503145 2
NT-P

GEORGIA INSTITUTE OF TECHNOLOGY
OFFICE OF CONTRACT ADMINISTRATION

NOTICE OF PROJECT CLOSEOUT

Closeout Notice Date 08/05/93
Original Closeout Started 07/26/89

Project No. E-24-524 Center No. 05716-0A0

Project Director MITCHELL C M School/Lab ISYE

Sponsor IBM CORPORATION/

Contract/Grant No. 7/22/87 AWD LTR Contract Entity GIT

Prime Contract No.

Title SPONSORED INSTRUCTION FOR MAN-MACHINE SYSTEMS AREA

Effective Completion Date 930630 (Performance) 930630 (Reports)

Closeout Actions Required:	Y/N	Date Submitted
Final Invoice or Copy of Final Invoice	Y	
Final Report of Inventions and/or Subcontracts	N	
Government Property Inventory & Related Certificate	N	
Classified Material Certificate	N	
Release and Assignment	N	
Other	N	

CommentsEFFECTIVE DATE 7-1-87. CONTRACT VALUE $75,000.

Subproject Under Main Project No.

Continues Project No.

Distribution Required:

Project Director	Y
Administrative Network Representative	Y
GTRI Accounting/Grants and Contracts	Y
Procurement/Supply Services	Y
Research Property Managment	Y
Research Security Services	N
Reports Coordinator (OCA)	Y
GTRC	Y
Project File	Y
Other CARL BAXTER-FMD	Y
FRED CAIN-OOD	Y

8-24-524

GEORGIA TECH 1885-1985

DESIGNING TOMORROW TODAY

Georgia Institute of Technology
School of Industrial and Systems Engineering
Atlanta, Georgia 30332-0205
(404) 894-2300

16 June 1988

Ms. Sandra P. Morris
IBM Corporation
1000 NW 51st Street MS 5010
P.O. Box 1328
Boca Raton, FL 33432

Dear Sandy:

Enclosed is an annual report for this year's activities conducted under the sponsorship of the IBM Department Grant to Georgia Tech's School of Industrial and Systems Engineering. Activities include technical interchanges with IBM staff at the Boca facility, sponsorship of a number of M.S. and Ph.D. research efforts, and faculty research in the areas of supervisory control and interactive optimization. Please let me know if you would like more detail or copies of related publications for any of these activities. I look forward to continuing our collaboration in the years to come.

Sincerely,

Christine M. Mitchell

encl.

cc: W. M. Sangster, Dean, College of Engineering
M. E. Thomas, Director, School of Industrial & Systems Engineering
S. P. Krosner, IBM
S. M. Belyeu, IBM

1987-1988 Research Activities conducted with the Sponsorship of IBM Department Grant

to Georgia Tech's School of Industrial & Systems Engineering

Georgia Tech-IBM Technical Interchange

This year, several members of the IBM staff visited Georgia Tech's research facilities in the Center for Human-Machine Systems Research (School of Industrial & Systems Engineering); and Georgia Tech faculty visited the IBM Boca facilities two times. In December, Dr. Christine M. Mitchell (Georgia Tech) and Steve Krosner (IBM employee and Georgia Tech Ph.D. student) made a presentation to Manufacturing Special Products staff, including Mr. John Klein. The purpose was to introduce the Georgia Tech GT-FMS (Georgia Tech Flexible Manufacturing System) research project and explore mutual interests. This meeting was successful and set the stage for a follow-on visit by Dr. Mitchell.

In March, Dr. Mitchell spent three days visiting the IBM Boca Raton facilities. Her visit included several presentations, a tour of the PS/2 manufacturing system, and meetings with several groups to explore the possibility of mutual research activities. The dialogue continues and is likely to lead to some joint activity next year.

