Winter 1993

The Use of a Factory Simulation to Evaluate a Flexible Control Structure for Integrated Manufacturing

Ling Ling Pan
Old Dominion University

Recommended Citation

Pan, Ling L.. "The Use of a Factory Simulation to Evaluate a Flexible Control Structure for Integrated Manufacturing" (1993). Doctor of Philosophy (PhD), Dissertation, Engineering Management & Systems Engineering, Old Dominion University, DOI: 10.25777/q53c-8j64
https://digitalcommons.odu.edu/emse_etds/167

THE USE OF A FACTORY SIMULATION TO EVALUATE A FLEXIBLE CONTROL STRUCTURE FOR INTEGRATED MANUFACTURING

by

Ling Ling Pan
B.S.C.S. August 1983, Tianjin University
M.S. December 1988, Old Dominion University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

ENGINEERING MANAGEMENT

OLD DOMINION UNIVERSITY
December 1993

Approved by:

Dr. Laurence D. Richards (Co-Director)

Dr. Derya A. Jacobs (Co-Director)

Dr. Billie M. Reed

Dr. Larry W. Wilson

ABSTRACT

THE USE OF A FACTORY SIMULATION TO EVALUATE A FLEXIBLE CONTROL STRUCTURE FOR INTEGRATED MANUFACTURING

Ling Ling Pan
Old Dominion University, 1993
Co-Directors: Dr. Laurence D. Richards
Dr. Derya A. Jacobs

Once a control structure for an integrated manufacturing system is decided upon, manufacturing activities are limited by that structure. A flexible control structure is presented as an approach for accommodating a variety of manufacturing activities, without being limited to a single control structure. A flexible control structure is one that allows multiple types of control structure in the manufacturing process. For example, both hierarchical and non-hierarchical structures may be used in a flexible structure. The properties of a flexible control structure are discussed from the point of view of graph theory.

Control structures for automated manufacturing are difficult to evaluate without actually setting up a pilot production system. Since this is often not possible for reasons of expense or equipment availability, it would be advantageous to be able to simulate alternative control structures for their various characteristics. In this research, flexible control is demonstrated with a factory simulation of an automated on-line/post-process inspection system. Factory simulations present special problems when used for evaluation purposes. An approach to using a factory simulation is developed, and alternative control structures are evaluated with respect to their fault tolerance characteristics. The results of this research indicate that flexible control may be cost effective when a large variety of manufacturing activities must be accommodated, but further research is needed to confirm precisely how wide a range and what types of activities would justify this approach.

TABLE OF CONTENTS

Page

LIST OF TABLES ---iv

LIST OF FIGURES -- v

ACKNOWLEDGMENTS--vi

Chapter

1. INTRODUCTION -- 1

2. LITERATURE REVIEW ----------------------------------- 4

3. THE RESEARCH PROBLEM --------------------------------10

    Statement of the Problem ------------------------10
    Research Objectives -----------------------------12

4. A MATHEMATICAL MODEL FOR FLEXIBLE CONTROL ----------14

    Definition--------------------------------------16
    Properties--------------------------------------16

5. RESEARCH METHODOLOGY --------------------------------21

    Design of the Simulation ------------------------21

        Simulation Goals----------------------------22
        Input Data and Initial Conditions-----------22
        Inspection System---------------------------23
        RCV System with Different Control Structures ---------------------------------24
        Fault Tolerance ----------------------------32
        Output Data Analysis and Hypothesis Testing-------------------------------------36

    Simulation Features -----------------------------39
    Simulation Tool: PlantWorks 2.0------------------41
    Simulation Tasks --------------------------------42

6. EVALUATION OF THE CONTROL STRUCTURES ---------------48

Evaluation of Fault Tolerance ------------------48

System Data --------------------------------48
Sampling Technique -------------------------49
Data Analysis Methods-----------------------49

Evaluation of Features -------------------------71

7. CONCLUSION AND FUTURE RESEARCH ---------------------75

Conclusion -------------------------------------75
Future Research --------------------------------77

LIST OF REFERENCES ---------------------------------------78

APPENDICES

A. The Hardware Configuration ---------------------83
B. Program List -----------------------------------88
C. Preparation for Using the RCV Software ---------95
D. Simulation and Data Analysis Results -----------96
E. Menus ---------------------------------------106

AUTOBIOGRAPHICAL STATEMENT -------------------------------109

LIST OF TABLES

Table Page

1. Mean System Times--------------------------------------38

2. Mean System Times
(for ten part inspections)-------------------------50

3. Standard Error of System Time-------------------------51

4. Mean System Times for Eight Control Structures -------63

5. Mean System Times with Fault Tolerance Cases
(for eight control structures)--------------------64

6. Results of Analysis of Variance
(for eight control structures)--------------------65

7. ANOVA and Duncan Data Analysis Results
(for eight control structures)--------------------66

8. Mean System Times with Fault Tolerance Cases
(for seven control structures)--------------------68

9. Results of Analysis of Variance
(for seven control structures)--------------------69

10. ANOVA and Duncan Data Analysis Results
(for seven control structures)--------------------70

11. Number of Ways to Perform Operations------------------73

12. System Expandability----------------------------------74

LIST OF FIGURES

Figure Page

1. Flexible Control Graph----------------------------15
2. Central Control Structure-------------------------25
3. Sequential Control with Computer------------------27
4. Sequential Control with no Computer---------------28
5. Hierarchical Control Structure--------------------30
6. Flexible Control Structure------------------------31
7. RCV Inspection System Simulation------------------43
8. RCV with Sequential Control (no computer)---------54
9. RCV with Sequential Control (with computer)--------55
10. RCV with Central Control--------------------------56
11. RCV with Hierarchical Control---------------------57
12. RCV with Flexible Sub-Sequential Control (no computer)----------------------------------58
13. RCV with Flexible Sub-Sequential Control (with computer)--------------------------------59
14. RCV with Flexible Sub-Hierarchical Control--------60
15. RCV with Flexible Sub-Central Control-------------61

ACKNOWLEDGMENTS

I would like to acknowledge Dr. Laurence D. Richards and Dr. Derya A. Jacobs, co-supervisors of this dissertation, for their technical guidance and support, and Dr. Billie M. Reed and Dr. Larry W. Wilson, dissertation committee members, for their significant contributions. I also wish to express my appreciation to Dr. Han P. Bao and Dr. Mark M. Pardue for their support and the IBM PlantWorks support group for its help.

CHAPTER 1

INTRODUCTION

A Computer Integrated Manufacturing (CIM) environment consists of the integration and automation of product information flow from a product's conception through design, production, inspection, marketing, shipment, and support. Automation, as an important aspect of CIM, is the application of mechanical, electronic, and computer-based systems to accomplish the processing, assembly, material handling, and inspection that are performed on the product. Inspection is a quality control operation that involves the checking of parts, assemblies, or products for conformance to certain criteria generally specified by a design engineering department [1].

Automated on-line/post-process inspection implies that the measurement or gaging procedure is accomplished automatically and immediately following the production process. Even though it follows the production process, it is still considered an on-line method because it is integrated with the manufacturing workstation, and the results of the inspection can immediately influence the production operation

[2]. If the inspection system is structured flexibly, it directly enhances the integration of manufacturing activities.

There have been many kinds of control structures utilized in automated/integrated manufacturing environments [3, 4, 5, 6, 7]. Traditionally, once a control structure is decided upon, certain manufacturing activities are limited by that structure. And, this limitation can no longer be tolerated in the multi-product manufacturing systems of the future. Therefore, a flexible control structure has been proposed and studied in this research. With flexible control, multi-product manufacturing can be carried out because a flexible structure allows different structures to be applied, such as both hierarchical and non-hierarchical structures.

Following an overview of traditional control structures, the concept of a flexible control structure is introduced, and its properties are discussed from the point of view of graph theory. In this research, an inspection system, called RCV (Robot, Computer, Vision), is presented as an example of the flexible control application. The inspection system consists of a PUMA 762 Robot Arm, an IBM 7552 Industrial Computer, and an IRI D256 Machine Vision system (see Appendix A). The RCV system configuration provides a multiple task environment. The amount of integrated equipment in the system is expandable, and the structure of the system is flexible.

The application is simulated with a functional language supported by the IBM PlantWorks software. This kind of simulation is a factory simulation. Five different control structures are tested with the RCV inspection system:

1. Central control,
2. Sequential control with computer,
3. Sequential control with no computer,
4. Hierarchical control, and
5. Flexible control.

Another aspect of this study will be to evaluate the alternative control structures with respect to their fault tolerance characteristics.

CHAPTER 2

LITERATURE REVIEW

In 1989, the U.S. Air Force's San Antonio Air Logistics Center [4] (Kelly AFB, San Antonio), established a new integrated blade inspection system (IBIS) for jet engine turbine and compressor blades, which included two infrared inspection modules (IRIM) and automated fluorescent penetrant inspection modules to detect cracks and other abnormalities in the surfaces of used blades. The IBIS is designed as a central control structure. A single operator controls all IBIS functions, loading and unloading parts onto the system's conveyor, and monitoring the entire operation from a central console. The advantage of the IBIS is that it integrates different inspections within the blade production cycle so that the defects can be found and corrected as early as possible. For example, an IRIM inspection is added at the point of the drilling cell in the blade production cycle. The misdrilled or dogged blades are able to be identified. The necessary repairs can be made before additional processing such as coating or grinding takes place. The result is a substantial reduction in unnecessary labor, rework and scrap.

However, if something is wrong with the central control, all of the controlled functions and controlled activities stop.

The J. Lynmar Mfg. Co. (Masontown, PA) [5] developed an integrated robotics/vision automated handling system for flywheel processing. There are three sequential workstations in this process, a lathing station, a drilling and tapping station, and a balancing station. Each station has one robot to perform loading and unloading work. There is a conveyor between each workstation. The vision system provides accurate part offset information to the robots. Not only does the system locate and orient parts for automatic loading in the machine centers, but use of the system has also resulted in reduced fixturing requirements and minimized tool changeover when processing different parts. Because all three robots are controlled by the vision system, and because the vision system works with each robot in a sequential order, this configuration results in excessive process times, resource utilization, and troubleshooting.

Examples of inspecting system configurations are also found at the research level. A hierarchical structure has been implemented at the University of Missouri-Rolla for their flexible Automated Assembly Cell. However, they have concluded that their configuration is not suitable for large scale manufacturing [8]. Albert Jones and Abdol Saleh propose a decentralized control architecture composed of multi-level and multi-layered components [3]. Since the multi-level and multi-layered design requires information to flow through many interfaces, the system response is slow.

Nils R. Sandell, et al.'s survey [9] indicates that there is a trend toward decentralized decision making, distributed computation, and hierarchical control for economic and possibly reliability reasons. However, this trend for large scale systems does not "mesh" with the centralized methodologies and procedures associated with control theory. The following are additional examples of various control schemes.

Texas Instruments' HARM missile plant in Lewisville, Texas [11], uses a combination of robotics and vision systems to control a picking up process. First, a robot picks up a component, and then a vision system gets the robot to verify precisely that it was picking up the component in the proper position. On the other hand [11], those "highly engineered vision systems locate, orient and identify parts, and download that information through the system, to the robot controller. The robot's gripper can then be correctly positioned at the time of part pick up."

The B&S manufacturing engineering staff and the systems engineering group of ASEA Robotics (New Berlin, WI) [11] have developed a multiprocess robot cell. This production cell combines robot vision, material handling and deburring operations in a sequential order. The IVR 2500 robot vision with an IRB L6/2 robot is used to identify and properly locate incoming parts from a conveyor. The 64-level gray-scale vision system processes each "scene". The part is placed on the conveyor and proceeds to downstream operations. The robot proceeds to designated areas after locating, identifying, and picking up the part from the conveyor.

The Honeywell Production Technology Laboratory [12] has developed an integrated robot/machine vision station to automate the first pass inspection of solder joints on a sequentially controlled Honeywell product line. A computer image processor directs a PUMA 560 robot to position the circuit board in front of a TV camera. Then, the image processor takes a "snapshot" of the board area, inspects each solder joint in the scene individually and accumulates flaw data. After that, the image processor downloads the flaw data to a microprocessor while the robot sets the board down. The microprocessor processes the data and then prints out the inspection results for the board as the image processor inspects the next board.

One agricultural equipment manufacturer [13] prefers to automate the measurement process so that an operator does not have to monitor the process. This manufacturer uses a DEC PDP 11/23 as a central translator between the various controls involved. Controlled by the central translator, the gantry robot loads parts for machining, then brings the parts to a coordinate measuring robot for measurement. Once a part has been measured, it goes right into inventory if it is good.

Otherwise, the gantry robot dumps the bad part in a tub. The programmable controller keeps track of which of the machines the parts come from.

General Motors [14] uses the vision system to track trends in specific data points or relationships between specific data points on a relative basis by correlating absolute car body dimensional information. This information is provided by the coordinate measuring machine. The vision systems and the coordinate measuring machine complement each other. General Motors also plans to use the vision system to load, automatically, data from the process control to a personal computer for the purpose of process improvement analysis. Therefore, the computer is controlled by the vision system while the vision system and the coordinate measuring machine are controlling each other.

Zheng and his colleagues [15], from Tsinghua University in China, use an acoustic emission (AE) sensor as a central monitor. In the machine center, the AE signal distinguishes various kinds of cutting-tool breakage for turning, milling, drilling, and boring operations on CNC lathes.

In a review of the available literature to date, two approaches, hierarchical and heterarchical, have been proposed to control activities within a CIM environment [3, 9]. Whatever a control structure is, it is fixed. However, at the CIM supported shop floor, activities may need to be controlled differently to reach different manufacturing goals.

In the design of the system, a question must be asked: which control structure can meet the needs of various manufacturing activities?

In this dissertation, a flexible control structure is proposed and designed for an integrated manufacturing environment. This design combines different control structures: sequential, central, and hierarchical. Thus, the control is flexible. The inspection system (RCV) at Old Dominion University's Automated Manufacturing Laboratory was used as the subject for the design of a factory simulation experiment. The RCV inspection system constitutes an integrated manufacturing environment; thus, the control structures can play their roles within the environment. The objectives of this research are discussed in Chapter 3, and a descriptive model is presented in Chapter 4. The factory simulation of the RCV system and the research design will be described in Chapter 5. Chapter 6 will present the results of the evaluation of control structures using the research design.

CHAPTER 3

THE RESEARCH PROBLEM

Statement of the Problem

Results of the literature review indicate a need for a control structure to meet the requirements of various manufacturing activities in an integrated environment. Such a control structure should be flexible, i.e., one that can be changed in accordance with the variety and types of tasking required.

For example, for a system which is used primarily to monitor manufacturing output, a central control structure is reasonable. An industrial computer can be considered as a central controller. By analyzing a defective part reported from the vision system, the computer may trace causes to certain processes and send the diagnostic results to the operator. If a manufacturing activity is assembling a product which needs a critical operational order, the acceptable control structure would probably be sequential. Sometimes, different types of production, involving different products or different batch sizes, may need selective controls depending on the criteria. These criteria could include time limitations and manufacturing specifications, as well as management strategies. If the number of the RCV system components is increased, a change in the control structure will be more complicated.

A flexible control structure is introduced in this study. The concept of the flexible structure is to allow different control structures to apply according to the manufacturing activities required. The concept of flexible control in relation to the RCV system example will become apparent in the simulation discussion.

