Dynamic Scheduling of Manufacturing System with Stochastic Timed Petrinet: A Genetic Algorithm Approach

Jeetendra Vadher
javadher1@yahoo.com
Shantilal Shah Engineering College, Bhavnagar, Gujarat-India

M B Patel
mulchandbhaichand@yahoo.com
Government Engineering College, Modasa, Gujarat-India

Abstract

Dynamic Decision-making under the resource Breakdown - Repair at the various stages of manufacturing system such as loading scheduling and during processing. The present paper deals with how effectively Stochastic Petri net concept used for modeling of decision making real time manufacturing system. Genetic Algorithm is applied for stochastic time generation and Analysis. A Case study is discussed to understand the proposed concept and simulation of the system is carried out. The proposed work is developed using Object Oriented language and validated by existing problems.

Keywords: Stochastic Petri net, CP-Matrix, Dynamic Decision, Genetic Algorithm.

1. INTRODUCTION

The current highlighted interest in manufacturing is long overdue. It follows a period during which manufacturing was neglected, often being considered merely a necessary task. Because manufacturing has not been viewed as an intellectual challenge, its intellectual base not expanded as the products being manufactured have increasingly become more complex. The manufacturing enterprise is under going fundamental change; it both uses and produces sophisticated technology. It is as much software as it is a hardware system.

Decision making at the time of planning and design stages includes number and type of machine, time involved in performing operations, number of material handling devices and type of devices with cost involved, number of buffers and its availability, size of pallet pool and number of jigs and fixtures, best possible layouts, tool storage capacity, part type selection, machine grouping, batching, balancing, sequencing and scheduling priorities. During the operational stage of a manufacturing, performance modeling can help in making decisions related to finding the best routes in the event of breakdowns and selection of best possible way out of number alternatives, predicting the effect of adding or withdrawing resources and parts, obtaining optimal schedules in the event of machine failures or sudden changes in part mix or demands, and in avoiding unstable situations such as deadlocks.

Enhancements in latest developing tools have brought in a variety of applications. Petrinets is one of the important tools now a day for modeling, analysis and simulation of various systems. Petrinet have been introduced to by Carl Adam Petri in 1962. A. W. Holt's (1970) first demonstrated the capabilities of PN for modeling and analyzing systems with concurrent components. Petrinet is a graphical and mathematical modeling tool (Murata, 1989) that has been successfully applied in the areas of performance evaluation, communication protocol, legal system, and decision-making models (Brand, 1988 and Murata, 1989). Decision signal (Denial Tabak, 1985) models of routing de-multiplexers are added to the Petrinet formalism to represent internal decision-making in the model. High level Petrinet (Alast, 1994) concepts, tools and analysis methods of i.e. Petrinet extended with 'color', 'time' and 'hierarchy' can be used for the modeling and analysis of many complex systems encountered in industry, and also given some application prototyping of software, (re)design of logistic systems, (re)design of administrative organizations. Stochastic colored Petrinet model (Moore, Gupta, 1995) used for flexible manufacturing system and material handling systems and machining, they have transition firing time is exponentially distributed. Scheduling problems considered in the literature are often static (activities are known in advance and constraints are fixed). However, every real-life schedule is subject to unexpected events. In these cases, a new solution is needed in a preferably short time and as close as possible to the current solution. This part is also considered the paper. The paper (Fleury G., Lacomme P., and Sevaux M. 2004) addresses resolution of a robust scheduling problem arising at the French railways company. The objective is to reduce the immobilization of trains during the maintenance operations. The processing times of operations are submitted to hazards and the objective is to minimize both the total completion time (i.e. the make span) and to limit the random event consequences on solution quality. Their Solutions was robust when facing some variations of processing times. Due to periodic modifications in the composition of trains a computer aided design system is required to avoid periodic manual planning computation. The system developed was used for machine sequence of operations; make span evaluation and an evaluation of the processing time variation consequences. They have applied dedicated genetic algorithm for robust computations. Some of the researchers have applied genetic algorithm approach, (James C. Werner Mehmet E. Aydin Terence C.-2000) and addresses an attempt to evolve genetic algorithms by a particular genetic programming method to make it able to solve the classical Scheduling problem, which is a type of very well known hard combinatorial optimization problems. The aim is to look for a better GA such that solves scheduling with preferable scores. A genetic algorithm builds new sequences, (Jerker Bjorkqvist, 2005) by combining and mutating previous sequences of genes, i.e. chromosomes, into a new set of chromosomes. In this new set, only the fittest survive, and the procedure is repeated. As a schedule in chemical batch plant can be seen as a sequence of starting points for the batches, the methodology of genetic algorithms can be applied also to batch scheduling. (G. J. Tsinarakis, N. C. Tsourveloudis, 2005 and K. P. Valavanis, Vitali Volovoi-2004), addresses the dynamic modeling of degrading and repairable complex systems. Emphasis is placed on the convenience of modeling for the end user, with special attention being paid to the modeling part of a problem, which is considered to be decoupled from the choice of solution algorithms. Depending on the nature of the problem, these solution algorithms can include discrete event simulation or numerical solution of the differential equations that govern underlying stochastic processes. Such modularity allows a focus on the needs of system reliability modeling and tailoring of the modeling formalism accordingly. To this end, several salient features are chosen from the multitude of existing extensions of Petri nets, and a new concept of aging tokens (tokens with memory) is introduced. The above survey discuss about application of Petri net modeling, scheduling of system and genetic algorithm approach to make the system in real application. With this existing work the proposed work uses extension of Petri net for dynamic scheduling under the stochastic condition and genetic algorithm is combined, that helps in modeling, analysis and dynamic system for real life decision application. This paper is summarized in the following sections: section 2 Stochastic timed Petri net, section 3 Dynamic decision system modeling concept, section 4 modeling of time for transition, section 5 problem definition, section 6 is application of Genetic algorithm and dynamic scheduling, section 7 is CP-Matrix manipulation, section 8 conclusion and future scope with strong recommendation for dynamic situation for manufacturing system.

