

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

ABSTRACT

Timed Petri Net Simulation of Flexible Manufacturing Systems

**by
Yan Chen**

Standard Petri nets have been used to model and analyze Flexible Manufacturing Systems. The timed Petri net, which can incorporate the time delay associated with manufacturing events, provides additional information about real time behavior of practical systems. The Timed Petri Net Simulation Tool, a highly interactive graphical tool, is applied to simulate the modeled flexible manufacturing systems. Timed Petri net models are experimented with. Machine utilization data and throughput are obtained. Analysis of the results shows that the system performance can be optimized by choosing proper parameters.

**TIMED PETRI NET SIMULATION OF
FLEXIBLE MANUFACTURING SYSTEMS**

by
Yan Chen

**A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science**

Electrical Engineering

January, 1993

Blank Page

APPROVAL PAGE
Timed Petri Net Simulation of
Flexible Manufacturing Systems
Yan Chen

Dr. Anthony D. Robbi, Thesis Adviser
Associate Professor of Electrical and Computer Engineering and Director of
Computer Engineering Program, NJIT

Dr. MengChu Zhou, Committee Member
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Constantine Manikopoulos, Committee Member
Associate Professor of Electrical and Computer Engineering, NJIT

This thesis is dedicated to
my dear parents
Mr. Chen Jie & Mrs. Chen Bi-Rong

ACKNOWLEDGMENT

I would like to express my sincere gratitude to my advisor, Professor Anthony D. Robbi, for his guidance, support, kindness and encouragement throughout this thesis.

Special thanks to Professor MengChu Zhou and Professor Constantine Manikopoulos for kindly serving as members of the thesis committee. Professor Zhou gave many helpful suggestions.

Also, thanks to the friends who have kindly helped me.

TABLE OF CONTENTS

Chapter	Page
1 INTRODUCTION	1
1.1 Petri Net	1
1.1.1 Petri Nets Structure	1
1.1.2 Graphical Representation	1
1.1.3 Petri Net Marking	2
1.1.4 Execution Rules	2
1.2 Modeling by Petri Nets	3
1.2.1 Model	3
1.2.2 Petri Net Modeling Concept	3
1.2.3 A Modeling Example	4
1.3 Petri Nets Liveness	6
1.4 Petri Nets Applications	6
1.5 Research Objectives	7
2 TIMED PETRI NET SIMULATION TOOL	9
2.1 Timed Petri Nets	9
2.1.1 Timed Transition	9
2.1.2 Timed Transition Firing	9
2.2 Timed Petri Net Simulation Tool	10
2.2.1 User Interface	10
2.2.2 Drawing and Editing Petri Nets	11
2.2.3 Simulating Execution	13
2.2.4 Output Files	14
2.3 TPNS Applications	15
3 FLEXIBLE MANUFACTURING SYSTEM AND TPNS	16
3.1 Flexible Manufacturing Systems	16

3.2 Petri Nets and FMS	17
3.3 Petri Net Modeling Methodology	17
4 EXAMPLES	19
4.1 Example One	19
4.1.1 FMS System Description	19
4.1.2 Petri Net Representation for Case 1 Model	20
4.1.3 Simulation Results for Case 1	25
4.1.4 Description for Case 2	29
4.1.5 Simulation Results for Case 2	31
4.1.6 Description for Case 3	31
4.1.7 Simulation Results for Case 3	34
4.2 Example Two	35
4.2.1 System Description	35
4.2.2 Petri Net Model of FMC	38
4.2.3 Simulation Results for FMC	40
4.2.4 Petri Net Model of FMS	42
4.2.5 Simulation Results for FMS	44
5 CONCLUSION	46
5.1 Summary	46
5.2 Furture Version of TPNS	47
APPENDIX	48
REFERENCES	57

LIST OF TABLES

Table	Page
1.1 Typical Interpretations of Transitions and Places	4
4.1 Places and Their Representations in FMS of Figure 4.2	22
4.2 Transitions and Their Representations in the FMS of Figure 4.2	23
4.3 Places and Their Representations in FMC	40
4.4 Transitions and Their Representations in FMC	41
4.5 Utilizations of Lathes with <i>One</i> Tokens in p_1 & p_2	42
4.6 Utilizations of Lathes with <i>Two</i> Tokens in p_1 & p_2	42
4.7 Places and Their Representations in FMS	43
4.8 Transitions and Their Representations in FMS	44
4.9 Utilizations of Lathes with <i>Two</i> Tokens in p_{14}	44
4.10 Utilizations of Lathes with <i>Four</i> Tokens in p_{14}	45

LIST OF FIGURES

Figure	Page
1.1 A Simple Example of A Robot Moves Two Steps	6
2.1 A Timed Transition Firing Sequence	10
2.2 TPNS Tool Layout	12
2.3 A Pop-up Window for Terminating the Simulation	12
2.4 A Pop-up Window for Choosint the Output Files	14
4.1 A Block Diagram of the Underlying FMS Layout	20
4.2 A Petri Net Model of FMS in Figure 4.1	21
4.3 Verify File for the Petri Net in Figure 4.2	24
4.4 Utilization File for Net in Figure 4.2, $t_7 = 9$	26
4.5 Machine 1 and Machine 2 Utilizations for the Net in Figure 4.2.	28
4.6 Utilizations for the Net in Figure 4.2 with 6 Tokens in Place p_3 to Start with	29
4.7 A Petri Net Model for t_1 and t_2 are Timed Transitions	30
4.8 Utilizations for Net in Figure 4.7	31
4.9 The Position Buffer is Added	32
4.10 A Petri Net Model of the FMS with Position Biffers Added	33
4.11 Utilizations with One and Two Tokens in Each Position Buffer	34
4.12 Westinghouse EMD FMS Layout	36
4.13 System Operation Procedure.	37
4.14 The Gantry Robot Working Routine.	38
4.15 A Petri Net Model of FMC.	39
4.16 A Petri Net Model of FMS.	43

CHAPTER 1

INTRODUCTION

1.1 Petri Net


Petri net theory was developed by Carl Adam Petri in 1962[1]. It allows a discrete event system to be modeled by a bipartite graph net. A Petri net is a mathematical representation of such a system, designed for modeling analysis, performance evaluation and control of interacting concurrent discrete event systems. Analysis of a Petri net can reveal important information about the dynamic behavior of the modeled system. This information can then be used to evaluate the modeled system and suggest improvements or changes. Petri nets are useful for modeling and analyzing systems involving concurrent activities.

1.1.1 Petri Net Structure

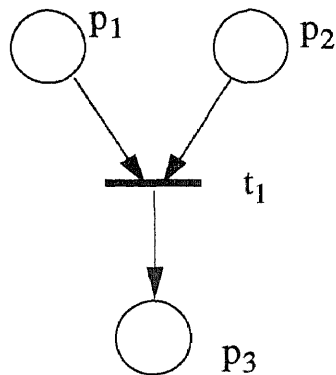
A Petri net is composed of four parts: a finite set of places P , $P = \{p_1, p_2, \dots, p_n\}$, where $n \geq 0$; a finite set of transitions T , $T = \{t_1, t_2, \dots, t_m\}$, where $m \geq 0$; an input function I , and an output function O . The set of places and the set of transitions are disjoint, $P \cap T = \emptyset$.

$I: T \rightarrow P^\infty$ is the input function, a mapping from a transition t_j to a collection of places $I(t_j)$. $O: T \rightarrow P^\infty$ is the output function, a mapping from a transition t_j to a collection of places $O(t_j)$, ($0 \leq j \leq m$). A place p_i is an input place of a transition t_j if $p_i \in I(t_j)$, p_i is an output place of a transition t_j if $p_i \in O(t_j)$.

1.1.2 Graphical Representation

In a Petri net graph, a place is represented by a circle ; a transition is rep-

represented by a solid bar $\rule{0.5em}{0.4pt}$ or a hollow bar \square depending on the type of the transition. The details about transition type will be given in Chapter 2. The places and the transitions are connected by directed lines called arcs \rightarrow . The number of arcs between a place and a transition can be multiple. If p_i is an input place of a transition t_j , the connecting arc is pointed to transition, if p_i is an output place of a transition t_j , then the connecting arc is pointed away from the transition. For example: if p_1 and p_2 are input places of t_1 and p_3 is an output place of t_1 , the graph representation is:



1.1.3 Petri net Marking

A marking μ is an assignment of tokens to the places of a Petri net. Tokens reside in the places in a Petri net. Pictorially tokens are represented by dots \bullet , or by an integer. The number and position of tokens normally change during the execution of a Petri net.

1.1.4 Execution Rule

A transition $t_j \in T$ in a marked Petri net with marking μ is enabled if each of its input places has at least as many tokens in it as arcs from the places to the transition. That is: for all $p_i \in P$,

$$\mu(p_i) \geq \#(p_i, I(t_j))$$

A transition is fired once if it is enabled. Firing a transition will in general change the marking μ of the Petri net to a new marking μ' .

A transition fires by removing an input token per arc from its input places and then depositing into each of its output places one token for each arc from the transition to the place. That is,

$$\mu'(p_i) = \mu(p_i) - \#(p_i, I(t_j)) + \#(p_i, O(t_j))$$

1.2 Modeling by Petri nets

1.2.1 Model

In many fields, a phenomenon is not studied directly but indirectly through a model of the phenomenon. The model is used to describe the system and to predict what happens if certain actions are taken. Then we can observe its behavior, and infer the behavior of the actual object or system under the same circumstances. Since the model will represent only certain features of the original system, it will usually be less complex and easier to study. We tend to manipulate simple models of the real world.

Systems can be modeled by using differential equations, by using queueing theory, etc. One of the important and powerful methods of modeling is the Petri net. Many discrete systems especially those with independent components, can be modeled by Petri net.

1.2.2 Petri Net Modeling Concept

The major strength of Petri nets is, in the modeling of systems which may exhibit concurrence; concurrence is modeled in a natural and convenient way. A Petri net model can be used to represent and communicate the design of a concurrent system.

In Petri net modeling, using the concept of conditions and events, places

represent conditions, and transitions represent events. A transition (an event) has a certain number of input and output places. The input places of a transition represent the preconditions of the corresponding event and the output places the postconditions. The concurrence of events corresponds to the simultaneous firing of the corresponding transitions.

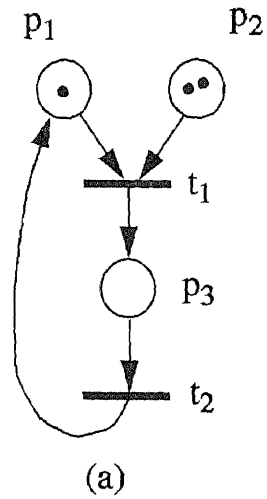
The presence of tokens in a place is interpreted as holding the truth of the condition associated with the place. In another interpretation, k tokens are put in a place to indicate that k data items or resources are available. When a transition fires it removes the tokens representing the truth of the precondition and creates new tokens which represent the truth of postconditions. Some typical interpretation of transitions and their input places and output places are shown in Table 1.

Table 1.1: Typical Interpretations of Transitions and Places

Input Places	Transitions	Output Places
Precondition	Event	Postconditions
Input data	Computation step	Output data
Input signals	Signal processor	Output signals
Resources needed	Task or job	Resource release
Conditions	Clause in logic	Conclusion(s)
Buffers	Processor	Buffers

1.2.3 A Modeling Example

A simple example in Figure 1.1 shows that a robot moves in two steps. Places p_1 and p_2 indicate the availability status of the robot and parts respectively, and p_3 indicates the operational status of the robot, t_1 represents the robot



p_1 : Robot available

p_2 : Part in position

t_1 : Robot moves part

t_2 : Robot returns

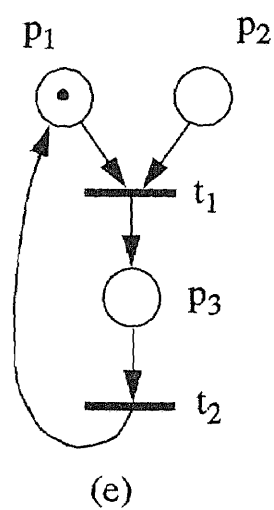
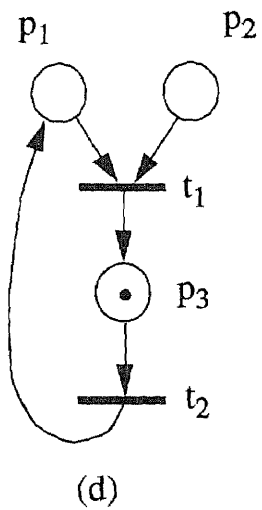
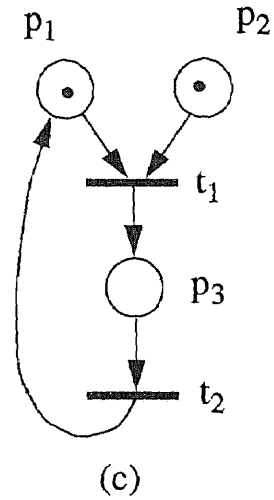
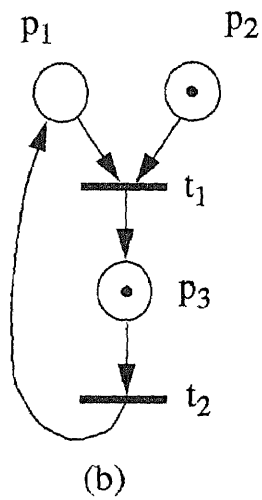


Figure 1.1 A Simple Example of Robot Moving Two Parts

moving a part, and t_2 the return of the robot. If there are more parts to be moved, then more tokens appear in p_2 , and the robot will keep moving parts continuously.

1.3 Petri Net Liveness

A deadlock in a Petri net is a marking in which no transition can fire. A Petri net is live if it cannot deadlock. The execution of a Petri net is the firing of all enabled transitions in one state (marking) to reach another state. A live Petri net guarantees deadlock-free operation, no matter what firing sequence is chosen. If a Petri net represents a working system, then it should be kept live.

1.4 Petri Net Applications

The knowledge of the fundamentals of Petri net theory is becoming mandatory for the computer science, system analysis, and engineering professions.

