Deadlock-free IIoT Controlled Systems

CALIN CIUFUDEAN and CORNELIU BUZDUGA
Faculty of Electrical Engineering and Computers Science
University Stefan cel Mare
University str. 13, Suceava
ROMANIA
calin@eed.usv.ro and cbuzduga@eed.usv.ro

Abstract: - This paper deals with a new issue encountered in industry, e.g. the bottlenecks of industrial internet of things` (IIoT) automation devices, such as flexible manufacturing systems (FMS). We model such systems with Petri nets and we impose a restrictive policy in order to reject bottlenecks in the system. Deadlocks in automations driven through IIoT applications occur mainly when two or more devices try to access the same resource. In order to solve these issues we use Petri net models which ensure reversibility of the net, respectively to grant the recoverability property of the modelled system. We deal with reversibility of a net not necessary to find returning routes from bottlenecks, but also by finding alternative routes for avoiding deadlocks in the IIoT controlled system. We emphasize our approach with an example, and we propose a general policy for these systems in order to ensure their vivacity.

Key-Words: - Industrial Internet of things, flexible manufacturing systems, Petri net, reversibility, vivacity.

1 Introduction

Nowadays Internet has technical capabilities for controlling industrial machines, and further expectations evolved through a novel kind of human-machine interactivity. Cloud data, modelling and predicting efficient methods of maintenance and control, and many new ways, e.g. new intelligent throughput roadmap, are now on the way to be extensively implemented on a new generation of industrial machine [1]. For controlling flexible manufacturing line by IIoT we need specifications for the controlled system, in order to ensure it with a high degree of free choice decisions, as we have to deal with highly interactive, highly connected platforms designed with flexibility and versatility. They are user friendly and remotely accessible, and nowadays they are known as "human-connected machines." For these systems the main goal for accomplishing a required throughput is to synthesise the controller for a specific technological algorithm, which has a certain operation schedule that determines the characteristic of the technological process, i.e. the specific allocation of resources. The lack of clear priorities of resources allocation in a FMS may lead to a deadlock situation. As FMSs are by excellence resource-shared systems, bottleneck and deadlocks issues must be considered and fixed starting from the design phase.

IIoT on security work already is reported to determine impacts unexpected and serious systemic effects by increasing the risk of diminished resilience. For example, Google and Facebook are using IIoT to limit certain kinds of content, and that can be an issue for computers, which do not understand cultural norms in the population [2-4]. Although exemplifying the advantages of IIoT driven products we cite: "With an Industry 4.0-enabled factory, Harley-Davidson can build 1,700 bike variations on one production line and ship an individualized bike approximately every 90 seconds. At the same time, the company has brought costs down 7%, increased net margin by 19%, and slashed the locked schedule to build a bike from 21 days to 6 hours" [5].

Although the IIoT systems delivers information in real time about the production flow, dealing with a bottleneck or a deadlock situation requires time and effort, i.e. the desired production throughput might be unaccomplished, or it might be finally accomplished with supplementary costs. As it is well known prophylaxis is better than cure. For us, in this work, prophylaxis is synonym with a well design model of the IIoT controlled production line. We propose a discrete event approach, i.e. Petri nets model and simulation with restrictive policies for avoiding deadlock in the system. For our Petri net approach this issue can be solved by ensuring the reversibility of the net, respectively to grant the recoverability property of the modelled system. We deal with reversibility of a net not necessary to find returning routes from bottlenecks, but also by finding alternative routes for avoiding deadlocks in the IIoT controlled system.

For dealing with the deadlock avoidance in FMSs we remember the works [6,7]. In [8] there is proposed an approach using un-timed PN for avoiding deadlock in complex manufacturing systems. Also, [9,11] deal with the bottleneck issue introducing siphons in the PN model, and [12] proposed a type of Petri net modelling concurrency between technological process flows. We mention that, stochastic models are used also for solving deadlock issue, especially where different concurrent operations of a FMS are down. Dealing with these situations involves Markov chain models studied in literature with a Markov chain as breakdowns in an industrial system respect the lows of distribution characteristic for stochastic process [13, 14].