2. STOCHASTIC TIMED PETRI NET

When time delays are modelled random variables, or probabilistic distribution are added to the deterministic timed Petri net models for the conflict resolution, stochastic timed Petri net models are yielded. In such models, it has become a convention to associate time delays with the transition only. When random variables are of general distribution or both deterministic and random variables are involved, the resulting net models cannot be solved analytically for general cases. Thus simulation or approximation methods are required. The stochastic timed Petri nets in which the time delays for each transition is assumed to be stochastic and exponentially distributed are called stochastic Petrinets. Now from the above concept a S*tochastic timed Petri net* is a six tuple TPN = (*P, T, I, O, TS, D*) satisfying the following requirements:

(i) *P* is a finite set of *places*.

(ii) *T* is a finite set of *transitions*.

(iii) $I \in T \rightarrow P(P)$ is a function which defines the set of *input places* of each transition.

(iv) $O \in T \rightarrow P(P)$ is a function which defines the set of *output places* of each transition.

(v) *TS* is the *time set*.

(vi) $D \in T \rightarrow TS$ is a function which defines the *firing delay* of each transition.

In the next section the above concept is used for modelling with some advancement to encounter dynamic situation.

3. DYNAMIC DECISION SYSTEM MODELING

The transition firing rules applicable in the case of Stochastic timed Petrinet while simulating dynamic behavior of systems are:

(i) Activities of the manufacturing system are modeled by Transitions that is vertical bar, and stochastic firing time is associated with each transition. There will be input place and output place for each transition.

(ii) A token, represented by small filled circle inside the place shows dynamic status of the system. Whenever a transition completes its firing duration, it updates the relevant attributes of the tokens to it's outplace place, that is represented by circle. The transition is represented by T and place is represented by P. Fig. 1(a) and (b) shows that transition T1 with input place P1 and output place P2, after firing transition T1 deposits token to its output place P2. A token in place P1 indicates that the precedence conditions of T1 is satisfied.