M.S. and Ph.D. Thesis Research

The majority of the grant funds this year have been used to support students and research activities related to GT-FMS. This project involved four students, three at the masters level and one doctoral student. One masters student, Dean Hettenbach, is developing an interactive scheduling system for GT-FMS cell level control. His system will enable a human operator to 'tune' a heuristic scheduling system based on real-time feedback of current cell status and system goals. Mr. Hettenbach will evaluate his proposed system with an experiment in which human subjects are trained and control the GT-FMS scheduling system for 10 to 12 hours. His research may give some insight into the effectiveness of human supervision over the parameters of real-time, state-based scheduling systems. Such a system, if successful, could offer substantial improvements over the simple dispatch rules, e.g., first-come-first-served, currently used. This research constitutes Mr. Hettenbach's M.S. thesis. He will complete his degree (M.S. in Industrial & Systems Engineering with a Computer Integrated Manufacturing Systems Certificate) in December.

Charlene Benson is another M.S. student whose thesis will examine the design and evaluation of direct manipulation interfaces for monitoring and supervising the control of predominantly automated manufacturing systems. Her design will be compared experimentally to a more conventional operator workstation. Ms. Benson's research will be completed in Spring 1989.

Steve Krosner is a full-time IBM employee and a part-time Georgia Tech graduate student. He has completed all the degree requirements for a Ph.D. except his thesis. His thesis uses GT-FMS with a configuration based on electronics assembly data from an IBM manufacturing facility. His research proposes a model-based, hierarchical design for the cell-level supervisory controller; this design will constitute a theory of the type of control, display, and information requirements that effective human supervisory control of an

FMS might require. To evaluate the effectiveness of his design, he will run an experiment that compares his proposed design to a conventional operator interface. Mr. Krosner's research will be completed this year.

Faculty Support

Some of the grant funds were used for faculty released time and to support computer system laboratory personnel. Drs. Platzman and Mitchell both used the released time from teaching to carry on an on-going discussion addressing the issues of interactive optimization and control, with emphasis on electronics assembly. Some funds were also used to support Richard Robison, the system manger for the Center for Human-Machine Systems, School of Industrial & Systems Engineering.

Next Year

This grant has been approved for another year. Next year's funds will support the completion of the projects described above. In addition, two new projects will be initiated. One will implement and evaluate the use of an integer program to schedule automated guided vehicles (AGVs) in the context of GT-FMS. The project will examine the mechanisms required for implementation and the effectiveness of this type of optimization when applied to a realistic scheduling problem. The other new project will involve a masters student, Ms. Sally Cohen, who will begin her program in fall. The subject of her research is still undetermined at this time, but will be in the general area of supervisory control of manufacturing systems.

E-24-524

Center for Human-Machine Systems Research

School of Industrial and Systems Engineering

Georgia Institute of Technology *Atlanta, Georgia 30332-0205*

Christine M. Mitchell, mitchell@chmsr.gatech.edu, (404) 894 4321

Trudy - Sent to sponsor! C

June 30, 1989

Dr. Stan Belyeu
IBM Corp.
Internal Zip 5229
1000 NW 51st Street
Boca Raton, FL 33429-1328

Dear Dr. ~~Belyeu~~ Stan:

Enclosed is the annual report for this year's activities conducted under the sponsorship of the IBM Department Grant to Georgia Tech's School of Industrial and Systems Engineering. Activities include technical interchanges with IBM staff and sponsorship of a number of M.S. and Ph.D. research efforts in the area of supervisory control and interactive optimization. Please let me know if you would like more detail or copies of related publications for any of these activities. I look forward to continuing our collaboration in the years to come.

Sincerely,

Christine M. Mitchell

cc: W. M. Sangster, Dean, College of Engineering
M. E. Thomas, Director, School of Industrial & Systems Engineering
J. J. Jarvis, Acting Director, School of Industrial & Systems Engineering
S. P. Krosner, IBM
Jerry Woolf, IBM
Robert R. Leavitt, IBM

encl.

1988-1989 Research Activities conducted with the Sponsorship of IBM Department Grant

to Georgia Tech's School of Industrial & Systems Engineering

Georgia Tech-IBM Technical Interchange

This year several members of the IBM staff visited Georgia Tech's research facilities in the Center for Human-Machine Systems Research (School of Industrial & Systems Engineering). Visitors included Dr. Stan Belyeu who participated in a Center manufacturing research group meeting, IBM CIMS representatives, and Dr. Bob Leavitt (Watson Research Center)

In addition, Dr. Mitchell has spoken with several IBM staff members to identify IBM manufacturing sites related to the on-going Center for Human-Machine Systems Research activity.