2 Modeling deadlock avoidance for shared resource systems

Our model of an automation manufacturing system controlled through IIoT involves resources: $vB_{1,\,B}vB_{2,\,B}\ldots, v_{m\,B}$and finite products: w_{1B}, $wB_{2,\,..B}$, wB_{n}. $_{T}$We assume that scheduling an IIoT FMS is based on sequences of resources processing; for example $s(wB_{iB})$ is the sequence necessary to complete technological process for product type wB_{iB}. In order to simplify our approach we assume that we deal with four automatic machines MB_{1B}, MB_{2B}, $MB_{3\,B}$and MB_{4B} , an automated guided vehicle (AGV) system TS, two types of finite products wB_{1B}, wB2, input and output buffers with capacity of five for each machine: $IBB_{1,\,B}\ldots$, IBB_{4}, respectively OBB_{1B}, …, OBB4. Technological process implies the following handling schedule of FMS`s resources; machines MB_{3B}, MB_{1B}, MB_{2B}, MB1 delivers finite product $w_{1,B}$ B and MB_{2B}, MB_{3B}, MB_{4B} delivers finite product wB_{2B}: $s(wB_{1B})$ = (IBB_{3B}, OBB_{3B}, IBB_{1B}, OBB_{1B}, IBB_{2B}, OBB_{2B}, IBB_{1B}, OBB_{1B}), and $s(w_{2B})$ = (IBB_{2B}, OBB_{2B}, IBB_{3B}, OBB_{3B}, IBB_{4B}, OBB_{4B}).

In the PN transitions firing model activity of AGV, while tokens in place p_{0i} mark the unfinished products type w_i, and tokens in places p_{ni} model the number of finite products w_i. LSB_i is the schedule row finite product w_i. Places p_{1i}, p_{2i}, p_{LSi} of the Petri net model are modelling the schedule of technological process, and each one requires a specific resource. A place a_{ii} models the buffer type places for each type of resource i, where i=1, …, m; where the initial marking is $m_0(a_i) = v_i$. Tokens in places $aB_{i\,B}$show availability of resources i, and we have the initial marking $mB_{0B}(aB_i) = A_{iB}$, where AB_{iB} is the capacity of resources of type i.

The Petri net model of our FMS is depicted in Fig.1.

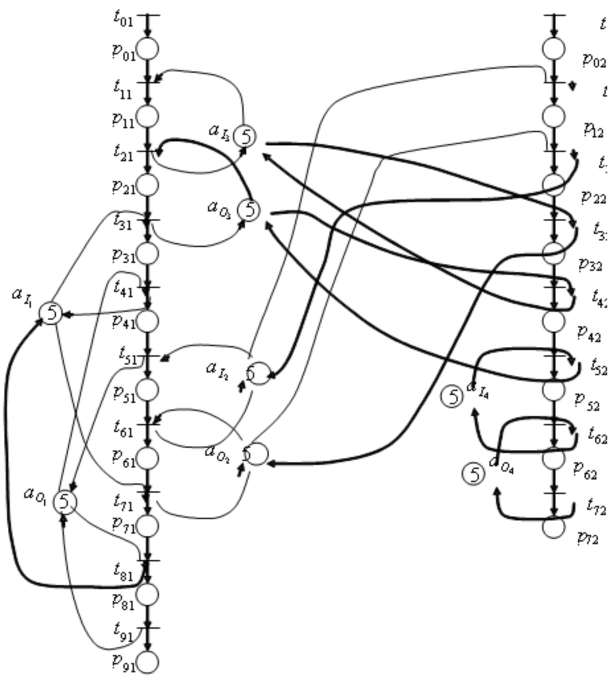

Fig. 1 Example of an IIoT controlled FMS modelled with Petri net

Petri net model of an IIoT controlled FMS has machines` places ($P_p$), the resources` places (P_r), the input/output buffers (IB)/(OB), and the transitions (T), as follows:

$$P_p = \{p_{ji}, i = 1,2,...,n,\ j = 0,1,..., LS_i + 1\} \quad (1)$$

$$P_r = \{a_{vi}, i = 1,2,...,m\} \quad (2)$$

$$T = \{t_{ij}, i = 1,2,...,n,\ j = 0,1,...LS_i B + 1\} \quad (3)$$

$$IB = \{(p_1 ji, t_1(j+1i)), i = 1,2,...n,\ j = 0,1,... \\ ...,[(LS)]_{\downarrow} i\} \quad (4) \\ \{(av(p_{1ji})Bt_{1ji}), i = 1,2,..n,\ j = 1,...[(LS)]_{\downarrow} iB\}$$

$$OB = \{(t_1 ji, p_1 ji, i = 1,2,...n,\ j = 0,1,... \\ ...,[(LS)]_{\downarrow} i + 1B\} \quad (5) \\ \{(t_{j+i}, aBv(p_{ji})B), i = 1,2,...n,\ j = 1,...LS_i\}$$