If a hierarchical, sequential, or central control structure is needed at different times and under different circumstances, then the flexible structure can adapt by selecting one of these traditional control structures, i.e., the flexible control structure allows different control structures to be applied. Since the IRI D256 vision system, the PUMA 762 robot, and the IBM 7552 industrial computer all have their own microprocessors, they are internally controlled machines. This provides an opportunity for each component of the RCV system to control or to be controlled. The goals of a flexible control structure include:

1. To increase the flexibility of a manufacturing system,
2. To increase real time control abilities by permitting manufacturing activities to obtain the right control structures at the right time,

3. To make efficient use of the system resources (equipments, materials, etc.) by accommodating parallel processing of activities, and
4. To reduce overall manufacturing times through a reduction of communication and memory requirements, which occurs when the right control structure is applied to the right manufacturing activities.

Research Objectives

The objectives of this study are:

1. To develop a mathematical model for a new control structure, flexible control, for an integrated manufacturing environment,
2. To demonstrate the different control structures in an integrated manufacturing environment with a factory simulation, and
3. To evaluate the flexible control structure and the other control structures with respect to their fault tolerance characteristics.

One way to describe the flexible control structure is to explore the mathematical properties of the structure. Therefore, in Chapter 4, a mathematical model is developed using basic graph theory. When the flexible control structure is graphically described, it is possible to compare it with the other control structures. The efficiencies of different control structures can be evaluated by measuring the time that the RCV system takes to inspect each part under the different controls. The effectiveness of the control structures can also be evaluated with respect to their fault tolerance characteristics. Control structures for automated manufacturing are difficult to evaluate without actually setting up a pilot production system. The methodology, here, is to use a factory simulation to establish an integrated real-time inspection system. Factory simulations present special problems when used for evaluation purposes. Chapters 5 and 6 present an approach for using a factory simulation to evaluate and compare alternative control structures.

CHAPTER 4

A MATHEMATICAL MODEL FOR FLEXIBLE CONTROL

By Aho's definition [35], a graph $G = (V, E)$ consists of a finite, non-empty set of vertices V and a set of edges E. Each $e \in E$ is an ordered pair (v_i, v_j) where v_i and v_j are in V. A path from v_1 to v_n in a graph is a sequence of edges of the form $(v_1, v_2), (v_2, v_3) \ldots (v_{n-1}, v_n)$.

Control structures can be described using graphs and subgraphs. Each vertex of a graph represents a control point (or working unit) of the control structure. Each edge of a graph represents a connection between two control points in the control structure. For example, the graph in Figure 1 describes the flexible control structure for the RCV system. The vertices represent the control points: ROBARM (robot arm), CONVIN, CONVOUT (for the same conveyor), FEEDER (feeder), GATE (gate), IC (industrial computer), ARTIC (ARTIC card), VISION (vision system). The edge between the vertices of VISION and GATE represents the communication between the two control points of vision system and gate. In Figure 1, whenever an edge exists between two vertices, data and signals can be transmitted between those points. For instance, the fact that

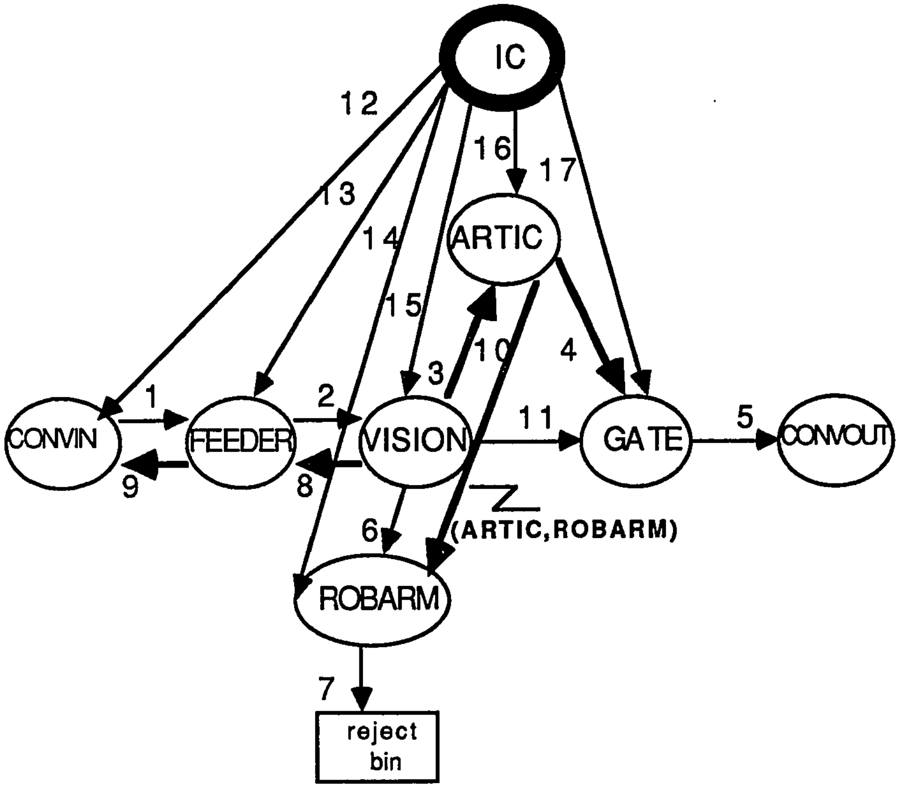


Figure 1. Flexible Control Graph.

ARTIC data can be transmitted to ROBARM is represented by the edge of (ARTIC, ROBARM).

When a control structure is represented by a graph, subcontrol structures appear as subgraphs. A sub-graph which corresponds to a subcontrol structure is called a unique sub-graph. For the RCV inspection system, the graph of the flexible control structure contains unique sub-graphs in the form of a line, a tree, and a star. In the RCV system, the unique sub-control structures are sequential control with no computer, sequential control with computer, hierarchical control, and central control.

Definition

A control structure is flexible if and only if:

1. The control structure includes more than two unique sub-control structures, and,
2. The control structure has an ability to switch from one sub-control to another sub-control as required.

Properties

If the following symbols are defined, then the RCV flexible control structure can be described in terms of a graph as follows:

$F=(V_F, E_F)$---The graph for the flexible control structure, where V_F={ROBARM, CONVIN, CONVOUT, FEEDER, GATE, IC, ARTIC, VISION} and E_F = {1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17 }, where 1=(CONVIN, FEEDER), 2=(FEEDER,VISION), 3=(VISION,ARTIC), 4=(ARTIC,GATE), 5=(GATE,CONVOUT), 6=(VISION, ROBARM), 7=(ROBARM, REJECTBIN), 8=(IC,ROBARM), 9=(FEEDER, CONVIN), 10=(ARTIC,ROBARM), 11=(VISION,GATE), 12=(IC,CONVIN), 13=(IC,FEEDER), 14=(IC,ROBARM), 15=(IC,VISION), 16=(IC,ARTIC), and 17=(IC,GATE).

Further, the subcontrol structures will be represented by subgraphs as detailed below.

$G_0=(V_{G0}, E_{G0})$---Unique sub-graph for sequential control with no computer component within the flexible control structure, where V_{G0}={CONVIN, CONVOUT, FEEDER, VISION, GATE, ROBARM, REJECTBIN} and E_{G0}={1,2,11,5,6,7}. The primary command routes start at CONVIN and proceed through 1, 2, and either 11, 5 or 6, 7.

$G_1=(V_{G1},E_{G1})$---Unique sub-graph for sequential control with computer within the flexible control structure, where V_{G1}={CONVIN, CONVOUT, FEEDER, VISION, ARTIC, GATE, ROBARM, REJECTBIN} and $E_{G1}=\{1,2,3,4,5,6,7,10\}$. The primary command routes start at CONVIN and proceed through 1, 2, 3, and either 4, 5 or 10, 7.

$G_2=(V_{G2},E_{G2})$---Unique sub-graph for hierarchical control within the flexible control structure; the set V_{G2} is the same as V_{G1} and $E_{G2}=\{3,4,5,7,8,9,10\}$. The primary command routes start at VISION and proceed through 8, 9, and either 3, 4, 5 or 3, 10, 7.

$G_3=(V_{G3},E_{G3})$---Unique sub-graph for central control within the flexible control structure; V_{G3} is the same as the set V_{G1} with the addition of the IC vertex and $E_{G3}=\{5,7,12,13,14,15,16,17\}$. The primary command routes start at the central controller IC and proceed through 12, 13, 15, and either 17, 5 or 14, 7.

The systems ability to select a particular subcontrol scheme to be active and the ability to change this selection is represented by the function σ.

$\sigma(t)$---Function for switching to one unique subset of the sub-graphs to another one. For example, if A is the set of subcontrol structures required by the manufacturing process at time t the $\sigma(t)$ will be equal to A. In this research, the switching function is implemented in the following way at times t_i (i=0, 1, 2, or 3) and

$$\sigma(t_i) = \{ G_i \bmod 3 \}.$$

This function is invoked whenever the RCV system has finished the inspections for one batch of parts and prior to starting the inspections for a new batch of parts.

Therefore, the flexible Control Structure for the RCV system may be represented by the Six tuple = $<F, G_0, G_1, G_2, G_3, \sigma>$. This indicates which subcontrol structures are available within the RCV system and the particular switching function.

Property 1: The flexible control structure in the RCV system (Figure 1) has distinct but overlapping subcontrol structures.

For example, the structures of sequential control with computer G_1 and hierarchical control G_2 are distinct, E_{G1} is not the same as E_{G2}, but overlapping, the edges of 3, 4, 5, 7, and 10 belong to both E_{G1} and E_{G2}.

Property 2: The graph F can be considered as a unique subgraph of a larger graph.

For example, graph F becomes a unique sub-graph of a larger graph, G, if graph G describes a manufacturing control structure that includes part processing subcontrol structure, part assembly subcontrol structure, and part inspection subcontrol structure and the inspection control structure is for the RCV inspection system. Property 2 indicates that a flexible control structure can be designed as a subcontrol structure of a larger control structure.

Property 3: Evaluating the switching function $\sigma(t_i)$ corresponds to switching from subcontrol structure G_i to subcontrol structure $\sigma(t_i)$.

Property 3 explains that with the switch function, the unique subcontrol structures are selectable within a flexible control structure.

It is understood that if one has different configurations for the control scheme the flexible control structure is different. Properties 1 and 2, which correspond to the first part of the flexible control structure definition, provide a static description, and property 3, which corresponds to the second part of the definition, provides a dynamic description for any flexible control scheme.

CHAPTER 5

RESEARCH METHODOLOGY

The flexible control structure proposed here is designed for a real-time manufacturing operation, e.g., inspection, that imposes a strict requirement on the system process time. The RCV system supports the execution of real-time inspection and ensures that the requirements of process time (or working time) are met. This is referred to as a real-time system. However, the control structures for the RCV inspection system are difficult to evaluate without actually setting up a pilot production inspection system. Thus, a factory simulation is used as a tool which can help assess the differences between control structures by subjecting each to a variety of conditions. The variety of conditions are captured in the variety of fault tolerance treatments used in the research design.

Design of the Simulation

The factory simulation design is discussed in detail in the following sections.

Simulation Goals

The purpose of the study is to evaluate the flexible control structure and other alternative control structures with respect to their fault tolerance characteristics. Since any control structure can be represented as a graph, according to the definition of the flexible control structure and property 1 mentioned in Chapter 4, the flexible control structure should be considered as one of the alternative control structures. In order to evaluate the flexible control structure with respect to the others, two goals of the factory simulation need to be achieved:

1. To apply different control structures to the integrated RCV inspection system, and
2. To test the fault tolerance of the RCV system with the different control structures.

When the two goals are met, the first will show the validity of each control scheme for the RCV inspection system in an integrated manufacturing environment. The second will provide a comparison of the control structures with respect to their fault tolerance characteristics.

Input Data and Initial Conditions

The input data include the part sizes that are randomly generated in an uniform distribution, and batch sizes.

In this study, the RCV system inspects one dimension of a manufactured object. For purposes of the simulation the specification for this dimension is 10 cm, with a tolerance of ±0.0568. Hence, if the randomly generated part size falls within the range of

$$10-0.0568 < \text{part size} < 10+0.0568$$

then the inspection result is PASS; otherwise it is FAIL.

Inspection System

The components of the RCV inspection system include a conveyor, feeder, sensor, camera, vision machine, gate, robot, PC and industrial computer. The feeder adds parts to the conveyor at some specified rate. The conveyor and the robot do the material handling work. After inspection, a good part continues along the conveyor through the gate, and a bad part is picked up by the robot. At the vision station, a sensor detects an arriving part; a camera takes a snapshot; then, the vision machine processes the part image information. The part sizes are generated from a **Random Size Generator**. The primary function of the computer is to analyze and report the quality results; but, sometimes, the computer can also perform a control function. Personal computers (PCs) are used for the central, sequential, and hierarchical control structures. An industrial computer (IC) is applied to the flexible structure.

The inputs for simulation include batch size and part specification; the outputs of the simulation contain the inspection results in the form of a "PASS" or "FAIL" message, the number of total inspected parts, the number of good parts, the number of bad parts, percentages of passed and failed parts, and types of manufacturing operations. Different control procedures correspond to the different control structures.

RCV System with Different Control Structures

To reach the goal of applying different control structures to the RCV system, the following control procedures are needed.

Central Control

In the central control structure, all the components, i.e., robot, conveyor, vision machine, etc., are controlled by a personal computer (Figure 2). The conveyor is ordered to deliver the parts provided from the feeder to the vision machine and carry the "PASS" inspection parts away. Through the computer, the vision machine sends the messages of "PASS" to the gate and "FAIL" to the robot after its inspection. The gate opens to allow the good parts through. The robot picks up the "FAIL" inspection parts from the vision workstation and unloads them to the reject bin. For the details of the algorithm, refer to the LLP:CENTC program list in Appendix B.

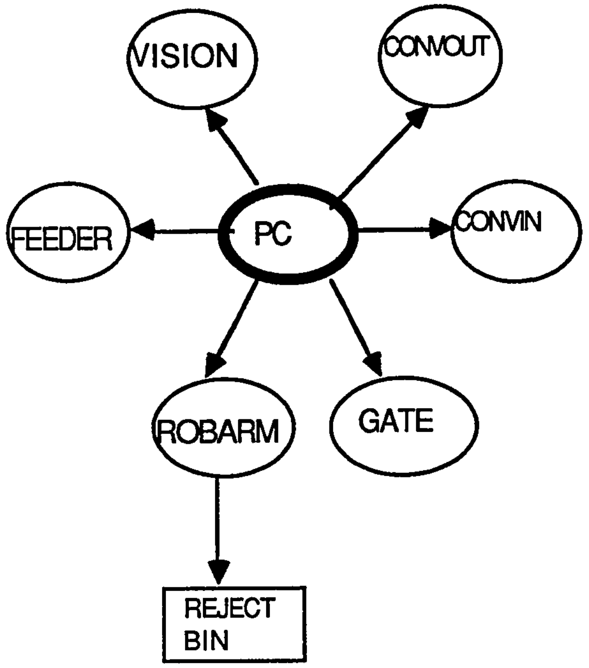


Figure 2. Central Control Structure.

For simulation purposes, the following information is provided as input: batch size and inspection specification. The outputs obtained are: inspection result, part inspection time, and batch inspection time. The operations performed include: inspecting and material handling.

Sequential Control with Computer

The components in the sequential control system interact with each other in a successive order (Figure 3). The conveyor delivers the parts fed by the feeder to the vision machine. The vision machine, after its inspection, sends the message of "PASS" or "FAIL" to the computer. The computer interprets the "FAIL" message and sends an instruction to the robot to pick up the bad part from the vision workstation and unload it to the reject bin. However, the good parts remain on the conveyor and go through the gate. Refer to the LLP:SEQC program in Appendix B.

Similar to the central control scheme, the inputs are batch size and inspection specification. The outputs are inspection result, part inspection time, and batch inspection time. And, the operations are inspecting and material handling.

