DOI: 10.2507/28th.daaam.proceedings.021

HUMAN-OPERATOR BEHAVIOUR MONITORING IN IT-SYSTEMS

Alexey Shumsky, Alexey Zhirabok & Nadezhda Kalinina

This Publication has to be referred as: Shumsky, A[lexey]; Zhirabok, A[lexey] & Kalinina, N[adezhda] (2017). Human-Operator Behaviour Monitoring in IT-Systems, Proceedings of the 28th DAAAM International Symposium, pp.0160-0167, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-11-2, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/28th.daaam.proceedings.021

Abstract

This work deals with IT-systems that include human-operator. It is well known that just human-operator incorrect actions in many cases resulted in fatal accidents. Due to this, the problem arises to timely detect and isolate the human-operator behaviour errors. The objective of present paper is the generalization and extension of former results proposed by authors within the framework of the above problem solution. To solve the problem, the algebra of partitions is involved. The model in the form of nondeterministic finite automation is used to describe human-operator behaviour. The method for monitor design based on the nondeterministic finite automata determinization is proposed. The procedure for the determinization that guarantees the minimal loss of information is developed. The facilities of the monitor to detect and isolate human-operator errors are investigated. Illustration is given for a case of the change management process in IT-systems.

Keywords: IT-systems; Algebra of partitions; Nondeterministic finite automata; Determinization.

1. Introduction

Strict demands imposed on the modern systems of critical purposes require timely detection of such system behaviour deviation from the prescribed order that may cause the no desirable accidents. This work deals with IT-systems that include human-operator. It is well known that just human-operator incorrect actions in many cases resulted in fatal accidents. That is why this work focuses solely on the diagnosis of human-operator behaviour errors. Related problems were considered in several papers dealing with human-operator monitoring in human-machine systems (HMS). For example, [1] deals with the task to timely detect and prevent consequences of unexpected situations whose solution is based on direct measurements of human emotional and psychological states. But existing sensors technological limitations may hinder the performance of such approaches at practice. Due to this, when human-operator errors are not immediately related to human life, health or significant financial loss, so-called behavioural approach looks reasonable. So, when direct measurements are impossible or difficult, an intuitive solution is to observe variables that are directly linked to the unobservable information we need to assess. In order to do so, qualitative or quantitative models of human-machine systems are required to estimate the human state, the performance or the behaviour [2].

Within the behavioural approach the solution to the problems of unexpected situation detection and abnormal operation diagnosis in HMS was given for the train-driving context in [3-5]. The feature of this solution is the use of the human-operator behaviour model (HBM) in the form of nondeterministic finite automaton (NFA). Then, solution is reduced to this model determinization involving mathematical techniques of the pair algebra of partitions [6].

The objective of present paper is the generalization and extension of the results proposed in [4, 5] to IT-systems. At the beginning, the modelling approach proposed in above papers is generalized for a case of IT-systems. It is the first contribution of the paper. The second contribution is the condition to check the facilities of the new monitor to detect and isolate human-operator errors.

The rest of the paper is organised as follows. Firstly, in Section 2, designing procedure for HBM in the form of NFA with application to IT-systems is considered. Then, in Section 3, an approach to monitor design in the form of a bank of deterministic finite automata (FA) is developed. Section 4 is devoted to human-operator errors detection and isolation facilities analysis. All the theoretical results from Sections 2-4 are illustrated by examples related to a case of the change management process in IT-systems. Section 5 concludes the paper.

2. The HBM design for IT-systems

2.1. Nondeterministic modelling approach

An approach to design the HBM in the form of NFA [4, 5] involves the procedure including the following steps:

1) design the HBM in the form of deterministic FA which describes the ”desired” behaviour according to existing operator’s guidelines;

2) extend this model up to nondeterministic one, taking into account the actions of operator under possible troubles that involves the link: action – observation –reaction;

3) augment the obtained model taking into account the possible human-operator errors resulting in deviations from the prescribed operator actions.

Notice that application of the two first steps of this procedure results in HBM in the form of NFA:

$$M = (I, Q, \delta) \qquad (1)$$

where I and Q are the finite sets of the NFA inputs and states respectively, δ is the function of transitions. In contrast to the conventional deterministic FA, the model (1) is nondeterministic, since for some state $q \in Q$ and some input $i \in I$ several transitions may exist and what transition should take place in real situation becomes clear only when this transition has occurred. This fact can be explained as follows: in general case one can admit existence of different results of observations and human-operator reactions which does not conflict with operator’s guidelines in the main.