The initial marking $mB_{0B}(p_i) = A_{iB}$, where $pi_i \in \{aB_{v_i B}, i = 1, 2, \dots, n\}$, otherwise $mB_{0B}(p_i) = 0$.
In Fig. 1, we have: i = 1, 2; j = 1, 2, …, 9; $A_{iB} = 20$.
Performing the technological process it is possible to encounter a bottleneck or a deadlock situation, and we exemplify such situations using the following marking of places in Fig.1:

$$m(p) = \begin{cases} 5, \; for \; p \in \{p_{31}, p_{41}, p_{51}, p_{61}, a_{I3}, \\ , a_{I4}, a_{O3}, a_{O3}\} \\ 0, otherwise \end{cases} \quad (6)$$

this marking transitions t_{41}, t_{51}, t_{61}, and t_{71} cannot fire and the Petri net encounter a deadlock. We adopt the following definition of deadlock [15, 16]: transitions TD are in deadlock situation if and only if there are met both the following topological and marking conditions:

1) Each transition in TD has input process place enabled by marking m, and
2) Each transition in TD with input resource place has no valid marking.

Also, we have the following observation derived from the above given definition: when two sets of transitions TD_1 and TD_2 are deadlocked by marking m, then the set of transitions $TD_1 \cup TD_2$ is deadlocked by marking m.

In [8] is proved that for avoiding net`s blocking we will restrict the number of tokens in every deadlocking marking; e.g. the number of tokens in places which belong to deadlock marking PD should not be greater than $\sum_{t \in R(TD)} C_r - 1$.

Following this assumption, for the Petri net in Fig. 1, we will exemplify a method to avoid deadlock. The deadlocking topology and marking of the left side PN depicted in Fig. 1 are the following ones:

$TD_1 = \{t_{41}, t_{51}, t_{61}, t_{71}\}$,
$TD_2 = \{t_{51}, t_{61}, t_{71}, t_{81}\}$,
$TD_3 = TD_1 \cup TD_2 = \{t_{41}, t_{51}, t_{61}, t_{71}, t_{81}\}$.

Since $TD_1 \subseteq TD_3$, $TD_2 \subseteq TD_3$, the deadlock avoidance policy will restrict the tokens in the places of TD_3, as follows:

$$\sum_{r \in R(D_3)} C_r - 1 = (C_{I_1} + C_{I_2} + C_{O_1} + C_{O_{21}}) - 1 = 19 \quad (7)$$

Let us consider the following marking m:

$$m(p) = \begin{cases} 5, if \;\; p \in \{p_{31}, p_{41}, p_{51}, p_{61}, a_{I3}, a_{O3}\} \\ 0, else \end{cases} \quad (8)$$

We remember that transitions t_{41}, t_{51}, t_{61}, t_{71} are in deadlock, and $\{t_{41}, t_{51}, t_{61}, t_{71}\} = TD_1$ is a bottleneck structure and $TD_1 \subset TD_3$. Therefore, marking m from relation (8) cannot provide an efficient, because there are maximum 19 tokens which belong to the entrance places in transition, respectively $TD_3 = \{p_{31}, p_{41}, p_{51}, p_{61}, p_{71}\}$. The 19 resources represent the maximum number of tokens (e.g. resources of FMS) which can enter in places of TD_3 in order to avoid deadlock. The Petri net structure for avoiding deadlock is depicted in Fig. 2.