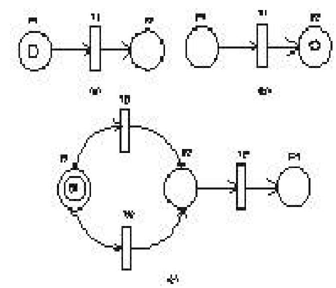

Figure 1: Basic Modeling Concepts

(iii) Decision nodes in the system are indicated by decision places, which are represented by two concentric circles. Fig.1(c) shows a decision place, and T1 and T2 are the two transitions emerging from this place. From any decision place only one transition will be enabled that depends on the token generated inside the circle with minimum time firing. Here stochastic time is considered using beta distribution. This concept is used for modeling Breakdown - Repair situation arrives. So from the figure 1(c) only one transition will fire.

(iv) A transition T is said to be enable if each input place P of T is marked with at least W (P, T) tokens, where W (P, T) is the weight of the arc from P to T.

(v) An enabled transition may or may not fire, that depend on whether the event actually takes place.

(vi) A firing of an enable transition T removes W (P, T) tokens from each input P of T, and adds W (T, P) tokens to each output place P of T where W (T, P) is the weight of the arc from T to P.

(vii) A transition without input place is called "Source Transition" and without output place is called" Sink Transition". A pair of place P and Transition T is called "Self Loop" if P is both an input and output place of T.

(viii) A net is said to be "Pure" if it does not have self-loops. A net is called "Ordinary" if all arcs have weight 1.

4. MODELING OF TIME FOR TRANSITION

For real systems it is often important to describe the *temporal behaviour* of the system, i.e. we need to model durations and delays. Since the classical Petri net is not easily capable of handling quantitative time, we add a timing concept. There are many ways to introduce time into the classical Petri net (Murata, 1989). In this work a timing mechanism is used where time is associated with tokens, and transitions determine delays (Murata, 1989, Jeetendra Vadher 2000). Each token has a *timestamp* which models the time the token becomes available for consumption. Since these timestamps indicate when tokens become available, a transition becomes enabled the earliest moment for which each of its input places contains a token which is available. The timestamp of a produced token is equal to the *firing time* plus the *firing delay* of the corresponding transition. In this work two types of timing can be assigned: one is deterministic discussed above and another is stochastic. The initial effort at simulating a Manufacturing System of a valid model of the distribution of transition times. Several probability density functions have been proposed for modelling non-deterministic activity times. These include the normal distribution, uniform distribution, lognormal distribution, and beta distribution (A. B. Badiru, 1990). The beta distribution is by far the most used for representing the probabilistic nature of activity times. To simulate stochastic situation we take a look at the statistical aspect of the Beta distribution. The standard equation for expected time is given by:

$$T\_Exp = (T\_Min+4T\_Most+T\_Max)/6$$

While the theoretical mean of the general beta distribution is given by

$$\mu y = T\_Min + (T\_Max - t\_Min) * (\alpha/\alpha+\beta)$$

Where α and β are the shape parameter of the Beta distribution. The value of α and β is given by this equation,

$$\alpha=\Phi\ \beta$$

$$\Phi = (5T\_Min - 4T\_Most - T\_Max)/(Tt\_Min+4Tt\_Most-5T\_Max);$$

$$\beta= - (\Phi 2 +2\Phi +1) / (\Phi+1)3$$

This is accomplished by letting

$$T\_Exp = T\_Min +(T\_Max - T\_Min)* x$$

Where x is the standard beta random variable between 0 and 1.this random numbers are used for stochastic time generation for transition.

5. PROBLEM DEFINITION

To demonstrate the methodology, we consider the case problem of the system as shown in figure 2. The manufacturing system consists of two machining tools (M1 and M2), two robot arms, and two conveyors. Each machining tool is serviced by a dedicated robot arm which performs load and unload tasks. One conveyor is used to transport work pieces, the other one is used to transport empty pallets. Each work piece is machined on M1 and M2, in this order with related time period. Suppose that the machining tool M1 is faster then M2, however subject to failures. M2 and the two robots are failure free. Given the processing rates of the machining tools and robots, as well as the failure and repair rates of M1, and the production rate of the system (throughput), assuming that only one palate is available. Figure 3 and 4 shows Petri net representation of the manufacturing system. Place represents precedence relationship and status of the system, Transition represents manufacturing systems activities and dot in place known as token which shows status of resources whether available or non available.