The function of transitions for NFA is written as follows

$$\delta(q, i) \in \{q^+ \mid q^+ \in Q^+_{q,i} \subseteq Q\} \qquad (2)$$

where $Q^+_{q,i}$ is the NFA subset of states that are admissible after transition from the state $q \in Q$ initiated by the input $i \in I$. It is assumed that the function of transitions is specified by appropriate table or directional graph.

Human-operator errors e_j, $1 \le j \le N$, where N is a number of errors, may result in additional transitions which are not initially assumed in (2). The model with the error e_j is written in the form

$$M_j = (I, Q, \delta_j) \qquad (3)$$

with the function of transitions δ_j which contains additional (with respect to the function δ) transitions corresponding to the error e_j. The model (3) is also nondeterministic.

2.2. Application to IT-systems

Consider the change management process in IT-systems. This process is the most important for IT-systems: it is responsible for living cycle management for all changes and facilitates of the useful change realization with minimal interruption of the IT-services [7]. The individuals of the process are:

1) the initiator is the member of IT department who realizes the initial processing, assignment and checking the process fulfilling;

2) the executor is an engineer who fulfills the changes in the elements of configuration or coordinates the work of the contractor;

3) the consulting committee which fulfills the scheduled estimating and planning the changes;

4) the process manager is the member of the IT department who monitors the change management process and forms

the proposals for its improving.

The prescribed order of the process assumes fulfilling the following steps:

1) the initiator formulates the task and sends it for confirmation to the consulting committee;
2) the consulting committee makes a decision to confirm this task and to send the task to the process manager or to return the task for its modification to the initiator;
3) the process manager makes appropriate mark and sends the task to the executor;
4) the executor performs the task and informs the process manager when the task is completed;
5) the process manager checks the result and completes the task or sends the information about necessary task modification to the initiator.
6) the task is completed.

Consider HBM design in detail taking into account prescribed order of the process given above. This design follows the steps of HBM designing approach described above.

Step 1: Construct the model in the form of the deterministic FA. Denote the FA states q_i, $1 \le i \le 6$, according to the steps of above order. The FA inputs generated by the process manager are introduced as follows: i_1 is generated when the task formulation is accomplished, i_2 is generated when the task is confirmed by the consulting committee, i_3 is generated when the consulting committee returns the task for its modification to the initiator, i_4 is generated when the task is sent to the executor, i_5 is generated when the task is completed by the executor, i_6 is generated when the task is completed by the process manager, and i_7 is generated when the information about necessary task modification is sent to the initiator. Appropriate model description is given in Table 1.

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
q_1	q_2	–	–	–	–	–	–
q_2	–	q_3	q_1	–	–	–	–
q_3	–	–	–	q_4	–	–	–
q_4	–	–	–	–	q_5	–	–
q_5	–	–	–	–	–	q_6	q_1
q_6	–	–	–	–	–	–	–

Table 1. The deterministic model

Step 2: Extend the above deterministic model up to nondeterministic. To do this, consider possible situations which violate the prescribed order. These situations may occur because of insufficiently accurate order description, approbation absence of the order for all possible situations or private reasons of the process participants.

Due to this, additional transitions arise that correspond to the following situations:

1) the initiator does not send the task for confirmation to the consulting committee but sends it to the process manager;
2) the executer does not send the task to the process manager but completes the process;
3) the process manager returns the confirmed task for additional confirmation to the consulting committee;
4) the process manager returns the confirmed task to the initiator;
5) the executor returns the confirmed task to the initiator;
6) the executor returns the confirmed task to the consulting committee.

Additional transitions, corresponding to the situations 3-6, are the reactions of the process participants aimed at making the task clearer. So, from the deterministic model (Table 1) one obtains the basic model (Table 2).

