Internet of Things for Flexible Manufacturing Systems` Diagnosis

Calin Ciufudean and Corneliu Buzduga
Stefan cel Mare University, 13 University, Suceava, Romania

Keywords: Internet of Things, Flexible Manufacturing System, Discrete Event Model, Memory Buffer, Markov Chain.

Abstract: This paper deals with an actual topic concerning the diagnosis of Internet of Things (IoT) controlled flexible manufacturing systems (FMS). We focus on models realized with Markov chains of FMS with stochastic and not equal throughput rates. Discrete-event models assume that FMS is decomposed, and we study the following events: an Internet server fails, an Internet server is repaired, an Internet server memory buffer fills up, an Internet server memory buffer empties. The IoT diagnosis is performed with by calculating the time to absorption in Markov model of the IoT controlled FMS. Future development of IoT diagnosis of FMS are also discussed in this work.

1 INTRODUCTION

In this work, we assume that a flexible manufacturing system controlled and monitored by Internet of Things (IoT) is similar to a discrete event system (DES) and we model it in a discrete stochastic space.

Absorbing states of Markov chain models display a steady-state i.e., the absorbing state attended after time T; therefore, only transient analysis displays the system performance. Our approach deals with an IoT controlled system which displays in time a trajectory modelled with a Markov chain $\{x(t); t \geq 0\}$ with state space $S = \{0, 1, ...\}$ and space generator W. Let i, j $\in$ S and, we have (Viswandham, 1992), (Kemeny, 1960):

$$p_{ij}(t) = P\{x(t) = j / x(0) = i\} \quad (1)$$

$$A(t) = [p_{ij}(t)] \quad (2)$$

The following equations describe the behavior of the above mentioned Markov chain (Buzacott, 1993), (Narahari, 1994), (Ciufudean, 2008), (Viswandham, 1994):

$$\frac{d}{dt}[A(t)] = A(t) \cdot W \quad (3)$$

$$\frac{d}{dt}[A(t)]^* = W \cdot A(t) \quad (4)$$

Where A(0) = I. For matrix components we have:

$$\frac{d}{dt}[p_{ij}(t)] = w_{ij} \cdot p_{ij}(t) + \sum_{k=j} w_{kj} \cdot p_{ik}(t) \quad (5)$$

$$\frac{d}{dt}[p_{ij}(t)] = w_{ii} \cdot p_{ij}(t) + \sum_{k=i} w_{ik} \cdot p_{kj}(t) \quad (6)$$

The solution is:

$$A(t) = e^{W \cdot t} \quad (7)$$

$$e^{W \cdot t} = \sum_{k=0}^{\infty} \frac{(W \cdot t)^k}{k!} \quad (8)$$

The state probabilities $Y(t) = [p_0(t), p_1(t), ...]$ where $p_j(t) = P\{x(t) = j\}$, $j \in S$, are given by the following equation:

$$\frac{d}{dt}[Y(t)] = Y(t) \cdot W \quad (9)$$

The solution is:

$$Y(t) = Y(0) \cdot e^{W \cdot t} \quad (10)$$

$$p_{ij}(t) = P\{X(t) = j / X(0) = i\} \quad (11)$$

For t > 0, and T the time to reach the absorbing state, we obtain:

$$P\{T > t\} = P\{X(t) \notin (m+1, ..., m+n)\} \quad (12)$$

Where $m \geq 0$, $n > 0$, we have (m+1) states, and the next states are absorbing ones.

$$P\{T > t\} = 1 - \sum_{j=1}^{n} p_{0,m+j}(t) \quad (13)$$

Ciufudean, C. and Buzduga, C.
Internet of Things for Flexible Manufacturing Systems' Diagnosis.
In *Proceedings of the 5th International Conference on Smart Cities and Green ICT Systems (SMARTGREENS 2016)*, pages 468-471
ISBN: 978-989-758-184-7

Then time interval T may be displayed by:

$$F_T(t) = \sum_{j=1}^{n} p_{0,m+j}(t) \quad (14)$$

Where $p_{0,m_{+j}}(t)$ is given by equation (3) (Ciufudean, 2008), (Viswandham, 1994), (Dallery, 1992).