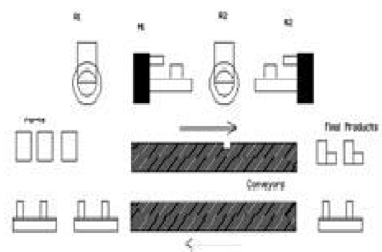

Figure 2: A Manufacturing System of two Machines, Two robots, and two conveyors

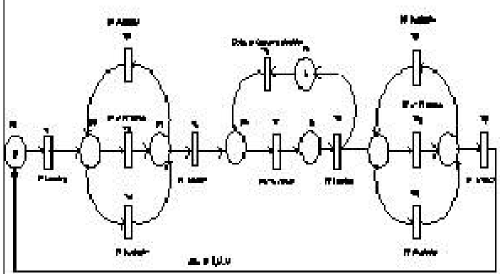

Figure 3: Petri Net Representation of a Manufacturing System

6. GENETIC ALGORITHM AND DYNAMIC SCHEDULING

GA approaches to the dynamic scheduling modeling problem are scarce. In this work GA is capable of solving the dynamic problem is described. It combines well known components adopted from previous research in the fields of Operations Research and Evolutionary Computation into a very efficient algorithm. First, the concept of GA is used here for the generation of number of iteration and best time for transition firing. Modeling of time is done with beta distribution, under which number of transition times are generated and simulated for 100 times and results are identified. GA is applied to code each time generated by random number with:

$$f(x) = T\_Exp = T\_Min + (T\_Max - T\_Min)$$

Dynamic scheduling of transitions under the circumstances like transition is not able to fire due to activity breakdown, failure in machine (maintenance), resource availability constraint, and transportation delay. Modeling of this situation is carried out with break down and repair transition. A special arc is provided that activates this transition and make able to do performance. The activation of this transition is done through new heuristic rule with modification in CP-Matrix. The modeling is shown in the fig. 4.

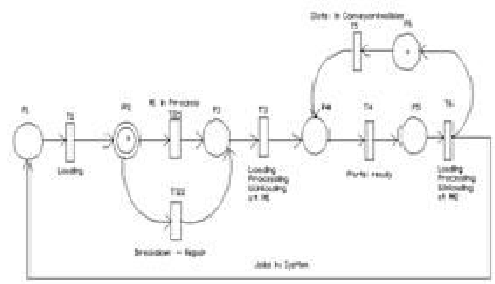

Figure 4: Stochastic Timed colored Petri net Model for the Manufacturing System

Transition Interpretations:

Transition	Interpretations
T1	Loading
T21	M1 in Process
T22	M1 in Breakdown-Repair status
T3	Loading Processing and Unloading at M1
T4	Parts Ready
T5	Slots in Conveyor available
T6	Loading Processing and Unloading at M2

Transition and stochastic firing time:

Transition	Stochastic time(Min)	
	Minimum	Maximum
T1	20	40
T21/T22	30/40	50/80
T3	10	15
T4	12	18
T5	0	0
T6	20	40

PRELIMINARY
PRELIMINARY

6. 1. Encoding:

The binary coding used to represent variable. In the calculation here, 7 bits are chosen for each variable, thereby making the total string length equal to 10. With 7 bits, accuracy designed here is based on following equation:

$$xi = T\_Min + (T\_Max - T\_Min)/2li - 1 \quad \text{decoded value } (si)$$

The operation like single point crossover and mutations are applied on it. The random population created using above beta distributation and initialize counter t=0.