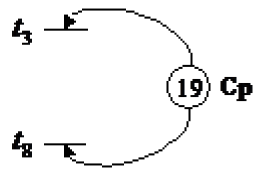

Fig. 2 PN basic structure for deadlock avoidance policy

Considering the Petri net from Fig. 1, following this judgment, we have the next deadlock structures:
$TDB_{0B} = \{t_{41}, t_{51}, t_{61}, t_{71}\}$,
$TDB_{1B} = \{t_{51}, t_{61}, t_{71}, t_{81}\}$,
$TDB_{2B} = \{t_{41}, t_{51}, t_{61}, t_{71}, t_{22}\}$,
$TDB_{3B} = \{t_{51}, t_{61}, t_{71}, t_{81}, t_{22}\}$,
$TDB_{4B} = \{t_{21}, t_{31}, t_{41}, t_{51}, t_{22}, t_{32}\}$,
$TDB_{5B} = \{t_{31}, t_{41}, t_{51}, t_{61}, t_{32}, t_{42}\}$,
as well as unions of this structures.
The PN basic deadlock structure of the Petri net depicted in Fig. 1 is given in Fig. 3.

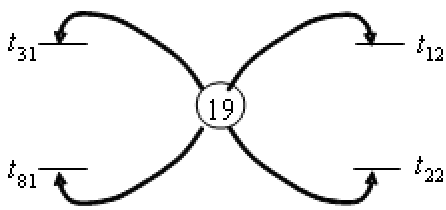

Fig. 3 The PN structure avoiding bottleneck for the net in Fig. 1

3 IIoT`s stochastics controllers

The promises and warnings of IIoT are not virtual but new technologies make them more pressing. One warning concerns the efficient scheduling of limited resources in order to ensure the required throughput of the driven systems and to avoid communication bottlenecks and production breakdowns. Therefore, faster and cheaper diagnostics and maintenance when FMS`s equipment fails, combined with real time machine debugging on the control and power buses will determine significantly improved systems` assembly and manufacturing times. It still remains the impossibility to predict unpredictable breakdowns of the IIoT or FMS equipment. This breakdown deals with a stochastic environment and therefore has to be analyzed with a proper formalism, e.g. has to be analyzed using Markov chains models and by assigning a firing probability to all transition in the net. Let us exemplify our approach with one IIoT controlled machine which may be in either of its two states: "functional" and "failure". We denote the probability of being functional with s, and the probability the machine is out of order (failure) we denote it by r. The only possible states of the machine are the following ones included in set X: X = {available, failure}, and the state transition matrix is T_X, where the elementary Markov chain modeling the our exemplifying system is depicted in Fig. 4:

$$T_X = \begin{bmatrix} s & 1-s \\ 1-r & r \end{bmatrix} \tag{9}$$

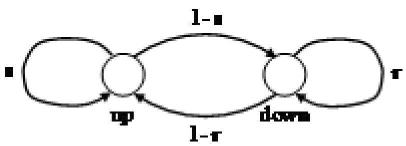

Fig. 4 Markov chain model of an IIoT controlled machine

The transition matrix is fully determined by the probability of being functional, and the probability of failure [17]. For the functional state, probability *s* is characterized by an increasing probability of failure, respectively by a decreasing probability of being functional. Oppositely, for the failure state, probability *r* displays a low probability of failure, which is independent of the probability of functioning, i.e the machine is in breakdown. Let $m \in [0,1]^n$ be row vector; and let have a Markov chain and its transition`s matrix $TM \in [0,1]^{n \times n}$, then we say that the Markov chain is safe for marking m if and only if the probability vector stays limited by m , i. e. $\Pi_0 \cdot TM^k \leq m$, where all $k \geq 0$ [18]. Let $\pi_m = \{\pi \in \Pi / \pi \leq m\}$ be the set of safe states given by probability distribution vectors. For the Petri net in Fig. 4 we impose that the machine`s probability of failure is less than 25 %; therefore m = [0.25,1], where $m_1 = 1$ means that the probability of being functional is high, and $m_2 = 0.25$ means that the probability of being in the failure state is 25%. Now we underline a condition for TM_A maintaining safe the state probability distribution of the controlled Markov chain [12]:

$$\pi_0 \cdot TM_A \in \pi_m \tag{10}$$

So we calculate *s* and *r* which satisfy relation (10), where $\pi_0 \in \pi_m$. For our example, we have as initial condition s ≥ 1-r, and the controller with state matrix established by relation (9) will deliver:

