

# Analysis of Flexible Manufacturing System using Petri Nets to design a Deadlock Prevention Policy

Shashank Satish More<sup>1</sup>  
Student of M.E. (MECH-PROD)  
K.I.T's College Of Engineering  
Kolhapur, India

Dr. S. G. Bhatwadekar<sup>2</sup>  
Professor  
Dept of Production Engg.  
K.I.T's College Of Engineering  
Kolhapur, India

**Abstract**—This paper illustrates the Petri net modeling of a conceptual model of flexible manufacturing system (FMS) and addresses the problem of deadlock by establishing deadlock prevention policy. The first part introduces the problem of deadlock in FMS, deadlock handling strategies and basic concepts of petri net modeling. Literature review is presented in the second part and third part shows an illustration of Petri net modeling of a conceptual model of FMS processing two part types, consisting of three machines, a load/unload station and single robots. It is assumed that each of the part has three machining operations to be performed. Reachability graph is then generated for this model and is used to detect the possible deadlock states. Finally deadlock prevention policy is established for this conceptual FMS model based on reachability graph analysis.

**Keywords**—Deadlock, FMS, Petri net, Reachability graph, deadlock prevention.

## I. INTRODUCTION

Manufacturers must adapt to changes in the production environment as well as in the market in order to achieve and maintain competitiveness. Effectively designing and operating an automated manufacturing system (AMS) is important for manufacturers to reach this goal. An AMS conglomerates of machine tools, robots, buffers, fixtures, automated guided vehicles (AGVs), and other material-handling devices. Different types of parts enter the system at discrete points of time and are processed concurrently; these parts cause a high degree of resource sharing [11].

It is a difficult to predict the behaviour of manufacturing systems without modelling, analyzing, and control techniques. Therefore, several techniques have been developed to describe the behaviour of manufacturing systems. One of which are Petri nets (PNs). Petri nets are a powerful graphical tool for modeling and analyzing concurrent, parallel, simultaneous, synchronous, distributed, and resource sharing systems. There are many advantages of using Petri nets such as enable an easy visualization of complex systems, can model a system hierarchically and can analyze qualitative and quantitative aspects of the system, qualitative analysis searches for structural properties like the absence of deadlocks, the absence of overflows or the presence of certain mutual exclusions in case of resource sharing. Quantitative analysis looks for performance properties such as throughput, utilization rates, average queue lengths, or average completion times.

### A. Deadlocks In FMS

Deadlock is a state where a set of parts are in “Circular waiting” situation, i.e., a situation where a set of two or more processes keep waiting indefinitely for other processes in the same set to release resources. But in context of FMS resources refer to machines, buffers, AGVs, robots, fixtures etc. and process refer to operations to be performed on the raw material at the work centers.

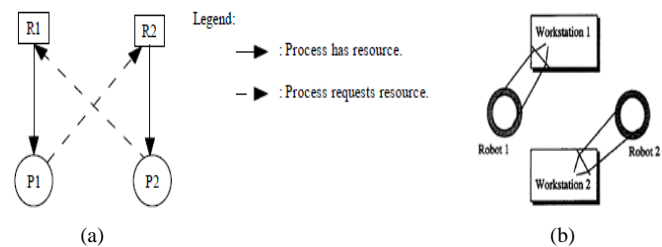


Figure 1: A typical deadlock scenario.

Fig. 1 shows an example of a typical deadlock scenario. The deadlock situation can disrupt the entire system and makes the automated system stop working. To recover from deadlock one has to shutdown the system where deadlock has occurred and forcibly remove the part which is being processed and then restart the system in order to return to normalcy. Thus occurrence of deadlock causes unpredictably long down-time and causes under utilization of resources and may also lead to catastrophic results in safety-critical systems. Deadlocks in a FMS are in general considered to be a result of: [9]

- Deficiency of system resources (Excessive sharing of resources).
- An inappropriate order of process execution, and
- Improper resource allocation logic.