Petri nets have been proposed for a very wide variety of applications. This is due to the generality and permissiveness inherent in Petri nets. They can be applied informally to any area or system that can be described graphically like flow charts and that needs some means of representing parallel or concurrent activities. Some areas like modeling and analysis of distributed-software systems, distributed-database systems, concurrent and parallel programs, flexible manufacturing/industrial control systems, discrete-event systems, multiprocessor memory systems, dataflow computing systems, fault-tolerant systems, programmable logic and VLSI arrays, asynchronous circuits and structures, compiler and operating systems, office-information systems, formal languages, and logic programs. However, careful attention must be paid to a trade-off between modeling generality and analysis capability[2].

1.5 Research Objectives

The use of modern computer-aided tools is a necessity for practical applications of Petri nets. Initial work towards the development of a graphical Petri net tool was made by Arsalan Gilanli[3]. Then Ashish Shukla integrated a simulation for Petri nets with the graphical editor to create PNS[4]. Later Javaid Siddiqi added time to Shukla's model to enable simulation of real systems that have time delays as an important feature, TPNS[5]. In TPNS new features were added to enhance the editing and simulation capabilities of the tool. Systems with deterministic and discrete stochastic delays can be simulated and many performance and throughput parameters can be derived. Petri modeling can solve problems that cannot be precisely modeled using queuing theory, avoiding the time consuming, trial and error approach of custom simulation.

This thesis is focused on timed Petri net applications to Flexible Manufacturing Systems with time constraints. Based on the existing TPNS, examples from real time Flexible Manufacturing Systems are simulated with the tool. System performance can be predicted from the output data.

Many researches have been done to analyze Flexible Manufacturing Systems. And there has been a significant increase in the use of simulation to design and analyze manufacturing systems. The Petri net applications to manufacturing systems include modeling of manufacturing system, qualitative analysis, performance evaluation, scheduling and control implementations.

The modeling and analysis of a flexible manufacturing system cell using Petri nets is described by Zhou et al.[6]. The paper describes the detailed modeling process; top-down refinement, system decomposition, and modular composition, to achieve a hierarchical model which preserves important system

properties.

An automatic method for computing the expected execution time of jobs is proposed by Someya, et al. in [7]. A job operation flow is modeled using the timed Petri net. Operation cases for faults are considered using the reachability tree of the Petri net and the expected execution time is computed automatically.

In a paper authored by Menga et al.[8] an FMS is modeled by closed queuing network analysis to estimate certain throughputs of the system, and to analyze the problem of scheduling concurrent lots of different components on a FMS structured with a closed transportation network.

In paper authored by Davis et al.[9] Markov models are used to describe an easily understood production process. Rework and work-in-process inventories are modeled, together with inspections and scrap rates, and a method is shown for taking tool wear into account. The scheduling impacts of tool wear are also analyzed using the Markov models developed. Markov models which can be derived from Petri nets suffer from the state explosion problem when the number of places and transitions exceeds a small limit.

The use of TPNS to model manufacturing systems offers the following: graphical representation to ease system description, visual debugging, mathematical basis to ensure liveness, flexible assumptions regarding delay times of events, and summary evaluation of system performance by simulation. The disadvantage of simulation in its cost of computer cycles is diminishing as the cost effectiveness of computers continues its rapid improvement.

CHAPTER 2

TIMED PETRI NET SIMULATION TOOL

2.1 Timed Petri Net

In conventional Petri nets, the firings of transitions are instantaneous events, and the analysis of such nets usually assumes that the firings are performed “one at a time.” In the real life, processes do take time. In manufacturing, a robot or an automated guided vehicle takes or moves an object, and a milling machine or a grinding machine takes a certain amount of time to perform work. Modeling of real systems must also take into account the duration of systems activities represented by transition firings.

2.1.1 Timed Transition

In TPNS timed Petri nets carry timing information as a part of the transition. Each timed transition takes a finite time to fire. The firing delay of a transition can be specified as deterministic or stochastic. In deterministic Timed Petri nets the delay is specified by a constant. In stochastic timed Petri nets, the delay is a random time that is generated from a user specified distribution.

2.1.2 Timed Transition Firing

In timed Petri nets, the firing of an enabled transition is composed of three “conceptual” steps; the first (instantaneous) removes tokens from the input places, the second (temporal) holds the removed tokens within the transition for the duration of the firing time, and the third step (instantaneous) moves tokens to all transition’s output places. Figure 2.1 shows a timed transition firing.

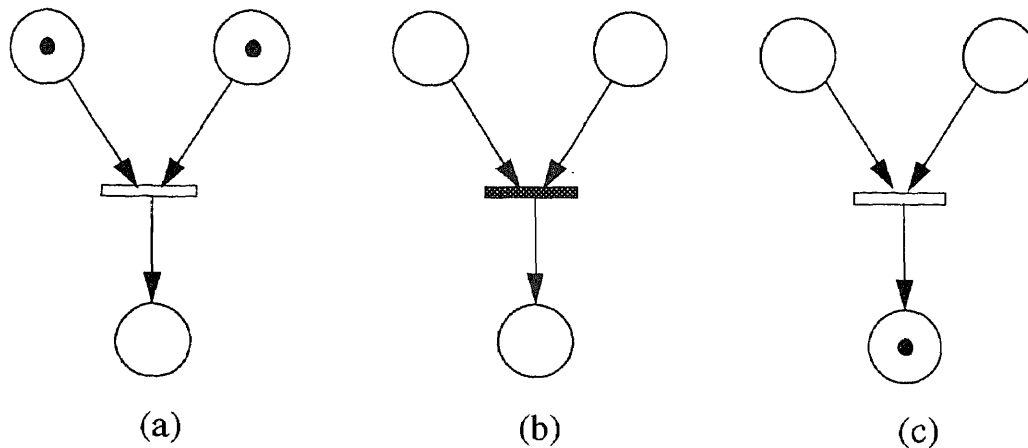


Figure 2.1 A Timed Transition Firing Sequence

2.2 Timed Petri Net Simulation Tool

Timed Petri Net Simulation Tool (TPNS) is a highly interactive graphical tool for the drawing and simulation of discrete event systems involving time constraints. It is capable of simulating systems with deterministic and stochastic delays.

2.2.1 User Interface

A user friendly interface is developed for effective interaction between the tool and the users. TPNS runs on SUN3 and SUN4 workstations under the Suntool utility based on SunView providing a graphical user interface. Windows provide visual access to several aspects of an application at the same time. The TPNS interface utilizes pop-up windows and on-line help facility. To draw immediate attention a pop-up window with a message is a very effective technique. Normally these windows suddenly appear in the middle of the active window. Message pop-up windows are extensively used for TPNS tool. They give a result, an error message, and a help message to inform

the user of an action.

Data Entry pop-up windows are used to request the user to input some data.

Alerts are pop-up windows that require mandatory user attention. They display a message and one or more action buttons. A user is asked to make one or one of the multiple choices to continue the work.

Information pop-up windows give information about a selected object.

2.2.2 Drawing and Editing Petri Nets

A timed Petri net can be drawn by means of TPNS on the working area. Figure 2.2 shows the appearance of the tool on the workstation screen[4, 5]. There are fourteen buttons on the Panel for net editing, simulation and results display. Place, transition, arc and token symbols are shown in the panel window of the upper screen as different objects. They can be chosen as needed by using the mouse. An immediate transition is represented by a thin solid bar (same as the transition in ordinary Petri nets). A timed transition is represented by a hollow bar. When a timed transition is chosen, the user is asked to specify the type, and its time delay parameters in a pop-up window. TPNS maintains an integer clock. Thus deterministic delays are represented as integers. Stochastic delays are computed from an exponential probability density function with a modeler-chosen average delay as a parameter. The working area can be enlarged for nets which do not fit in one screen.

The ERASER can erase the unwanted objects. TAG shows information about places and transitions including transition or place number. For a transition, TAG shows whether it is an immediate transition or a timed one. If it is timed, it also indicates the time delay for deterministic or average delay for stochastic transitions.

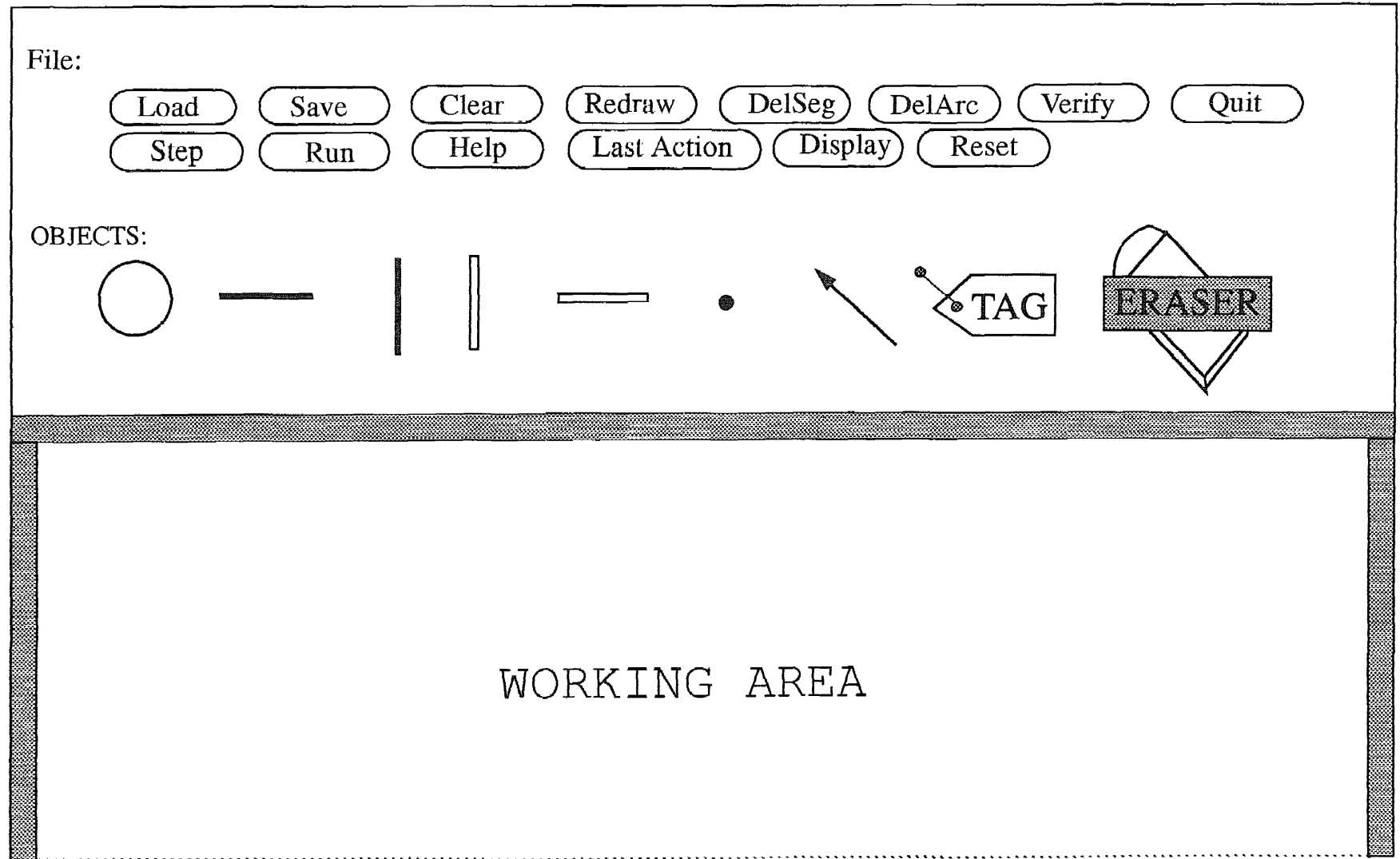


Figure 2.2 TPNS Tool Layout

Once a net is drawn, it can be saved to a file. An extension *.pic for picture will automatically be appended to the name. A backup file with extension *.bak and the same file name will hold the previous version.

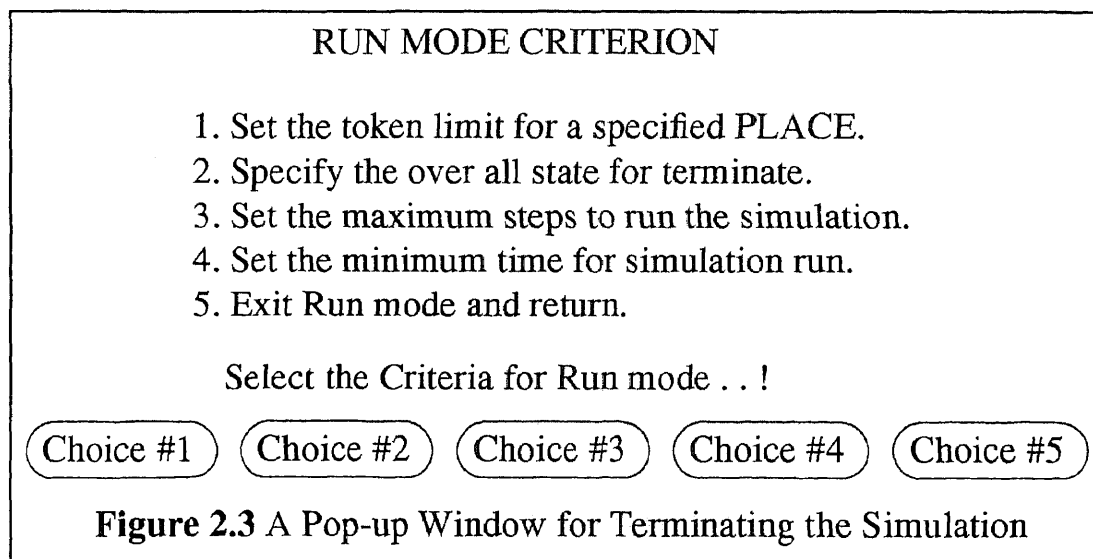
To load an existing picture file, the file name should be typed without extension followed by pressing the **Load** button. Loading a file clears any previously edited net. If the specified file does not exist, then a pop-up window will give a message. The other buttons for editing the nets are described in detail elsewhere[4, 5].

2.2.3 Simulating Execution

The tool has **Step** and **Run** simulation modes.

A **Step** is defined in TPNS as the firing of at least one transition. The clock may advance. The detail of this definition is found in[5].