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
q_1	q_2	–	–	–	–	–	–
q_2	–	q_3	q_1	–	–	–	–
q_3	–	–	–	q_1, q_2, q_4	–	–	–
q_4	–	–	–	–	q_1, q_3, q_5	–	–
q_5	–	–	–	–	–	q_6	q_1
q_6	–	–	–	–	–	–	–

Table 2. The basic nondeterministic model

Step 3: Additional transitions, corresponding to the situations 1 and 2, may be considered as the initiator and the executer errors respectively. So, the augment of the basic model by appropriate additional transitions results in the models M_1 and M_2 related to the above errors. Descriptions for the errors containing models are given in tables 3 and 4.

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
q_1	q_2, q_3	—	—	—	—	—	—
q_2	—	q_3	q_1	—	—	—	—
q_3	—	—	—	q_1, q_2, q_4	—	—	—
q_4	—	—	—	—	q_1, q_2, q_5	—	—
q_5	—	—	—	—	—	q_6	q_1
q_6	—	—	—	—	—	—	—

Table 3. The model M_1

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
q_1	q_2	—	—	—	—	—	—
q_2	—	q_3	q_1	—	—	—	—
q_3	—	—	—	q_1, q_2, q_4	—	—	—
q_4	—	—	—	—	q_1, q_2, q_5, q_6	—	—
q_5	—	—	—	—	—	q_6	q_1
q_6	—	—	—	—	—	—	—

Table 4. The model M_2

3. Monitor design

3.1. Description of the monitor scheme

The monitor scheme applied below is the special case of one proposed in [8, 9]. The last scheme is an original extension of the scheme usually used for fault isolation in engineering dynamic systems (a bank of unknown input observers) [10]. The structure of the scheme in use is given in Fig. 1.

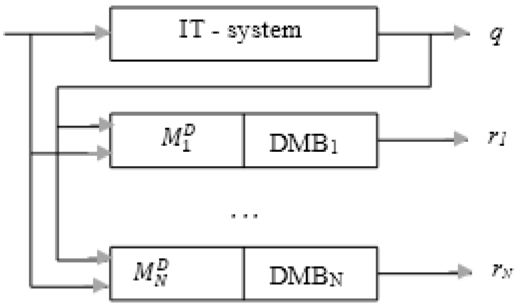

Fig. 1. The structural representation of the monitor

The deterministic FA M_j^D , $1 \le j \le N$, from this bank is constructed by determinization of the automation M_j such that the behavior of this automation is invariant (insensitive) with respect to the human-operator error e_j and sensitive (as soon as it is possible) to the rest errors. For the model M_j^D the following description is used

$$M_j^D = (I \times Q, Q^{(j)}, \delta^{(j)}) \tag{4}$$

with the transition function

$$q^{(j)+} = \delta^{(j)}(q,i), \quad q^{(j)+} \in Q^{(j)}, \quad q \in Q, \quad i \in I \tag{5}$$

and $Q^{(j)}$ is the model M_j^D states set

Consider the link between the models M_j^D and M_j. Assume existence of the map $\varphi^{(j)}: Q \to Q^{(j)}$ such that under absence of human-operator errors $e_k,\ k \neq j,$ the following equality holds

$$q^{(j)} = \varphi^{(j)}(q). \qquad (6)$$

Taking into account (2), (3), (5), and (6), one can write defining relation for the transition function of the model M_j^D:

$$\delta^{(j)} = \varphi^{(j)}(\delta_j). \qquad (7)$$

Decision making block *DMB*$_j$ checks equality (6). By construction, the model M_j^D is invariant to human-operator error e_j. As a result, *DMB*$_j$. forms the residual $r_j = 0$ under presence of this error. Also, if M_j^D is invariant with respect to some error e_i and this error has occurred, *DMB*$_j$ forms the residual $r_j = 0$ again. To detect the rest human-operator errors via the residual r_j, *DMB*$_j$ should form $r_j = 1$ for all these errors that assumes additional demand imposed on the map $\varphi^{(j)}$ (see Proposal given below). The correspondence between human-operator errors and the residuals is specified by the human-operator error syndrome matrix (ESM). An example of such matrix for $N = 3$ is given in Table 5.

	e_1	e_2	e_3
r_1	0	0	1
r_2	0	0	1
r_3	1	1	0

Table 5. Example of the ESM

The cell of ESM related to the pair r_j and e_k contains “1” if the error e_k violates equality (6), otherwise it contains ”0”. The ESM is a tool for decision making via residual vector $r = col(r_1, r_2, ..., r_N)$. Errors are isolated if every two columns of this matrix are different. It follows from above example that errors e_1 and e_2 are indistinguishable. As a result, the monitor scheme can be simplified by excluding one of the model (M_1^D or M_2^D) and appropriate decision making block.