2 THE MODEL FOR IOT DIAGNOSIS OF FMS

The basic cell of the IoT system diagnosis of a FMS consists of a computer e.g. server connected to Internet, S_i, with memory buffer and its downstream machine from the FMS. In figure 1 we depicted the Markov chain model of the one of the n identical cells of our model for IoT control and diagnosis of a FMS, where n represents the number of servers necessary to control the FMS (Ciufudean, 2008).

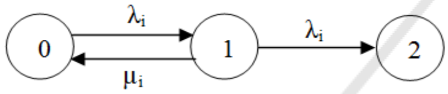

Figure 1: The basic model of Markov chain for IoT diagnosis.

The meaning of the cell depicted in figure 1 is that the server S_i is in state 0 when there is no information to process, and there is the transfer of the information to/from machines of FMS. In state 1, we process information, and a deadlock occurs in state 2. Information bits transfer rate is λ_i and the servers processing rate of information bits is μ_i. The Markov chain model for IoT diagnosis of FMS is depicted in figure 2.

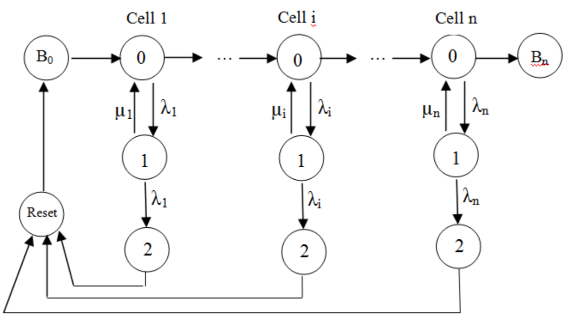

Figure 2: Markov chain model for IoT diagnosis of FMS.

Here T is the time elapses until deadlock occurs, and deadlock is a probability $Da_T(t) = p_{02}(t)$. In order to determine $p_{02}(t)$ we will use the generator W of the above depicted Markov chain:

$$W = \begin{bmatrix} -\lambda_i & \lambda_i & 0 \\ \mu_i & -(\lambda_i + \mu_i) & \lambda_i \\ 0 & 0 & 0 \end{bmatrix} \quad (15)$$

From equation (6) we have the probability $p_{02}(t)$:

$$\frac{d}{dt} p_{02}(t) = w_{00} \cdot p_{02}(t) + w_{01} \cdot p_{12}(t) + w_{02} \cdot p_{22}(t) \quad (16)$$

But we also have $w_{02} = 0$, and therefore:

$$\frac{d}{dt} p_{02}(t) = -\lambda_i \cdot p_{02}(t) + \lambda_i \cdot p_{12}(t) \quad (17)$$

Similar, for $p_{12}(t)$ we have:

$$\frac{d}{dt} p_{12}(t) = w_{10} \cdot p_{02}(t) + w_{11} \cdot p_{12}(t) + w_{12} \cdot p_{22}(t) \quad (18)$$

And $p_{22}(t) = 1$, and therefore:

$$\frac{d}{dt} p_{12}(t) = -\mu_i \cdot p_{02}(t) - (\lambda_i + \mu_i) \cdot p_{12}(t) + \lambda_i \quad (19)$$

Where $p_{ij}(s)$ is the Laplace transform of $p_{ij}(t)$:

$$sp_{02}(s) = -\lambda_i p_{02}(s) + \lambda_i p_{12}(s) \quad (20)$$

$$sp_{12}(s) = \mu_i p_{02}(s) - (\lambda_i + \mu_i) \cdot p_{12}(s) + \frac{\lambda_i}{s} \quad (21)$$

Equations (20) and (21) determine:

$$p_{02}(s) = \frac{\lambda_i^2}{s\left[s^2 + s(2\lambda_i + \mu_i) + \lambda_i^2\right]} \quad (22)$$

And equations (22) have as solution the probability $p_{02}(t)$:

$$p_{02}(t) = A + B \cdot e^{-at} + C \cdot e^{-bt} \quad (23)$$

Where, a, b, A, B, C are [2, 3]:

$$a = \frac{2\lambda_i + \mu_i + \sqrt{\mu_i^2 + 4\lambda_i\mu_i}}{2} \quad (24)$$

$$b = \frac{2\lambda_i + \mu_i - \sqrt{\mu_i^2 + 4\lambda_i\mu_i}}{2} \quad (25)$$

$$A = \frac{\lambda_i}{ab}; B = \frac{\lambda_i(b - 2a)}{ab(b - a)}; C = \frac{\lambda_i}{b(b - a)} \quad (26)$$