In **Run** mode, the steps are repeated until a user specified terminating condition is reached or a deadlock occurs. The terminating condition is specified by the user by making a choice in a pop-up window associated with the selection of **Run**. The window is shown in Figure 2.3.



To simulate a net, the more steps assigned, the more reliable the statistical results are. However the cost of CPU time imposes practical limits. This will be discussed later.

2.2.4 Output Files

Certain files are generated automatically when the net is simulated. They are *.vfy, *.mark, *.log, *.utilize. The output files option window is shown in Figure 2.4.

Vfy stands for verify. It is created in **Run** mode and contains the net connection matrix for every place and transition. The net connection can be obtained independently at any time by selecting the Verify button.

Mark stands for the marking. This file contains the state information at each step.

Log is a history file that is created in **Run** mode. This file contains all the information about state transitions after each step.

The *.utilize file is created at the end of **Run** mode only. It is used to log summary transition and place utilization and behavior. The information can be used for performance and throughput analysis.

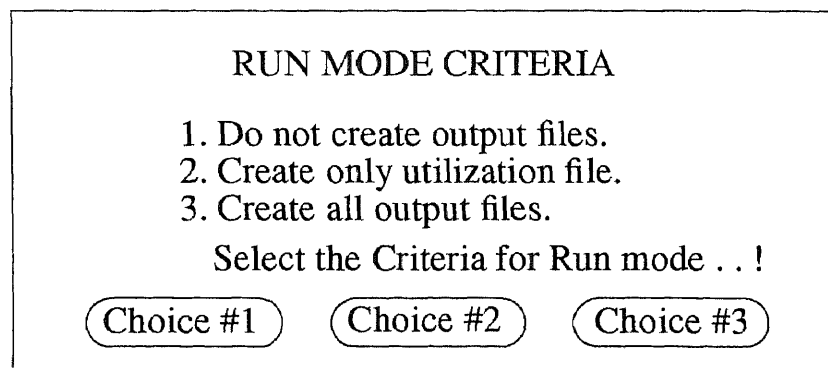


Figure 2.4 A Pop-up Window for Choosing the Output Files

All these files are illustrated later.

In the utilization file, transition utilization logs the number of firings, the percent of time busy, the number of conflicts, and the number of firings in conflict situations for each transition. Place utilization counts the total number of tokens entering and percent of time not vacant for each place. Cycles counts the number of times the initial marking is repeated.

2.3 TPNS Applications

Simulation is a useful method for estimating performance of new systems before they are built. It also can use to improve the existing system performance. Simulating a modeled system can help design future systems and improve existing systems.

In a timed Petri net, a transition may represent part of all of manufacturing process. With TPNS, designers can analyze and evaluate the system performance for different parameters for each process.

CHAPTER 3

FLEXIBLE MANUFACTURING SYSTEMS AND TPNS

3.1 Flexible Manufacturing Systems

A flexible manufacturing system (FMS) is a highly computerized and automated production unit capable of producing a range of discrete products with a minimum of manual intervention. It consists of production workstations where processing and/or assembly of parts occurs. A material handling system moves parts from one workstation to another. An FMS operates as an integrated system under full programmable control. An FMS is capable of adapting its processing automatically, whereas hard automation requires down-time and manual intervention.

In manufacturing a process plan for each product is essential to achieve coordination. The process plan details all the activities, the resources required to perform them and the order in which they must occur. It may include specific machining operations and which machines can perform them, part routing, assembly sequencing, and so on. The process plans embody a complex set of precedence relations between activities, and can give rise to a challenging resource allocation problem because of the high degree of concurrency and interaction.

An automatic transport system is required in a FMS in order to move parts from machine to machine. Such transport devices could be a robot, an automated guided vehicle (AGV) or an automated storage and retrieval system (ASRS). Frequently, a flexible manufacturing system is structured into manufacturing cells. A cell is an elementary manufacturing system consisting of a flexible machine tool (or an assembly device, or any complex device dedicated to a complex manufacturing operation), and a material handling

interface.

3.2 Petri Nets and FMS

The Petri net is good basis for the modeling, control, and performance analysis of the FMS. An FMS system involves concurrent and interactive activities. Petri nets are well suited for modeling an FMS because they can capture the precedence relations and interaction among these events.

The flow of tokens through the net models the sequence of events to carry out a specific task, such as the manufacturing of a particular part. When time is added to the firing of the transitions, it becomes possible to calculate temporal measures that analyze the merit of a particular production system. Different product processing takes different time, and the timed Petri net model can truly represent such a system.

3.3 Petri Net Modeling Methodology

Modeling methodology is key to the application of Petri nets to manufacturing systems[10]. In the examples given in the next chapter, the modeling is basically approached as follows:

1. Identify the activities and resources required for the production of one item of each product.
2. Order activities by the precedence relations as given in the process plans.
3. For each activity in order: create a place to represent the status of that and a transition to represent the activity processing. Specify the type of the transition and the parameter, if it is a timed transition, according to the specific process.
4. Connect all appropriate resources available with arcs.

5. Verify the net.

6. Specify an initial marking for the system.

At this point, a system model has been created. Now results are obtained as follows:

1. Execute the net using step mode to observe that token flow is proper, a net debug of sorts.

2. Set up run mode parameters and simulate the system.

CHAPTER 4

FMS APPLICATIONS

In this chapter TPNS is used to model and simulate the behavior of real time FMS systems. One objective of the simulation is to examine the utilizations of the timed transitions. Percent utilization is defined in TPNS as:

$$\frac{\text{Total transition delay} \times 100}{\text{Total simulation time}}$$

For timed transitions which represent machines, the greater the utilization, the better.

The log file shows every step of the simulation. It can be used to help understand the system behavior. But for large numbers of steps, utilization files are more significant.

Another datum in the output file indicates the duration of a place was occupied by tokens. This datum describes the percentage of time parts or material reside in a certain buffer or location during the simulation run.

4.1 Example One

4.1.1 FMS System Description

A factory floor model of an FMS layout is shown in Figure 4.1. The system is operated to process unfinished material in central storage to finished parts in one of the storage buffers. This system was examined briefly in[5].

The system comprises a central storage facility connected to two material buffers, labeled WORKSTATION in the figure, by two conveyer belts. An Automatic Guided Vehicle (AGV) moves material from the buffers to two distant associated processing machines. The processed parts are transported by conveyers to output buffers A and B. To increase productivity and cost

ratio, one AGV is used to transport material in two different directions.

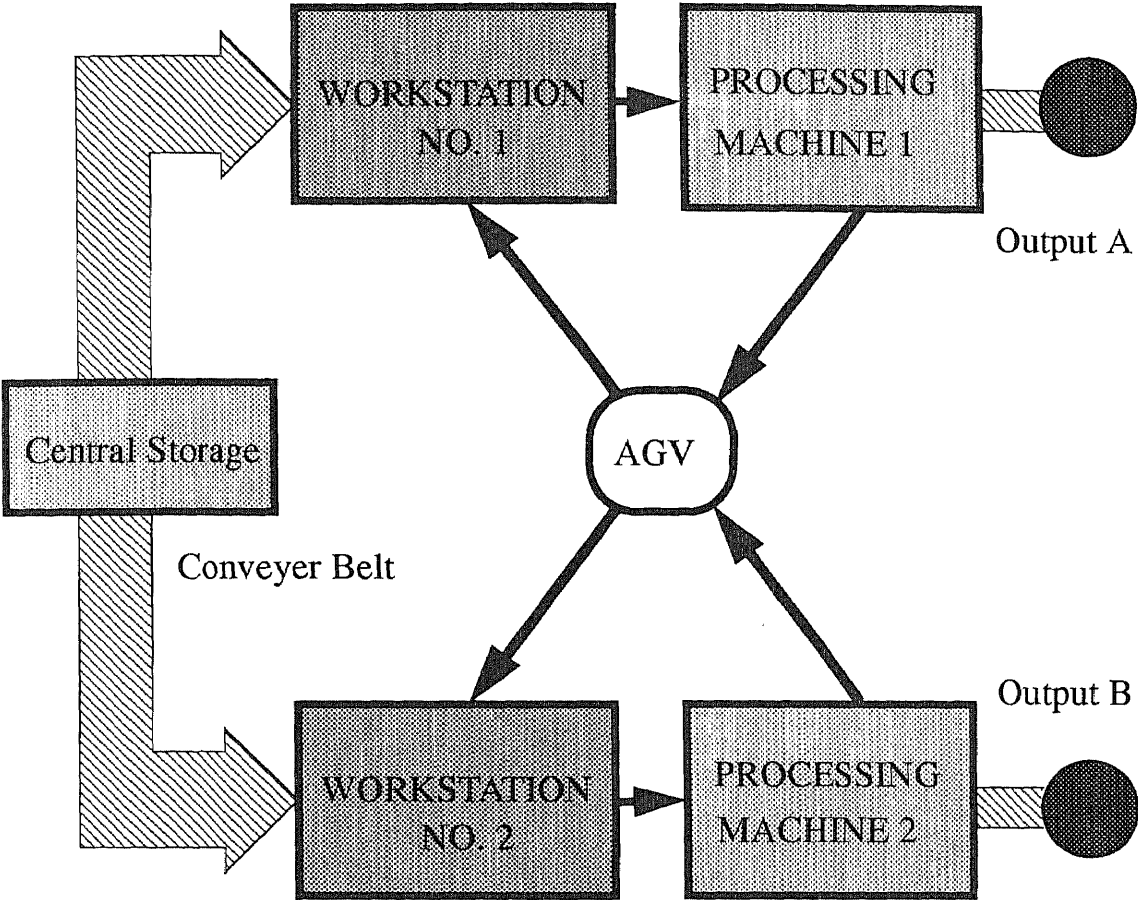


Figure 4.1 A Block Diagram of the Underlying FMS Layout

4.1.2 Petri Net Representation of Case 1

Petri net methodology permits modeling the potential competition for the services of the AGV, the concurrent behavior of the two machines, and the finite capacities of the buffers to be modeled. The time feature permits finite time delays to be associated with various activities.

A Petri net model for the system is given in Figure 4.2. The places and transitions are defined in Tables 4.1 and 4.2. The types and the parameters

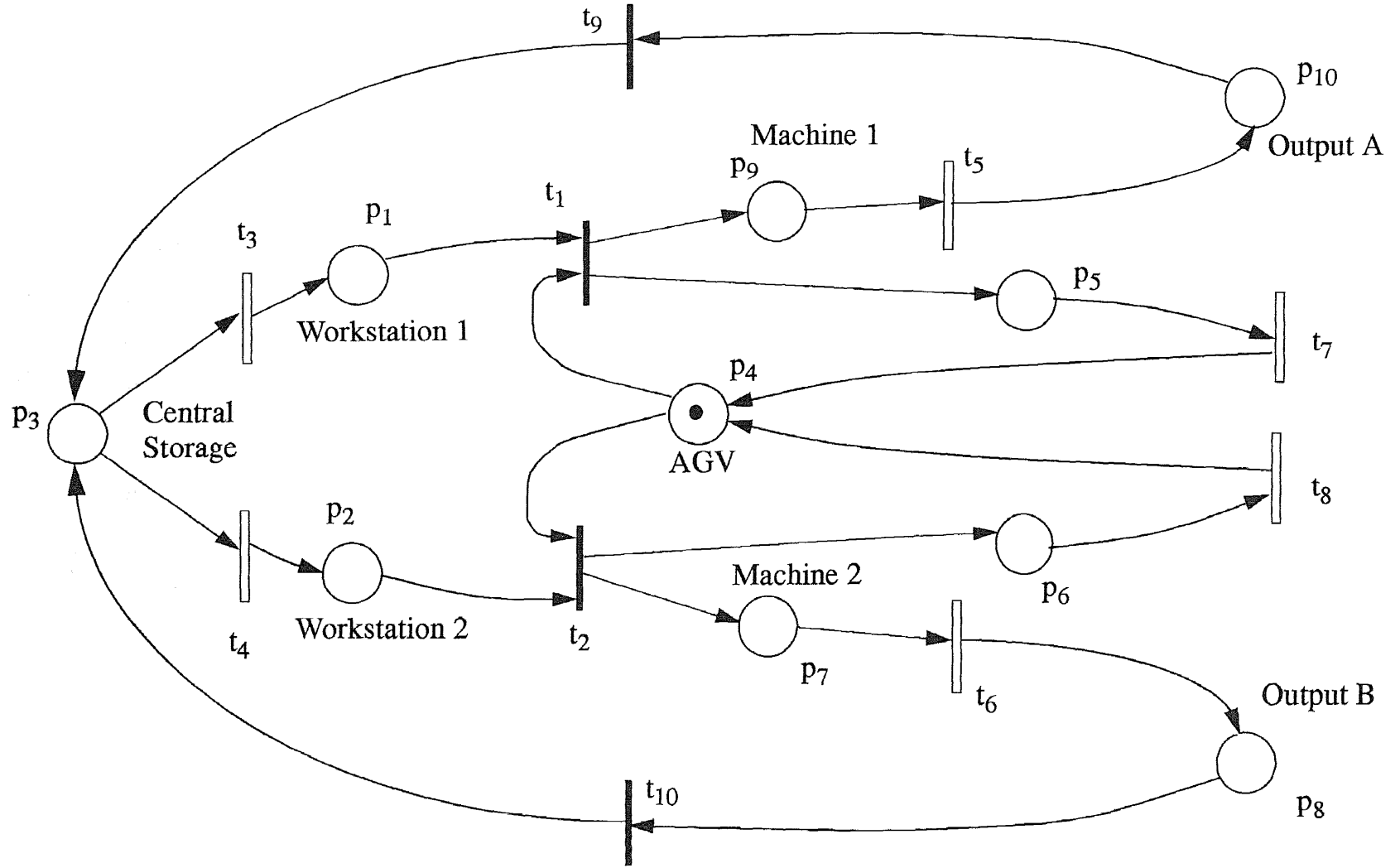


Figure 4.2 A Petri Net Model of FMS in Figure 4.1

associated with the transitions are also given in Table 4.2. Figure 4.3 shows the verify file for the net. It displays the connection matrix for places and transitions.