To find the maps $\varphi^{(j)},\ 1 \le j \le N,$ and to construct the ESM, pair algebra of partitions proposed by Hartmanis and Stearns [6] is essentially used.

3.2. Mathematical background

The main elements of algebra in use are partitions of the set Q. Remind that partition of the set Q is the set of its subsets (blocks of partition) $\{B_{\pi_1}, B_{\pi_2}, ..., B_{\pi_n}\}$ such that

$$B_{\pi_i} \subseteq Q, \quad B_{\pi_i} \cap B_{\pi_j} = \varnothing,\ i \neq j, \quad \bigcup_{i=1}^{p} B_{\pi_i} = Q.$$

Let π and σ be partitions of Q. One says that partition π is less or equal to σ and denote $\pi \le \sigma$ if for every B_{π_i} there exists B_{σ_k} such that $B_{\pi_i} \subseteq B_{\sigma_k}$, where B_{π_i} and B_{σ_k} are appropriate blocks of the partitions π and σ, respectively. One can say that if we know the partition π, then we have information about states of the FA with an accuracy up to partition π; if $\pi \le \sigma$, then one says that the partition π has the same or more information than σ.

There are two special partitions denoted by **0** and **1**: each block of the partition **0** contains the only element of the set Q; the partition **1** has single block containing all the elements of Q. For arbitrary partition π of Q the evident inequality holds: $0 \le \pi \le 1$.

It is known [6] that a set of all partitions of Q with relation of partial order is a lattice. Therefore, for each pair of partitions (π, σ) of Q one can find two partitions *inf* (π, σ) and *sup*(π, σ). It is common practice to denote these partitions by $\pi \times \sigma$ and $\pi + \sigma$.

To compute the result of operations × and +, one may use the following simple rules [6]: each block of the partition $\pi \times \sigma$ is intersection of some blocks of the partitions π and σ; each block of the partition $\pi + \sigma$ is union of all intersected blocks of the partitions π and σ.

Let π and σ be partitions of *Q*. Binary relation Δ is formed from all pairs (π, σ) of partitions of *Q* satisfying the following condition:

$$q \equiv q^{'}(\pi) \Rightarrow \delta(q,i) \equiv \delta(q^{'},i)(\sigma), \quad \forall i \in I$$

In this case one says that partitions π and σ form a pair. Given a partition π, several partitions σ may exist such that $(\pi, \sigma) \in \Delta$, in particular, $(\pi, \mathbf{1}) \in \Delta$ is valid for all π. Define the operator **m** as follows: **m**(π) is a partition satisfying conditions

$$(\pi, m(\pi)) \in \Delta, (\pi, \sigma) \in \Delta \Rightarrow m(\pi) \leq \sigma.$$

Thus, the partition **m**(π) is the smallest one which the partition π forms a pair with. Because $(\pi, \mathbf{1}) \in \Delta$ for each π, the partition $\mathbf{m}(\pi)$ exists for every π, at least, $\mathbf{m}(\pi) = \mathbf{1}$. The following formula of the operator **m** computation for NFA is the extension of known (for FA) formula [6] and has been proposed in [5]:

$$\mathbf{m}(\pi) = \sum_{j} \sigma_j \tag{8}$$

where σ_j is the minimal partition such that

$$q \equiv q^{'}(\pi) \Rightarrow \delta(q,i) \equiv \delta(q^{'},i)(\sigma_j) \tag{9}$$

for all $\delta(q,i) \in Q^{+}_{q,i}, \delta(q^{'},i) \in Q^{+}_{q^{'},i}$.

3.3 The model determinization

Solution of this problem for some model $M_j, \quad 1 \leq j \leq N,$ assumes finding the map $\varphi^{(j)}$, followed by the transition function $\delta^{(j)}$ determination from defining relation (7).

Notice, in all cases the determinization of nondeterministic model results in loss of information about the next system state as soon it is unknown what possible transition of the nondeterministic system will realize at practice. To guarantee the maximal accuracy of monitoring (human-operator errors detectability and isolability), information loss should be reduced to minimal one.