Sequential Control without Computer

Another sequential control structure, but without a computer (Figure 4), has similar features to the sequential

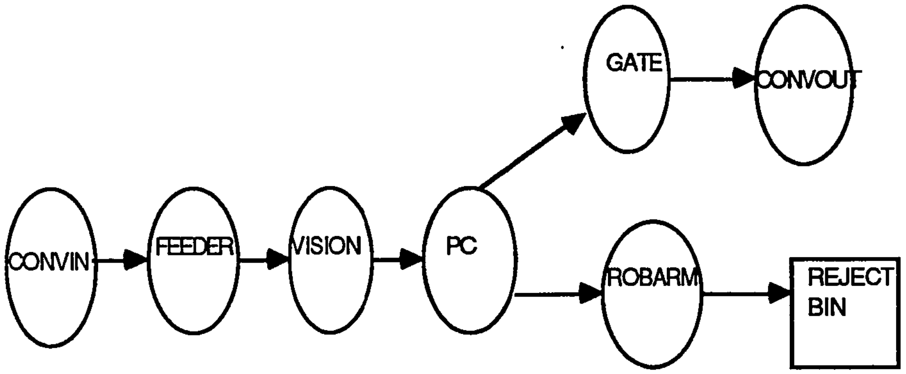


Figure 3. Sequential Control with Computer.

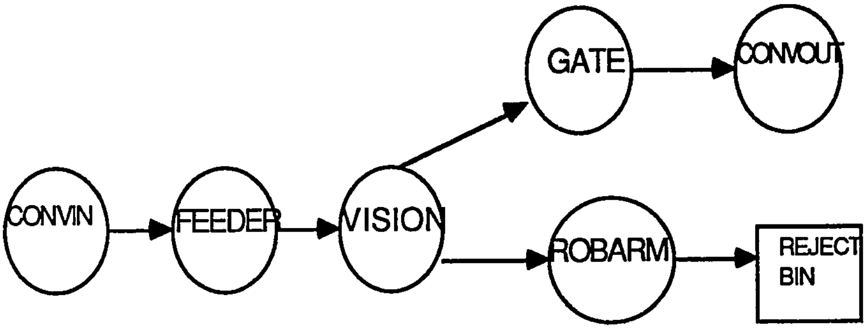


Figure 4. Sequential Control with no Computer.

control structure described above. Program LLP:SEQVR is listed in Appendix B. The input and output data and the operations are the same as those for the sequential control with computer.

Hierarchical Control

The components of the hierarchical control scheme are organized in a tree structure (Figure 5). The vision system sensor initiates a command to the feeder to drop a part onto the conveyor. After finishing the inspection, the vision machine sends the "PASS" or "FAIL" messages to the computer. The computer, then, asks the gate to open for the good parts to pass through and instructs the robot to pick up the failed parts from the vision workstation and unload them to the reject bin. The program LLP:HYRC is in Appendix B. Again, the inputs are batch size and inspection specification; the outputs are inspection result, part inspection time, and batch inspection time; and the operations include inspecting and material handling.

Flexible Control

The flexible control structure is a combination of the central, sequential (two types of sequential control mentioned above), and hierarchical control structures (Figure 6). It is important to recognize that a PC is used in the control structures mentioned in the above cases. In the flexible

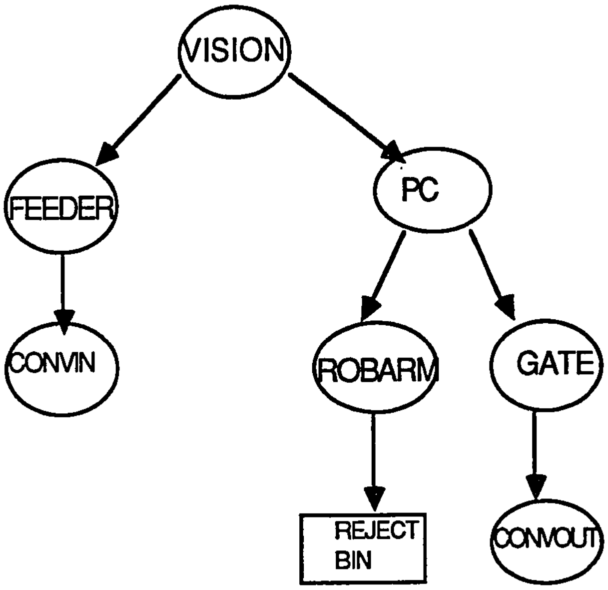


Figure 5. Hierarchical Control Structure.

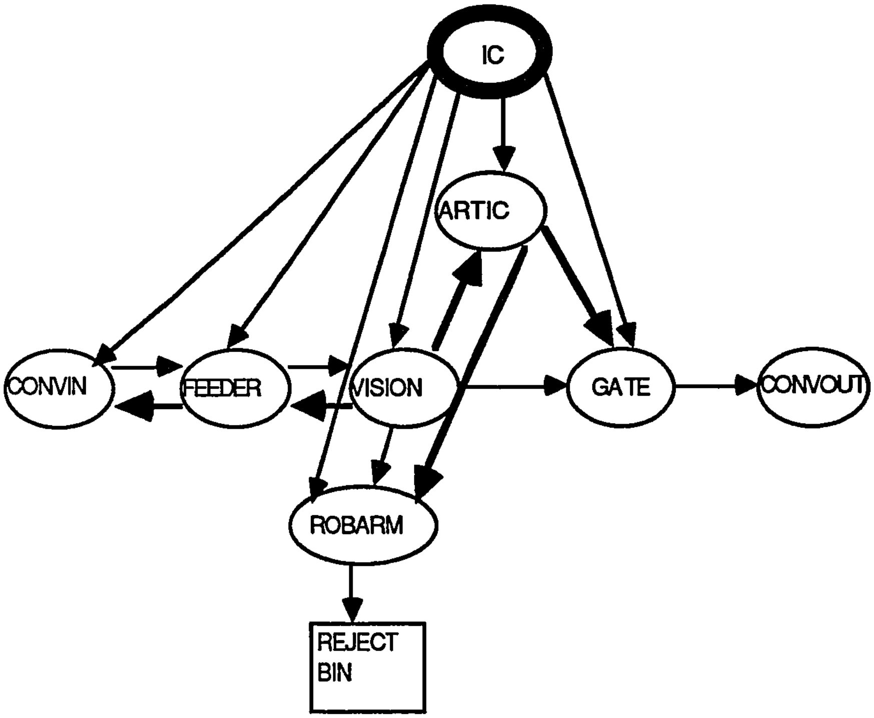


Figure 6. Flexible Control Structure.

control structure, an industrial computer is employed. The industrial computer allows IBM's Real-time Interface Co-processor (ARTIC) to be installed. With this ARTIC card, the principle of flexible control can be applied.

Within the flexible control structure, the industrial computer acts as a central controller when central control is required, and the ARTIC card plays the PC role when the sequential and hierarchical controls are needed. When sequential control without the industrial computer and ARTIC card is the appropriate control structure, the computer and the ARTIC card become available to perform other manufacturing operations. When this occurs, the flexible control structure permits a form of multi-tasking environment.

For flexible control, the inputs include the control structure requirement in addition to batch size and inspection specification. The outputs include inspection result, part inspection time, batch inspection time, manufacturing activities, and information on the current control structure. The last two outputs do not appear in any other control scheme. The operations in the flexible control structure are not only inspecting and material handling, but also data processing and assembling (see Appendix A).

Fault Tolerance

According to Anderson and Lee [46] and Merenbloom [47], fault tolerance implies the capability of a system to remain functional even when faults develop. To meet the second goal of testing the fault tolerance of the RCV system with the different control structures, a fault tolerance design is needed.

First, faults are defined as either component faults or communication faults. They are recognized as faults when the component function times or the communication times exceed certain time constraints for individual part inspection. For example, a component fault, such as a sensor fault, occurs when the sensor function time exceeds a given time constraint (limitation). In a real situation, the time delay may be caused by the wrong position of the sensor. A similar concept can be applied to the communication faults. For instance, the communication between the vision machine and the robot arm takes a certain amount of time. If the communication time is greater than a specified requirement, a communication fault is detected. One practical case could occur when the robot receives an improper instruction from the vision machine, which delays the feedback time.

Second, the fault tolerance testing should be able to generate the fault (called **Fault Generator**), detect the fault, access the fault, and correct the fault. Four fault cases for each control structure are used as treatments in this study:

1. The RCV system has no fault (NF),
2. The RCV system has one component fault (CPF), such as a conveyor fault,

3. The RCV system has one communication fault (CMF), such as a "starting system" fault, and
4. The RCV system has one component fault and one communication fault (CPMF), such as a conveyor fault and a "starting system" fault.

One could also assume other fault cases, such as more than one component fault, more than one communication fault, or different combinations of more than one component and one communication fault.

Fault Generator

From the fault classification, two fault generators are developed for the fault tolerance simulation. One is called Component Fault Generator and another one is called Communication Fault Generator. The principle of generating faults is to measure the actual simulated component function time or communication time, and then add a sensitive Δt (for example, $\Delta t=1$ second) to the component time or the communication time. If the sum of the two times is greater than the relative time constraint, then a component fault or communication fault occurs. Since the RCV system includes 10 components and 28 types of communications, the possible component faults are 10 and the possible communication faults are 28. For the list of faults, please refer to Appendix B.

Fault Monitor

The RCV system efficient working time is defined as the sum of system component function times and communication times. If the system efficient working time is greater than some time limit, an alarm is sounded and fault detection is started.

Fault Tolerance Program

Fault detection. Timing checks are applied for fault detection. If the specification of a component includes timing constraints on the provision of service, then a timing check can determine whether the operation of the component meets those constraints or not. If the constraints are not met, then the timing check can raise a "time out exception" to indicate a component fault. Timing checks are usually sufficient to reveal the presence of faults in a system. Similarly, if the specification of a connection between two components includes timing constraints on the provision of service, then a timing check can determine whether the communication between the components meet those constraints or not. If the constraints are not met then the timing check can raise a "time out exception" to indicate a communication fault.

Fault assessment. The fault access program continues to search for the exact location of the fault, such as a fault in the robarm or in the communication between feeder and sensor.

After this search is completed successfully, the detected fault is recorded and reported in a printout and video display. The number of faults is also counted.

Fault recovery. A **Reset** technique is employed as fault recovery in the simulation. When a **Reset** of the system is invoked, the initial states are recovered.

Fault treatment. From step 2, the fault can be located. Then, the system repair should be in function. It is assumed here that system repair does not involve making repairs internally to a suspect component. The technique for system repair will be based on the dynamic and spontaneous reconfiguring of the RCV system by restarting the current control procedure. Having dealt with the faults in the system by locating the faults responsible and effecting appropriate repair, the RCV system returns to its normal operation again.

Output Data Analysis and Hypothesis Testing

Having inspected one part, the RCV system generates a report of system working times and inspection results. Analysis of the inspection results can provide insights on quality control. However, this analysis is not a major aspect of our research. In this research, the analysis of system working times is of central concern and is discussed in detail. With this analysis, the alternative control structures can be evaluated with respect of their fault tolerance characteristics. The fault tolerance of the different control structures is measured by taking the mean of the RCV system working times for different fault cases. For example, if the RCV system with one component fault is operating under the central control structure, then the mean of the system working times for inspecting ten parts characterizes the fault tolerance of the RCV system with the central control scheme.

By analyzing the system working times using the method of one-way Analysis of Variance (ANOVA), the following hypothesis can be tested:

Hypothesis: There are no significant differences in the fault tolerance characteristics of the RCV inspection system under different control schemes.

Next, if the hypothesis is not true, i.e., there are significant differences between the control structures, then the degree of fault tolerance of the different control structures can be evaluated by Duncan's multiple-range test.

In summary, the ANOVA method can test whether there are **any** differences between the structures, and Duncan's multiple-range test can then rank the control structures with respect to their fault tolerance characteristics. Within the domain of mean system working times (see Table 1), the hypothesis can be tested with an F value, where

F = (Between Structure Variance)/(Within Structure Variance)

In terms of Table 1, **between** is between different columns and **within** is within one column. If the variation **between** the structures is large compared to the variation **within** the structures, the obtained F ratio will be greater than 1 and the hypothesis should be rejected. That is, the alternative control structures would not be equivalent. If the hypothesis is true, the F statistic will be equal to 1.00 on the average.

Table 1. Mean System Times.

	CONTROL STRUCTURES							
FAULT CASES								
			MEAN SYSTEM TIME					

The variable of **control structure** is independent and the variable of **mean system time** is dependent. The values of **control structure** can be sequential control without computer, sequential control with computer, hierarchical control, central control, and the four sub-control structures of the flexible control scheme. For each control structure, there are four mean system working times, one for each of the four fault cases. The fault cases are: no fault (NF), one component fault (CPF), one communication fault (CMF), and both a component fault and a communication fault (CPMF).

With one control structure, the RCV system has four system working times corresponding to the four fault cases. Regression analysis is used to discover the closeness of the system working times between "no fault" and "with some fault" (such as a component fault, a communication fault, or a component fault and a communication fault) for each control structure. The measure of closeness is the **standard error** (stder). The smaller the **standard error,** the closer the system working times with no fault are to the system working times with a fault.

Simulation Features

In this research, the measures of system performance are defined relative to the interval of simulation time during which a specified activity occurs. For example:

1. The system working (processing) time interval is measured by the total of the component function times and the communication times for an individual part inspection. The component function time or the communication time is defined as the time interval between the start of the function or the communication and the end of the function or communication.
2. Time delay intervals are inserted to represent real-world manufacturing conditions. For example,

when the sensor detects an arriving part, if two seconds are needed for the detection, the time delay interval, 2 seconds, is added into the sensor function time. A similar principle is applied to the communication case. For instance, if the communication between the computer and the robot requires a one second delay, the time delay interval, 1 second, is added into the communication time.

3. The concept of component or communication faults is based on the determination of whether the component function time intervals or communication time intervals exceed certain time constraints.

It is important to point out that this type of factory simulation is not intended for mathematical modeling purposes. The output system times are influenced by computer runtimes, not strictly by model times, and thus serve only as estimates of the true characteristics of the system being modeled. The factory simulation used had to be adapted to analytically evaluate the performance of the real-time RCV system with different control structures with respect to their fault tolerance characteristics. Altogether, 37 I/O points, 94 working variables/records, 77 programs (or chains), 8 reports, and 3 screen display designs have been developed for the simulation. The descriptive graph of each control structure was used to help set up the control procedures.

Simulation Tool: PlantWorks 2.0

IBM PlantWorks, released in 1992, was selected as the simulation tool. This is a software package which provides a functional language for simulation programming. This application package, supported by the OS/2 operating system, runs on the industrial computer (IBM model 7561). The features of PlantWorks are (as listed in the manual):

1. Store and manipulate data values,
2. Communicate with input and output devices,
3. Control manufacturing operations,
4. Dynamically represent information on displays,
5. Produce and manage alarms,
6. Generate reports,
7. Maintain a history of information, and
8. Provide C++ language environment.

The algorithms for the various controls--central, sequential, hierarchical, and flexible--are developed using PlantWorks features.