Table 4.1: Places and Their Representations in FMS of Figure 4.2

Places #	Representations
p ₁	Workstation 1
p ₂	Workstation 2
p ₃	Central Storage
p ₄	AGV available
p ₅	AGV returns from Machine 1
p ₆	AGV returns from Machine 2
p ₇	Machine 2
p ₈	Output B
p ₉	Machine 1
p ₁₀	Output A

The supply of the raw material to workstations is carried by conveyer belts which are represented by stochastic transitions t_3 and t_4 . The supply to these buffers is assumed to have exponential distribution to represent some randomness in material supply. The other transitions are deterministic.

The material in Central Storage is equally likely to go to Workstation 1 or 2, whenever t_3 and t_4 are not ticking (occupied with tokens). Over time Machine 1 gets more work because the average firing rate of t_3 is higher than that of t_4 . The availability of the AGV is represented by a token in place p_4 . When the AGV is available, if there are tokens in either place p_1 or p_2 , the

AGV transports the material to either Machine 1 or Machine 2 by firing either t_1 or t_2 . If there are tokens in both p_1 and p_2 , material in both workstations is waiting to be transported, and a conflict situation occurs. Priorities of t_1 and t_2 are equal and one is enabled at random. After the AGV finishes every transport and returns from either machine, a token appears in place p_4 , indicating that it is available for the next transport.

Table 4.2: Transitions and Their Representations in the FMS

Transitions #	Values or Means	Representations	Types
t_1	N/A	Commits AGV to Workstation 1	Immediate
t_2	N/A	Commits AGV to Workstation 2	Immediate
t_3	$\bar{3}$	Conveyer belt 1 to Workstation 1	Stochastic
t_4	$\bar{4}$	Conveyer belt 1 to Workstation 2	Stochastic
t_5	5	Machine 1 processing	Deterministic
t_6	3	Machine 2 processing	Deterministic
t_7	Variable	Time required for AGV to return from Machine 1	Deterministic
t_8	2	Time required for AGV to return from Machine 2	Deterministic
t_9	N/A	Feedback from Output A	Immediate
t_{10}	N/A	Feedback from Output B	Immediate

```

*****
Petri Net File name :  ex1-6_2.vfy
*****
Trans#           I/P Places           O/P Places
-----
1                 1
-----
1                 4
-----
1                 9
-----
1                 5
-----
2                 4
-----
2                 2
-----
2                 7
-----
2                 6
-----
3                 3
-----
3                 1
-----
4                 3
-----
4                 2
-----
5                 9
-----
5                 10
-----
6                 7
-----
6                 8
-----
7                 5
-----
7                 4
-----
8                 6
-----
8                 4
-----
9                 10
-----
9                 3
-----
10                8
-----
10                3

```

Figure 4.3 Verify File for the Petri Net in Figure 4.2

4.1.3 Simulation Results of Case 1

The Petri net in Figure 4.2 is simulated for 2000 steps. It was found that 500 steps are insufficient for accurate results. The number of tokens in p_3 at simulation start is 4 (later 6). Four is a minimum to permit concurrent operation of all devices.

AGV return times from the machines are represented by t_7 and t_8 . If the delay time of t_8 is fixed at 2, and that of t_7 is changed, which implies that AGV return time varies, then the utilizations of processing Machine 1 and 2, which are represented by t_5 and t_6 , will vary. One of the utilization files is illustrated in Figure 4.4, and Figure 4.5 shows the utilizations for both machines as a function of t_7 (connected points).

It can be seen from Figure 4.5 that the utilization of Machine 1 is higher than that of Machine 2 for all the values of t_7 because Machine 1 is slower. The utilizations for both machines generally decrease while the time delay of t_7 increases. The utilization of Machine 2 reaches a peak at t_7 equal to 3. Conveyer belt 1 transports material with an average time delay of 3 which is one unit faster than belt 2, and Machine 1 processes material for 5 units of time, two units slower than Machine 2. In the path to Machine 1, the supply of material is faster, and machine processes is slower; in the path to Machine 2 supplies material relatively slowly, and since this Machine needs less time to process, it is idle much of time waiting for material, therefore the utilization of Machine 1 is much higher than Machine 2 for the same AGV delay time. As the return time of AGV from Machine 1 increases, the more time for AGV to get ready to transport for any path supply, so the utilizations for both machines are generally decreased.