There are four necessary conditions for a deadlock to occur [12], known as the Coffman conditions:

- Mutual exclusion condition: a resource can only get involved in one process at a time.
- Hold and wait condition: processes (operation) already holding resources may request new resources;
- No preemption condition: no resource can be forcibly released from a process holding it, when processing is under progress and resources can be released only by the explicit action of the process;

- d) Circular wait condition: two or more processes form a circular chain where each process waits for a resource that the next process in the chain holds.

A deadlock will never occur if one of these conditions is not satisfied. The physical characteristics and technical background of an FMS which is a discrete event system, show that the first three deadlock conditions always hold and any method which guarantees that the fourth condition (the circular wait) does not hold, is a valid solution to tackle problem of deadlock in FMS.

### B. Deadlocks Handling Strategies

There are three major approaches to manage a deadlock situation [12] namely deadlock prevention, deadlock avoidance, and deadlock detection and recovery. Deadlock prevention refers, to static resource allocation policies to eliminate completely the possibility of deadlock occurrence. This approach is too much conservative and leads to underutilization of system. Deadlock avoidance involves dynamic resource allocation, so as to avoid deadlock just before it occurs. This approach is widely accepted because it leads to better utilization of resources than prevention policies.

In deadlock detection and recovery, however, resources are granted to a process without any check. A deadlock detection algorithm determines whether a set of processes is deadlocked. If a deadlock is found, the system recovers from it by aborting one or more deadlocked processes. A good deadlock detection algorithm is the one, which detects all possible deadlocks and does not report nonexistent deadlocks.

### C. Petri Nets as a Modeling Tool

Petri nets are bipartite directed graphs, in which there are two types of vertices known as places and transitions. The places are represented by hollow circle and transitions are represented by vertical bar. Places usually represent a state of the system, conditions or a resource and transitions represent events. A token is a small solid dot (or solid circle), presence of token in a place means the condition represented by the place is satisfied or a resource represented by the place is available. Places are connected to transitions by directed arcs and they represent input and output function.

A firing of transition indicate occurrence of an event represented by it and transition is said to be enabled for firing, if all the input places of a transition are having tokens in them, in other words an event can occur only if all the conditions are satisfied or all the resources required for an event are available. Petri nets, as graphical and mathematical tools, provide a uniform environment for modeling, analysis and design of discrete event systems [13]. It is their simplicity and ability to model sequential execution, concurrency, conflict, synchronization and merging like situations which makes them suitable to model FMS.

## II. LITERATURE REVIEW

Narahari and Viswanadham (1985) [1] present an approach for modelling and analyzing flexible manufacturing systems (FMSs) using Petri nets. In this approach, they first build a (Petri net model ~PNM) of the given FMS in a bottom-up fashion and then analyze important qualitative aspects of FMS behaviour such as existence/absence of deadlocks and buffer overflows. They illustrate their approach using two typical manufacturing systems: an automated transfer line and a simple FMS. Author also discusses how Petri net invariants are useful in showing absence of deadlocks in a given FMS. Wu and Zhuang (1995) [4] study that Flexible Manufacturing Systems (FMS's), belong to the class of discrete event dynamic systems. In such a system, potentially conflicting events may occur due to concurrency. The problem of collision and deadlock avoidance can be investigated using Petri nets which are powerful techniques suitable for modeling concurrent processes. A Petri net based approach is employed in this paper to model, detect and avoid collisions and deadlocks in FMS's. An approach for deadlock and collision avoidance in a flexible manufacturing system has been proposed in this paper.

Lawley et al. (1998) [5] describes Configuration flexibility, the ability to quickly modify manufacturing system components and their logical relationships, requires automatic generation of control executables from high level system specifications. These control executables must guarantee deadlock-free operation. The objective of their research is to develop rapidly configurable control policies that guarantee deadlock free FMS operation. In this paper, authors presented the Resource Order Policy; a correct, configurable, and scalable Deadlock avoidance policy. Li et al. (2012) [9] investigate research surveys the state-of-the-art deadlock control strategies for automated manufacturing systems by reviewing the principles and techniques involved in preventing, avoiding, and detecting deadlocks. The primary focus of this research is deadlock prevention due to a large and continuing stream of efforts on it. A control strategy is evaluated in terms of computational complexity, behavioral permissiveness, and structural complexity of its liveness-enforcing or deadlock-free supervisor.