Let $\pi_{\varphi(j)}$ be the partition specified by the map $\varphi^{(j)}$ according to the rule

$$q \equiv q'(\pi_{\varphi(j)}) \Leftrightarrow \varphi^{(j)}(q) = \varphi^{(j)}(q') \tag{10}$$

Zero partition of the set *Q* contains the whole information about nondeterministic model M_j state. Thus, according to the definition of operator **m**, the rule of the next state computation for the model M_j with the maximal accuracy is

$$\pi_{\varphi(j)} = \mathbf{m}(0) \tag{11}$$

where operator **m** is applied to the model M_j. So, relation (11) gives the rule for the model determinization with minimal loss of information.

3.4 *Monitor design for IT-system*

Consider the model M_1^D design on the base of the model M_1 (Table 3). The first step is computation of the partition **m**(**0**). From (9) one has (rows of the Table 3 are considered; the states from the same cell of the table are combined within the same block of partition): $\sigma_1 = \{(q_1);(q_2,q_3);(q_4);(q_5);(q_6)\}$; $\sigma_2 = \sigma_5 = \sigma_6 =$ $\{(q_1),(q_2),(q_3),(q_4),(q_5),(q_6)\}$; $\sigma_3 = \{(q_1,q_2,q_4);(q_3);(q_5);(q_6)\}$; $\sigma_4 = \{(q_1,q_2,q_5);(q_3);(q_4);(q_6)\}$. After this, applying formula (8) that assumes combining the intersected blocks of above partitions, one obtains $\pi_{\varphi(j)}$ = **m**(**0**) =

$\{(q_1,q_2,q_3,q_4,q_5);(q_6)\}$. The second step involves finding transition function for the model M_1^D (Table 3). Solution of this task consists in:

1) put the states of the model M_1^D into correspondence to appropriate blocks of the partition $\pi_{\varphi^{(1)}}$: $q_{1,1} \in Q^{(1)}$ corresponds to the block (q_1,q_2,q_3,q_4,q_5) and $q_{1,2} \in Q^{(1)}$ corresponds to the block (q_6);

2) rewrite the matrix of transitions (Table 3) according to the new definitions (see Table 6).

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
q_1	$q_{1,1}$	–	–	–	–	–	–
q_2	–	$q_{1,1}$	$q_{1,1}$	–	–	–	–
q_3	–	–	–	$q_{1,1}$	–	–	–
q_4	–	–	–	–	$q_{1,1}$	–	–
q_5	–	–	–	–	–	$q_{1,2}$	$q_{1,1}$
q_6	–	–	–	–	–	–	–

Table 6. The model M_1^D

The partition $\pi_{\varphi^{(2)}} = \{(q_1,q_2,q_4,q_5,q_6);(q_3)\}$ and the table of transitions for M_2^D (Table 7) are given without additional illustration

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
q_1	$q_{2,1}$	–	–	–	–	–	–
q_2	–	$q_{2,2}$	$q_{2,1}$	–	–	–	–
q_3	–	–	–	$q_{2,1}$	–	–	–
q_4	–	–	–	–	$q_{2,1}$	–	–
q_5	–	–	–	–	–	$q_{2,1}$	$q_{2,1}$
q_6	–	–	–	–	–	–	–

Table 7. The model M_2^D

4. Error syndrome matrix construction

Construction of the ESM is based on finding conditions that guarantee violation of (6) in a case of human-operator error. Assume that human-operator error e_k instead of transition into state q results in transition into state q'. Put these states into the same block of partition π_{e_k} whose rest of blocks contain only single state from the list that does not contain q and q'.

Proposal. Human-operator error e_k results in violation of condition (6) if

$$\pi_{e_k} \times \pi_{\varphi^{(j)}} = 0 . \qquad (12)$$

Proof. As soon by construction the states q and q' are contained in the same block of partition π_{e_k}, from (12) it follows that these states should be contained in different blocks of partition $\pi_{\varphi^{(j)}}$. Then, from (10) one obtains