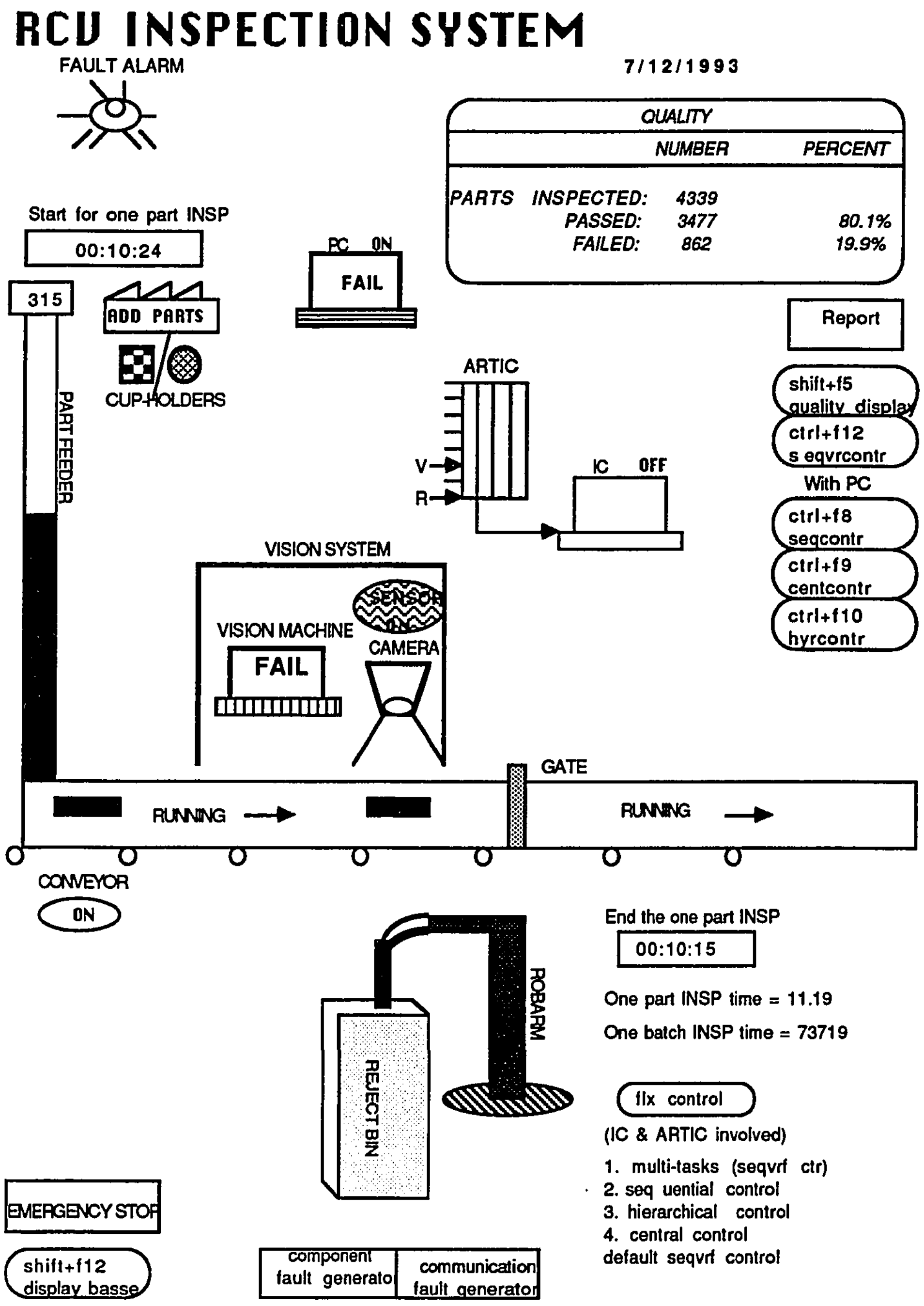
Simulation Tasks

The simulation model is developed using IBM's PlantWorks functional language. Part size data are generated by a program called Random Generator (Appendix B). This data is then used as the basis for the inspection operation for the 10 cm object manufactured. If the part exceeds a given tolerance (such as 0.0568), then the inspection result is "FAIL", otherwise it is "PASS". The inspection program is in Appendix B.

With the RCV simulation system, a user is able to perform the following tasks (Figure 7):

1. Input the type of product to be inspected.
2. Input the batch size (1000 is defined as the maximum size).
3. Select the control structures. There are five options:

   a. Sequential control with PC,
   b. Sequential control without PC,
   c. Central control,
   d. Hierarchical control, and
   e. Flexible control.



RCU INSPECTION SYSTEM

7/12/1993

QUALITY		NUMBER	PERCENT
PARTS	INSPECTED:	4339	
	PASSED:	3477	80.1%
	FAILED:	862	19.9%

Start for one part INSP: 00:10:24

End the one part INSP: 00:10:15

One part INSP time = 11.19

One batch INSP time = 73719

(IC & ARTIC involved)

1. multi-tasks (seqvrf ctr)
2. seq uential control
3. hierarchical control
4. central control

default seqvrf control

Figure 7. RCV Inspection System Simulation.

If the flexible control structure is selected, in accordance with property 3 discussed in Chapter 4, four cases can be considered:

a. Sequential control with R and V (the robot and the vision system).
b. Switching from sequential control with R and V to the sequential control with R, V, and ARTIC (ARTIC card).
c. Switching to hierarchical control with R, V, and ARTIC.
d. Switching to central control with IC, R, and V.

All the switches can be either automatic or manual. Here, the control structure is changed automatically when a batch of parts have been completely inspected. When the control involves the ARTIC card, the ARTIC card symbol and its connections to the R and V will change color or blink on the simulation screen, in order to show that the ARTIC card is processing and communicating.

4. Initiate or terminate the RCV inspection system with the different control structures. The aim of this task is to start or terminate the RCV system when one of the control structures is selected. Different equipments can be key factors to the different control schemes. For example, the conveyor is the initiating equipment for the sequential control scheme. The PC (or the industrial computer) is the critical equipment in the central control scheme. And, the sensor is the root equipment in the hierarchical control scheme.
5. Inspect the products. For simulation runs under all control structures the RCV system is used for the part inspection. The feeder drops the part onto the conveyor. The conveyor delivers the part to the vision station. The sensor and the camera of the vision system flash on and off to obtain the part size image data. Then, the vision machine processes the data and prepares the inspection result. This result may or may not be sent to the PC or the industrial computer, depending on the control structure used. If the "FAIL" message is sent from the vision machine, PC, or the industrial computer to the robot, the robot will pick up the bad part from the conveyor and unload it into the reject bin. By the same principle, the gate will

open and allow the good part to continue to move along the conveyor when the inspection result is "PASS".

6. Obtain the quality control report. The system allows the user to display a statistical report on the following items:

    a. Total parts inspected,
    b. Total parts passed,
    c. Percentage of passed parts,
    d. Total parts failed, and
    e. Percentage of failed parts.

7. Monitor the part inspection time and the batch size inspection time. A special function and program have been developed to keep track of the time spent on single part inspection and on batch inspection in order to compare the effectiveness of different control structures. The user can then make a decision on a satisfactory control structure by comparing the time tests.
8. Secure the simulation. A blue button, called "display base", is designed for securing the simulation procedure. If this button is selected, the animation simulation of the RCV system is hidden by the PlantWorks base screen.

9. Select type of fault. By simulating fault tolerance, the time behavior of the RCV system under different control structures can be evaluated. Users can test the RCV system with the different control structures by introducing component faults (CPF) or/and communication faults (CMF). There are two buttons called "component fault generator" and "communication fault generator" designed for this purpose.

CHAPTER 6

EVALUATION OF THE CONTROL STRUCTURES

In this chapter, an evaluation of the central, sequential (with computer and without computer), hierarchical, and flexible control structures is presented. The evaluation focuses primarily on the fault tolerance characteristics of the RCV system under the different control structures. Other features of the control structures are also examined.

Evaluation of Fault Tolerance

From the data on RCV system working times, the alternative control structures can be evaluated with respect to their fault tolerance characteristics.

System Data

For each control structure, the fault tolerance simulation generates two types of data about system working time (see Appendix D). One is the system working time with no faults (NF) introduced and the other one is with faults introduced. To generate data on a range of faults, three fault cases were used: 1) one component fault (CPF, e.g., the conveyor), 2) one communication fault (CMF, e.g., the communication for the system start), and 3) one component and one communication fault (CPMF, e.g., the conveyor and the communication for the system start).

Sampling Technique

For each control structure, four kinds of simulation run on the batch size of 200 parts were completed (see Appendix D). Each run is related to a particular fault case. Since there are four fault cases and eight control structures (four traditional control structures and four flexible sub-control structures), altogether there were 32 runs.

Part numbers 96 to 105 (10 out of 200 parts), were selected as the sample by the Judgement Sampling Technique [49, 50]. This sample selection was necessary because the PlantWorks software would not allow use of the same parts and part order in a given batch. The ten units serve as representatives of the part population since the RCV system working times for the ten parts are assumed to be stabilized in the middle of the run. These emulation software anomalies can and did produce inconsistencies in the system working time data during simulation.

Data Analysis Methods

In order to evaluate the alternative control structures with respect to their fault tolerance characteristics, two analyses were involved: analysis of the closeness of the system working times with a fault to the system working times with no fault and analysis of the effectiveness of the RCV system fault tolerance corresponding to the different control structures.

Since the mean of the system working times for different fault cases is important to the analyses, the table of mean system working times for the ten part inspections is provided in Table 2.

Table 2. Mean System Times (for ten part inspections).

Fault Cases	Control Structures							
	seqvr	seqc	centc	hyrc	seqvrf	seqfc	hyrfc	centfc
NF	23.2	16.6	70.1	31.0	37.6	32.4	83.5	124.4
CPF	15.6	11.9	84.5	29.5	48.8	39.4	87.2	471.1
CMF	20.4	17.7	67.3	33.2	45.2	60.4	80.1	147.4
CPMF	15.6	11.6	67.9	35.92	36.4	35.2	72.7	108.5

Ideally, the system working time should be constant. Thus, linear regression is adopted to calculate a mean and standard error. The regression line should be a horizontal line. The closeness of the RCV system working times is measured by the Standard Error of system time estimation (see Appendix D) with the QuattroPro software tool [36]. The standard errors of system working times are summarized in Table 3. The apparent inconsistencies in Table 2 for the mean system time for fault and no fault cases are attributed to the batch specification difficulties discussed earlier. These inconsistencies do not detract from the methodology discussed herein.

Table 3. Standard Error of System Time.

	Control Structures							
NF vs	seqvr	seqc	centc	hyrc	seqvrf	seqfc	hyrfc	centfc
CPF	6.8	4.3	16.6	4.3	15.2	9.8	11.5	180
CMF	6.1	10.2	9.33	6.3	9.03	9.9	31.9	10.7
CPMF	1.2	0.85	7.2	8.7	5.68	3.3	63.6	32.0

seqvr----sequential control with no computer,
seqc-----sequential control with computer,
centc----central control,
hyrc-----hierarchical control.
seqvrf---sequential control with no computer,
seqfc----sequential control with computer,
centfc---central control,
hyrfc----hierarchical control,
NF-------no fault,
CPF------one component fault,
CMF------one communication fault,
CPMF-----one component and one communication fault.

In this table, each column corresponds to different control structures, and each row relates to different fault cases.

Each numerical value in the table represents a standard error. If one takes the system working times with no fault as a normal system working time and the system working time with a fault (one component fault, one communication fault, or one component and one communication fault) as an abnormal system working time, the estimated standard error of the RCV system working times represents the deviation of the abnormal system working times from the normal system working times. For instance, "6.8" in the first cell is a standard error between the normal system working time (system time with no fault) and the abnormal system working time (system time with one component fault) when the RCV system is under the sequential control without computer. In the fourth column, the RCV system is operating with the hierarchical control structure and the standard error of the system working times between the cases of no fault and one component fault is 4.3, the error between the cases of no fault and one communication fault is 6.3, and the error between the cases of no fault and one component fault and one communication fault is 8.7. Differences between the error values can be attributed to the nature of the control structures and/or the characteristics of the faults.

All the standard errors in the given table are the deviations between the normal system times and the abnormal system times. The smaller the standard error values, the closer the system working times with a fault are to those with no fault. According to Francis [37], graphics, "that can show whether the structure fits well, can suggest how it might be changed to fit better and can make us qualitatively aware of the behavior of the data as a whole" (p.27) The figures (Figure 8, 9, 10, 11, 12, 13, 14, 15) depicting system working time and part number provide such a graphic. As an example, in Figure 11, RCV with Hierarchical Control, the piecewise line with "■" represents the system working times taken with no faults. Similar to the other cases, the piecewise line with "+" is for system working times taken with one component fault. The piecewise line with "*" is for system times with one communication fault. And, the piecewise line with "□"is for system working times taken with one component and one communication fault. In each graph, some point(s) deviate substantially from the average or norm. The explanation for this deviation is that when a part fails the inspection and is picked up by the robot arm, the system working times will be longer than when the part passes the inspection (refer to Chapter 5 on the control procedures).

The fault tolerance effectiveness of the RCV system with the different control structures was studied through two steps: 1) testing the hypothesis to see if there was any significant difference between the control structures with respect to their fault tolerance characteristics, and 2)

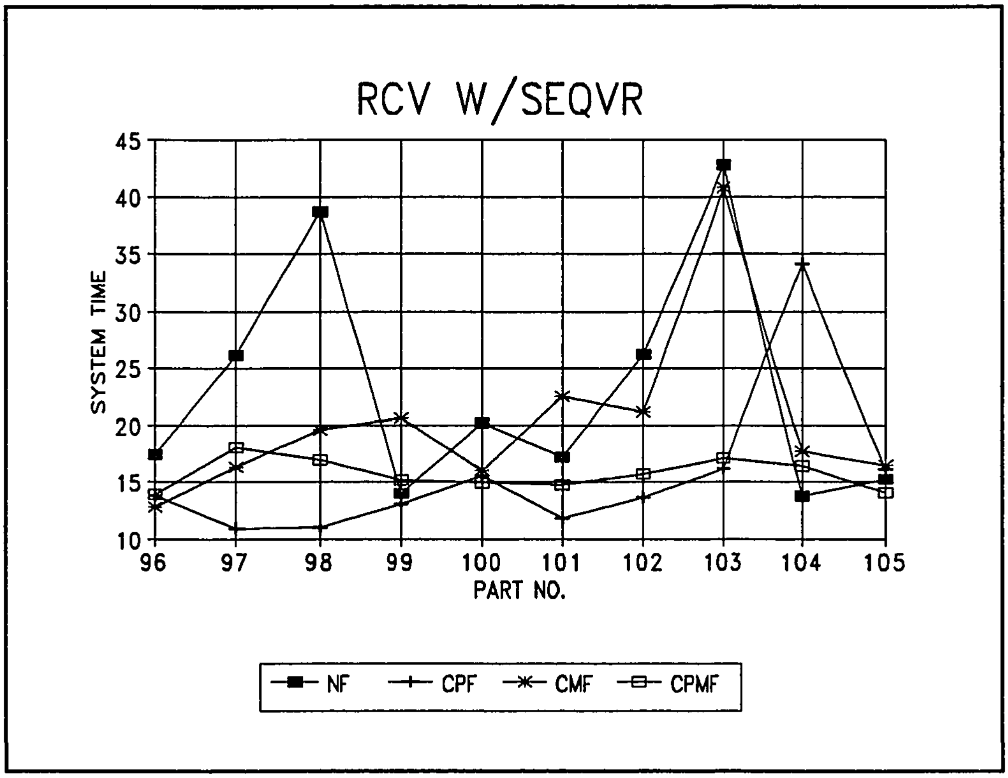


Figure 8. RCV with Sequential Control (no computer).