6. 2 Fitness Function:

GAs mimics the survival of the fittest principle of nature to make a search process. A fitness function $F(x)$ is first derived from the objective function and used in successive genetic operations. The fitness function can be considered as same as objective function $F(x) = f(x)$. The following fitness function is used in this work:

$$F(x) = 1 / (1 + f(x))$$

The operation of GAs begins with a population of random strings representing design or decision variables. Therefore, each string is evaluated to find the fitness value. The population is then operated by three main operators – reproduction, crossover and mutation – to create a new population points. The new population is further evaluated and tested for termination. If the termination criterion is not met, the population is iteratively operated by the above three operators and evaluated. This procedure is continued until the termination criterion met. One cycle of these operations and the subsequent evaluation procedure is called generation in GA. In this work reproduction operator selects good strings and crossover operator recombines good substrings from good strings together to hopefully create a better sub string. The mutation operator alters a string locally to hopefully create a better string. The entire process is done with software, developed in object oriented paradigm.

6.3 GA Population:

Genetic population is tabled for transition 1 only, consider transition T1 firing timing are (20, 40). The string length is seven and generates random population of n chromosomes. Here ten different random numbers are chosen and represented in a binary form. The string number column represents the selected string based on random number. After this selection procedure is repeated n times, the number of selected copies for each string is counted, that is shown in true count in mating pool. In the next, the string in the mating pool are used in the cross over and mutation operation. The sample calculation is shown in the *Annexure 1*. In the next generation counter t=1 and proceed further that calculation are not shown here.

7. CP-MATRIX MANIPULATION

In the P-matrix an element '1' at the position (i, j) indicates that activity in j^{th} column is a predecessor to the activity in i^{th} row. A transition is said to be enabled when all its preceding transitions are completed and a token is generated in its input place. This can be identified by the row sum of the P-matrix. All those transitions for which row sum is zero are enabled ones. After an enabled transition in i^{th} row completes its firing '1' is placed in the position (i, i), and the precedence constraints ('1') in the column of that i^{th} transition are removed enabling its successors. This process is repeated until all the elements of the leading diagonal get '1's, indicating that the project is completed.

Proposed work expose the colored token concept for the modeling of the manufacturing system network during rescheduling type situations arrives. For each activity after completion, a color token named 'Green' is generated and deposited to output places after firing completes at specified time duration associated with it. A green token in a place is the indication of the completion of the preceding activities without any breakdowns, accidents or delay.

The activity network is encountered with breakdown, delays, accidents or any problem occur during its firing, under this situation a red token generated in its output place is the indication of delay in the completion of the preceding activities. Under this condition the network modeled with decision place, from the decision place there will be choice, which transition will be activated. Figure 4 indicates that there will be two path choices, whether machine M1 is ready or in repair, according to that, decision takes place. If machine is ready then it will activate transition T_2, and if it is not ready due to problem then it has to pass through Break down and repair transition and token is generated in P_4 Place. After repair it will generate 'Green' token to place P_2 and then transition T_2 is activated.

Rule:

1) Row sum = ∑Tij = 0, Transitions are enabled and token is generated after Completing

2) Row sum = ∑Tij ≥ 1, Transitions are enabled and may fire as per precedence rule or require to reschedule and Green or Red token is generated

3) If = ∑Tii = 1, Transitions have completed firing (diagonal element).

The rescheduled manufacturing system is shown with green token is shown in *Annexure 2*. The token movement according to CP-Matrix manipulation is highlighted in the Annexure table by a, b, c, d, e, and f.

8. CONCLUSION AND FUTURE SCOPE

This work proves the power of Petrinets for modeling the rescheduling of the Manufacturing system. Petrinets are emerging as a powerful tool for modeling and analysis of many realistic systems. The proposed work PNRS-Net is developed in Object Oriented Paradigm and validated by using different case study problems from existing literature and modeling capability has been proved. In the above work dynamism of manufacturing system is established using Petri net extension called stochastic colored Petrinets. Genetic algorithm is applied for time generation and optimum time modeling with beta distribution. This proposed PNRS-Net is tested, validated and following points are identified with case problem discussed above:

i. No dead locks occurs as long as token exist
ii. The net is K-bounded
iii. Conflicts are resolved according to system net states.
iv. Net can also applied under deterministic condition with some modification.
v. Resource analysis with multimode is also possible
vi. Substitution, Preemption may be modeled.