$$\varphi^{(j)}(q) \neq \varphi^{(j)}(q') .$$

Using this result, illustrate ESM construction for above IT-system. Denote human-operator errors in this system as e_1 and e_2 (initiator and executor errors respectively, see Subsection 2.2). So, the error e_1 results in transition from the state q_1 to the state q_3 under input i_1 while under absence of this error transition to the state q_2 should take place. Respectively, the error e_2 results in transition from the state q_4 to the state q_6 under input i_5 while under absence of this error transition to the state q_5 should take place. As a result, one obtains partitions $\pi_{e_1} = \{(q_1);(q_2,q_3);(q_4);(q_5);(q_6)\}$ and $\pi_{e_2} = \{(q_1);(q_2);(q_3);(q_4);(q_5,q_6)\}$. Using partitions $\pi_{\varphi^{(1)}} = \{(q_1,q_2,q_3,q_4,q_5);(q_6)\}$ and $\pi_{\varphi^{(2)}} =$

$\{(q_1, q_2, q_3, q_4, q_6); (q_3)\}$ obtained above, it is easily to show that $\pi_{e_1} \times \pi_{\varphi(2)} = 0$ and $\pi_{e_2} \times \pi_{\varphi(1)} = 0$. The first equality means that the first error appearance will result in violation of (6) and generation $r_2 = 1$ while the second equality means that the second error also will result in violation of (6) and generation $r_1 = 1$. Thus, ESM takes the following form:

	e_1	e_2
r_1	0	1
r_2	1	0

Table 8. ESM for IT-system

As soon the columns of this matrix are different, the isolation of human-operator errors is possible.

5. Conclusion

The problem of human-operator errors monitoring has been considered with application to IT-systems. Solution of the problem assumes the use of behavioral human-operator model in the form of NFA and involves the determinization of this model. Following new results have been obtained:

1) the method of nondeterministic behavioral human-operator model design for IT-systems;

2) the method of nondeterministic finite automata determinization that guarantees the minimal loss of information and, in contrast to known method [3, 4], allows to decrease the computation expanses;

3) the method of human-operator errors analysis. Application of all above methods has been illustrated for the change management process monitoring in IT-systems.

It is assumed to improve the obtaining results by application of the mathematical tools of the algebra of covers instead of the algebra of partitions.

6. Acknowledgments

This work was supported by Russian Scientific Foundation (project No 16-19-00046).

7. References

[1] Wilson, J.R. & Norris, B.J. (2005). Rail human factors: Past, present and future, Applied ergonomics, Vol. 36, 2005, pp.649-660.

[2] Mceod, R.W.; Walker, G.H. & Moray, N. (2005). Analyzing and modelling train driver performance. Applied ergonomics, Vol. 36, 2005, pp.671–680.

[3] Ouedraogo K.A.; Enjalbert, S. & Vanderhaegen, F. (2011). A state of the art n feedforward-feedback learning control systems for human errors prediction, Preprints of the 18th IFAC World Congress. Milano, Italy, 2011.

[4] Berdjag, D.; Vanderhagen, F.; Shumsky, A. & Zhirabok, A. (2014). Unexpected situation diagnosis: a model-based approach for human-machine systems, Proceedings of 19th IFAC Congress, Cape Town, South Africa, 2014, pp. 3545-3550.

[5] Berdjag, D.; Vanderhaegen, F.; Shumsky, A. & Zhirabok, A. (2015). Abnormal operation diagnosis in human-machine systems, Proceedings of 10th Asian Control Conference, Kota-Kinabalu, Malayzia, 2015.

[6] Hartmanis, J. & Stearns, R. (1966). The algebraic structure theory of sequential machines. N. Y.: Prentice-Hall Inc. 1966, 211 p.

[7] IT infrastructure library version 3, office of Government commerce (UK), Available from: https://www.axelos.com/best-practice-solutions/itil, Accessed: 2016-01- 26.

[8] Patton, R. (1994). Robust model-based fault diagnosis: the state of the art, Proceedings of IFAC symposium Safeprocess, Espoo, Finland, 1994, pp.1-24.

[9] Zhirabok A.; Filaretov V. & Pavlov S. (2015) Fault Isolation in Technical Systems by Data-Driven Method, CD ROM Proceedings of 26th International DAAAM Symposium, Zadar, Croatia, 2015.

[10] Zhirabok, A. & Shumsky, A. (2012). Fault accommodation in discrete-event robots, Advances in Robotics (Edited by: C. Ciufudean), iConcept Press, Hong Kong, 2012, pp.323-339.