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 2514
Total number of steps : 2000
*****

```

Transition Utilization / Behavior

Transition Tag	No. of times Fired	Busy*		Involved in conflict	Success
T1	235	11.75%	***	264	131
T2	198	9.90%	***	264	133
T3	235	28.40%		306	151
T4	199	36.16%		306	155
T5	235	46.74%		0	0
T6	196	23.39%		0	0
T7	235	84.13%		0	0
T8	197	15.67%		0	0
T9	235	11.75%	***	0	0
T10	196	9.80%	***	0	0

* computed as: (total transition delayX100)/total simulation time.

*** immediate transition computed as: (total firing w.r.t total steps)

Place Utilization / Behaviour

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	235	77.72%
P2	199	69.69%
P3	431	27.72%
P4	432	0.20%
P5	235	58.23%
P6	198	2.70%
P7	198	14.84%
P8	196	0.00%
P9	235	20.84%
P10	235	0.00%

Initial marking not reached during net execution.

Figure 4.4 One of the Utilization Files for Net in Figure 4.2

If there is at least one token in both workstations, then both t_1 and t_2 have equal chance to fire if there is a token in p_4 . Since belt 1 is one unit faster than belt 2, Workstation 1 (p_1) has more chance to have a token in it than Workstation 2 (p_2), then t_1 fires immediately as soon as AGV is available while Workstation 2 is still waiting for the token from belt 2. If AGV returns very fast, Machine 1 keeps a high utilization and Machine 2 has low utilization. As t_7 return time increases, it takes longer time to enable both t_1 and t_2 , then the faster speed of belt 1 is less significant, and the chance of firing of t_1 and t_2 tends to equalize. This situation benefits the utilization of Machine 2 as shown in Figure 4.5 at t_7 equal to 3. But it reaches a saturation point as the delay time of t_7 gets larger.

Switching the mean return times for t_3 and t_4 , that is t_3 equal to 4 and t_4 equal to 3, will lower the utilization of Machine 1 and raise that of Machine 2 as indicated in Figure 4.5 with ● for Machine 1 and ■ for Machine 2. In this case, belt 1 is one unit slower than belt 2, Workstation 2 has more chance to have a token in it when AGV is available while no token in Workstation 1. Now Machine 1 utilization peaks at t_7 equal to 2.

Place p_3 in the model represents the Central Storage in the FMS, and the number of tokens describes the quantity of material supply. From the data of utilization files, the duration that p_3 is occupied during the simulation varies from 62.48% to 27.72%, as t_7 goes from 1 to 9. This means the Central Storage is empty as much as 72.28% of the simulation time. During the empty period there is some material already on the workstations or the belts. But if the percentage of storage occupancy could be increased, the workstations could be more likely occupied when the AGV is available. This would improve machine utilizations.

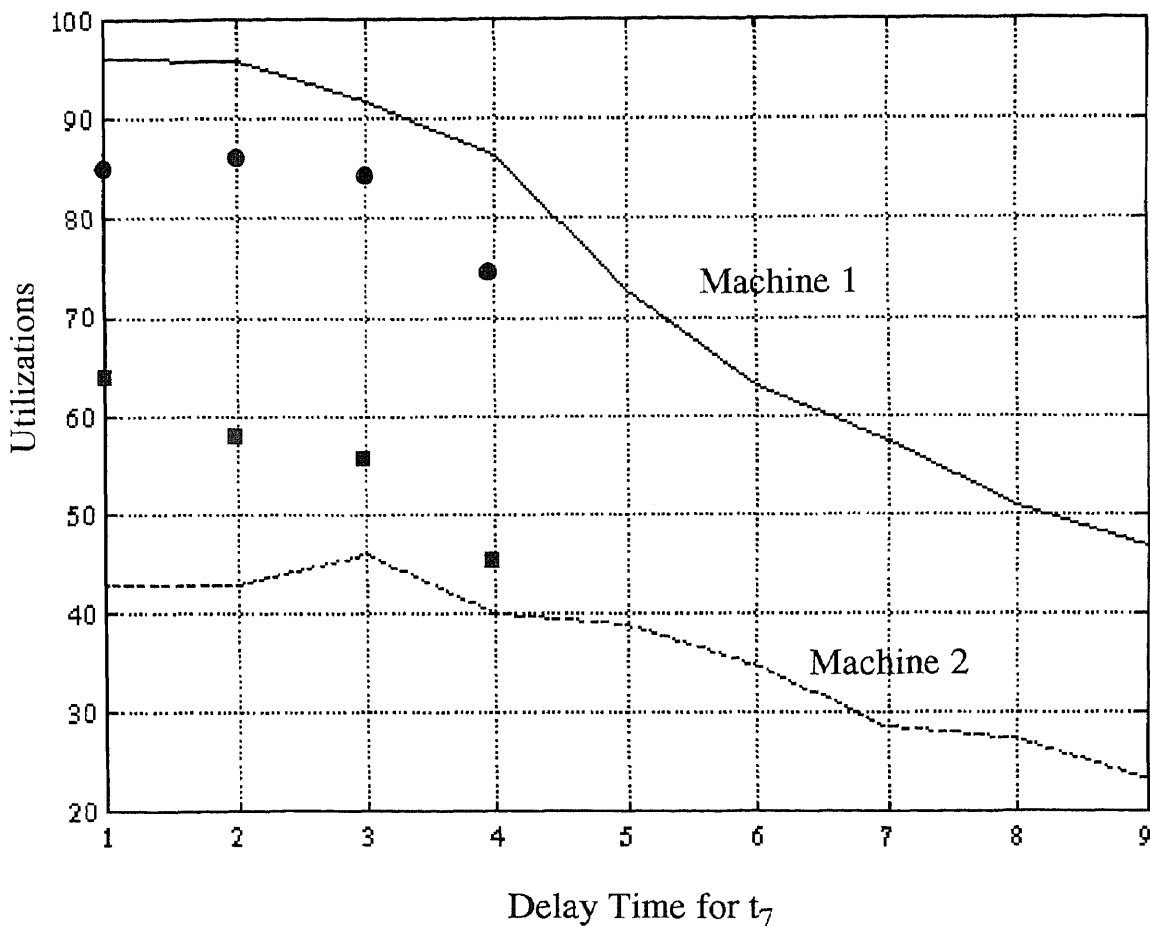


Figure 4.5 Machine 1 and Machine 2 Utilizations for the Net in Figure 4.2

The same net was simulated with six tokens in the place p_3 to start with. Place p_3 occupancy is now from 67.07% to 31.81%, and Figure 4.6 shows that the utilizations of Machine 1 and 2 are increased. The peak in Machine 2 utilization is more pronounced because more material comes to it. In fact the total of parts produced by the two machines peaks at a t_7 of 3. This can be deduced from the utilization files presented in the Appendix- Figures 1,2,3, and 4. This is a seemingly analysis result, but it arises here because slowing down the robot prevents work from piling up at the slower machine.

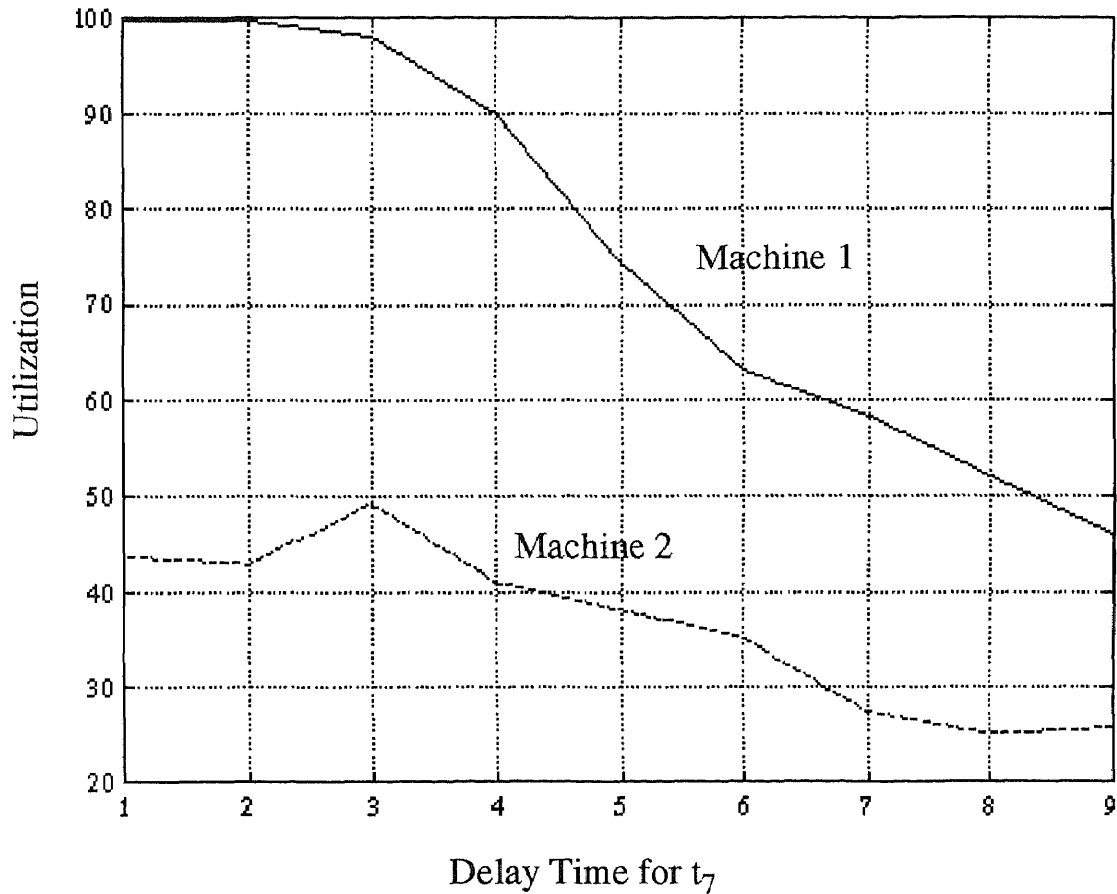


Figure 4.6 Utilizations for Net of Figure 4.2 with 6 Tokens in p_3 to Start with

4.1.4 Description of Case 2

In the Figures 4.2, transitions t_1 and t_2 , which represent the AGV transport the material to workstations, are immediate types. The AGV delay is totally modeled by t_7 and t_8 . Practically, the AGV delay has two components, so t_1 and t_2 are represented better by deterministic timed transitions in Figure 4.7 with equal time delays of 2 for t_1 and t_2 . Figure 4.8 shows the simulation results.

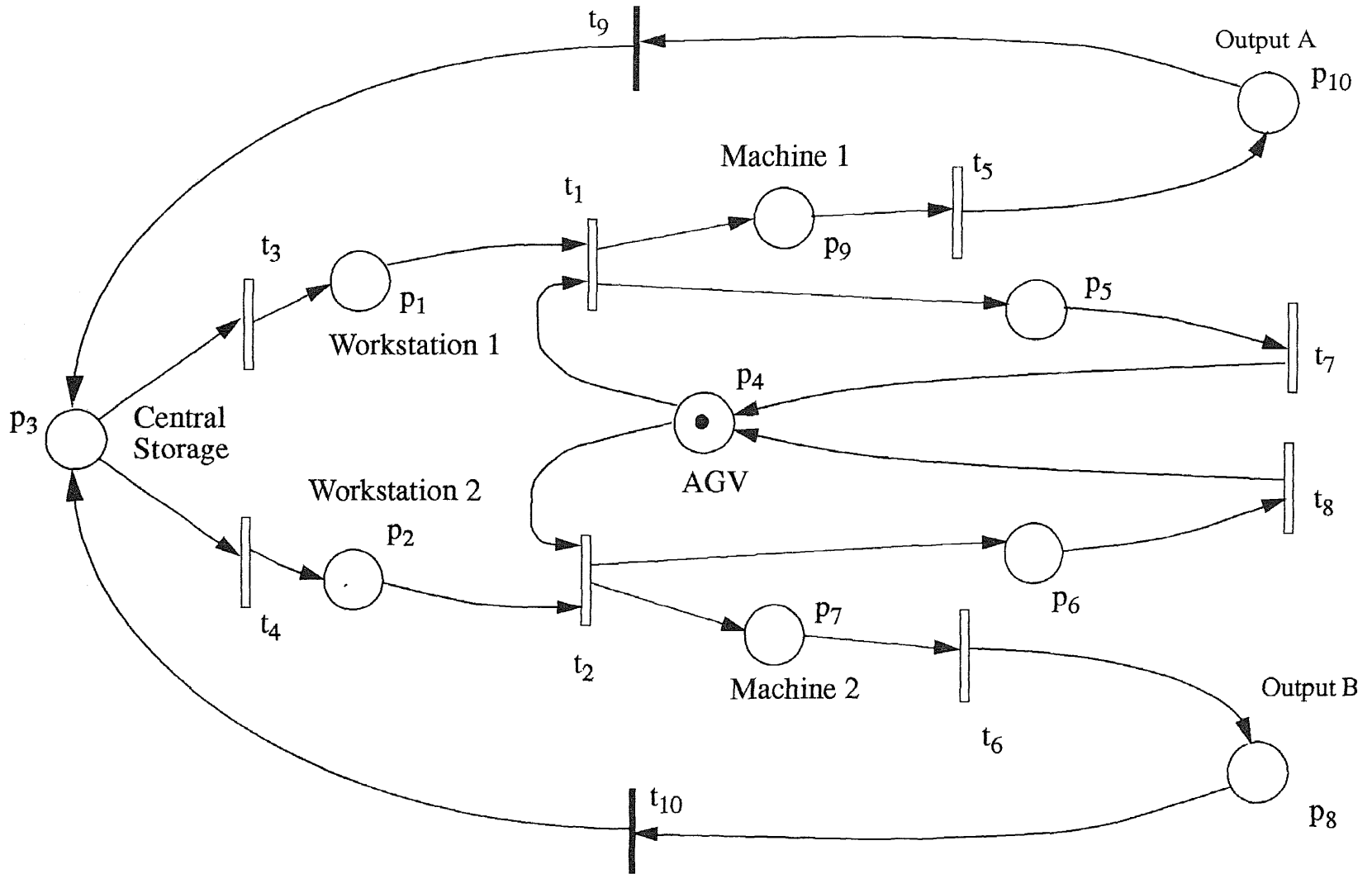


Figure 4.7 A Petri Net Model for t_1 and t_2 to Timed Transitions

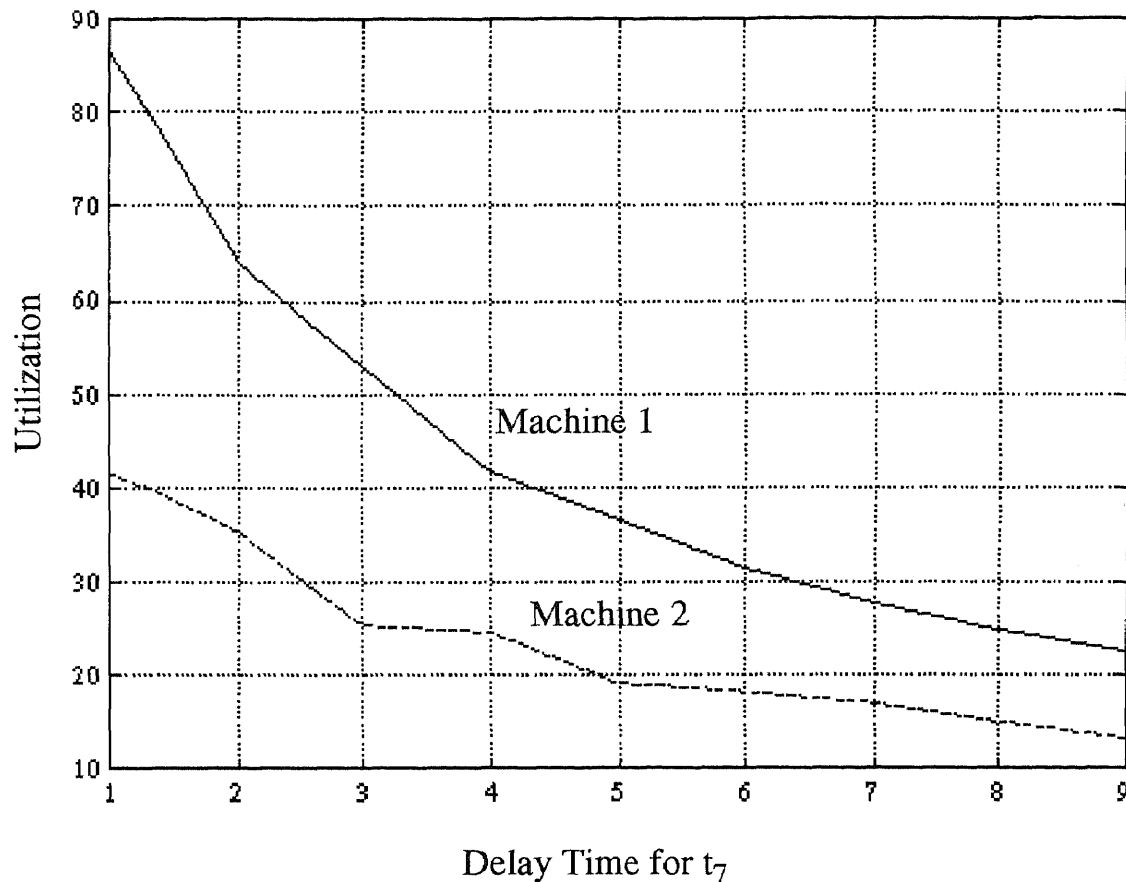


Figure 4.8 Utilizations for Net in Figure 4.7

4.1.5 Simulation Results of Case 2

The Machine utilizations of the net in Figure 4.7 have decreased compared to that in Figure 4.6, because the machines need to wait longer for material to process. Utilizations decrease immediately as t_7 increases because the AGV delay is comparable to the conveyer delay even at small t_7 .

4.1.6 Description of Case 3

A position buffer is used to limit the number of tokens in a place. It is accomplished by adding a path with a place antiparallel to the path with the original place as shown in Figure 4.9. Transitions t_1 and t_2 are both timed.

They may represent processing or transport delays. If the time delay of t_2 is greater than that of t_1 , then tokens could possibly accumulate in p_1 . Transition t_1 can fire whenever it is enabled no matter how many tokens are already in p_1 . After the path with p_2 is added, then t_1 can fire only if there is at least one token, which is generated by firing t_2 , in p_2 . Therefore the firing of t_2 is a precondition of enabling t_1 , and the number of tokens in p_1 is limited. The device which is represented by t_1 has to wait for the device which is represented by t_2 to complete the processing before it begins. The number of tokens in p_2 represents the number of parts that the device represented by p_1 can handle. The number of tokens in the p_2 limits the number of tokens in place p_1 .

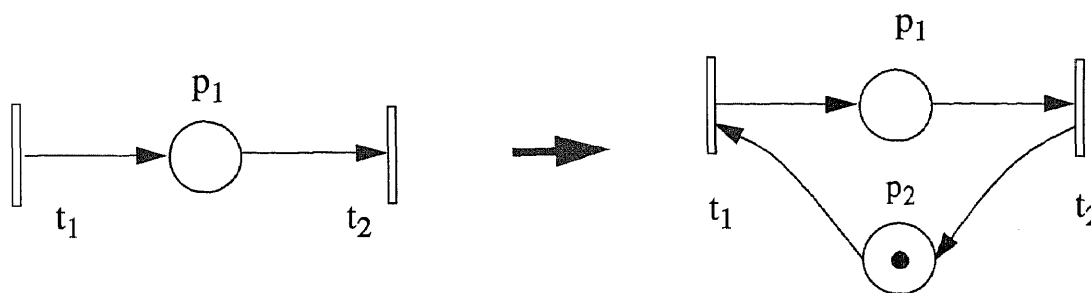


Figure 4.9 The Position Buffer is Added

Tokens in the places may represent material in the devices. In simulating the net shown in Figure 4.7, it can be observed during the simulation that 4 or 5 tokens can accumulate in the places p_1 , p_2 , p_7 and p_9 which represent the workstations and processing machines. The speed of processing is less than the transport speeds. In a machine shop, the number of parts waiting at a certain device must be limited to avoid overloading. The workstations and the processing machines also have a limited material handling capability. Therefore, the position buffers are added to workstations and processing machines as shown in Figure 4.10.