REFERENCES:

A, Seidmann (1978) "Online Scheduling of Robotic manufacturing cell with stochastic sequence-department processing rates" International Journal of Production Research, Vol. 25, No. 6, pp. 907-924.

A. Seidmann (1988) "Optimal Dynamic Routing in Flexible Manufacturing System with Limited Buffers" Annals of Operations Research, 15, pp. 291-311.

Adedeji B. Badiru (1991) "Starc 2.0 : An Improved PERT Network Simulation Tool", Computers and Industrial Engineering, Vol. 20, No. 3, pp. 389 - 400.

Ann E Gray, A. Seidmann, Kathryn E Stecke (1993) "A Synthesis of Decision Models for tool Management in Automated Manufacturing" Management Science, Vol. 39, No. 5, pp. 549-67.

Carman Veronica, Bobeanu, Eugene J H Kerckhoffs, and Hendrik Van Landeghem (2004) "Modeling of Discrete Event Systems: A Holistic and Incremental Approach using Petri Nets" ACM Transactions on Modeling and Computer Simulation, Vol. 14, No. 4, pp. 389-423.

Chrisian B, Dirk C, M, (1999). "Production Scheduling and Re-Scheduling with Genetic Algorithms" Evolutionary Computation 7(1), Massachusetts Institute of Technology, pp. 1-17

Cohen, G., Moller, P., Quadrat, J. –P., and voit, M. (1989) "Algebraic tools for the performance evaluation of discrete event systems" Proceedings of the IEEE 77(1): pp. 99-109

Daniel Tabak, Alexander H. Levis (1985) "Petrinet Representation of Decision Models" IEEE Transaction on Systems, Man, and Cybernetics, Vol. SMC - 15, No. 6, pp. 812 -818.

PRELIMINARY

Desrochers, A. A., & Al-Jaar, R.Y. (1995) "Application of Petri nets in manufacturing systems: modeling, control, and performance analysis". New York: IEEE Press.

Dong-Soon Yim, Thomas A. Barta "A Petrinet Based Simulation Tool for the Design and Analysis of Flexible Manufacturing Systems" Journal of Manufacturing Systems, Vol. 13, No. 4, pp. 251 – 261.

Edward Willims, Ramu Narayanaswamy (1999), "Application of simulation to scheduling, Sequencing, and material handling", Proceeding of the 1999 winter simulation conference, New Albany, USA.

Fleury G., Lacomme P., and Sevaux M. (2004), "Stochastic Maintenance Scheduling Problem", Proceeding of nineth international conference on Project Management & Scheduling, PMS 2004, Nancy France, Apr, 26-28, pp. 405-409.

G. J. Tsinarakis, N. C. Tsourveloudis, and K. P. Valavanis (2005). "Petri Net Modeling of Routing and Operation Flexibility in Production Systems" Cape forum 2005, Romania cluj-Napoca feb. pp. 352-357.

James c Werner, Mehmet E Avdin, Terece C Fogart (2000) "Evolving genetic algorithm for job shop scheduling problems", Proceeding of ACDM 2000, PEDC, University of Plymouth, UK.

James C. Werner Mehmet E. Aydin Terence C (2000). "Evolving genetic algorithm for Job Shop Scheduling problems Fogarty School of Computing, Information Systems and Mathematics" Proceedings of ACDM 2000 PEDC, University of Plymouth, UK.

Jeetendra Vadher (1998) "Development of Project-Net and Resource Leveling" Master's Thesis, Indian Institute of Technology, Chennai (Madras),

Jeetendra Vadher (2003) "Modeling of Manufacturing System, A Petri Nets Approach" National Conference on Automation in Manufacturing (NCOAIM), GHRCE, Nagpur.pp.37-45

Jeetendra Vadher, H S Trivedi, Prof. M B Patel (2006) "A Color Petrinets Application To Re-Scheduling Of An Activity Network" International Journal of Computational Mathematics, New Delhi, Vol. 6 No. 3-4, pp. 459-460.