M.S. and Ph.D. Thesis Research

All of the grant funds this year have been used to support graduate students and research activities related to GT-FMS. This project involved four students, three at the masters level and one doctoral student. One masters student, Dean Hettenbach, completed a thesis concerning interactive scheduling for a GT-FMS cell level control. His system enabled a human operator to 'tune' a heuristic scheduling system based on real-time feedback of current cell status and system goals. Mr. Hettenbach evaluated his system with an experiment in which human subjects in the Computer Integrated Manufacturing Systems (CIMS) graduate program were trained and controlled the GT-FMS scheduling system for 10 to 12 hours. His research gives a great deal of insight into the effectiveness of human supervision over the parameters of real-time, state-based scheduling systems. Dean completed his degree with partial support of the IBM grant (M.S. in Industrial & Systems Engineering with a Computer Integrated Manufacturing Systems Certificate) in March 1989. Dean's thesis was the recipient of the 1989 Industrial and Systems Engineering Outstanding M.S. thesis award.

Charlene Benson is another M.S. student who completed her masters degrees (thesis option) under the sponsorship of the IBM grant. Her thesis examined the design and evaluation of direct manipulation interfaces for monitoring and supervising predominantly automated manufacturing systems. Her interface was compared experimentally to a more conventional operator workstation. Ms. Benson evaluated her system with human subjects enrolled in Georgia Tech's Computer Integrated Manufacturing Systems (CIMS) graduate program; these students were trained and controlled the GT-FMS system with one of the two interfaces for approximately 10 to 12 hours. Charlene's research showed a significant positive effect in overall system performance for subjects using the direct manipulation interface. This research constitutes one of the first rigorous empirical examinations of the effect of advanced human-computer interaction techniques on performance of operators in complex control tasks. Charlene completed her degree (M.S. in Industrial & Systems Engineering with emphasis in human-machine systems) in June 1989.

A third M.S. student, Joe Krebbs, with interests in the application of optimization to manufacturing scheduling and control began his thesis research this year. Joe's research implements and evaluates the use of integer programming to schedule automated guided vehicles (AGVs) in the context of GT-FMS. The project examines the mechanisms required for implementation and the effectiveness of this type of optimization when applied to a realistic scheduling problem in real time. In addition, Joe has decided to pursue doctoral work in the area of optimization applied to real-time manufacturing scheduling and control; he applied for an IBM fellowship to help sponsor his education.

Finally, Steve Krosner is a full-time IBM employee and a part-time Georgia Tech graduate student. He has completed all the degree requirements for a Ph.D. except his thesis. His thesis uses GT-FMS with a configuration based on electronics assembly data from an IBM manufacturing facility. His research proposes a model-based, hierarchical design for the cell-level supervisory controller; this design will constitute a theory of the type of control, display, and information requirements that effective human supervisory control of an FMS might require. To evaluate the effectiveness of his design, he will run an experiment that compares his proposed design to a conventional operator interface. Mr. Krosner's research will be completed this year.

Next Year

This grant has been approved for another year. Next year's funds will support the completion of Joe Krebb's and Steve Krosner's research as described above. In addition, a new project is underway. This project involves a masters student, Ms. Sally Cohen, a CIMS M.S. student, and two new Ph.D. students. They are defining an object-oriented manufacturing simulator that will support more comprehensive research and evaluation in human decision making, interactive optimization, and artificial intelligence in manufacturing scheduling and control.

Notes

The two masters theses supported by this grant were published as technical reports; copies are contained the attached appendices. The research results will also be presented at international engineering conferences, and submitted for publication in refereed journals.