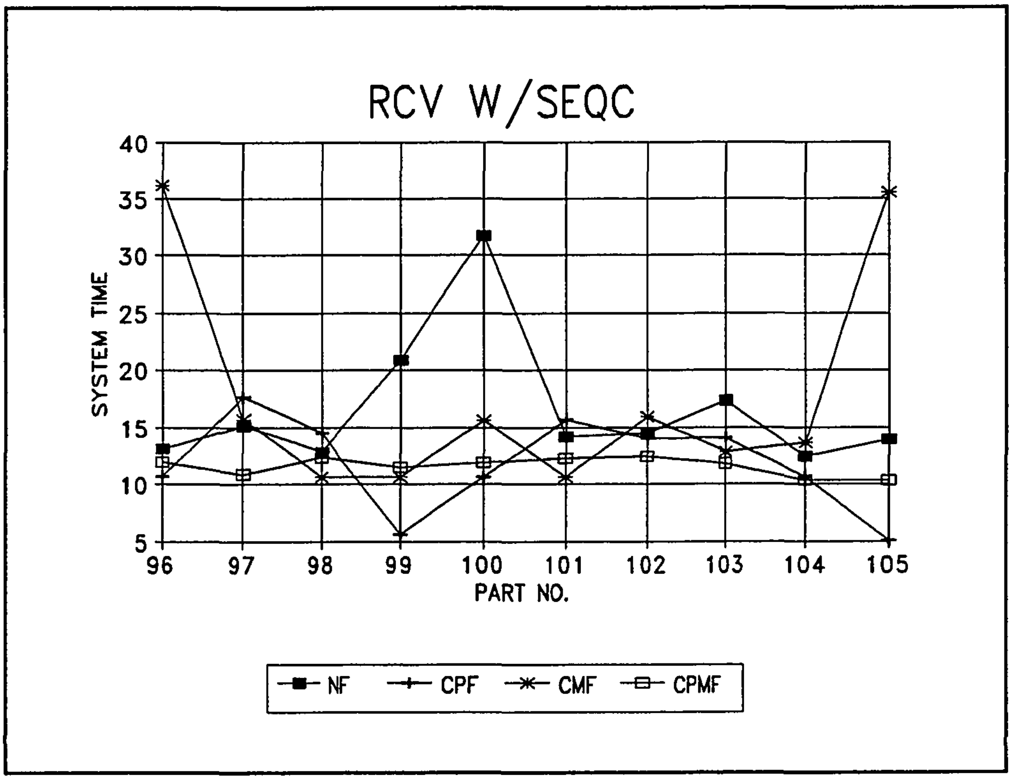


Figure 9. RCV with Sequential Control (with computer).

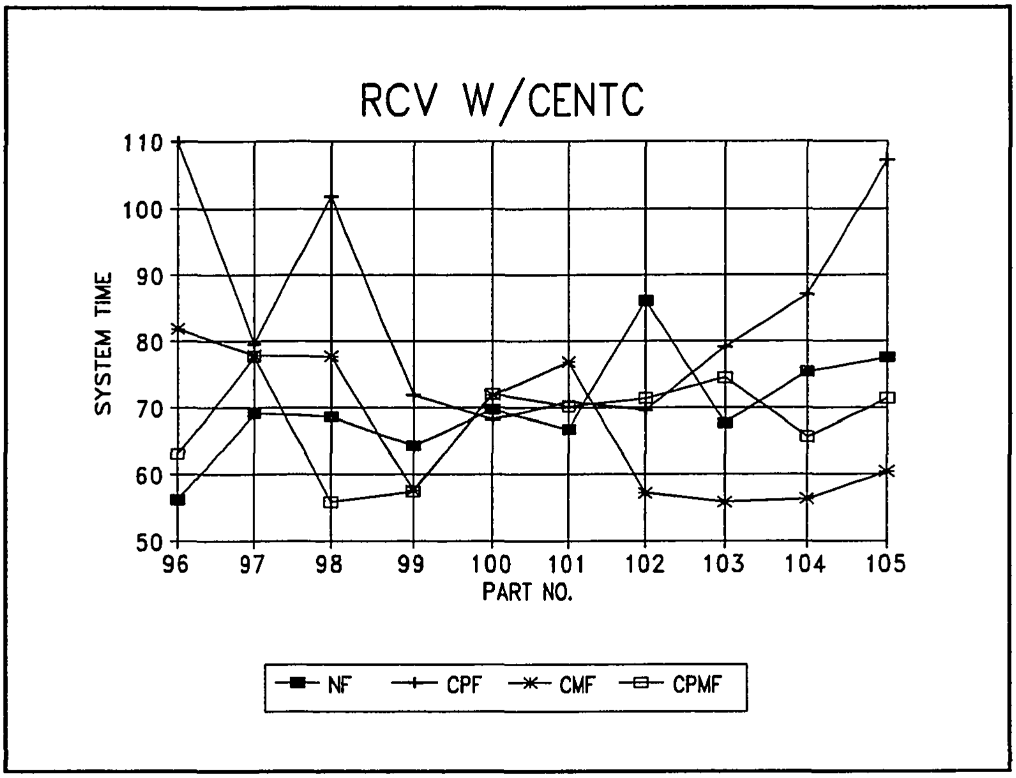


Figure 10. **RCV with Central Control.**

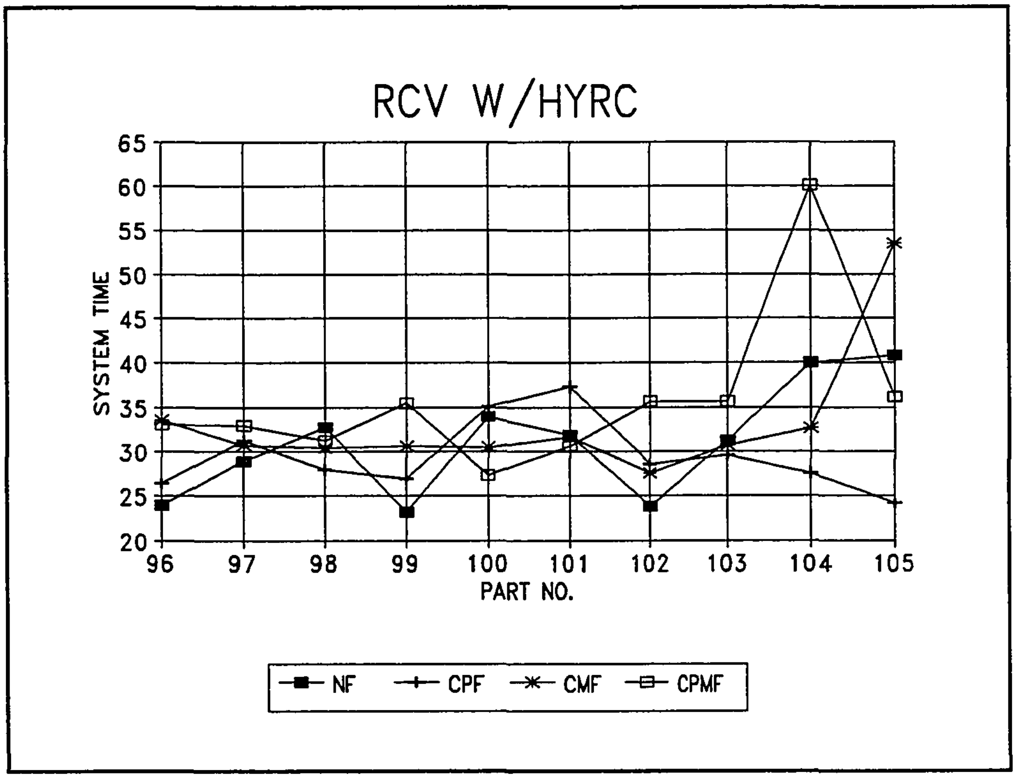


Figure 11. **RCV with Hierarchical Control.**

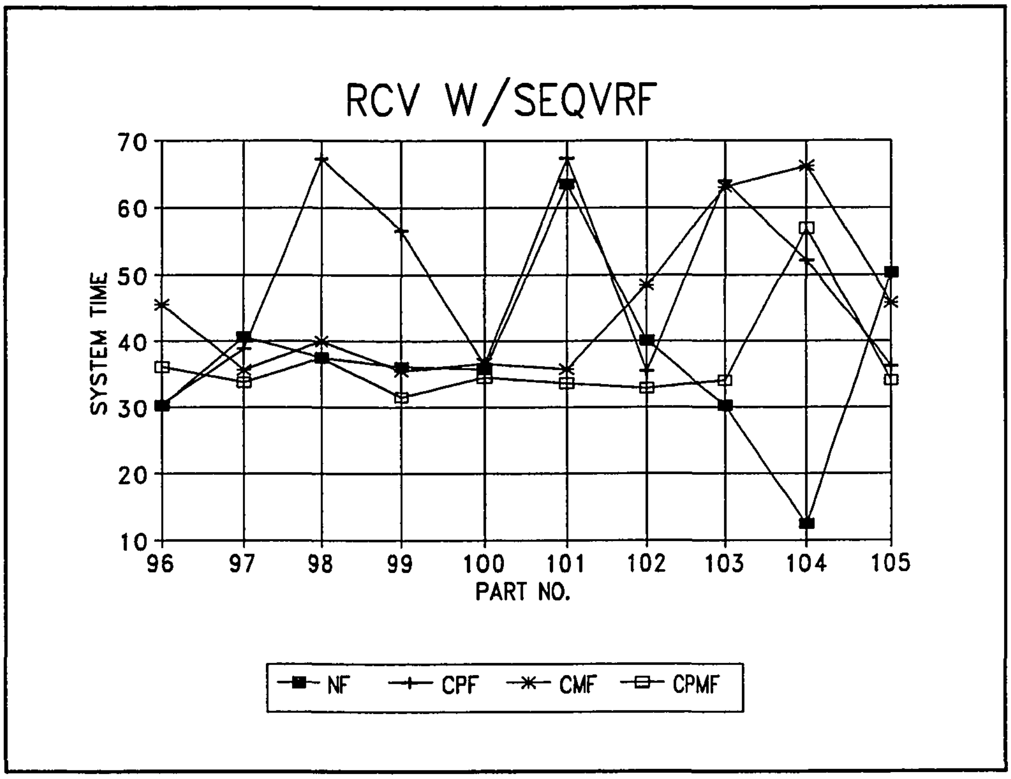


Figure 12. RCV with Flexible Sub-Sequential Control (no computer).

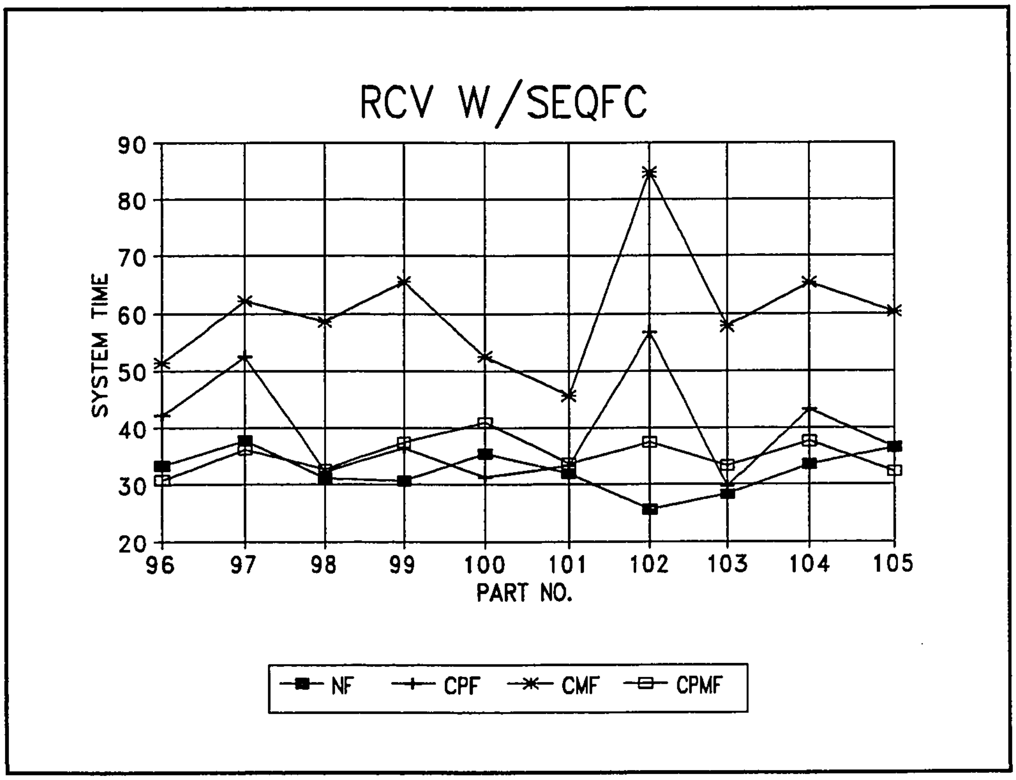


Figure 13. RCV with Flexible Sub-Sequential Control (with computer)

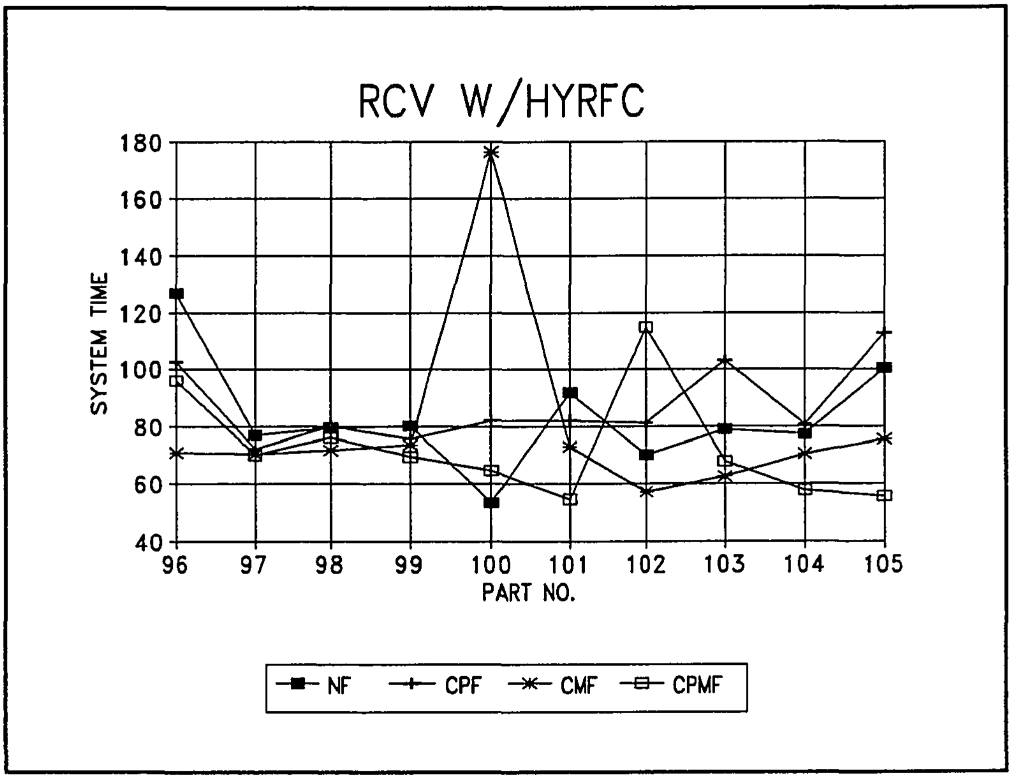


Figure 14. RCV with Flexible Sub-Hierarchical Control

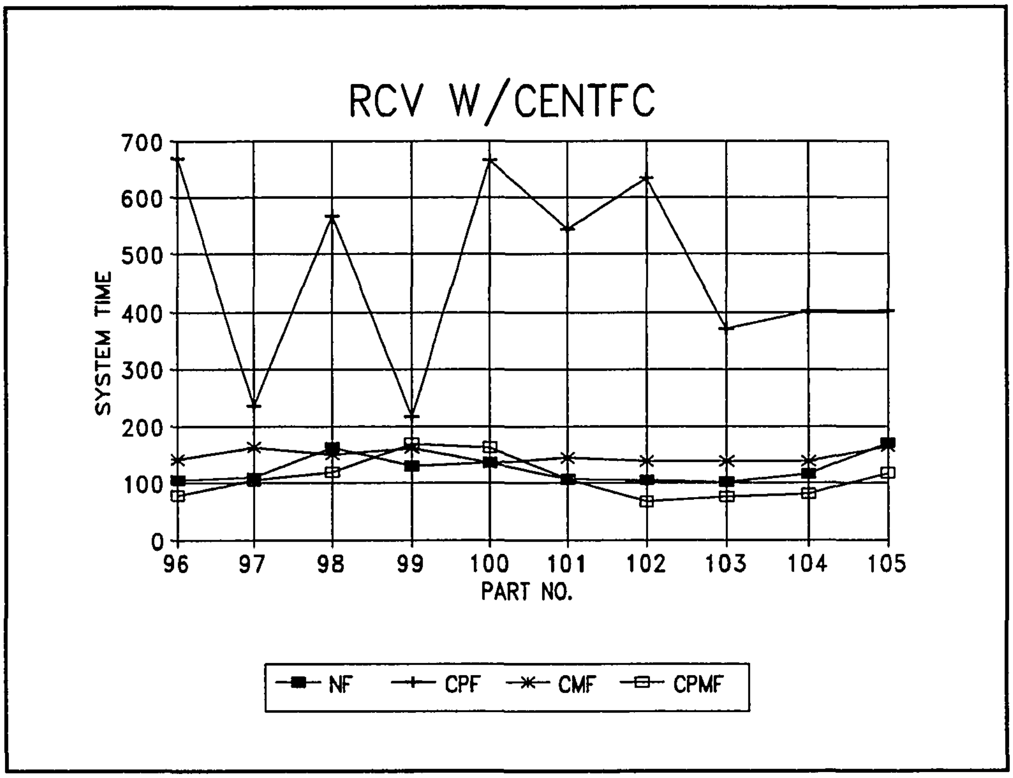


Figure 15. RCV with Flexible Sub-Central Control.

discovering the different levels of effectiveness in the fault tolerance among the control structures when the hypothesis was not true. The first step was carried out with a one-way Analysis of Variance (ANOVA). The second step was accomplished using Duncan's multiple-range test [51, 52]. These analyses were performed with a program developed in SAS [52, 53, 54].