3 THE EVENTS OF MEMORY BUFFERS

For components manufactured in FMS, the transition from one event to next event depends on current state and on the generator W of the FMS. So, we may say that in a FMS controlled by IoT deadlocks have mainly two possibilities of diagnosis: a blocked server empties its memory or information less (e.g. empty server) commands its downstream machine. Therefore the events dynamic is determined by information which flow both way from S_i to the downstream machine. We consider a FMS controlled by servers S_{i-1}, S_i and S_{i+1}, and the memory buffers B_{i-1} and B_i.

We assume that an event occurs at time t and let TA be the apparent time of the next event. We have:

μ_i is the information processing speed (bits-unit/time-unit) of server S_i, i = 1, ... , n.

$$S(i,\ t) = \begin{cases} 1, \text{ if server } S_i \text{ is functional} \\ 0, \text{ if } S_i \text{ is under repair} \end{cases}$$

$$B(j,t) = \begin{cases} 0, \text{if buffer } B_j \text{ is empty} \\ 2, \text{if buffer } B_j \text{ is full} \\ 1, \text{otherwise state} \end{cases}$$

$$BEj(t) = \begin{cases} 1, \text{if } B_j \text{ empties at time } t \\ 0, \text{otherwise} \end{cases}$$

$T_{1j}(t)$ Time necessary to store information in B_j

$T_{2j}(t)$ Time necessary to deliver information from B_j

We have the following scenarios:

First scenario: server S_{i+1} is faster than S_{i-1}. This is modeled in figure 3 and we have (Ciufudean, 2008), (Viswandham, 1994), (Dallery, 1992):

$$(T_{21} > T_{1,i\text{-}1} + \frac{1}{\mu_i}) \cap (\mu_{i+1} > \mu_{i-1}) \tag{27}$$

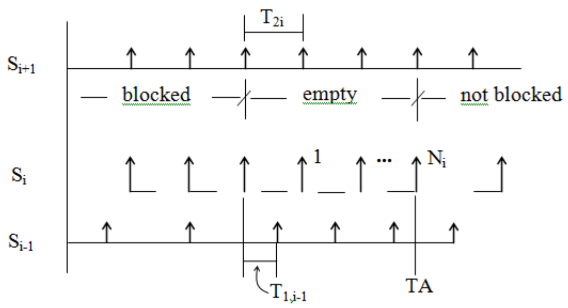

Figure 3: Dynamic of servers when S_{i+1} is faster than S_{i-1}.

In figure 3 and 4 we depicted with continuous line server data processing and with arrows we depicted data flow (Ciufudean, 2008), (Viswandham, 1994), (Dallery, 1992). Intervals blank mark the idle processing time due to blockage/repair of servers. Memory buffer B_i empties from full memory. The end of processing time of the (N_i+1) bits on server S_i is greater than the time when memory B_i empties. The dual case is for the first N_i bits. Therefore we have (Martinelli, 2001):

$$t + T_{2i} + \frac{N_i}{\mu_{i+1}} < t + T_{1,\,i\text{-}1} + \frac{N_i}{\mu_{i-1}} + \frac{1}{\mu_i} \tag{28}$$

and

$$TA = t + T_{2i} + \frac{N_i - 1}{\mu_{i+1}} \geq t + T_{1,\,i\text{-}1} + \frac{N_i - 1}{\mu_{i-1}} + \frac{1}{\mu_i} \tag{29}$$

For B_i equations (27) and (28) estimate the bits to next event:

$$N_i = 1 + \mathrm{Int}\left\{ \frac{T_{2i} - T_{1,i-1} - \frac{1}{\mu_i}}{\frac{1}{\mu_{i-1}} - \frac{1}{\mu_{i+1}}} \right\} \tag{30}$$

Another scenario studied here is dual to first discuss: server M_{i-1} process data faster than server S_{i+1} and the empty server S_i fills its memory buffer B_i (Di Benedetto, 2001), (Harrell, 2014), (Dolin, 2015), (Storey, 2014). After that, server S_{i-1} processes N_{i-1} bits, and blockage is modeled in figure 4.