Jeetendra Vadher, O V Krishnaiah Chetty, J P Reddy (2000) "Petrinets for Project Management and Resource Leveling" International Journal of Advanced Manufacturing Technology, Springer-Verlag London Limited, pp. 516 to 520,

Jeetendra Vadher, O V Krishnaiah Chetty, S Kumanan (1998) "Petrinet Application to Project Management" Proceeding of 14th International Conference on CAD, CAM, Robotics and Factories of Future, Coimbatore (India).pp. 853 to 860.

Jerker Björkqvist (2005), "Scheduling batch processing: Genetic Algorithms Versus Mathematical Programming", Intelligent Control System, Proceeding of the 2005 IEEE International Symposium on, Mediterrean Conference on control & Automation Volume issue 27-29 Jun. pp. 352-357

Jigfu Tang, Kai-Leung Yung, Andrew (2004) "Heuristic Based Integrated Decisions for Logistics Network Systems" Journal of Manufacturing Systems, Vol. 23, No. 1, pp. 1-13.

Julie N, William R. Lilegdon (1997) "Making Better Manufacturing Decisions with AIM", Proceeding of 1997 Winter Simulation Conference, pp. 552-558

K Venkatesh, M. Kaighobadi, MengChu Zhou, R. J. Caudill (1994) "Augmented Timed Petrinets for Modeling, Simulation, and Analysis of Robotic Systems with Breakdowns" Journal of Manufacturing Systems, Vol. 13, No. 4, pp. 289 - 301.

K. E. Moore, S. M. Gupta (1995), "Stochastic Colored Petrinet Models of Flexible Manufacturing Systems: Material Handling Systems and Machining", Computers Ind. Engg. , Vol. 29, No. 1 - 4, pp. 333 - 337.

K. Ravi Raju and O.V. Krishnaiah Chetty (1993), "Priority nets for scheduling Flexible Manufacturing systems" Journal of Manufacturing Systems. Vol. 12 no. 4, pp. 326-340.

L E Holloway, B H Krogh, A Giua, (1997) "A Survey of Petrinet Methods for Controlled Discrete Event Systems", Discrete Event Dynamic Systems: Theory and Applications, Vol. 7, Kulwer Academic Publishers, Boston. pp. 151-190,

L. Peterson, (1981) "Petrinet Theory and Modeling of systems", Englewood Cliff, NJ, Prentice Hall, Inc.

Leo J De Vin, Amos H. C. Ng, Jan Oscarsson, (2004) "Simulation based decision Support for Manufacturing System life Cycle Management" Journal of Manufacturing Systems, Vol. 3, No. 2, pp. 115-128

M Heiner, P Deussen and J Spranger (1999) "A Case Study in Design and Verification of Manufacturing System Control Software with Hierarchical Petrinets", International Journal of Advance Manufacturing Technology, Springer-Verlag London Limited, pp. 139-152.

Martin I. R. Lawrence M. W. (1998). "Dynamic Scheduling of a two class queue with setups". Operations Research, Vol. 46, No. 4, pp: 532-546.

N, Viswanandhan, Y, Narahari, (1994) "Performance Modeling of Automated Manufacturing Systems", Prentice-Hall of India Private Limited, New Delhi –India.

Ramadge P. J. and Wonham, W. M. (1987) "Supervisory control of a class of discrete event processes" SIAM J. of control and optimization 25(1): pp. 1202-1218.

Ramadge, P,J, (1989) "Some traceable supervisory control problems for discrete event systems modeled by buchi automata" IEEE transaction on Automatic control 34(1): pp.10-19

Rene David, Hassane Alla, (2005) "Discrete, Continuous, and Hybrid Petrinet", Springer, Germany.

Richard Zurawski, MengChu Zhou (1994) "Petri Nets and Industrial Applications: A Tutorial", IEEE Transactions on Industrial Electronics, Vol. 41, No.6, pp. 567-583

S. Cem Karacal (1997), "Evaluating embedded decision processes of manufacturing systems through simulation" proceeding of the 1997 winter simulation conference, USA, Edwardsville.

S. Shalev-Oren, A. Seidmann, P. J. Schweifzer (1985) "Analysis of Flexible Manufacturing Systems with Priority Scheduling: PMVA", Annals of Operations Research, 3, pp. 115-139.