Abdul-Hussin (2014) [10] describes that the deadlock prevention method is caused by the unmarked siphons, during the Petri nets are an effective way to model, analyze, simulation and control deadlocks in FMS is present in this work. Author says the deadlock prevention method are cause by the unmarked siphons, during the Petri nets are an effective way to model, analyze, simulation and control deadlocks in FMS is present in this work. Deadlocks can be avoided by adding a control place,

associate with the arcs to each emptyable siphon to prevent it from being unmark. Kaid et al. (2015) [11] demonstrate a comprehensive overview for applications of Petri nets and their extensions in modeling, analyzing, and control of manufacturing systems are present. More than 25 major production, manufacturing, operations management, and control journals published in years 1988–2015 has been review. The survey is classify into two fields, applications Petri nets with modeling and analyzing and applications Petri nets with control, each field is classify into three groups, and additionally a historical progression in these fields was emphasize. Authors find that most usage of Petri net is in control application specifically in deadlock prevention due to the large and continuing stream of efforts.

Richard Zurawski et al. (1995) [13] presented a tutorial paper on Petri Nets and Industrial Applications, an overview of applications of Petri nets in industry with a description of Petri nets properties, and analysis methods and further deals with modeling, formal analysis, and design of discrete event systems.

From the literature review one can infer that most of the works addressing deadlock problem using Petri nets are based on structural analysis of Petri nets and few works are based on reachability graph analysis and are restricted to very small systems. Structural methods eliminate the derivation of the state space, so they avoid the “state explosion” problem, but they cannot provide as much information as the reachability approach does.

### III. PETRI NET MODELING OF A CONCEPTUAL MODEL OF FMS

A conceptual model of FMS consisting of four machines, two robots processing two part types is considered for illustration using Petri net. The Fig. 2 shows the schematic layout of the conceptual FMS model.

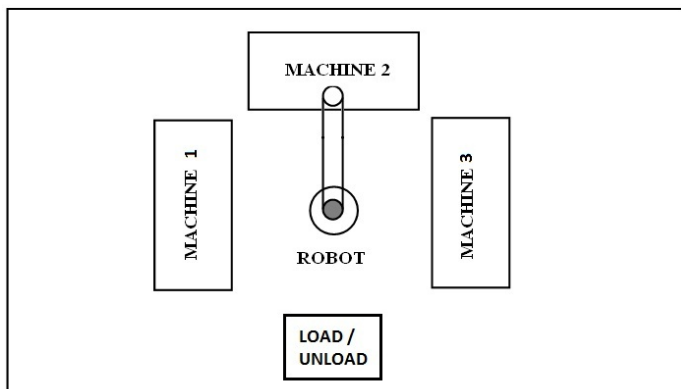


Figure 2: Conceptual model of FMS processing two parts A and B. Part A having sequence M1, M3, M2 and part B having sequence M2, M1, M3.

Each of the two machines in the above manufacturing cell is assumed to process one part at a time. The robot is meant to load/unload the parts on/from the machines and load/unload station; it is assumed that robot can handle one part at a time. Also it is assumed that the raw materials are always available in the load/unload station but once a particular part type is under processing the system doesn't start processing the same type of new part. For example when processing of part A is under progress till the completion of all three machining operation of part A. The system will not accept raw material of A to start machining new part A. The figure 3 shows the Petri net model of the conceptual model of the manufacturing system. The Petri net model has been developed using PIPE v4.3 software [14]. The places are used to represents the resources (robots, machines) and state of process. Whereas firing of transitions represents the occurrence of activities or events. The table 1 describes each of the places and transition.