The means of the system working times for the eight control structures with respect to their fault tolerance characteristics are listed in Table 4. As a result (see Tables 5, 6, and 7.), the F value is 4.38 and the probability of obtaining a value this large by chance is 0.0029. In Table 6, the F value and probability value are of primary interest. Therefore, the hypothesis is rejected since F=4.38 is greater than 1. It is concluded that with the different control structures, the fault tolerance effectiveness of the RCV system is not equivalent for all control structures. In another words, there is a significant difference between the control structures with respect to their fault tolerance characteristics. Furthermore, the Duncan multiple range test (p=0.05) shows that in fault tolerance, the effectiveness of the RCV system with the flexible sub-central control structure is significantly less than for the other seven control structures (see Table 7).

Whether the other seven control structures are significantly different from each other or not was tested

Table 4. Mean System Times for Eight Control Structures.

CONTROL STRUCTURE	SYSTEM TIME
1) Flexible Sub-Central Control	212.85
2) Flexible Sub-Hierarchical Control	80.91
3) Central Control	72.45
4) Flexible Sub-Sequential Control with No Computer	41.93
5) Flexible Sub-Sequential Control with Computer	41.85
6) Hierarchical Control	32.41
7) Sequential Control with No Computer	18.71
8) Sequential Control with Computer	14.45

Table 5. Mean System Times with Fault Tolerance Cases (for eight control structures).

CASES	OBS	STRUCTURES	SYSTEM TIME
NF	1	seqvr	23.153
CPF	2	seqvr	15.622
CMF	3	seqvr	20.404
CPMF	4	seqvr	15.647
NF	5	seqc	16.624
CPF	6	seqc	11.854
CMF	7	seqc	17.739
CPMF	8	seqc	11.593
NF	9	centc	70.073
CPF	10	centc	84.485
CMF	11	centc	67.310
CPMF	12	centc	67.937
NF	13	hyrc	31.042
CPF	14	hyrc	29.488
CMF	15	hyrc	33.185
CPMF	16	hyrc	35.920
NF	17	seqvrf	37.645
CPF	18	seqvrf	48.436
CMF	19	seqvrf	45.199
CPMF	20	seqvrf	36.422
NF	21	seqfc	32.390
CPF	22	seqfc	39.409
CMF	23	seqfc	60.379
CPMF	24	seqfc	35.218
NF	25	hyrfc	83.546
CPF	26	hyrfc	87.235
CMF	27	hyrfc	80.143
CPMF	28	hyrfc	72.723
NF	29	centfc	124.381
CPF	30	centfc	471.114
CMF	31	centfc	147.408
CPMF	32	centfc	108.506

Table 6. Results of Analysis of Variance (for eight control structures).

Source	DF	Sum of Square	Mean Square	F Value	Pr > F
Model	7	115981.01	16568.717	4.38	0.0029
Error	24	90689.01	3778.709		
Corrected Total	31	206670.02			
R-Square	C.V.	Root MSE	SYST Mean		
0.561189	95.38599	61.47120	64.444688		

Table 7. ANOVA and Duncan Data Analysis Results (for eight control structures).

Duncan Grouping	Mean	N	STRUCTURES
A	212.85	4	centfc
B	80.91	4	hyrfc
B	72.45	4	centc
B	41.93	4	seqvrf
B	41.85	4	seqfc
B	32.41	4	hyrc
B	18.71	4	seqvr
B	14.45	4	seqc

Note: Means with the same letter are not significantly different.

further with the same method discussed above (see Tables 8, 9, and 10). This time, the F value is 53.99 and the probability of obtaining a value this large by chance is 0.0001. Therefore, there is also a significant difference between the remaining seven control structures with respect to their fault tolerance characteristics since the F=53.99 is greater than 1. Furthermore, the Duncan multiple range test (p=0.05) shows that in fault tolerance, the effectiveness of the RCV system with the flexible sub-hierarchical and central control structures is significantly less than for the other five control structures (see Table 10). However, for the other five control structures, the effectiveness of the RCV system with the flexible sub-sequential control (without computer), flexible sub-sequential control (with computer), and hierarchical control structures is significantly less than for the sequential control (without computer) and the sequential control (with computer) structures.

Therefore, in this research, the RCV system with flexible sub-central control structure had significantly lower fault tolerance than the other seven structures did. For the remaining seven control structures, sequential control with and without computer exhibited the best; hierarchical control and flexible sequential control with and without computer the next best; and central control and flexible hierarchical control the least effective fault tolerance. In each table,

Table 8. Mean System Times with Fault Tolerance Cases (for seven control structures).

CASES	OBS	STRUCTURES	SYSTEM TIME
NF	1	seqvr	23.153
CPF	2	seqvr	15.622
CMF	3	seqvr	20.404
CPMF	4	seqvr	15.647
NF	5	seqc	16.624
CPF	6	seqc	11.854
CMF	7	seqc	17.739
CPMF	8	seqc	11.593
NF	9	centc	70.073
CPF	10	centc	84.485
CMF	11	centc	67.310
CPMF	12	centc	67.937
NF	13	hyrc	31.042
CPF	14	hyrc	29.488
CMF	15	hyrc	33.185
CPMF	16	hyrc	35.920
NF	17	seqvrf	37.645
CPF	18	seqvrf	48.436
CMF	19	seqvrf	45.199
CPMF	20	seqvrf	36.422
NF	21	seqfc	32.390
CPF	22	seqfc	39.409
CMF	23	seqfc	60.379
CPMF	24	seqfc	35.218
NF	25	hyrfc	83.546
CPF	26	hyrfc	87.235
CMF	27	hyrfc	80.143
CPMF	28	hyrfc	72.723

Table 9. Results of Analysis of Variance (for seven control structures).

Source	DF	Sum of Square	Mean Square	F Value	Pr > F
Model	6	15296.197	2549.3666	53.99	0.0001
Error	21	991.622	47.220		
Corrected Total	27	16287.819			
R-Square	C.V.	Root MSE	SYST Mean		
0.939119	15.89064	6.871687	43.243607		

Table 10. ANOVA and Duncan Data Analysis Results (for seven control structures).

Duncan Grouping	Mean	N	STRUCTURE
A	80.912	4	hyrfc
A	72.451	4	centc
B	41.926	4	seqvrf
B	41.849	4	seqfc
B	32.409	4	hyrc
C	18.707	4	seqvr
C	14.453	4	seqc

Note: Means with the same letter are not significantly different.

the variable of SYST is the mean system times for each fault case.

The flexible control scheme did not improve the fault tolerance of the RCV system because the complexity of the structure and, in particular, the control switching process, increased the time consumption. However, the flexible control scheme may be significant to the integrated manufacturing environment because its subcontrol structures can be interchanged to accommodate different manufacturing processes at different times. Similarly, flexible control may be cost effective when a large variety of manufacturing activities need to be coordinated simultaneously. Under these circumstances, the fault tolerance of the flexible control scheme could be expected to exceed that of the other structures. Special features of flexible control are not possessed by the other control schemes. The features of the control structures are discussed next.

Evaluation of Features

Two features of the control structures for the RCV system are worth examining: system operability and system expandability.

System operability is defined as the number of ways to execute one operation. The number of ways is decided by Larsson's counting technique [61]: "If a first operation can be performed in any of n_1 ways and a second operation can then be performed in any of n_2 ways, both operations can be performed (the second immediately following the first) in n_1*n_2 ways." (p.43)

The RCV inspection system consists of four major operations: initiating the system, processing the part image data, handling the good parts, and handling the bad parts. Referring to the earlier structure figures, the results can be derived in Table of Number of Ways to Perform Operations (Table 11). The flexible control structure has a higher number of ways to execute the four operations than do the other control schemes.

System expandability is defined as the number of additional components which can be added to the system. This feature is examined for the following two cases:

1. Adding a component to the existing system. If a control structure permits this "adding", then the system expandability is "1", otherwise it is "0". The results of Case 1 show that all five control structures allow expansion by one additional component. This is made possible by utilizing one of the two free communication ports offered by the vision system.

Table 11. Number of Ways to Perform Operations.

Operations Structures /# of ways	1) Initiating System	(2) Processing Data	(3) Handling Good Parts	(4) Handling Bad Parts
seqvr/1	conveyor	vision	vision-gate	vision-robarm
seqc/4	conveyor	vision-PC	1 vision-gate 2 PC-gate	1 vision-robarm 2 PC-robarm
centc/4	pc	vision-PC	1 vision-gate 2 PC-gate	1 vision-robarm 2 PC-robarm
hyrc/4	sensor	vision-PC	1 vision-gate 2 PC-gate	1 vision-robarm 2 PC-robarm
flxc/81	1 conveyor 2 sensor 3 IC	1 vision 2 vision-ARTIC 3 vision-IC	1 vision-gate 2 ARTIC-gate 3 IC-gate	1 vision-robarm 2 ARTIC-gate 3 IC-robarm

2. Adding a maximum number of components to the existing system. The results for Case 2 show that each traditional control structure has a maximum system expandability of two components. The maximum expandability for the flexible control structure is 31. This is possible because the vision system has two extra ports (total four ports), and the industrial computer allows four

ARTIC cards to be installed. Each ARTIC card has eight communication ports. Since three ports of one ARTIC card are connected to the R, V, and IC, the system expandability is 2+(4x8-3)=31. Thus, the table for system expandability is established.

Table 12. System Expandability.

Control Structures	Case1 add one	Case2 add max
seqvr	1	2
seqc	1	2
centc	1	2
hyrc	1	2
flxc	1	31

CHAPTER 7

CONCLUSION AND FUTURE RESEARCH

Conclusion

The flexible control theory proposed and studied in this research has been examined by employing a factory simulation of an RCV inspection system. With the factory simulation approach developed in this research, a flexible control structure and alternative control structures have been evaluated with respect to their fault tolerance characteristics.

In the research, a mathematical model of the flexible control structure has been defined, and its properties have been derived. Properties 1 and 2, which correspond to the first part of the flexible control structure definition, provide a static description, and property 3, which corresponds to the second part of the definition, provides a dynamic description for any flexible control scheme.

Because the PlantWorks software anomalies can and did produce seemingly inconsistent results in the system working time during simulation, statistical inference cannot be drawn as to the efficiency of fault tolerance for different control structures. Within the constraints of the data analyzed, it is believed that the flexible control structure is as effective as the central or hierarchical control structures in the RCV system with respect to fault tolerance, with the exception of the flexible sub-central control structure which is not as effective. Examination of control structure features indicates that if the RCV system is managed by the flexible control structure, the RCV system possesses operability and expandability characteristics that the alternative control structures do not.

It is concluded that the proposed flexible control structure may be useful in integrated manufacturing design, particularly when a large variety of manufacturing activities must be accommodated (see Appendix A). This research has contributed to the mathematical models for description of manufacturing control schemes and to the evaluation methods required for assessment of integrated manufacturing control structures. The idea of flexible control structure appears to be significant in integrated manufacturing and automated production system design. Considering the results reported from this research, factory simulations can effectively contribute to additional studies in manufacturing efficiency. Further research is needed to confirm precisely how wide a range and what types of activities would justify this approach.

Future Research

Does the flexible control structure have real potential for meeting the needs of many types of manufacturing activities? Given the theoretical description and both analytical and simulation results discussed here, the answer is more "yes" than "no". Even though the RCV system with flexible control is theoretically sound and appears to be quite practical, some future research is suggested to address the following: 1) from property 1 mentioned in Chapter 4, how to obtain an optimal design of the flexible control structure from time, cost, and reliability points of view, 2) how to simulate a system with flexible control to solve a variety of manufacturing problems, 3) how to implement an uniform interface between each pair of integrated components, 4) how to make a flexible control scheme intelligent enough to switch to different control structures in order to address different problems, and 5) how to manage an efficient coordination among the elements of an integrated system in order to execute multiple tasks simultaneously.

LIST OF REFERENCES

[1] Mikell P. Groover **Automation, Production Systems, and Computer Integrated Manufacturing.** New Jersey: Prentice-Hall, 1987.

[2] Mikell P. Groover, Mitchell Weiss, Roger N. Nagel, and Nicholas G. Odrey, **Industrial Robotics: Technology, Programming, and Applications**, New York: McGraw-Hill Book Company, 1986.

[3] Albert Jones and Abdol Saleh, "A Decentralized Control Architecture Computer Integrated Manufacturing System," **Proceedings of the IEEE International Symposium on Intelligent Control**, Albany, New York, September 1989, pp. 56-59.

[4] Richard Burkel and Jimmy Murphy, "Infrared Imaging Systems Automate Aircraft Engine Inspection At General Electric," **Industrial Engineering**, April 1989, pp. 23-28.

[5] "Westinghouse Electric Asks 'What if?' With Modeling At Its Productivity Center," **Industrial Engineering**, June 1990, pp. 22-24.

[6] Stanley L. Robinson and Richard K. Miller, **Automated Inspection and Quality Assurance**, New York: Marcel Dekker, SQC Quality Press, 1989.

[7] Richard C. Dorf, **Robotics and Automated Manufacturing**, Reston, Virginia: Reston Publishing Company, 1983.

[8] Rashmi Roy and Sema Alptekin, "Development of a Flexible Automated Assembly Cell in a Multitasking Environment," **American Society for Engineering Management**, 1990 Annual Meeting, 1990, pp. 40-43.

[9] Nils R. Sandell, Pravin Varaiya, Michael Athans, and Michael G. Safonov, "Survey of Decentralized Control Methods for Large Scale Systems," **IEEE Transactions on Automatic Control**, Vol. 23, No. 2, April, 1978, pp. 108-125.