As background information, it might be helpful for you to know that we work closely with several other programs: 1) CIMS (Computer Integrated Manufacturing Systems) graduate certificate program--most of our masters students also receive CIMS graduate certificates; 2) Material Handling Research Center (MHRC)--there are many overlapping interests with students and faculty affiliated with MHRC though our programs are separate; and 3) Manufacturing Research Center--this is an Institute wide effort that is just getting started and we hope to work closely with it as its research agenda takes form. IBM is actively involved in all three of these Georgia Tech efforts.

Our research is conducted at a much smaller scale and focused in a specialized area. We are interested in human-computer interaction in the control of predominantly automated manufacturing processes. This area of research is unique at Georgia Tech and separate from the various other manufacturing research and educational entities. Our focus is system design that specifically addresses the advantages and problems of human operators responsible for the productivity and safety of real-time manufacturing processes. Our research is both theoretical and empirical; we almost always collect human performance data in order to evaluate experimental system designs.

IBM's grant has greatly advanced our research. Without your support, our project would have involved fewer people and produced fewer research results. The grant also facilitated numerous technical interchanges with IBM personnel. As you may recall, I had an opportunity to visit you and your colleagues at Boca Raton several times as well as the IBM facilities in Lexington and Atlanta to share our research goals and results.

Thesis Research Supported by IBM Department Grant

Center for Human-Machine Systems Research
School of Industrial & Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332

June 1989

Appenidix A

Hettenback, Dean, "An Investigation of Decision Making in Supervisory Control of a Flexible Manufacturing System," M.S. thesis, Center for Human-Machine Systems Research, School of Industrial and System Engineering, Georgia Institute of Technology, chmsr 89-2, March 1989.

Appenidix B

Benson, Charlene, "The Use of Single-Page, Direct Manipulation Interfaces in Real Time Supervisory Control Systems," M.S. thesis, Center for Human-Machine Systems Research, School of Industrial and System Engineering, Georgia Institute of Technology, chmsr 89-3, June, 1989.

E-24-524

Center for Human-Machine Systems Research

School of Industrial and Systems Engineering

Georgia Institute of Technology *Atlanta, Georgia 30332-0205*

Christine M. Mitchell, cm@chmsr.gatech.edu, (404) 894 4321, fax (404) 894 2301

July 13, 1991

Dr. Stan Belyeu
IBM Corp.
Internal Zip 5229
1000 NW 51st Street
Boca Raton, FL 33429-1328

Dear Dr. Belyeu:

Enclosed is the annual report for this year's activities conducted under the sponsorship of the IBM Department Grant to Georgia Tech's School of Industrial and Systems Engineering. At my request, IBM granted us a no-cost extension. Thus this report summarizes the activities carried out with IBM's support from June 1990 to June 1991. Activities include technical interchanges with IBM staff and sponsorship of a number of M.S. and Ph.D. research efforts in the area of human supervisory control in manufacturing systems. Please let me know if you would like more detail or copies of related publications for these activities.

Sincerely,

Christine M. Mitchell

cc: John A. White, Dean, College of Engineering
J. J. Jarvis, Director, School of Industrial & Systems Engineering

encl.

July 1989-June 1991 Research Activities conducted with the Sponsorship

of the

IBM Department Grant

to Georgia Tech's School of Industrial & Systems Engineering

Georgia Tech-IBM Technical Interchange

This year several members of the IBM staff visited Georgia Tech's research facilities in the Center for Human-Machine Systems Research (School of Industrial & Systems Engineering). Visitors included Dr. Stan Belyeu who participated in a Center manufacturing research group meeting, IBM CIMS representatives, and Dr. Bob Leavitt (Watson Research Center)

M.S. and Ph.D. Thesis Research

All of the grant funds this year have been used to support graduate students and research activities related to GT-FMS. This year's research involved two students, one at the masters level and one at the doctoral level.