TABLE I. Definition of Places and Transition in Petri Net model

PLACES	
ROBOT	: Robot is available
Place M1,M2,M3:	Machine 1,2,3 are available respectively
Place A	: Raw material A is Available
Place 1	: Robot is loading raw material A on machine 1
Place 2	: Machining operation of machine 1 is under progress
Place 3	: Robot is loading raw material A on machine 3
Place 4	: Machining operation of machine 3 is under progress
Place 5	: Robot is loading raw material A on machine 2
Place 6	: Machining operation of machine 2 is under progress
Place B	: Raw material B is Available
Place 7	: Robot is loading raw material B on machine 2
Place 8	: Machining operation of machine 2 is under progress
Place 9	: Robot is loading raw material B on machine 1
Place 10	: Machining operation of machine 1 is under progress
Place 11	: Robot is loading raw material B on machine 3
Place 12	: Machining operation of machine 3 is under progress
TRANSITIONS	
T1	: Robot starts loading raw material A on machine 1
T2	: Machine 1 starts machining raw material A
T3	: Machining is over; robot unloads machine 1 and loads part A on machine 3
T4	: Machine 3 starts machining part A
T5	: Machining is over, robot unloads machine 3 and loads part A on machine 2
T6	: Machine 2 starts machining part A
T7	: All machining operation on part A are over and machine 2 is unloading
T8	: Robot starts loading raw material B on machine 2
T9	: Machine 2 starts machining part B
T10	: Machining is over, robot unloads machine 2 and loads part B on machine 1
T11	: Machine 1 starts machining part B
T12	: Machining is over, robot unloads machine 1 and loads part B on machine 3
T13	: Machine 3 starts machining part B
T14	: All machining operation on part B are over and machine 3 is unloading