[10] Bob Francis, "A New Vision of Quality Control," **Datamation**, Vol.36, April 1, 1990, pp. 70-72.

[11] Sandra Brooks Dornan, "Sensing Systems for the Real-Word Factory," **Production**, September 1987, pp. 62-68.

[12] William E. Macintosh, "Automating the Inspection of Printed Circuit Boards," **Robotics Today**, June 1985, pp. 75-78.

[13] Gary S. Vasilash, "When Do You Measure? And Do You?" **Production**, February 1988, pp. 39-49.

[14] W. J. Pastorius, "Vision Measures Up in Automated Assembly," **Manufacturing Engineering**, April 1989, pp. 81-83.

[15] James R. Koelsch, "Sensors-The Missing Link," **Manufacturing Engineering**, October 1990, pp. 53-56.

[16] Grary S. Vasilash, "Cell Control: Making Sense In CIM's Gary Zone," **Production**, December 1990, pp. 36-40.

[17] Edward Jones, **Using dBASE IV 1.1**, Berkeley, California: McGraw-Hill, 1990.

[18] "Puma Robot Models 761/762 Equipment Manual," Unimation, a Westinghouse Company.

[19] "IRI Vision Systems Equipment Manual," ALCYON Corporation.

[20] IBM ARTIC card Documentation.

[21] Dimitri Bertsekas and Robert Gallager, **Data Networks**, Englewood Cliffs, New Jersey: Prentice-Hall, 1987.

[22] Eric Teicholz, and Joel N. Orr, **Computer Integrated Manufacturing Handbook**, New York: McGraw-Hill, 1987.

[23] Andrew Kusiak, **Intelligent Manufacturing Systems**, Englewood Cliffs, New Jersey: Prentice Hall, 1990.

[24] L.J. McGuffin, L. O. Reid, and S. R. Sparks, "MAP/TOP in CIM Distributed Computing," **IEEE Network**, Vol.2, No. 3, May 1988, pp. 23-31.

[25] Carl Kaltwasser, "Know How To Choose The Right CIM System Integrator," **Industrial Engineering**, July 1990, pp. 27-29.

[26] John M. Martin, "Personal Computers in Manufacturing," **Manufacturing Engineering**, June 1989, pp. 16-18.

[27] Gerald Boblitz, "Simulation Eliminates Need For An $80,000 Machine And Conveyor Investment," **Industrial Engineering**, March 1991, pp. 26-28.

[28] Dr Paul Ranky, **The Design and Operation of FMS**, Kempston, Bedford, UK: IFS (Publications) Ltd, 1983.

[29] Chern-Lin Chen and Yuan-Yih Hsu, "Applying Two Numerical Methods to the Decentralized Modal Control Problem," **IEEE Transactions on Automatic Control**, Vol. 36, No. 5, May 1991, pp. 627-629.

[30] AI Wrigley, "New GM Machine Vision System Checks Surface Quality," **Metalworking News**, Vol. 16, January 1, 1989, p. 10.

[31] Lauri Giesen and Dave Fusaro, "Robots, Vision Systems Reappear at Assembly Technology Expo.," **Metalworking News**, September 1989, p. 15.

[32] Richard Layne, "Computer-Integrated Manufacturing: Can You Afford (to miss) It?" **D & B Reports**, Vol.34, July-August 1986, p. 5.

[33] Ling Ling Pan, Steve Henderson, and Tom Ruff, "RCV Project Report," Norfolk, Virginia, Fall 1991.

[34] Laurence D. Richards, **Constraint Theory**, Lanham, MD: University Press of America, 1983.

[35] Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman, **The Design and Analysis of Computer Algorithms**, Reading, Mass: Addison-Wesley Publishing Company, 1974.

[36] **Quattro Pro User's Guide**, Version 2.0, Borland International, Inc., 1987.

[37] Francis John Anscombe, **Computing in Statistical Science through APL**, New York: Springer-Verlag, 1981.

[38] Wayne J. Graybeal and Udo W. Pooch,**Simulation: Principles and Methods**, Cambridge, Massachusetts: Winthrop Publishers, 1980.

[39] J. Schwarzenbach and K. F. Gill, **System Modelling and Control**, New York: Halsted Press, 1978.

[40] Ernst J. Schuegraf and Martin F. Van Bommel, "Automatic Indexing of Document Databases by Cooperating Expert System," **Database and Expert Systems Applications Proceedings of the International Conference** in Vienna, Austria, 1990, Springer-Verlag, Wien, pp. 8-11.

[41] Elizabeth C. Scheyder, "Relational Database Applications in Manufacturing System Design," **Database and Expert Systems Applications Proceedings of the International Conference** in Vienna, Austria, 1990, Springer-Verlag, Wien, pp. 96-100.

[42] C. Bolte, K. Kurbel, and C. Raulenstranch, "Integration of Knowledge-Based Modules into a Distributed Production Planning and Control System," **Database and Expert Systems Applications Proceedings of the International Conference** in Vienna, Austria, 1990, Springer-Verlag, Wien, pp. 101-104.

[43] Henk de Swaan Arons and Eric-Paul Jansen, "Application of Hybrid Expert Systems in Computer Integrated Manufacturing," **Database and Expert Systems Applications Proceedings of the International Conference** in Vienna, Austria, 1990, Springer-Verlag, Wien, pp. 90-95.

[44] Stefan Bottcher, "A Tool Kit for Knowledge Base Production Planning System," **Database and Expert Systems Applications Proceedings of the International Conference** in Vienna, Austria, 1990, Springer-Verlag, Wien, pp. 84-89.

[45] T. Anderson and P.A. Lee, **Fault Tolerance: Principles and Practice**, London: Prentice Hall International, 1981.

[46] Paul Merenbloom, "So You Need/Want Fault Tolerance? Don't Jump to Solutions," **InfoWorld**, Jan 25, 1993, p. 6.

[47] Des Raj, **Sampling Theory**, New York: McGraw-Hill Book Company, 1968.

[48] William H. Von Alven, **Reliability Engineering**, Englewood Cliffs, New Jersey: Prentice-Hall, 1965.

[49] John M. Chambers, **Computational Methods for Data Analysis**, New York: John Wiley and Sons, 1977.

[50] Fred N. Kerlinger, **Foundations of Behavioral Research**, New York: Holt, Rinehart and Winston, 1965.

[51] Ronald P. Cody and Jeffrey K. Smith, **Applied Statistics and the SAS Programming Language**, New York: Elsevier Science Publishing Co., 1985.

[52] **SAS Language**, Reference Version 6, First Edition, SAS Institute, Inc., 1992.

[53] **SAS Procedures Guide**, Reference Version 6, Third Edition, SAS Institute, Inc., 1992.

[54] Richard de Neufville and Joseph H. Stafford, **Systems Analysis for Engineers and Managers**, New York: McGraw-Hill Book Company, 1971.

[55] Rex Maus and Jessica Keyes, **Handbook of Expert Systems in Manufacturing**, New York: McGraw-Hill, 1991.

[56] J. N. Daigle, A. Seidmann, and J. R. Pimentel, "Communications for Manufacturing: An Overview," **IEEE Network**, Vol.2, No.3, 1988, pp. 6-13.

[57] Andrew Kusiak and Sunderesh S. Heragu, "Computer Integrated Manufacturing: A Structural Perspective," **IEEE Network**, Vol.2, No.3, 1988, pp. 14-22.

[58] **IBM CIM Architecture**, Printed in the U.S., 1989.

[59] Harold Lorin, "Application development, software engineering and distributed processing," **Computer Communications**, Vol. 13, No. 1, January/February 1990, pp. 4-15.

[60] Michael Quinn Patton, **Qualitative Evaluation and Research Methods**, Second Edition, Beverly Hills: Sage Publications, 1990.

[61] Harold J. Larson, **Introduction to Probability Theory and Statistical Inference**, Third edition, Wiley Series in Probability and Mathematical Statistics, New York: Wiley, 1987.

[62] Ling Ling Pan, Darrell Duane, Dale Geesey, "Final Report (Part I): Integration of PUMA 762 Robot & IRI Vision System D256 via IBM ARTIC Card," Old Dominion University, Norfolk, Virginia, April 1991.

APPENDIX A

THE HARDWARE CONFIGURATION

The configuration of the RCV system is such that there is a communication link between each pair of integrated components. In this system, an IBM 7552 COMPUTER, an IRI D256 VISION MACHINE, and a PUMA 762 ROBOT can control each other or be controlled by each other. The three pieces of hardware pieces are integrated as a team in real-time situations.

The hardware link between the robot and the vision machine cannot be implemented if the robot is already physically connected to the industrial computer. This is because the robot does not possess multiple ports to connect more than one of the integrated equipments. The link between the robot and the vision machine is achieved by an IBM Real-time Interface Co-processor (ARTIC) card which is installed in the industrial computer [62].

The RCV system can be configured in the context of a flexible control structure by employing an IBM Real-time Interface Co-Processor (ARTIC) card. The ARTIC card has two SEQUENTIAL-232 compatible serial interface ports available. The ARTIC card also has its own co-processor so that it may process its own information.

The ARTIC card in the IBM 7552 Industrial Computer is a communication bridge from the IRI D256 Vision System to the PUMA 762 Robot. Within the Industrial Computer, the CPU is also able to communicate with the ARTIC card. The industrial computer, the robotics arm, and the vision system are integrated in this way so that they can work as a team within a flexible control structure in real-time situations. The control structure can be flexible because there are always communication paths to support whatever fixed control structure (hierarchical, network, or sequential) is needed The industrial computer provides a common linguistic environment through which all three systems can communicate. The features of the RCV inspection system when operated by the flexible control structure include:

Flexibility

In real time situations, the structure of the system will be changed dynamically. For example, when all of the integrated components connected to the industrial computer are allowed to communicate with the industrial computer, the structure of the system is a central control structure. When both the robot and the vision machine are under the industrial computer's control, the structure is hierarchical.

Automated Inspection Process

Once the integrated system has been developed, it could be used to do the quality inspection automatically. For example, the vision machine examines parts and then sends the information to the robot. The robot will allocate the good parts or bad parts to the respective places with the information provided by the vision system. Next, the robot feeds back a "done" message to the vision machine. The configuration of the integration should also allow the IBM 7552 computer to interrupt the communication between the PUMA Robot and the IRI D256 Vision Machine for certain reasons. Thus, the computer can also communicate with the robot and the vision machine.

Multiple Task Environment

The structure allows the robot, industrial computer, and vision machine to perform their jobs simultaneously since the configuration allows each workstation to be independent from the others. For instance, the robot arm can be a material handler, the industrial computer a data processor, and the vision machine an image processor.

If the robot arm and the industrial computer are used together, they can do the assembly work, material handling, and storage. If the robot is associated with the machine vision system and ARTIC card, the inspection task can be carried out automatically, which is an original intent for the system. If we use both the industrial computer and the vision machine, we may analyze the defective image data and make decisions about production process control. It is important to point out that the tasks can be executed either simultaneously or sequentially.

Expandability of the System

The reasons that the system is expandable are that the ARTIC card can be expanded to eight I/O ports and the vision machine to four I/O ports. This allows for significant expansion with small modifications to the programs. For example, a computer controlled conveyer, a computer-aided design system, and a different kind of robot, etc., are all, potentially, additional components.

Hence, this configuration can be expanded to include all of the necessary components to perform the physical activities related to the production processes that take place in a factory. And, the system will become an automated production system including all manufacturing processes, assembly operations, material handling tasks, and inspections that are performed on the product. This automated production system offers a future opportunity for computer integrated manufacturing (CIM) that portends the pervasive use of computers for designing products, planning production, controlling operations, and performing the various business related functions needed by a manufacturing firm.

Comparing flexible control to the traditional controls, the following conclusions are derived:

1. Flexibility is superior.
2. The average time of operation is similar.
3. System expandability is greatly increased and multiple tsk performance is possible.
4. Interchangeability of manufacturing activities is enhanced.

However, switching to different control structures in the flexible control scheme may involve a tradeoff with operating time for some applications. Three points can be made:

1. The integrated RCV inspection system with the flexible control structure is a new application of integrated manufacturing,
2. The mathematical model for the flexible control structure offers a descriptive approach for studying manufacturing control schemes, and
3. The fault tolerance test with a factory simulation provides a methodology for exploring alternative control structures with respect to their fault tolerance characteristics.

APPENDIX B

PROGRAM LIST

Control Procedures

The algorithms for the different control procedures are:

1. Central Control
2. Sequential Control with Computer
3. Sequential Control with no Computer
4. Hierarchical Control
5. Flexible Control

Switching Control

The algorithm of switching control operation, σ, is shown as the following. The language used is a functional language from IBM PlantWorks:

```
1.  CHNSTART
2.  CASETEST CASE_NUMBER
     IF CASE_NUMBER = 1, THEN CHNTRIG LLP:SEQVR
     IF CASE_NUMBER = 2, THEN CHNTRIG LLP:SEQFC
     IF CASE_NUMBER = 3, THEN CHNTRIG LLP:CENTFC
     IF CASE_NUMBER = 4, THEN CHNTRIG LLP:HYRFC
3.  END
```

Within the flexible control procedure, the σ is decided by the case number. It can be switched from one sub-control structure to another by indicating the case number. There are many ways to design σ. This will depend on the particular requirements.

Inspection

The detailed inspection algorithm written by the functional language is:

```
1.   CHNSTART
2.   RANDOM = RAND(1)
3.   LLP:SIZE1 = 10. + (RANDOM/500000.)
4.   LLP:SIZE2 = 10.
5.   DIF = | LLP:SIZE1.VALUE-LLP:SIZE2.VALUE |
6.   IFTEST DIF < 0.0567
     YES---LLP:PASSFAIL.VALUE=TRUE
     NO---LLP:PASSFAIL.VALUE=FALSE
7.   END
```

Fault Tolerance

Component Fault Generator LLP:CPFTG. The Component Fault Generator produces faults in the following components, depending on the particular control structure in use:

1. Conveyor,
2. Feeder,
3. Sensor,
4. Camera,
5. Vision machine,
6. Gate,
7. Robarm,
8. PC,
9. IC (Industrial Computer), and
10. ARTIC.

The first seven components are used in all control structures. A PC is used in the hierarchical and central control structures. An IC and ARTIC card are needed in the flexible control structure.

Communication Fault Generator LLP:CMFTG. The Communication Fault Generator produces faults in the links between components (labeled from 1 to 28), which vary from one control structure to another.

Sequential control with no computer (seqvr):

Feeder<---Conveyor (1)
Sensor<---Feeder (2)
Camera<---Sensor (3)
Sensor<---Camera (4)

Vision<---Sensor (5)

Gate<---Vision (6)

Robarm<---Vision (7)

Sequential control with computer (seqc):

Feeder<---Conveyor (1)

Sensor<---Feeder (2)

Camera<---Sensor (3)

Sensor<---Camera (4)

Vision<---Sensor (5)

PC<---Vision (8)

Gate<---PC (9)

Robarm<---PC (10)

Central control (centc):

Conveyor<---PC (11)

Feeder<--->PC (12)

Sensor<--->PC (13)

Camera<--->PC (14)

Vision<---Sensor (5)

PC<---Vision (8)

Gate<---PC (9)

Robarm<---PC (10)

Hierarchical control (hyrc):

Conveyor<---Sensor (15)
Feeder<--->Conveyor (16)
Camera<--->Sensor (17)
Vision<---Camera (18)
PC<---Vision (8)
Gate<---PC (9)
Robarm<---PC (10)

Flexible control (flxc):

case1: Seqvrf is same as seqvr.
case2: Seqfc is similar to seqc except that PC is replaced by ARTIC.
case3: Hyrfc is similar to hyrc except that PC is replaced by ARTIC.