$$(s \geq 1-r) \cap (3s-r) \geq 2 \tag{11}$$

These equations are valid for the next state probability relation:

$$\pi_m = \left\{ \pi \leq \left[\frac{1,1}{4} \right] \right\} \tag{12}$$

We refer to the PN model above discussed, where we proposed following condition, related to expression (10): 1 ≤ Number of tokens of a vivace Petri net ≤ $\sum_{t \in R(TD)} C_r - 1$. We deal now with the information traffic as a main component of an IIoT controlled system, and we propose an approach for avoiding bottlenecks and deadlocks in such systems. As it is well known classic control system, as well as an IIoT control system is responsible for sequencing the machine through its different operational sub-states [19]. The difference is the velocity and accuracy of this control, and here is the reason why one may transfer classic control to IIoT one.

Therefore, we propose a modular system modelled with Markov chains for deadlock avoidance which starts with a basic cell of the IIoT depicted in Fig. 5:

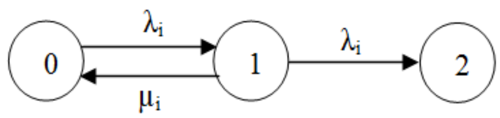

Fig. 5 The basic cell of Markov chain of an IIoT controlled system

In Fig. 5 the server is idle, respectively the state is in state 0 when there are no information to process, the server is active, respectively the server is in state 1 when we have a bi-directional transfer of the information to/from machines of FMS, and a deadlock occurs in state 2. Transfer rate of data is λ_i and the servers processing rate of data is μ_i [19]. By connecting basic cells we obtain a Markov chain model of the IIoT model as displayed in Fig. 6.

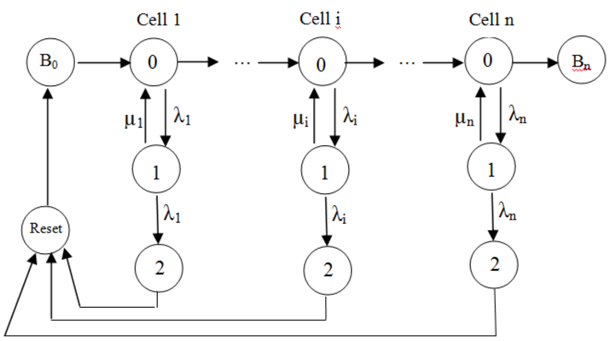

Fig. 6 Markov chain model of IIoT controlled FMS

In [19] we proved that the deadlock probability $p_{02}(t)$ is given by the following relation:

$$p_{02}(t) = A + B \cdot e^{-at} + C \cdot e^{-bt} \qquad (13)$$

Where, a, b, c, A, B, C have the following expressions:

$$\frac{\mu_i + \sqrt{\mu_i^2 + 4\lambda_i\mu_i}}{2};$$

$$b = \frac{2\lambda_i + \mu_i - \sqrt{\mu_i^2 + 4}}{2} \qquad (15)$$

$$B = \frac{\lambda_i(b-2a)}{ab(b-a)};\ C = \frac{\lambda_i}{b(b-a)} \qquad (16)$$

4 Conclusion

Industrial work on the flow philosophy imposes development of new control and decision-making tools. The SOM`s approach essentially turns highly complex subsystem design challenges into much simpler off-the-shelf subsystem integration challenges. We adopt this philosophy and we model it using a discrete event framework, i.e. Petri nets and Markov chains models [20, 21].

New demands for higher and higher throughputs determine more stresses applied to structural components, we mention here among many others the interaction man-machine, that lead to premature failure of the components and shorter service life for the manufacturing systems. Therefore, modelling and simulating production`s schedule in order to avoid the work`s on the flow bottlenecks we see it as mandatory in order to achieve higher performance using new technologies for control and monitoring the industrial manufacturing systems. Here, we proposed a strategy, and other related policies and modelling formalisms may extend the proposed one.

References:

[1]. *Ciufudean, C., Filote, C.*, Safety discrete event models for Holonic cyclic manufacturing systems in Holonic and Multi-Agent Systems for Manufacturing, *Springer Berlin Heidelberg*. 2009, pp. 225-233.

[2]. Harrell, C., The Internet of Things and Control System Architecture, http://blog.aac. advantech.com/the-internet-of-things-and-control-system-architecture, 2014.