During this year, the masters student, Ms. Sally Cohen, completed her master's thesis in the area of modeling expert troubleshooting for circuit board assemblies (PCB). Her research involved field study at a PCB assembly plant. She developed a model of expert troubleshooters. The model served as a basis of an interactive PCB troubleshooting computer program--CIMTEM (computer-based interactive model of troubleshooting in electronics manufacturing) . She validated her model in two ways. First, she compared model-generated troubleshooting actions and strategies to those of an expert; the CIMTEM was approximately 80 to 90% successful in matching the experts' activities. Second, bringing her system into the plant, on-line PCB troubleshooting activities were again compared to her model's output resulting in approximately 90% agreement. Finally, plant operations personnel not trained in troubleshooting, used CIMTEM as a on-line tutor. The development of accurate on-line tutors would be a major milestone in electronics assembly as the expected turnover is high and there is a desire to have all personnel cross trained. Ms. Cohen's thesis will be published as a technical report. In addition, CIMTEM's structure and initial verification were the subject of a paper presented at an international meeting on Human Factors in Design for Manufacturability and Process Planning (sponsored by the International Ergonomics Association). The paper was included in the conference proceedings; and an extended version of the conference proceeding paper will appear in text published by Taylor and Francis. We also anticipate the journal publication describing CIMTEM's validation and application to intelligent tutoring.

Steve Krosner is a full-time IBM employee and a part-time Georgia Tech graduate student. He has completed all the degree requirements for a Ph.D. except his thesis. His thesis uses GT-FMS with a configuration based on electronics assembly data from an IBM manufacturing facility. His research proposes a model-based, hierarchical design for the cell-level supervisory controller; this design will constitute a theory of the type of control, display, and information requirements that effective human supervisory control of an FMS might require. To evaluate the effectiveness of his design, he ran an experiment that compares his proposed design to a conventional operator interface. Mr. Krosner's research will be completed this year.

Notes

The GT-FMS research was summarized in a recently prepared chapter that will appear in the Academic Press Volume "Advances in Manufacturing and Automation Systems" edited by Professor C. T. Leondes, the Boeing Professor of Aerospace Controls and Professor of Electrical Engineering at the University of Washington, Seattle, WA. A copy of our chapter, entitled Human Supervisory Control of Predominantly Automated Manufacturing Processes: Conceptual Issues and Empirical Investigations, is included in the appendix. We think it provides a nice summary of the Georgia Tech research to date on human operators in highly automated manufacturing systems.

As background information, it might be helpful for you to know that we work closely with several other programs: 1) CIMS (Computer Integrated Manufacturing Systems) graduate certificate program--most of our masters students also receive CIMS graduate certificates; 2) Material Handling Research Center (MHRC)--there are many overlapping interests with students and faculty affiliated with MHRC though our programs are separate; and 3) Manufacturing Research Center--this is an Institute wide effort that is just getting started and we hope to work closely with it as its research agenda takes form. IBM is actively involved in all three of these Georgia Tech efforts.

Our research is conducted at a much smaller scale and focused in a specialized area. We are interested in human-computer interaction in the control of predominantly automated manufacturing processes. This area of research is unique at Georgia Tech and separate from the various other manufacturing research and educational entities. Our focus is system design that specifically addresses the advantages and problems of human operators responsible for the productivity and safety of real-time manufacturing processes. Our research is both theoretical and empirical; we almost always collect human performance data in order to evaluate experimental system designs.

IBM's grant has greatly advanced our research. Without your support, our project would have involved fewer people and produced fewer research results. The grant also facilitated numerous technical interchanges with IBM personnel.

Appendix

Mitchell, C. M., Govindaraj, T., Armstrong, J. E., Benson, C. R. and Hettenbach, D. (1991). Human Supervisory Control of Predominantly Automated Manufacturing Processes: Conceptual Issues and Empirical Investigations, Professor C. T. Leondes (Ed.). *Advances in Manufacturing and Automation Systems.* in press.

E 24-524

Center for Human-Machine Systems Research

School of Industrial and Systems Engineering

Georgia Institute of Technology *Atlanta, Georgia 30332-0205*

Christine M. Mitchell, cm@chmsr.gatech.edu, Phone: (404) 894 4321 Fax: (404) 894 2301

July 30, 1993

Dr. Stan Belyeu
IBM Corp.
Internal Zip 5229
1000 NW 51st Street
Boca Raton, FL 33429-1328

Dear Dr. Belyeu:

Enclosed is the final report for the activities conducted under the sponsorship of the IBM Department Grant to Georgia Tech's School of Industrial and Systems Engineering. Activities include technical interchanges with IBM staff and sponsorship of a number of M.S. and Ph.D. research efforts in the area of human supervisory control in manufacturing systems.