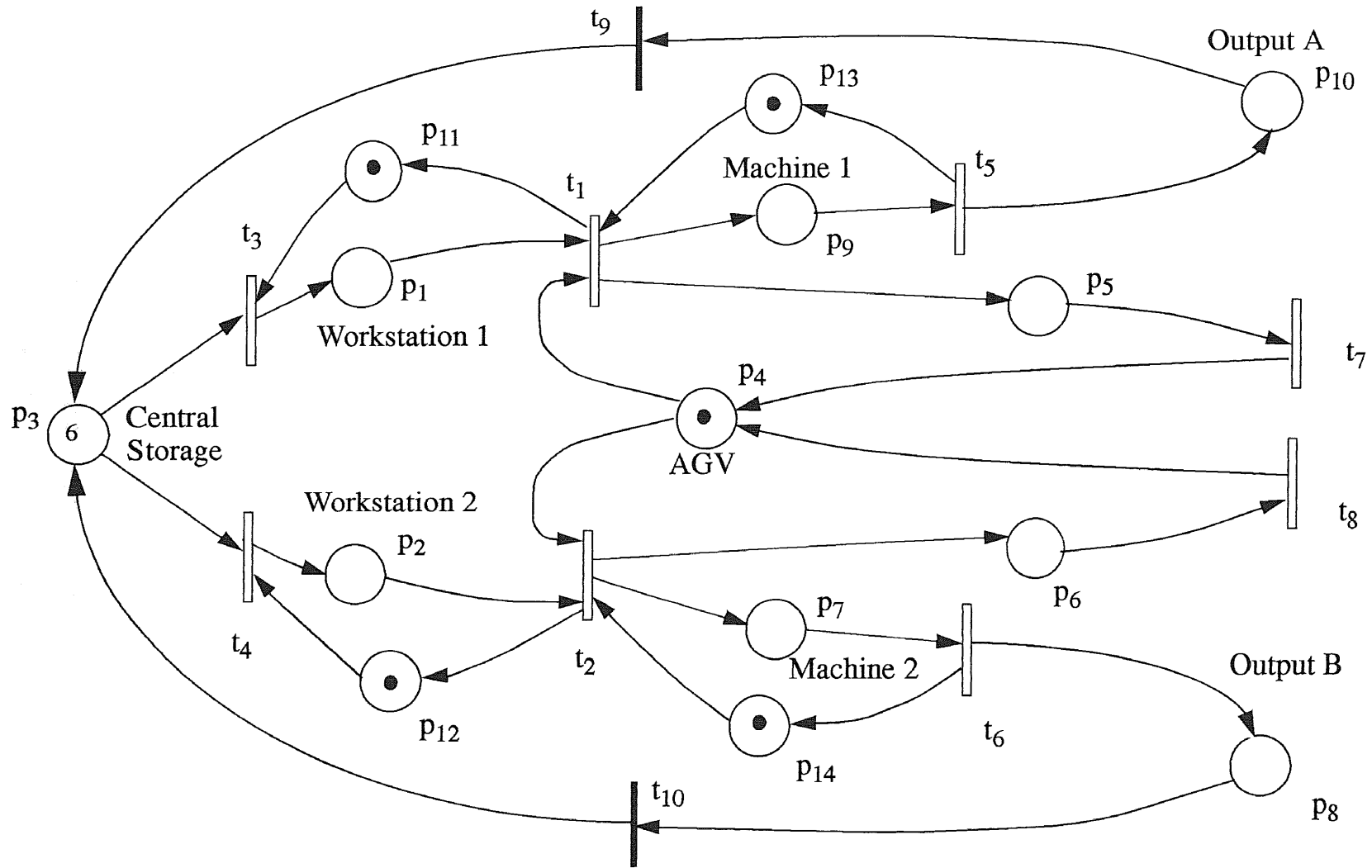


Figure 4.10A Petri Net Model of the FMS with Position Buffers Added

4.1.7 Simulation Results for Case 3 (6 tokens in p_3 to start)

The number of tokens in a position buffer represents the buffer capacity. If two tokens are placed in each position buffer, more parts can be held. Figure 4.11 shows the simulation results for the net in Figure 4.10 with one and with two tokens in each position buffer. The utilizations for both machines drop as t_7 delay increases. The two token case has a net higher production rate at all t_7 because the machines are less likely to wait for parts. At higher t_7 Machine 2 benefits from more conflict situations in the two-token case.

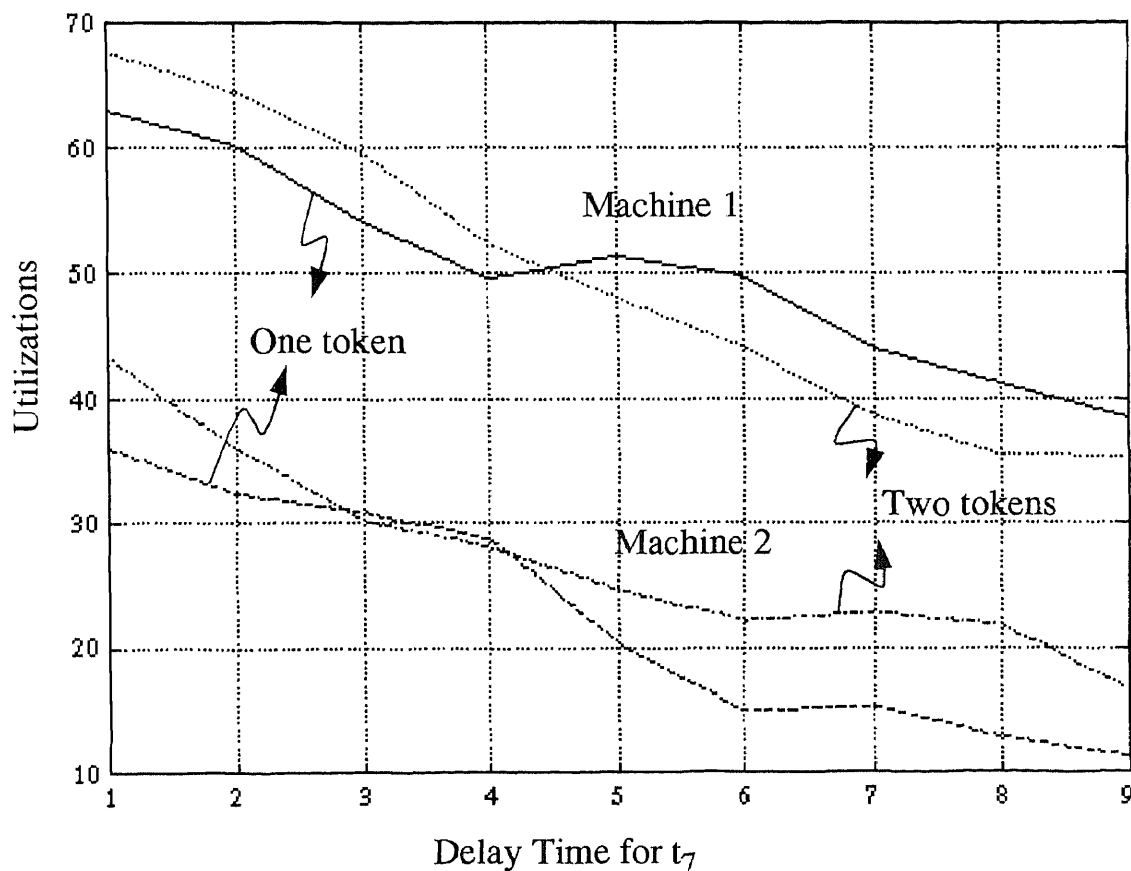


Figure 4.11 Utilizations with One and Two Tokens in Each Position Buffer

With one-token position buffers added, the place p_3 is occupied for 100% of simulation time. In a practical system, with output emerging at t_5 and t_6 , and

input material entering p_3 , the position buffer limits force excess material to accumulate in Central Storage.

As seen from this case, choosing a set of parameters for a Manufacturing system can be complex. For example, slowing down an AGV actually increased the production rate, showing that the basic design needed change. The trade-off between increasing local buffer capacity and system cost must be quantified. Given limited resources the designer has to consider where improvements are most cost-effective: buffer capacity, conveyer speed, AGV speed, machining speed, etc. Carefully chosen simulation runs can help the designer approach an optimal result.

4.2 Example Two

4.2.1 System Description

The system is adopted from a Westinghouse Electro-Mechanical-Division (EMD) FMS[11, 12]. The Westinghouse FMS integrates an entire manufacturing system which is capable of manufacturing from raw material to finished product. This system consists of an automated storage and retrieval system (ASRS), a flexible manufacturing cell (FMC), manual machining operations, assembly operations, seal testing, and inspection. The machining of characteristics that do not fit the unattended environment of the FMC are completed in the manual areas. Figure 4.12 shows a layout of the system. The ASRS stores raw material, work-in-process material, finished components and assemblies, and fixtures. A component can be delivered anywhere within the system through the ASRS. The functions of the overall manufacturing processes are under a central system control. Several manual processes within the system, such as assembly, lapping, grinding, deburring, and packing, could not be economically automated. An automated flexible machining cell

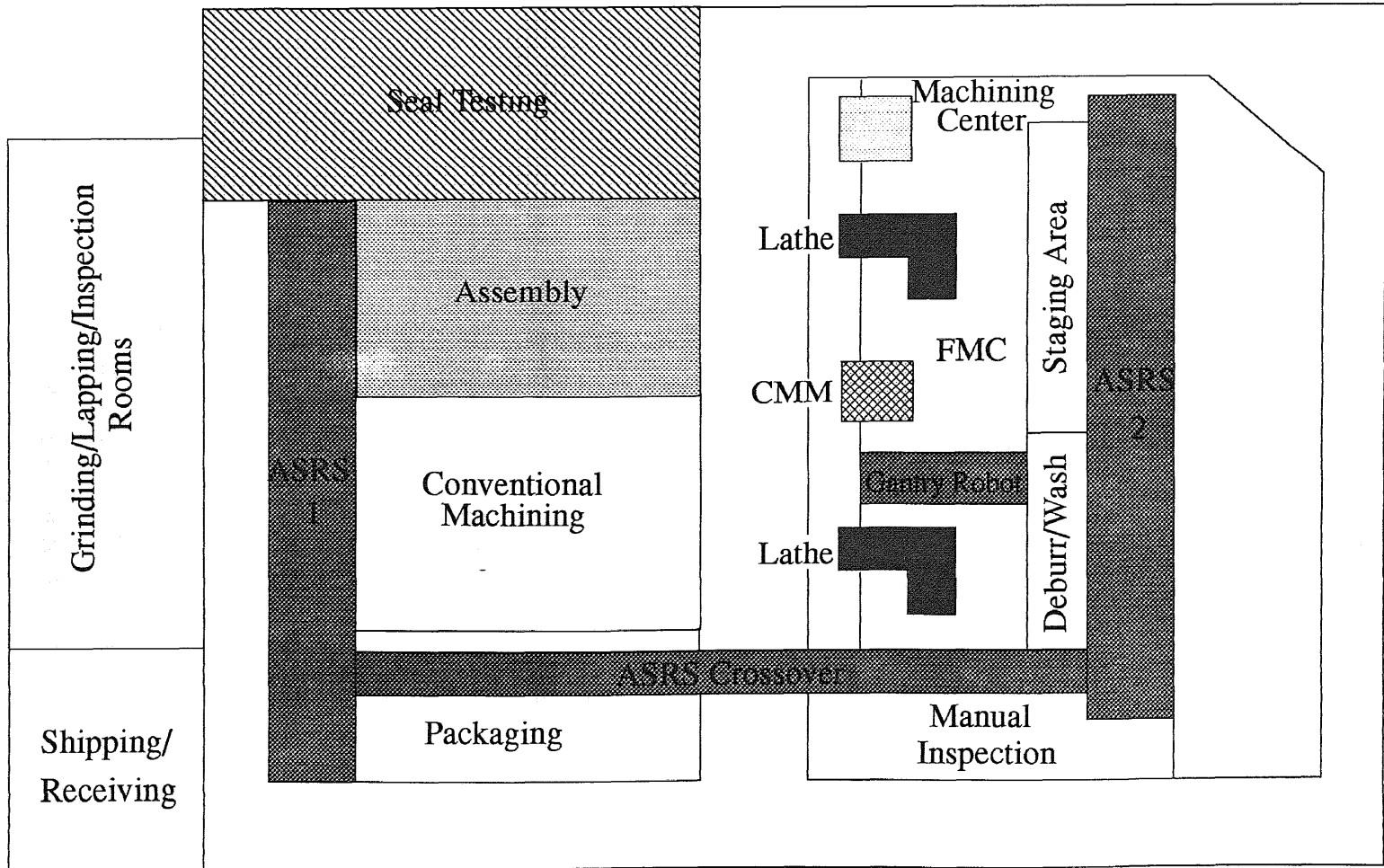


Figure 4.12 Westinghouse EMD Flexible Manufacturing System Layout

appears in the center of the layout. Raw material flows to conventional machining, grinding, lapping, and inspection operations first. Then parts are transported to the FMC for further work. After the FMC, parts are transported to the next manual area for further operations. The system operation procedure is depicted in Figure 4.13.

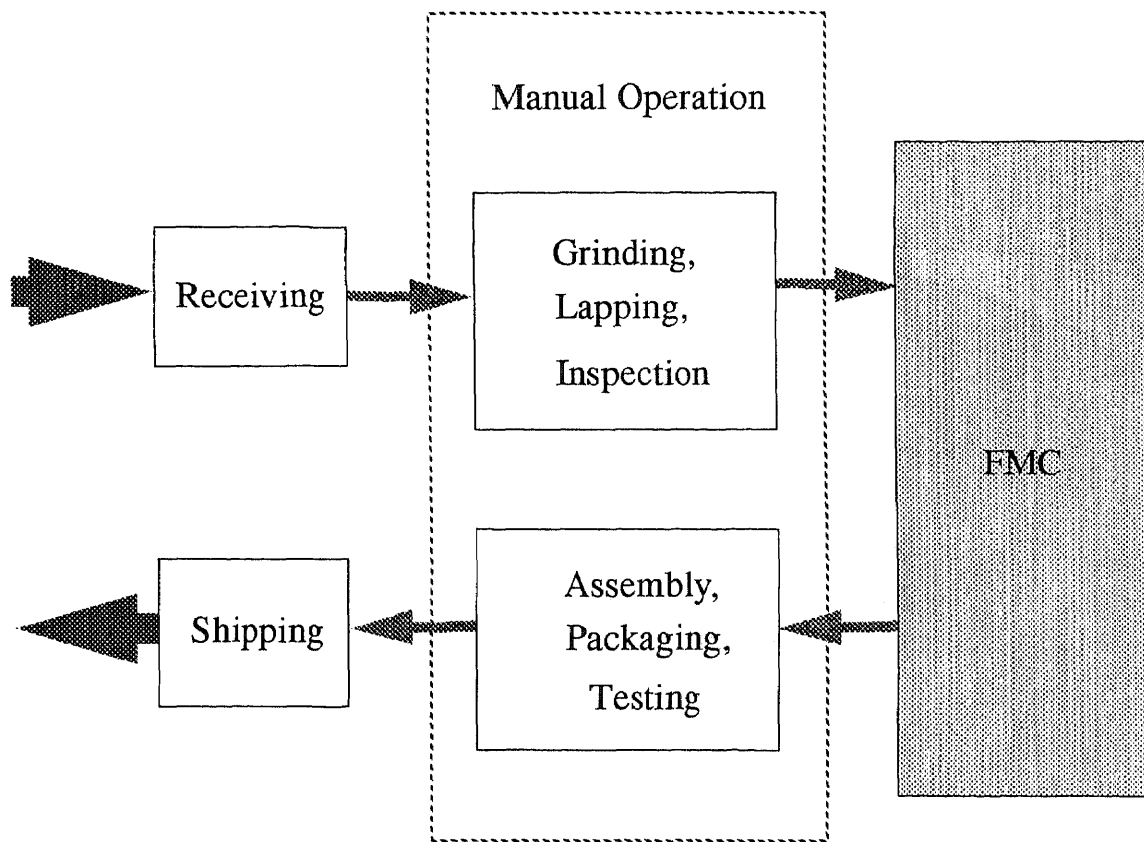


Figure 4.13 System Operation Procedure

The FMC cell is comprised of two lathes, one five-axis machining center, and a coordinate measuring machine (CMM). A gantry robot moves material within the cell. The general single-pass process for a part in the FMC is: load a part onto the buffer, perform a lathe operation, perform a machining center operation, wash the part, perform a CMM inspection, and unload the part. The

number of tools in the lathe and machining center must be limited. One gantry robot transports material to either lathes. Upon robot return from the lathe, it transports the products from the lathes to the following operations. The robot return times affect the utilizations of both lathes. The robot operation routine is shown in Figure 4.14. For simplicity, the operations after the lathes are represented only by “next operation.”

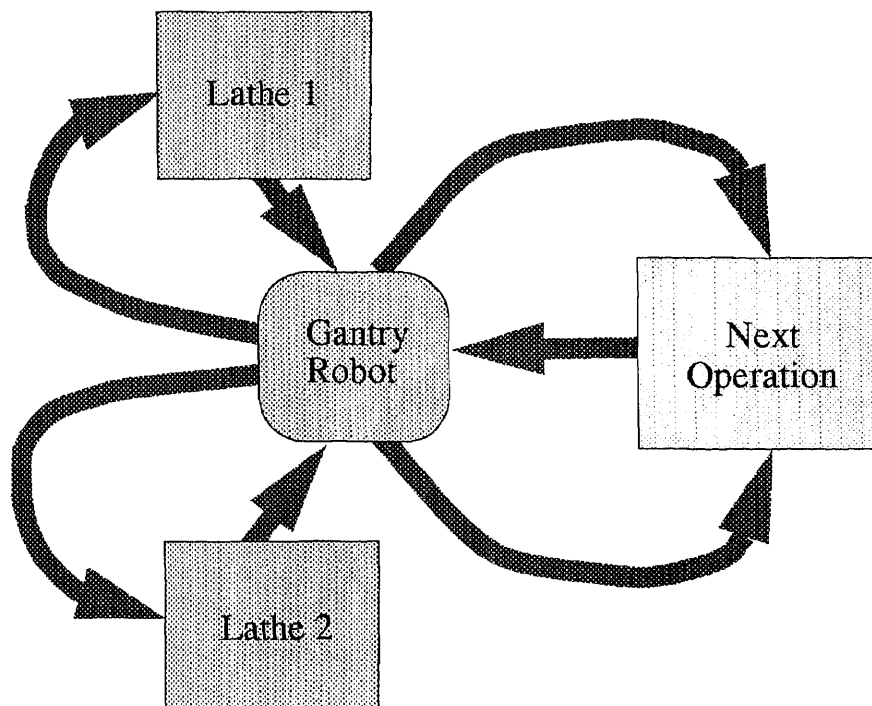


Figure 4.14 The Gantry Robot Working Routine

4.2.2 Petri Net Model of the FMC

Figure 4.15 shows the Petri net model of the FMC. Transitions t_3 and t_4 represent the return times from transport to the lathes, and transitions t_7 and t_8 represent the robot return times from transport to next operation. The representations of places and transitions are given in Tables 4.3 and 4.4; the parameters associated with the transitions are also given in Table 4.4. In this

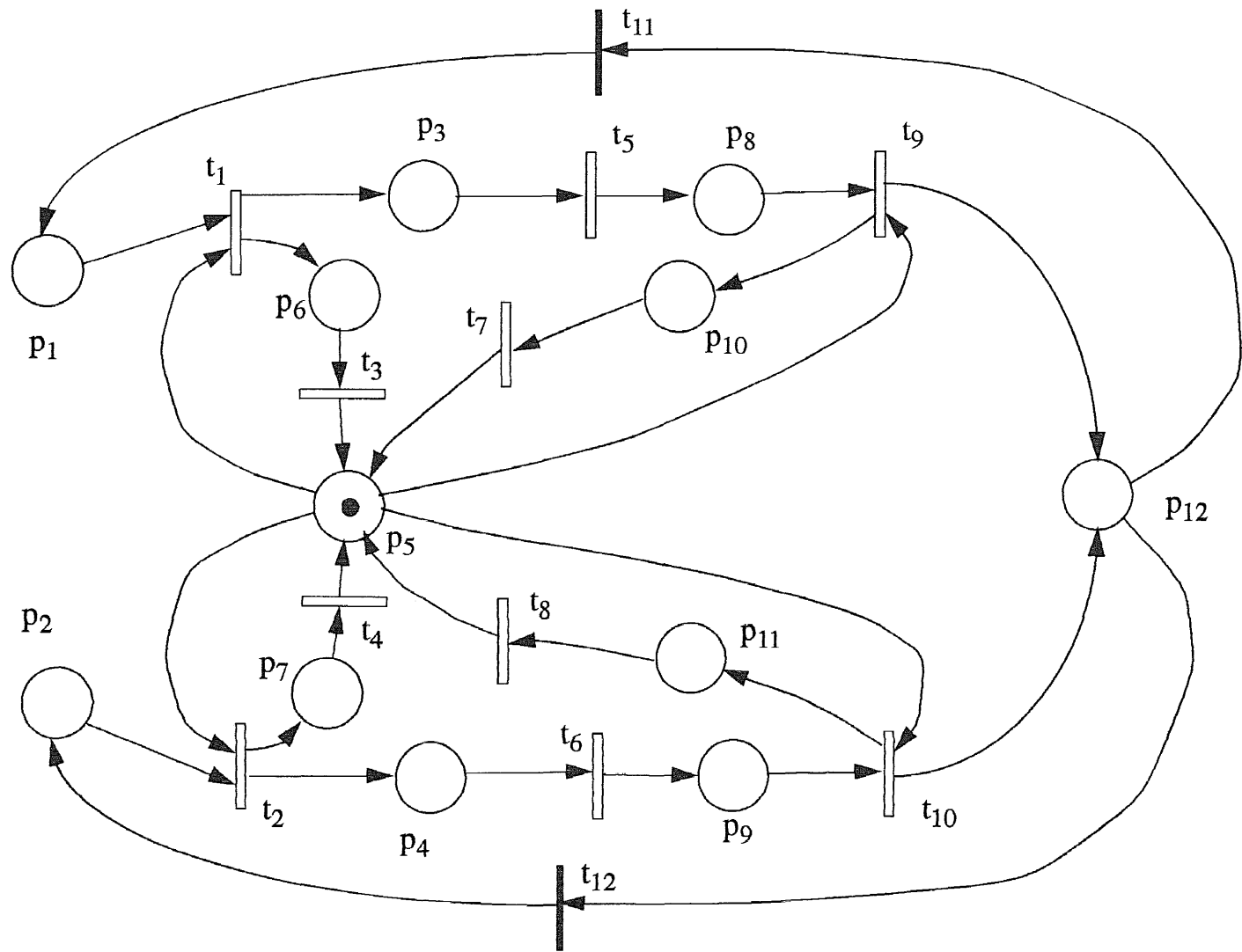


Figure 4.15 Petri Net Model for FMC

example the robot serves four distinct operations, unlike the previous example in which an AGV served two operations. The material supply to the FMC is represented by places p_1 and p_2 fed randomly from p_{12} to keep the simulation running. In this case there is a random pattern of arrivals at t_1 and t_2 , but not a continuously variable time of arrival.

Table 4.3: Places and Their Representations in FMC

Places #	Representations
p_1	Material supply to FMC
p_2	Material supply to FMC
p_3	Lathe 1
p_4	Lathe 2
p_5	Gantry robot
p_6	Buffer for robot return
p_7	Buffer for robot return
p_8	Lathe 1 output buffer
p_9	Lathe 2 output buffer
p_{10}	Buffer for Robot Return
p_{11}	Buffer for robot return
p_{12}	Next operation

4.2.3 Simulation Results for FMC

The net is simulated with different robot return times and different numbers of tokens in p_1 and p_2 at the start, which represent the material supplied to the FMC. The material supply to FMC will effect both lathe utilizations, because the more material supply, the more likely work comes to a lathe quickly when

a job completes. The net is simulated with one and two tokens in p_1 and p_2 each, (total of two and four), because the cell cannot hold more than 4 parts. The robot return time is either one or two units. The results are shown in Tables 4.18 and 4.19. The utilizations of both lathes decrease as robot return time goes from one to two. And the utilizations increase as number of tokens increase. The utilization of Lathe 1 is lower than that of Lathe 2 because its

Table 4.4: Transitions and Their Representations in FMC

Transitions #	Values	Representations	Types
t_1	1	Robot transport material to Lathe 1	Deterministic
t_2	1	Robot transport material to Lathe 2	Deterministic
