conclusion we can say based on the simulation results, that the supervisory controller has the intended effect on the system. With the help of it, the system can avoid most of the emergency situations while keeping the quality of the operation on the desired level.

4.3 Results of the reachability analysis of the supervised system

The reachability analysis of the supervised system was completed similarly to the original case. The system is started from each possible state and the simulation is completed for each disturbance value. Because of this, the capability of the supervisory controller to bound the disturbance value according to the state of the system is not visible. Therefore, the constructed reachability graph is to be considered as a worst-case result.

The result of the state space analysis can be seen in Fig. 10. The resulting graph shows similar structure as in the case without the supervisory controller.

Within the reachability graph, a subgraph can be formed considering the disturbance bounding capability of the supervisory controller, which operates effectively in the case of simulations over longer time windows. The bounded disturbance subgraph consists of the states which are reachable in the case when the disturbance bounding is in operation. The subgraph representing the formed reachability subset can also be seen in Fig. 10 in the shadowed ellipse region.

Considering the above, the following can be concluded. The supervisory controller which is designed using the partitioning of the state space is operating in a desired way. The original structure of the state space is preserved showing that the supervisory controller can drive the pressure controller effectively without serious changes in the existing structure of the system. Using the disturbance bounding capability of the supervisory controller, the reachability set of the system can be narrowed to a subset resulting further improvements in the control quality.

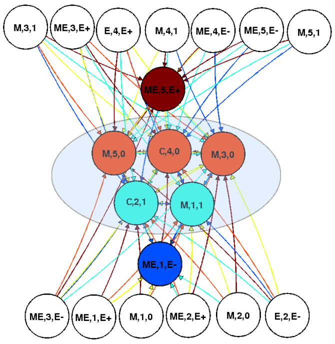

Fig. 10. Result of the reachability analysis of the supervised system. The marked set in the center is the subset in which the system stays if the disturbance values are bounded by the supervisory controller. Node labelling scheme is similar as in Fig. 6.

5. CONCLUSION AND FUTURE WORK

In this paper a coloured Petri net based model of a dynamic system has been presented. By using coloured Petri nets as a special type of discrete event systems, it is shown that a simple qualitative dynamic model of a controlled pressurizer can be represented with this tool.

The formulated CPN model is used to map the structure of the state space completing a reachability analysis of the system. The obtained reachability graph is partitioned into several sets to separate the states into functionally meaningful groups.

The created sets and the corresponding data were utilized as a knowledge base of a supervisory controller in a CPN form, which aims at avoiding the emergency states using the prediction capability gained from the reachability analysis.

Future work includes three main directions. First, the model of the pressurizer can be developed using more sophisticated dynamic model in order to describe more effects appearing in the real system. As a second direction, reducing the size of the temperature intervals by increasing the number of tokens representing them leads to more realistic simulation results. The expanded token set is also necessary at the examination the proper operation of the supervisory controller. Finally, the model has to be tested using real data sets in order to prove that an increased performance can be achieved with the help of the proposed architecture.

REFERENCES

Aveyard, R.L. (1974). A Boolean model for a class of discrete event systems. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-4(3), 249–258.

Blanchini, F. and Miani, S. (2008). *Set-theoretic methods in control*. Birkhäuser.

Dotoli, M. and Fanti, M.P. (2004). An urban traffic network model via coloured timed Petri nets. In *Discrete event systems*.

Fanni, A. and Giua, A. (1998). Discrete event representation of qualitative models using Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 28, 770–780. doi:10.1109/3477.735387.

Fazekas, C., Szederkényi, G., and Hangos, K. (2007). A simple dynamic model of the primary circuit in VVER plants for controller design purposes. *Nuclear Engineering and Design*, 237, 1071–1087. doi: 10.1016/j.nucengdes.2006.12.002.

Giua, A. and Seatzu, C. (2001). Design of observers/controllers for discrete event systems using petri nets. In B.C.X. Xie, P. Darondeau, and L. Lavagno (eds.), *Synthesis and Control of Discrete Event Systems*, 167–182. Kluwer.

Giua, A. (1992). *Petri Nets as Discrete Event Models for Supervisory Control*. Ph.D. thesis, Rensselaer Polytechnic Institute, Troy, New York.

Hangos, K., Lakner, R., and Gerzson, M. (2001). *Intelligent Control Systems: An Introduction with Examples*. Kluwer Academic Publisher.

Hanzalek, Z. (2003). *Parallel Algorithms for Distributed Control A Petri Net Based Approach*. Ph.D. thesis, Czech Technical University, Prague.

Iordache, M.V. and Antsaklis, P.J. (2005). *Supervisory Control of Concurrent Systems, a Petri net Structural Approach*. Birkhauser.

Jensen, K. (1997). *Coloured Petri nets*, volume 1. Springer Berlin / Heidelberg.

Luspay, T. (2011). *Advanced Freeway Traffic Modeling and Control*. Ph.D. thesis, Budapest University of Technology and Economics.

Murata, T. (1989). Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4), 541–580.

Petri, C.A. (1966). Kommunikation mit automaten. *New York: Griffiss Air Force Base, Technical Report*, 1, 1–Suppl. 1. English translation.

PetriSimM (2010). *PetriSimM Toolbox*. TU Wien. URL `http://seth.asc.tuwien.ac.at/petrisimm/`.

Su, H.Y., Wu, W.M., and Chu, J. (2005). Liveness problem of Petri nets supervisory control theory for discrete event systems. *Acta Automatica Sinica*, 31(1).

Szabó, Z., Szederkényi, G., Gáspár, P., Varga, I., Hangos, K.M., and Bokor, J. (2010). Identification and dynamic inversion-based control of a pressurizer at the Paks NPP. *Control Engineering Practice*, 18, 554–565. doi: 10.1016/j.conengprac.2010.02.009.

Vázquez, C., Sutarto, H., Boel, R., and Silva, M. (2010). Hybrid Petri net model of a traffic intersection in an urban network. In *2010 IEEE Multiconference on Systems and Control*. Yokohama, Japan.

Wang, J., Deng, Y., and Xu, G. (2000). Reachability analysis of real-time systems using time Petri nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 30, 725–736. doi:10.1109/3477.875448.