[3]. Dolin, R., Building an IoT for industrial control: Part 1 – What is Industrial IoT? ,http://www.embedded. com/design/real-world-applications/ 4426952/Building an IoT for industrial control. Part 1 What is Industrial IoT?, 2015.

[4]. Storey, H., Bullotta, R., Drolet, D., The industrial internet of things, http://www.controleng.com/industry-news/single-article/the-industrial-internet-of-things/, 2014.

[5]. Vermesan, O., Friess, P., (Eds.),. Internet of Things – From Research and Innovation to Market Deployment, *River Publishers, Aalborg*, Denmark, 2014.

[6]. Minoura, C., Ding, A., deadlock prevention method for a sequence controller for

manufacturing, *Int. J. Robotics Automat*., vol. 6, 1991, no. 3, pp.234-240.

[7]. Buzacott, J., A., Shantikumar, J., G., Stochastic Models of Manufacturing System, Englewood Cliffs, NJ: *Prentice Hall*. 1993.

[8]. S.P.Sethi, .Zhon, "Hierarchical production controls in a stochastic two machine flow-shop with a finite interval buffer", *IEEE Trans. Robotics Automat.*, vol. 13, 1997, no. 1, pp.112-116.

[9]. Narahari, J., Viswandham, N., Transient Analysis of Manufacturing System Performance, *IEEE Trans on Rob and Autom*., vol. 10, 1994, no. 2.

[10]. Ciufudean, C., Satco, B., Algebraic formalism for modelling the deadlock in flexible manufacturing systems, Jurnal of Applied Mathematics, vol.1, no.3, 2008, pp. 157-165.

[11]. J.Ezpeleta, J.M.Colom, J.Martinez, "A Petri net baqsed deadlock prevention policy for FMS", *IEEE Trans. Robotics Automat*., vol.11, no. 2, 1995, pp.232-240.

[12]. Dallery, J., Gershwin, S., B., Manufacturing flow line systems: A review of models and analytical results, Technical Report 91-002, 1992, *Laboratory for Manufacturing and Productivity*, MIT.

[13]. Martinelli, F., Shu, C., Perkins, J.R., On the Optimality of Myopic Productions Controls for Single-Server Continuous-Flow Manufacturing Systems, *IEEE Trans. Autom. Contr*., vol.46, no.8, 2001, pp.1269-1273.

[14]. Di Benedetto, M., D., Vintecentelli, A., S., Villa, T., Model Matching for Finite State Machines, *IEEE Trans. Autom. Contr*.,vol.46, no.11, 2001, pp.1726-1743.

[15]. K.Y.Xing, B.S.Hu, H.X.Chen, "Deadlock avoidance policy for PN modeling with shared resources", *IEEE Trans. Robotics Automat*., vol. 41, no. 2, 1996, pp.302-308.

[16]. Ciufudean, C., Filote, C., Amarandei, D., Measuring the Performance of Distributed Systems with Discrete Event Formalisms, *Proc. of The 2nd Seminar for Advanced Industrial Control Applications, SAICA*, 2007, Madrid, Spain.

[17]. Ciufudean, C., Graur, A., Filote, C., Turcu, C., Popa, V., Diagnosis of complex systems using ant colony decision petri nets, Availability, *Reliability and Security, ARES* 2006, The First International Conference on, Vienna, Austria.

[18]. Ciufudean, C., Satco, B., Filote, C., Reliability Markov chains for security data transmitter analysis, *Availability, Reliability and Security. ARES 2007*. The Second International Conference on, pp. 886-894.

[19]. Ciufudean, C., Buzduga, C., "*Chapter 7 Modelling the Diagnosis of Industry Internet of Things*", Springer Nature, 2017.

[20]. Ciufudean, C., Larionescu, A.B., "Estimation of the Performances of the Discrete Event Systems", *Advances in Electrical and Computer Engineering*, vol.3(10), no.2(20), 2003, pp.30-35, University of Suceava, Romania.

[21]. Ciufudean, C., Popescu, D., "Modelling Digital Signal Perturbations with Stochastic Petri Nets", *Advances in Electrical and Computer Engineering*, vol.4(11), no.1(21), 2004, pp.71-75, University of Suceava, Romania.