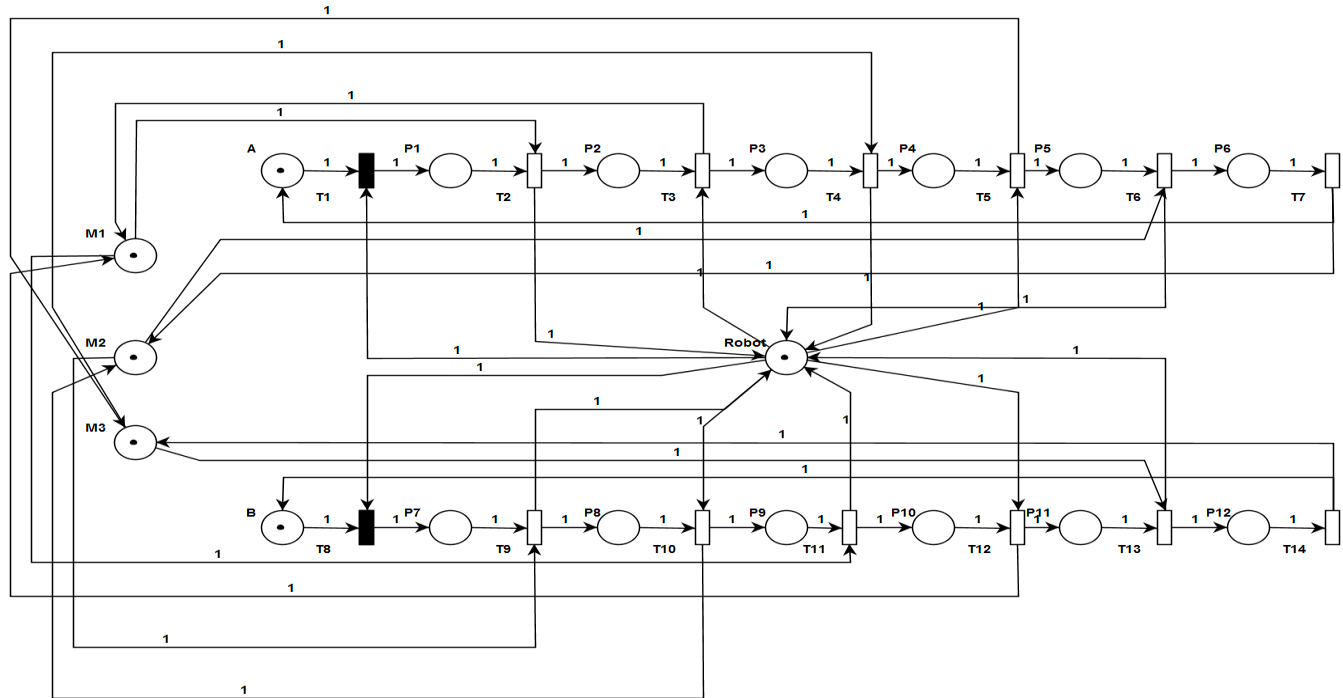


Figure 3: Petri Net model of the conceptual model of FMS

### A. Establishing Deadlock Prevention Policy

The Fig. 5 shows the reachability graph of Petri net model of FMC. The first step in establishing deadlock prevention policy is to detect the possible deadlock states. In this work the reachability graph is produced by using PIPE v4.3 software. From the reachability graph shown in figure 4, one can easily detect four such deadlock states namely, S9, S11, S15, S19. The corresponding critical states are S7, S10, S13, S16. Deadlock prevention policy is when system reaches critical state, avoid ring those transitions which leads to deadlock states. For example when the system is in state S7, two transitions T3 and T10 are enabled. If ring of T10 occurs, then system reaches deadlock state S9.

Hence our deadlock prevention policy would be to avoid ring T10. The physical interpretation of this is, when system is at state S7, actually both M1 and M2 are busy in machining. Firing of T10 indicates, Robot unloads machine M2 and starts loading on M1 since machining at M2 is over. Firing of T3 indicates, robot unloads machine M1 and starts loading on M3 since machining at M3 is over. Suppose ring of T10 occurs, i.e. if robots unloads machine M2 starts loading on M1, however M1 in is waiting for the same robot to unload it and since robot can handle only one part at a time the system will be deadlocked. Thus deadlock prevention policy would be at state S7, robot should always choose to unload machine M1, rather than

unloading M2. Table II shows the deadlock prevention policy for all possible deadlocks in the system. Fig. 4 shows basics of petri net model.

The first column in the table shows one possible sequence of transition ring for a particular deadlock state; however there can be many possible paths or sequence of transition ring which will lead to the same deadlock state. A state in a reachability graph is represented by a vector having elements equal to number of places and the value of each element corresponds to the number of tokens held by respective places at that instant. Places are in the order {A, B, M1, M2, M3, P1, P10, P11, P12, P2, P3, P4, P5, P6, P7, P8, P9, Robot}.