Tadao Murata (1989) "Petri Nets: Properties, Analysis and Applications" Proceedings of the IEEE, Vol. 77, No. 4, pp. 541-580.

Tzong, Ming Cheng (2004) "On-Line Deadlock Avoidance for Complex Routing Flexible Manufacturing Cells" International Journal of Applied Science and Engineering, pp. 163-176, 2004.

Vitali Volovoi (2004) "Modeling of System Reliability Using Petrinets with Aging tokens", Reliability Engineering and System Modeling, Vol. 84, pp. 149-161,

Annexure 1: Evaluation of reproduction phases on a random population

String No.	Random No, X	Initial Population (Randomly generated) T_Min +(T_Max - T_Min)*X	String	F(x)	Exp. Count	P. Select	Cu. Pro.	String No.	True count in Mating pool	Mating pool String
1	0.550	22	0010110	0.0434	1.2753	0.12753	0.12753	5	0	0100011
2	0.700	28	0011100	0.0344	1.0108	0.10108	0.22861	6	1	0100110
3	0.600	24	0011000	0.0400	1.1754	0.11754	0.34615	6	1	0100110
4	0.650	26	0011010	0.0370	1.0872	0.10872	0.45487	7	2	0100100
5	0.875	35	0100011	0.0277	0.8139	0.08139	0.53626	9	1	0101000
6	0.950	38	0100110	0.0256	0.7522	0.07522	0.61148	10	2	0011101
7	0.500	20	0100100	0.0476	1.3987	0.13987	0.75135	5	0	0100011
8	0.900	36	0100100	0.0270	0.7934	0.07934	0.83069	9	1	0101000
9	1.000	40	0101000	0.0243	0.7140	0.07140	0.90209	10	2	0011101
10	0.725	29	0011101	0.0333	0..9785	0.09785	1.00000	7	2	0100100

Annexure 2: CP-Matrix for token movement (a, b, c, d, e, f)

(a) T_1 fires

	T1	T21	T3	T4	T5	T6	RS	Token
T1	0	0	0	0	0	0	0	Enable
T21	1	0	0	0	0	0	1	--
T3	0	1	0	0	0	0	1	--
T4	0	0	1	0	0	0	1	--
T5	0	0	0	0	1	0	1	--
T6	0	0	0	1	0	0	1	--

(b) T_{21} fires

	T1	T21	T3	T4	T5	T6	RS	Token
T1	1	0	0	0	0	0	1	Green
T21	0	0	0	0	0	0	0	Enable
T3	0	1	0	0	0	0	1	--
T4	0	0	1	0	0	0	1	--
T5	0	0	0	0	1	0	1	--
T6	0	0	0	1	0	0	1	--

(c)T_3 fires

T1	1	0	0	0	0	0	1	Green
T21	0	1	0	0	0	0	1	Green
T3	0	0	0	0	0	0	0	Enable
T4	0	0	1	0	0	0	1	--
T5	0	0	0	0	1	0	1	--
T6	0	0	0	1	0	0	1	--

(d) T_4 fires, T_5 automatically fires as it is slots in conveyor

T1	1	0	0	0	0	0	1	Green
T21	0	1	0	0	0	0	1	Green
T3	0	0	1	0	0	0	1	Green
T4	0	0	0	0	0	0	0	Enable
T5	0	0	0	0	1	0	1	--
T6	0	0	0	1	0	0	1	--

(e) T_5 already having token and T_6 fires

T1	1	0	0	0	0	0	1	Green
T21	0	1	0	0	0	0	1	Green
T3	0	0	1	0	0	0	1	Green
T4	0	0	0	1	0	0	1	Green
T5	0	0	0	0	1	0	1	Green
T6	0	0	0	0	0	0	0	Enable

(f) All transition fired and completes the Manufacturing cycle

T1	1	0	0	0	0	0	1	Green
T21	0	1	0	0	0	0	1	Green
T3	0	0	1	0	0	0	1	Green
T4	0	0	0	1	0	0	1	Green
T5	0	0	0	0	1	0	1	Green
T6	0	0	0	0	0	1	1	Green