t_3	1, 2	Time required for robot to return from transport to Lathe 1	Deterministic
t_4	1, 2	Time required for robot to return from transport to Lathe 2	Deterministic
t_5	10	Lathe 1 operation	Deterministic
t_6	12	Lathe 2 operation	Deterministic
t_7	1, 2	Time required for robot to return from transport product	Deterministic
t_8	1, 2	Time required for robot to return from transport product	Deterministic
t_9	2	Robot transport material to next operation from Lathe 1	Deterministic
t_{10}	3	Robot transport material to next operation from Lathe 2	Deterministic
t_{11}	N/A	Feedback from output	Immediate
t_{12}	N/A	Feedback from output	Immediate

processing time delay, t_7 , is two units faster than t_8 , and the material supply has the same likelihood to both lathes. Lathe 1 finishes processing the material two units ahead and spends more time waiting for material.

Table 4.5: Utilizations of Lathes with *One* Tokens in p_1 & p_2

Robot return times (x)	Lathe 1	Lathe 2
1	59.74%	59.87%
2	56.94%	57.54%

Table 4.6: Utilizations of Lathes with *Two* Tokens in p_1 & p_2

Robot return times (x)	Lathe 1	Lathe 2
1	79.25%	86.45%
2	58.88%	83.13%

4.2.4 Petri Net Model for FMS

The Petri net model for entire the FMS is shown in Figure 4.17, feedback transitions t_{11} and t_{12} in Figure 4.16 are omitted. The manual operation between material receipt and the FMC is represented by one place p_{14} and the manual operation between the FMC and shipping is represented by p_{15} . The places from p_1 to p_{10} are as shown in Figure 4.16. The rest of the representations for places and transitions are given in Table 4.5 with the parameters for transitions.

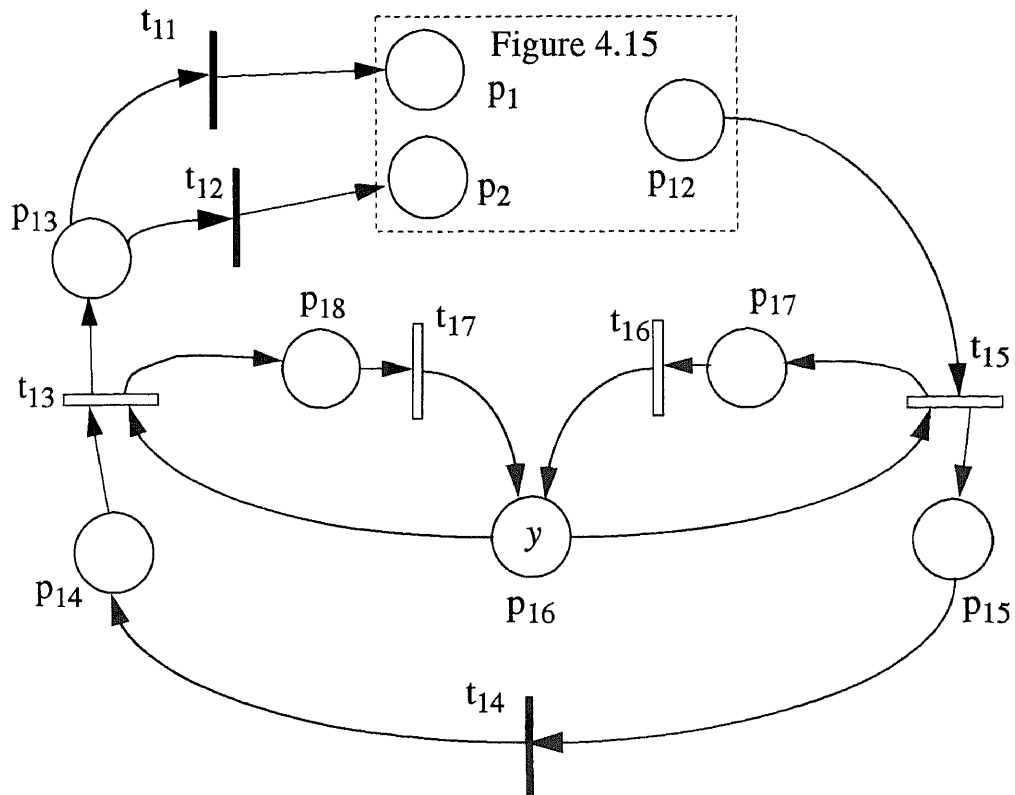


Figure 4.16 Petri Net Model of FMS

Table 4.7: Places and Their Representations in Figure FMS

Places #	Representations
p13	Material supply to FMC
p14	Operation between FMC and shipping
p15	Operation between receiving and FMC
p16	ASRS available
p17	ASRS return buffer
p18	ASRS return buffer

Table 4.8: Transitions and Their Representations in FMS

Transitions #	Values	Representations	Types
t_{11}	N/A	Commite material to FMC	Immediate
t_{12}	N/A	Commite material to FMC	Immediate
t_{13}	11	ASRS transports material to FMS	Deterministic
t_{14}	N/A	Feedback from output	Immediate
t_{15}	10	ASRS transport product from FMS	Deterministic
t_{16}	1	ASRS return time from t_{15}	Deterministic
t_{17}	1	ASRS return time from t_{13}	Deterministic

4.2.5 Simulation Results for FMS

The number of tokens in p_{12} and number of tokens in p_{14} represent the material supply to FMS and the capability of ASRS. The net is simulated with different numbers of token in p_{14} . and the different gantry robot return times. The simulation results are given in Tables 4.9 and 4.10.

Table 4.9: Utilizations of Lathes with y tokens in P_{14}

Tokens in p_{16} (y)	Lathe 1	Lathe 2
2	28.74%	32.52%
3	28.80%	32.58%

Table 4.10: Utilizations of Lathes with y tokens in P_{14}

Tokens in p_{16} (y)	Lathe 1	Lathe 2
2	46.93%	44.04%
3	45.91%	54.61%

The utilization of lathes is higher with four tokens in p_{14} than with two, because more tokens in p_{14} means the more parallel operations are possible. But the increases are not very significant. The quantity of material supply can raise the utilization more by permitting overlapping operations. This is a main cause for the utilization improvement. The utilization files can be found in the Appendix Figures A.7 to A.10.

CHAPTER 5

CONCLUSION

5.1 Summary

A flexible manufacturing system can perform more efficiently if it is optimized. The TPNS tool is useful for prototypes and later optimization of an FMS.

Using TPNS to execute a timed Petri net model of FMS, the behavior of every device or process which is represented by a place or transition in the Petri net is captured in the simulation results. The utilization file describes the system behaviors well. Analyzing the data from the output files, system critical parameters can be identified and beneficial changes can be made to them. A log file records all the steps of the simulation and well describes the system behavior in detail, when required. With the visual display of net status, intuition for the system behavior can be obtained. For example, net execution can reveal obvious problems such as buffer overloading.

In one example, the results of better performance with slower a AGV is explained by net analysis, revealing a poor design concept. The analysis includes the different aspects that effect device performance in the system in Westinghouse example.

During this work, several serious bugs in TPNS were identified, and some of them were fixed. One remaining problem is the allocation of memory without later release, resulting in eventual premature termination of a run. A second problem is improper conflict resolution with certain complex structures. The main structure of the program remains unchanged.

5.2 Future Version of TPNS

In developing of the future version of TPNS, better structured and documented source code is suggested. The object oriented program style will make changes easier.

The integration of color Petri net capability with TPNS would permit easier modeling of scheduling situation[]. The new tool will be developed with C++ under X-Window environment. The combination of TPNS and CPNS will make the tool more portable.

APPENDIX

Utilization Files for Some Points in Figures 4.2 and 4.15

Figure A.1 Utilization File of Net in Figure 4.2 at $t_7 = 2$ with 4 Tokens in p_3

Figure A.2 Utilization File of Net in Figure 4.2 at $t_7 = 3$ with 4 Tokens in p_3

Figure A.3 Utilization File of Net in Figure 4.2 at $t_7 = 4$ with 4 tokens in p_3

Figure A.4 Utilization File of Net in Figure 4.2 at $t_7 = 2$ with 6 Tokens in p_3

Figure A.5 Utilization File of Net in Figure 4.2 at $t_7 = 3$ with 6 tokens in p_3

Figure A.5 Utilization File of Net in Figure 4.2 at $t_7 = 4$ with 6 tokens in p_3

Figure A.7 Utilization of Figure 4.15 with *One* Token in p_1 & p_2 and $x = 1$

Figure A.8 Utilization of Figure 4.15 with *One* Token in p_1 & p_2 and $x = 2$

Figure A.9 Utilization of Figure 4.15 with *Two* Tokens in p_1 & p_2 and $x = 1$

Figure A.10 Utilization of Figure 4.15 with *Two* Tokens in p_1 & p_2 and $x = 2$

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 1426
Total number of steps : 2000
*****
Transition Utilization / Behavior