$$\left(T_{2i} > T_{1,\,i\text{-}1} + \frac{1}{\mu_i}\right) \cap (\mu_{i-1} > \mu_{i+1}) \tag{31}$$

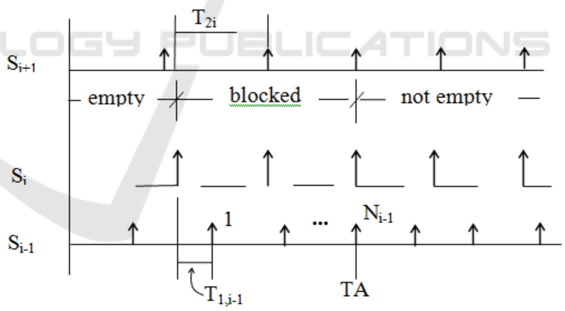

Figure 4: Dynamic of servers when S_{i-1} is faster than S_{i+1}.

Figure 4 shows that arrival time of $(N_{i-1} + 1)$ bits at buffer B_{i-1} is less than the processing time of server S_i. The dual case holds for the first N_{i-1} bits (Vermesan, 2014), (Ciufudean, 2009), (Ciufudean, 2007), (Ciufudean, 2006). Therefore we have:

$$t + T_{2i} + \frac{N_{i-1} - 1}{\mu_{i+1}} > t + T_{1,\,i\text{-}1} + \frac{N_{i-1}}{\mu_{i-1}} \tag{32}$$

and

$$TA = t + T_{2i} + \frac{N_{i-1} - 1}{\mu_{i+1}} \leq t + T_{1,\,i\text{-}1} + \frac{N_{i-1} - 1}{\mu_{i-1}} \tag{33}$$

Similar with the first scenario, for B_{i-1} we estimate the next event:

$$N_{i-1} = 1 + \mathrm{Int}\left\{\frac{T_{1,i-1} - T_{2i} - \frac{1}{\mu_{i+1}}}{\frac{1}{\mu_{i+1}} - \frac{1}{\mu_{i-1}}}\right\} \qquad (34)$$

Equations (29) and (33) allow us to avoid the above mentioned scenarios of deadlocks by fairly dimensioning the buffers, and taking into consideration flow rate of bits until next event: $T_{21} = p_{02}$ in relation (29) and, respectively, $T_{1,i-1} = p_{02}$ in relation (33); where p_{02} is given by relation (23) (Ciufudean, 2008), (Ciufudean, 2007).

As we proved in this paper the failure/blocking of servers can be avoided, if the buffer size is bigger than the critical size (e.g. the size determined with equations (30), (33), (34). The necessary and sufficient condition is to have an average time to repair a server smaller than the average time to fill the memory of server.

4 CONCLUSIONS

A model for IoT diagnosis of a FMS diagnosis has been proposed in this paper. The model may be obtained with our discrete-event approach or using heuristic models.

A discrete-event system formulation and FMS controlled by IoT connected by processing cells and fast determines an accurate diagnosis at an increased speed and costless. We observe that if the deadlock/repair time is known and the duration of diagnosis estimation is less than it, then transient analysis is more appropriate than the steady state analysis.

Further development of this approach should focus on intelligent flexible manufacturing systems modeled with Markov chains which have self-recovery algorithms from deadlock situations.

REFERENCES

Viswandham, N., Narahari, J., 1992. *Performance Modeling of Automated Manufacturing System*, Englewood Cliffs, NJ: Prentice Hall.

Kemeny, J., Snell, W., 1960. *Finite Markov Chains*, Van Nostrand.

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

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

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

Viswandham, N., Ram, R., 1994. Composite performance-dependability analysis of cellular manufacturing systems, *IEEE on Rob and Autom., vol. 10*, no. 2.

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

Martinelli, F., Shu, C., Perkins, J.R. 2001. On the Optimality of Myopic Productions Controls for Single-Server Continuous-Flow Manufacturing Systems, *IEEE Trans. Autom. Contr., vol.46*, no.8, pp.1269-1273, 200.

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

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

Dolin, R., 2015, 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?

Storey, H., Bullotta, R., Drolet, D. 2014, The industrial internet of things ,http://www.controleng.com/industry-news/single-article/the-industrial-internet-of-things/c98837a0efec387df9fc14c2de0a3b2f.ht,.

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

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

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

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

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