(a) Representation of Petri Net elements



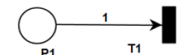
(b) Machine available



(c) Machine not available



(e) Buffer with 5 parts stored in it



(d) P1 is input place for transition T1

Figure 4: Representation of Petri nets

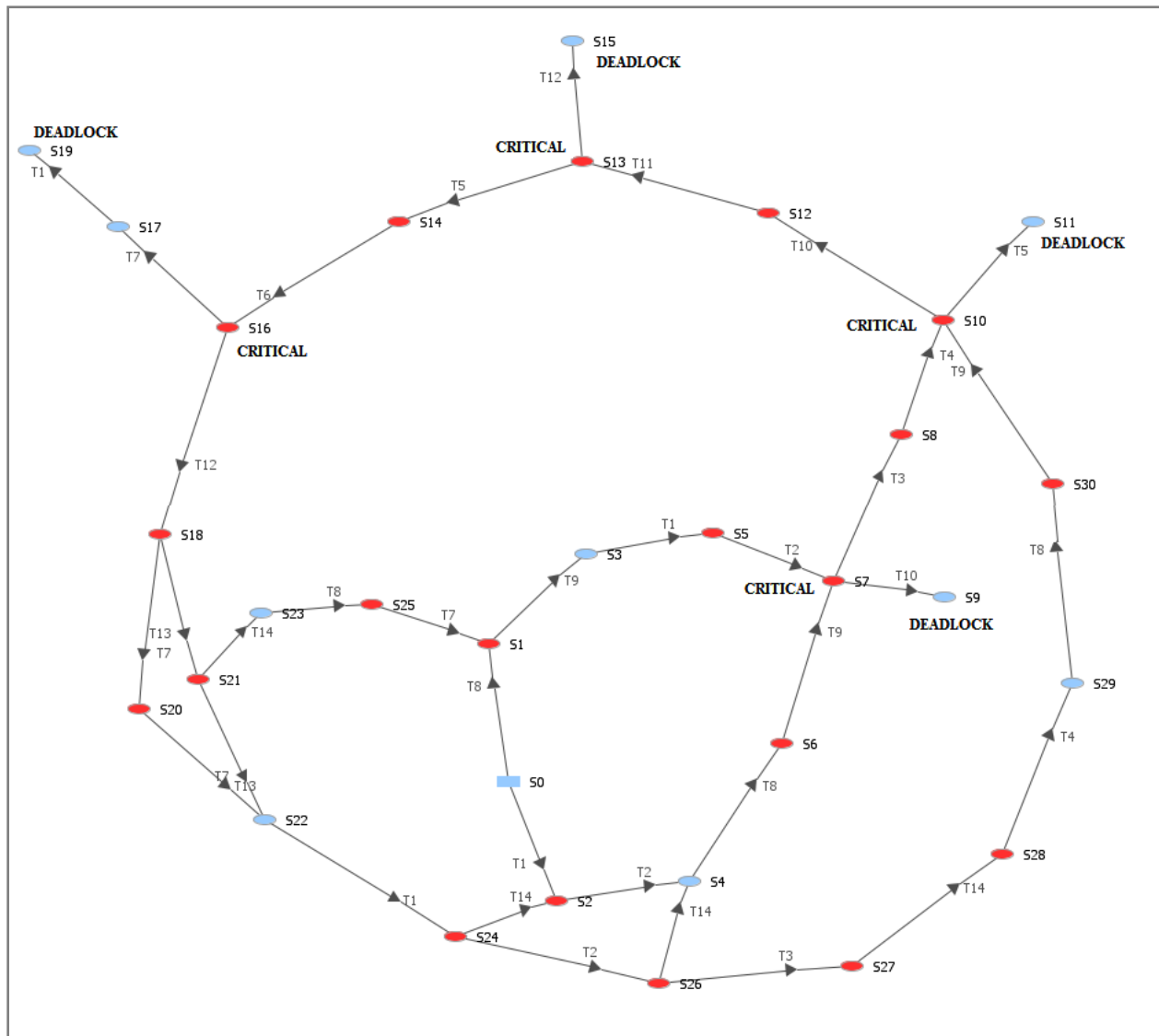


Figure 5: Reachability graph of Petri net model of FMS

TABLE II. Deadlock Prevention Policy for FMS

Sequence of transition firing	Deadlock state /Critical state	Deadlock Prevention Policy	Physical Interpretation
T8,T9,T1,T2, T10	S9/S7	At S7, do not fire T10 instead fire T3	When system is at state S7 there may be a situation to choose between 1. Robot assigned to unload M1 and 2. Robot assigned to unload M2 Deadlock prevention policy is, at this state in order to prevent deadlock always choose option 1.
T8,T9,T1,T2, T3, T4,T5	S11/S10	At S7, do not fire T5 instead fire T10	When system is at state S10 there may be a situation to choose between 1. Robot assigned to unload M2 and 2. Robot assigned to unload M3 Deadlock prevention policy is, at this state in order

			to prevent deadlock always choose option 1.
T8,T9,T1,T2, T3,T4,T10,T11,T12	S15/S13	At S13, do not fire T12 instead fire T5	When system is at state S13 there may be a situation to choose between 1. Robot assigned to unload M3 and 2. Robot assigned to unload M1 Deadlock prevention policy is, at this state in order to prevent deadlock always choose option 1.
T8,T9,T1,T2, T3,T4,T10, T11,T5,T6,T7	S19/S16	At S16, do not fire T7 instead fire T12	When system is at state S16 there may be a situation to choose between 1. Robot assigned to unload M1 and 2. Robot assigned to unload M2 Deadlock prevention policy is, at this state in order to prevent deadlock always choose option 1.