Thus, for case2 and case3 communication links (8), (9), and (10) become:

ARTIC<---Vision (19)
Gate<---ARTIC (20)
Robarm<---ARTIC (21)

case4: Centfc

Conveyor<---IC (22)
Feeder<--->IC (23)
Sensor<--->IC (24)
Camera<--->IC (25)
IC<---Vision (26)
Gate<---IC (27)
Robarm<---IC (28)

By using both component and communication fault generators, different combinations of faults can be produced.

Calculate Time Interval (LLP:INTV). Each component function time and each communication time are calculated by getting a pair of begin and end times. This happens after each part inspection.

Fault Monitor (LLP:FAULTM). Under real-time conditions, the monitor checks whether the system time is out of the specified time constraint or not. If YES, the monitor sounds an alarm, triggers the fault accessors, and sends a warning message to the video display. Otherwise, it keeps patrolling.

Fault Accessors (LLP:FAULTACS). The fault accessors search for and locate the fault(s), then trigger the fault treater to fix it. The location of the fault is found by comparing each component function time and communication time with their corresponding constraints. For those constraints not satisfied, a FALSE flag is given. Otherwise, they are in the TRUE status.

Fault Treater (LLP:FAULTREA). The treater resets the fault component function time and/or the fault communication time. Then, it finds out under which control structure the RCV system generated the fault(s) and re-configures the control structure by retriggering the control program. Finally, the treater sends a message of "RCV System Back to Normal" to the screen.

APPENDIX C

PREPARATION FOR USING THE RCV SOFTWARE

Before using the RCV software, one must: 1) Login to the OS/2 operating system and then the IBM PlantWorks system, 2) Select RUNTIME icon then select RUNTIME MONITOR item, 3) Choose SELECT DISPLAY at DISPLAY icon, and 4) Input the display name: LLP:HOME when the dialogue box shows up. After finishing the above four steps, the display called "RCV Inspection System Simulation" appears. Now, you are ready to use the RCV simulation package.

APPENDIX D

SIMULATION AND DATA ANALYSIS RESULTS

Under each control structure, the RCV system inspects 200 parts. The average runtime for each batch was one hour and fifteen minutes. Altogether, 32 batch runs were completed for eight control structures under four cases of NF, CPF, CMF, and CPMF. There are three types of report.

1. After each part inspection, the RCV System Report is automatically generated. From this report, the details of real-time information on the system component status and communication status can be understood.
2. After each batch inspection, the System Time Report is automatically generated. This report lists the RCV system times for each part inspection, along with the real-time control structure name and the batch size. It is necessary to point out that in this report, some part numbers have much higher system time values; this occurs when the robot arm is in function.

3. After each batch inspection, a Statistical Report can be generated. This report provides the real-time information on:

    a. Total inspected parts,
    b. Total passed parts and percentage,
    c. Total failed parts and percentage,
    d. Each control structure status, and
    e. total inspection time (in seconds).

Reports 1 and 2 are useful in the fault tolerance analysis and Reports 2 and 3 are useful in evaluating the RCV inspection system behavior.

Regression Output

RCV SYSTEM TIME WITH SEQVR

****

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	17.41	13.86	12.82	13.91
97	26.10	10.94	16.34	18.06
98	38.72	11.00	19.60	16.97
99	13.97	13.04	20.57	15.16
100	20.18	15.53	16.01	14.93
101	17.18	11.75	22.50	14.75
102	26.22	13.66	21.16	15.75
103	42.75	16.22	40.82	17.15
104	13.75	34.15	17.72	16.44
105	15.25	16.07	16.50	14.07
MEAN	23.15	15.62	20.40	15.72

```
  NF-CPF              Regression Output:

Constant                                           20.47375812
Std Err of Y Est                                   6.834834705
R Squared                                          0.101096716
No. of Observations                                         10
Degrees of Freedom                                           8

X Coefficient(s)                  -0.20955203
Std Err of Coef.                  0.220919899
```

```
  NF-CMF              Regression Output:

Constant                                            8.87583056
Std Err of Y Est                                   6.133311623
R Squared                                          0.440878354
No. of Observations                                         10
Degrees of Freedom                                           8

X Coefficient(s)                  0.497912557
Std Err of Coef.                  0.198244821
```

```
  NF-CPMF             Regression Output:

Constant                                            13.6769498
Std Err of Y Est                                   1.119922166
R Squared                                          0.425966933
No. of Observations                                         10
Degrees of Freedom                                           8

X Coefficient(s)                  0.088198082
Std Err of Coef.                  0.036198841
```

RCV SYSTEM TIME WITH SEQC

****

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	13.16	10.75	36.21	11.96
97	15.12	17.63	15.72	10.91
98	12.87	14.51	10.65	12.38
99	20.85	5.56	10.60	11.47
100	31.79	10.67	15.63	11.93
101	14.22	15.66	10.66	12.32
102	14.53	14.00	15.88	12.47
103	17.30	14.07	12.85	11.84
104	12.43	10.63	13.64	10.34
105	13.97	5.06	35.55	10.31
MEAN	16.62	11.85	17.74	11.59

NF-CPF Regression Output:

Constant		14.43554801
Std Err of Y Est		4.283669399
R Squared		0.048676428
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-0.15529042	
Std Err of Coef.	0.242719383	

NF-CMF Regression Output:

Constant		23.55455346
Std Err of Y Est		10.15078166
R Squared		0.044199009
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-0.34982877	
Std Err of Coef.	0.575159107	

NF-CPMF Regression Output:

Constant		11.23884124
Std Err of Y Est		0.850194319
R Squared		0.023863389
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	0.021304064	
Std Err of Coef.	0.048173335	

RCV SYSTEM TIME WITH CENTC

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	56.17	109.80	81.84	63.00
97	69.09	79.57	77.76	77.70
98	68.62	101.69	77.70	55.73
99	64.28	71.86	57.55	57.38
100	69.63	68.17	71.85	71.94
101	66.64	70.87	76.83	70.06
102	85.95	69.55	57.14	71.29
103	67.61	79.12	55.75	74.36
104	75.29	87.06	56.35	65.66
105	77.45	107.16	60.33	71.34
MEAN	70.07	84.49	67.31	67.85

NF-CPF	Regression Output:	
Constant		119.204927
Std Err of Y Est		16.58980402
R Squared		0.060909611
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-0.49548224	
Std Err of Coef.	0.687850227	

NF-CMF	Regression Output:	
Constant		121.1813263
Std Err of Y Est		9.333256382
R Squared		0.330363161
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-0.76878864	
Std Err of Coef.	0.386977599	

NF-CPMF	Regression Output:	
Constant		46.22102808
Std Err of Y Est		7.197768069
R Squared		0.11790545
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	0.308606338	
Std Err of Coef.	0.298435497	

RCV SYSTEM TIME WITH HYRC

****

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	23.99	26.45	33.50	33.03
97	28.84	31.20	30.70	32.88
98	32.73	27.99	30.40	31.15
99	23.20	26.91	30.62	35.50
100	33.94	35.06	30.51	27.36
101	31.83	37.33	31.65	30.58
102	23.81	28.61	27.55	35.68
103	31.18	29.56	30.73	35.68
104	40.03	27.56	32.66	60.13
105	40.87	24.21	53.53	36.21
MEAN	31.04	29.49	33.19	35.82

```
  NF-CPF            Regression Output:

Constant                                    29.93691845
Std Err of Y Est                            4.270104851
R Squared                                   0.000512497
No. of Observations                                  10
Degrees of Freedom                                    8

X Coefficient(s)               -0.01446165
Std Err of Coef.               0.225795687
```

```
  NF-CMF            Regression Output:

Constant                                     11.8895696
Std Err of Y Est                            6.262817201
R Squared                                   0.349128366
No. of Observations                                  10
Degrees of Freedom                                    8

X Coefficient(s)               0.686019921
Std Err of Coef.               0.331166836
```

```
  NF-CPMF           Regression Output:

Constant                                    17.11930897
Std Err of Y Est                            8.651152224
R Squared                                   0.178160301
No. of Observations                                  10
Degrees of Freedom                                    8

X Coefficient(s)                 0.6024319
Std Err of Coef.                0.45745782
```

RCV SYSTEM TIME WITH FLXC SEQVRF

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	30.18	30.46	45.52	36.04
97	40.67	38.81	35.55	33.73
98	37.46	67.20	39.87	37.47
99	36.00	56.61	35.46	31.41
100	35.70	36.50	36.56	34.40
101	63.43	67.31	35.75	33.62
102	40.07	35.45	48.41	32.78
103	30.24	63.83	63.00	34.00
104	12.34	52.09	66.12	56.80
105	50.36	36.10	45.75	33.97
MEAN	37.65	48.44	45.20	36.42

```
NF-CPF              Regression Output:

Constant                                          43.65123566
Std Err of Y Est                                  15.32441964
R Squared                                         0.013563284
No. of Observations                                        10
Degrees of Freedom                                          8

X Coefficient(s)                  0.127102254
Std Err of Coef.                  0.383230838
```

```
NF-CMF              Regression Output:

Constant                                           66.1015072
Std Err of Y Est                                    9.0316558
R Squared                                         0.430344855
No. of Observations                                        10
Degrees of Freedom                                          8

X Coefficient(s)                  -0.55525321
Std Err of Coef.                  0.225862323
```

```
NF-CPMF             Regression Output:

Constant                                           50.6253665
Std Err of Y Est                                  5.682581679
R Squared                                          0.46840108
No. of Observations                                        10
Degrees of Freedom                                          8

X Coefficient(s)                  -0.37729756
Std Err of Coef.                  0.142109169
```

RCV SYSTEM TIME WITH FLXC SEQFC

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	33.26	42.16	51.43	30.75
97	37.77	52.50	62.10	36.14
98	31.04	32.26	58.54	32.76
99	30.67	36.40	65.50	37.41
100	35.40	31.26	52.54	40.94
101	31.88	33.40	45.62	33.65
102	25.53	56.87	84.73	37.35
103	28.30	29.62	57.77	33.30
104	33.53	43.22	65.30	37.60
105	36.52	36.40	60.26	32.28
MEAN	32.39	39.41	60.38	35.22

NF-CPF Regression Output:

Constant		44.97482977
Std Err of Y Est		9.760819031
R Squared		0.004855657
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-0.17183791	
Std Err of Coef.	0.869747195	

NF-CMF Regression Output:

Constant		104.6587012
Std Err of Y Est		9.877734616
R Squared		0.231689351
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-1.36707938	
Std Err of Coef.	0.880165071	

NF-CPMF Regression Output:

Constant		34.58926471
Std Err of Y Est		3.341820294
R Squared		0.000530901
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	0.019411401	
Std Err of Coef.	0.297776121	

RCV SYSTEM TIME WITH FLXC HYRFC

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	126.80	102.72	70.64	96.12
97	77.18	72.00	70.54	70.02
98	79.60	80.17	71.76	76.23
99	80.23	75.64	73.50	69.47
100	53.27	82.20	176.37	64.90
101	91.77	82.13	72.82	54.52
102	70.17	81.20	57.21	114.99
103	78.88	103.00	62.70	67.76
104	77.26	80.54	70.36	57.67
105	100.30	112.75	75.53	55.55
MEAN	83.55	87.24	80.14	72.72

```
  NF-CPF              Regression Output:

Constant                                      52.77281783
Std Err of Y Est                              11.73987366
R Squared                                     0.346978958
No. of Observations                                    10
Degrees of Freedom                                      8

X Coefficient(s)                 0.412493503
Std Err of Coef.                 0.200070997
```

```
  NF-CMF              Regression Output:

Constant                                      150.4133947
Std Err of Y Est                              31.86200683
R Squared                                     0.230726127
No. of Observations                                    10
Degrees of Freedom                                      8

X Coefficient(s)                 -0.84109825
Std Err of Coef.                 0.542992512
```

```
  NF-CPMF             Regression Output:

Constant                                      63.55067651
Std Err of Y Est                              20.19062004
R Squared                                     0.012565649
No. of Observations                                    10
Degrees of Freedom                                      8

X Coefficient(s)                 0.109787704
Std Err of Coef.                 0.344088668
```

RCV SYSTEM TIME WITH FLXC CENTFC

PART NO.	SYST W/NF	SYST W/CPF	SYST W/CMF	SYST W/CPMF
96	104.90	670.02	140.88	77.29
97	110.24	235.80	162.41	104.90
98	162.76	566.77	150.75	119.65
99	129.66	216.87	161.55	170.27
100	136.45	665.36	134.23	162.87
101	106.06	545.11	143.51	106.59
102	104.56	635.02	138.62	67.42
103	102.43	370.50	138.73	77.01
104	115.40	402.71	138.16	81.60
105	171.35	402.98	165.24	117.46
MEAN	124.38	471.11	147.41	108.51

NF-CPF	Regression Output:	
Constant		489.7103338
Std Err of Y Est		179.6228847
R Squared		0.00049539
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	-0.14951105	
Std Err of Coef.	2.374361815	

NF-CMF	Regression Output:	
Constant		118.8157012
Std Err of Y Est		10.73143766
R Squared		0.247133563
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	0.22987674	
Std Err of Coef.	0.141854507	

NF-CPMF	Regression Output:	
Constant		15.66118438
Std Err of Y Est		31.97818127
R Squared		0.280470812
No. of Observations		10
Degrees of Freedom		8
X Coefficient(s)	0.74645497	
Std Err of Coef.	0.422706565	

APPENDIX E

MENUS

IBM PLANT FLOOR SERIES PlantWorks: Application Automation Edition

1. Introducing Plantworks
2. Installing and Configuring PlantWorks
3. Planning and Implementing PlantWorks Applications
4. Definition Service/2 Functions
5. Using Definition Services/2
6. System Concepts
7. Using Execution Services/2
8. Operating PlantWorks
9. Messages and Codes

DAE (Distributed Automation Edition)

1. Communication System/2 Utilities
2. Communication System/2 Installation and Configuration
3. Communication System/2 Device Programmer's Guide
4. Communication System/2 Using Logical Device/for MMS Device

5. Communication System/2 Distributed Screen Design Facility
6. Communication System Application Programming Volume 1
7. Communication System Application Programming Volume 2
8. Communication System Technical Guide and Reference
9. Communication System Device Data Management Support Programming
10. Communication Protocol Programs for Allem-Bradley Devices
11. Communication Protocol Programs/A for GE Fanuc Devices
12. Communication Protocol Programs/A for MODICON Devices
13. Communication Protocol Programs/A for Texas Instruments Devices
14. Host Link
15. Library

IBM Operating System/2 Extended Edition Version 1.3

1. Getting Started
2. User's Guide Volume 1: Base Operating System
3. User's Guide Volume 2: Communications Manager and LAN Requester

4. User's Guide Volume 3: Database Manager
5. Database Manger Exercises
6. Database Manager Structured Query Language (SQL) Concepts
7. Keyboard Layouts
8. Glossary
9. License Information
10. Guide to Information

AUTOBIOGRAPHICAL STATEMENT

Ling Ling Pan was born in Tianjin, China. She received her professional education from Tianjin Textile Technology School. After the two-year Textile Technology School, she was retained to serve the same school as a teacher. In 1983, she received her B.S. degree in computer science from Tianjin University. Then, she was invited to teach in the computer science department of Tianjin Institute of Technology for three years. She transferred from Beijing Software Graduate School to Old Dominion University and obtained the M.S. degree in computer science in 1988. Following her graduation, she continued her doctoral program in engineering management at the same university. She is interested in the fields of computer integrated manufacturing, automatic control, quality control, material handling, as well as artificial intelligence. She has taught courses in Engineering Economics, Calculus, Linear Algebra, and Fortran.