```

Transition Tag	No.of times Fired	Busy*	Involvement in conflict	Success
T1	286	14.30% ***	74	43
T2	213	10.65% ***	74	31
T3	287	50.37%	210	108
T4	213	61.67%	210	102
T5	284	95.83%	0	0
T6	213	42.97%	0	0
T7	286	38.47%	0	0
T8	213	28.65%	0	0
T9	284	14.20% ***	0	0
T10	213	10.65% ***	0	0

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behavior

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	287	12.31%
P2	213	11.43%
P3	497	62.27%
P4	499	32.89%
P5	286	12.51%
P6	213	7.60%
P7	213	24.48%
P8	213	0.00%
P9	286	94.22%
P10	284	0.00%

Initial marking not reached during net execution.

Figure A.1 Utilization File of Net in Figure 4.2 at $t_7 = 2$ with 4 Tokens in p_3

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 1426
Total number of steps : 2000
*****
Transition Utilization / Behavior

```

Transition Tag	No.of times Fired	Busy*	Involvement in Success conflict
T1	275	13.70% ***	75
T2	230	11.50% ***	62
T3	287	54.14%	104
T4	213	63.95%	105
T5	284	91.79%	0
T6	213	46.06%	0
T7	286	55.07%	0
T8	213	30.71%	0
T9	284	13.70% ***	0
T10	213	11.45% ***	0

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behavior

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	276	27.70%
P2	230	30.04%
P3	503	62.48%
P4	505	14.22%
P5	275	28.30%
P6	230	9.21%
P7	230	30.71%
P8	230	0.00%
P9	275	87.12%
P10	274	0.00%

Initial marking not reached during net execution.

Figure A.2 Utilization File of Net in Figure 4.2 at $t_7 = 3$ with 4 Tokens in p_3

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 1605
Total number of steps : 2000
*****
Transition Utilization / Behaviour

```

Transition Tag	No. of times Fired	Busy*	Involved in conflict	Success
T1	277	13.85%	217	103
T2	214	10.70%	217	114
T3	279	50.59%	207	101
T4	215	58.19%	207	106
T5	276	86.29%	0	0
T6	214	40.00%	0	0
T7	277	69.03%	0	0
T8	214	26.67%	0	0
T9	276	13.80%	0	0
T10	214	10.70%	0	0

* computed as : (total transition delay X 100) / total simulation time.

** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behaviour

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	105	56.57%
P2	99	46.11%
P3	105	58.26%
P4	99	4.30%
P5	407	43.30%
P6	105	7.91%
P7	99	26.17%
P8	105	0.00%
P9	99	77.20%
P10	99	0.00%

_Initial marking not reached during net execution.

Figure A.3 Utilization File of Net in Figure 4.2 at $t_7 = 4$ with 4 tokens in p_3

```

*****
          NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 1487
Total number of steps : 2000
*****

```

Transition Utilization / Behavior

Transition Tag	No.of times Fired	Busy*	Involvement	Success
T1	299	14.95% ***	89	52
T2	214	10.70% ***	89	37
T3	299	52.05%	195	97
T4	214	64.43%	195	98
T5	296	99.87%	0	0
T6	213	42.97%	0	0
T7	299	40.22%	0	0
T8	213	28.65%	0	0
T9	296	14.80% ***	0	0
T10	213	10.65% ***	0	0

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behavior

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	299	15.13%
P2	214	12.71%
P3	509	63.35%
P4	512	31.14%
P5	299	12.64%
P6	214	7.80%
P7	214	24.55%
P8	213	0.00%
P9	299	99.73%
P10	296	0.00%

Initial marking not reached during net execution.

Figure A.4 Utilization File of Net in Figure 4.2 at $t_7 = 2$ with 6 Tokens in p_3


```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 1473
Total number of steps : 2000
*****

```

Transition Utilization / Behavior

Transition Tag	No.of times Fired	Busy*	Involved in conflict	Success
T1	291	14.55% ***	255	138
T2	243	12.15% ***	255	117
T3	292	61.17%	173	85
T4	243	63.07%	173	88
T5	288	98.10%	0	0
T6	241	49.29%	0	0
T7	291	59.27%	0	0
T8	242	32.86%	0	0
T9	288	14.40% ***	0	0
T10	241	12.05% ***	0	0

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behavior

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	292	50.64%
P2	243	48.47%
P3	529	67.07%
P4	533	7.88%
P5	291	30.75%
P6	243	10.05%
P7	243	32.99%
P8	241	0.00%
P9	291	96.20%
P10	288	0.00%

Initial marking not reached during net execution.

Figure A.5 Utilization File of Net in Figure 4.2 at $t_7 = 3$ with 6 tokens in p_3

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 2079
Total number of steps : 2000
*****
Transition Utilization / Behaviour

```

Transition Tag	No. of times Fired	Busy*	Involved in conflict	Success
T1	309	15.45%	446	235
T2	265	13.25%	446	211
T3	311	44.59%	315	153
T4	267	54.16%	315	162
T5	309	74.31%	0	0
T6	265	38.24%	0	0
T7	309	74.31%	0	0
T8	265	25.49%	0	0
T9	308	15.40%	0	0
T10	265	13.25%	0	0

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behaviour

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	311	85.28%
P2	267	81.58%
P3	573	44.40%
P4	574	0.19%
P5	309	47.04%
P6	265	7.79%
P7	265	27.22%
P8	265	0.00%
P9	309	47.04%
P10	309	0.00%

_Initial marking not reached during net execution.

Figure A.6 Utilization File of Net in Figure 4.2 at $t_7 = 4$ with 6 tokens in p_3

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 3147
Total number of steps : 2000
Petri Net File name : sys92.utilize
*****
Transition Utilization / Behavior

```

Transition Tag	No.of times Fired	Busy*	Involved in conflict	Success
T1	188	5.97%	34	16
T2	157	4.99%	42	19
T3	188	5.97%	0	0
T4	157	4.99%	0	0
T5	188	59.74%	0	0
T6	157	59.87%	0	0
T7	187	5.94%	0	0
T8	157	4.99%	0	0
T9	188	11.95%	51	29
T10	157	14.97%	5	2
T11	187	9.35% ***	345	187
T12	158	7.90% ***	345	158

* computed as:(total transition delay X 100) / total simulation time

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behavior

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	187	7.91%
P2	158	7.02%
P3	188	59.74%
P4	157	59.87%
P5	689	78.11%
P6	188	5.97%
P7	157	4.99%
P8	188	15.13%
P9	157	15.54%
P10	188	5.94%
P11	157	4.99%
P12	345	0.00%

* Computed as : (total place occupancy X 100) / total simulation time.

Initial marking not reached during net execution.n.

Figure A.7 Utilization of Fig. 4.15 with *One* Token in p_1 & p_2 and $x=1$

NET UTILIZATION AND BEHAVIOR

CLOCK reached : 3337

Total number of steps : 2000

Petri Net File name : sys92_2.utilize

Transition Utilization / Behavior

Transition Tag	No.of times Fired	Busy* conflict	Involved in	Success
T1	190	5.69%	41	21
T2	160	4.79%	44	22
T3	190	11.39%	0	0
T4	160	9.59%	0	0
T5	189	56.94%	0	0
T6	160	57.54%	0	0
T7	189	11.33%	0	0
T8	159	9.53%	0	0
T9	189	11.33%	59	29
T10	159	14.29%	0	0
T11	189	9.45% ***	348	189
T12	159	7.95% ***	348	159

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behaviour

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	189	9.05%
P2	159	8.24%
P3	190	56.79%
P4	160	57.54%
P5	698	57.93%
P6	190	11.39%
P7	160	9.59%
P8	189	16.24%
P9	160	14.83%
P10	189	11.33%
P11	159	9.53%
P12	348	0.00%

* Computed as : (total place occupancy X 100) / total simulation time.

Initial marking not reached during net execution.

Figure A.8 Utilization of Fig. 4.15 with *One* Token in p_1 & p_2 and $x=2$

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 2410
Total number of steps : 2000
Petri Net File name : sys93.utilize
*****

```

Transition Utilization / Behaviour

Transition Tag	No.of times Fired	Busy*	Involved in conflict	Success
T1	191	7.93%	91	45
T2	171	7.10%	92	51
T3	191	7.93%	0	0
T4	171	7.10%	0	0
T5	190	79.25%	0	0
T6	169	84.65%	0	0
T7	190	7.88%	0	0
T8	168	6.97%	0	0
T9	190	15.77%	101	46
T10	169	21.04%	38	19
T11	189	9.45% ***	358	189
T12	169	8.45% ***	358	169

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behaviour

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	189	14.19%
P2	169	11.83%
P3	191	79.00%
P4	171	84.27%
P5	720	68.76%
P6	191	7.93%
P7	171	7.10%
P8	190	24.98%
P9	169	24.23%
P10	190	7.88%
P11	169	6.97%
P12	359	0.00%

* Computed as : (total place occupancy X 100) / total simulation time.

Initial marking not reached during net execution.

Figure A.9 Utilization of Fig. 4.15 with *Two* Tokens in p_1 & p_2 and $x=1$

```

*****
NET UTILIZATION AND BEHAVIOR
*****
CLOCK reached ..... : 2887
Total number of steps : 2000
Petri Net File name : sys93_2.utilize
*****
Transition Utilization / Behavior

```

Transition Tag	No.of times Fired	Busy*	Involved in conflict	Success
T1	170	5.89%	235	115
T2	200	6.93%	274	124
T3	170	11.78%	0	0
T4	200	13.86%	0	0
T5	169	58.88%	0	0
T6	199	83.13%	0	0
T7	168	11.64%	0	0
T8	198	13.72%	0	0
T9	168	11.64%	243	107
T10	199	20.68%	235	106
T11	168	8.40% ***	366	168
T12	198	9.90% ***	366	198

* computed as : (total transition delay X 100) / total simulation time.

*** immediate transition. computed as: (total firing w.r.t total steps)

Place Utilization / Behavior

Place Number	Total number of tokens entered this Place	% Duration for which Place was occupied
P1	168	25.63%
P2	198	31.94%
P3	170	58.71%
P4	200	82.99%
P5	736	47.14%
P6	170	11.78%
P7	200	13.86%
P8	169	35.54%
P9	199	40.87%
P10	168	11.64%
P11	199	13.72%
P12	367	0.00%

* Computed as : (total place occupancy X 100) / total simulation time.

Initial marking not reached during net execution.

Figure A.10 Utilization of Fig. 4.15 with *Two* Tokens in p_1 & p_2 and $x=2$

REFERENCES

1. Peterson, James L. "Petri Net Theory and the Modeling of Systems." *Prentice-Hall Inc. Englewood Cliffs, New Jersey* (1981).
2. Murata, Tadao. "Petri Nets: Properties, Analysis and Applications." *Proceedings of The IEEE, Vol. 77, No. 4, April* (1989): 541-579.
3. Gilanli, Arsalan. "A Graphical Editor for Petri Nets." *Master's Thesis, Electrical Engineering Department, New Jersey Institute of Technology*, (1989).
4. Shukla, Ashish and Anthony Robbi. "A Petri Net Simulation Tool." *Proceedings of 1991 IEEE Int. Conference on Systems, Man, and Cybernetics, Vol. 1, No.1, October* (1991): 361-366.
5. Siddiqi, Javaid Aslam. "A Graphical Tool for the Simulation of Timed Petri Nets." *Master's thesis, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, December* (1991).
6. Zhou, MengChu, Kevin McDermott, Paresh A. Patel and Tenian Tang. "Construction of Petri Net Based Mathematical Models of an FMS Cell." *Proceedings of 1991 IEEE Int. Conference on Systems, Man, and Cybernetics, Vol. 1, No.1, October* (1991): 367-372.
7. Someya, Harushi, Tsutomu Tashiro, Tomohiro Murata, and Norihisa Komoda. "Performance Evaluation of Job Operation Flows in Computer System by Timed Petri Nets." *Proceedings of the 3rd International Workshop on Petri Nets and Performance Models, Kyoto, Japan, December* (1989): 104-111.
8. Menga, Giuseppe, Giorgio Bruno, Renato Conterno, and Massimo Actis Dato. "Modeling FMS by Closed Queuing Network Analysis Methods." *IEEE Transactions on Components, Hybrids, and Manufacturing Technology, Vol. 7, No. 3, September* (1984): 241-248.
9. Davis, Robert P. and W. J. Kennedy Jr. "Markovian Modeling of Manufacturing Systems." *International Journal of Production Research, Vol. 25, No. 3, March* (1987): 337-351.
10. Dicesare, Frank and Alan A. Desrochers. "Modeling, Control, And Performance Analysis of Automated Manufacturing Systems." *Control and Dynamic Systems, Vol. 47, Academic Press Inc.* (1991).

11. Arrigo, Thomas J. "Planning for Flexible Manufacturing." *CASA/SME Flexible Manufacturing Systems Conference, March (1985)*.
12. Cable, Harold. "A FMS Tooling and Fixturing Package That Provides Flexibility and Chip Control." *SMF Advanced Machining Technology for Cells and FMS Conference, February (1986)*.
13. Desai, Sanjay. "Coloered Petri Net Simulator." *Master's Thesis, Electrical Engineering Department, New Jersey Institute of Technology, (1991)*.