#### IV. CONCLUSION

The paper presented an overview of the deadlock problem in FMS and Petri net as a modeling tool. A conceptual model of FMS having three machines, single robot processing two product types was analyzed for deadlock using reachability graph of Petri net model. Finally deadlock prevention policy was established which gives deadlock free resource allocation method. This work illustrates that the reachability graph technique can be effectively used for analyzing and establishing a deadlock prevention policy for an FMS.

#### REFERENCES

- [1] Y. Narahari and N. Viswanadham, "A petri net approach to the modelling and analysis Of flexible manufacturing systems", Analysis of Operations Research, Vol.3, 1985, pp. 449-472.
- [2] Tadao Murata, "Petri Nets: Properties, Analysis and Applications", IEEE, Vol.77/4, 1989, pp. 541-580.
- [3] Y. Narahari and N. Viswanadham et al., "Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models", IEEE transactions on robotics and automation, Vol.6, 1990, pp. 713-723.
- [4] Jie Wu, Hanqi Zhuang, "A petri-net-based collision and deadlock Avoidance scheme for FMS", IEEE 0-7803-2535-4, 1995, pp. 511-520.
- [5] Mark A. Lawley, Spyros A. Reveliotis, and Placid M. Ferreira, "A Correct and Scalable Deadlock Avoidance Policy for Flexible Manufacturing Systems", IEEE transactions on robotics and automation, Vol.14, 1998, pp. 796-809.
- [6] Sridhar Mohan, Ali Yalcin, Suresh Khator, "Controller design and performance evaluation for deadlock avoidance in automated flexible manufacturing cells", Elsevier Ltd. 0736-5845, 2004, pp. 1-11.
- [7] Sridhar Mohan, "Deadlock avoidance in mixed capacity flexible manufacturing systems", Graduate Theses and Dissertations 1164, 2004, pp. 1-69.
- [8] Farooq Ahmad, Hejiao Huang and Xiao-long Wang, "Analysis of Parallel Manufacturing Processes with Resource Sharing", International Journal of Computer Theory and Engineering, Vol.2/2, 2010, pp. 250-257.
- [9] Zhiwu Li, Naiqi Wu and Mengchu Zhou, "Deadlock Control of Automated Manufacturing Systems Based on Petri Nets-A Literature Review", IEEE Transactions on Systems Man and Cybernetics Part C 2011.2160626, 2012, pp. 1-50.
- [10] Mowafak Hassan Abdul-Hussin, "Petri Nets approach to simulate and control of Flexible Manufacturing Systems", International Journal on Software Engineering, Vol.1, 2014, pp. 1-10.
- [11] Husam Kaid, Abdulaziz M. El-Tamimi, Emad Abouel Nasr, and Abdulrahman Al-Ahmari, "Applications of Petri nets Based Models in-Manufacturing Systems: A Review", International Conference on Operations Excellence and Service Engineering Orlando (IEOM Society), 2015, pp. 516-528.
- [12] Y. Narahari and N. Viswanadham, "Performance modeling of Automated manufacturing Systems", Prentice Hall of India, New Delhi. ISBN 81-203-0870-0.
- [13] Zurawski Richard, Zhou MengChu., "Petri Nets and Industrial Applications: A Tutorial", IEEE. transactions on industrial electronics, VOL. 41, NO. 6, December 1994, pp. 567-583.
- [14] Platform Independent Petri Net editor version 4.3 (PIPE v4.3) software is a freeware which can be downloaded at <http://space.dl.sourceforge.net/project/pipe2/PIPEv4/PIPEv4.3/PIPEv4.3.zip> <accessed on 08 Aug 2016>