IBM's grant has greatly advanced our research. The research in human supervisory control of predominiantly automated manufacturing systems has been supported by groups both inside and outside Georgia Tech. Groups include the National Science Foundation, Georgia Tech's Material Handling Research Center, and Geogia Tech's Computer Integrated Manufacturing Systems program. Without IBM's support, our project would have involved fewer people and produced fewer research results.

Please let me know if you would like more detail or copies of related publications for these activities.

Sincerely,

Christine M. Mitchell

cc: John A. White, Dean, College of Engineering
J. J. Jarvis, Director, School of Industrial & Systems Engineering

encl.

Appendix

Copies of Publications Produced with Support of this Grant

July 1989-June 1991 Research Activities conducted with the Sponsorship

of the

IBM Department Grant

to Georgia Tech's School of Industrial & Systems Engineering

Final Report

July 1993

Georgia Tech-IBM Technical Interchange

Over the years several members of the IBM staff visited Georgia Tech's research facilities in the Center for Human-Machine Systems Research (School of Industrial & Systems Engineering). Visitors included Dr. Stan Belyeu who participated in a Center manufacturing research group meeting, IBM CIMS representatives, and Dr. Bob Leavitt (Watson Research Center). Our group also visited, presented research summaries, various IBM facilities, including Watson Research Center and the Boca manufacturing facility.

M.S. and Ph.D. Thesis Research

The grant funds have been used to support faculty, graduate students and research activities. Over the lifetime of the grant, it supported three M.S. (with thesis) students and two Ph.D. students.

Charlene Benson is a M.S. student who completed her master's thesis in the area of the design of training systems and direct manipulation interfaces to support operators in the control of predominantly automated manufacturing systems. Her research examined the effectiveness of direct manipulation interface technology for the operator supervising an integrated flexible manufacturing system. Her experimental environment was GT-FMS (the Georgia Tech Flexible Manufacturing System), a real-time interactive manufacutring system simulator reconfigurable to model a range of system configurations and controls. Charlene's configuration was based on an IBM electronics assembly facility. Her experiment evaluated both time-to-learn and expert control performance on two types of interfaces, direct manipulation and conventional command line. Her experimenal results showed that direct manipulation interaction facilitated operator learning and understanding of system operation. For trained operators the interface difference disappeared, demonstrating that trained operators almost always carry out their responsibilities quite effectively regardless of the conditions under which they work.

Dean Hettenback is an M.S. student (also receiving a CIMS (Computer Integrated Manufacturing Systems)certificate) who completed his master's thesis in the area of modeling operator decision making at the managerial level of flexbible manufacturing system control. Also using the GT-FMS simulator, Dean configured his system to represent a machining system under development by MTU. Rather than examine moment-to-moment, real-time operator control, Dean moved up a level in the supervisory control hierarchy examining decision making by a system supervisor who coordinated *orders* rather than individual *parts* through the manufacturing process. System control allowed the manager to refine a scheduling algorithm, that in turn handled lower level part movement. System performance was measured by order completion timeliness and minimization of costs due to lateness. The experimental evaluation showed that managers engaged in very analytic, methodical decision making. Verbal protocols were obtained from subjects during the experiment and modeled using Rasmussen's decision ladder framework. This research provides vital foundational material for understanding and aiding decision makers controlling real-time manufacturing systems.

Sally Cohen is another M.S. student (also receiving a CIMS (Computer Integrated Manufacturing Systems) certificate) who completed her master's thesis in the area of modeling expert troubleshooting for circuit board assemblies (PCB). Her research involved field study at a PCB assembly plant. She developed a model of expert troubleshooters. The model served as a basis of an interactive PCB troubleshooting computer program--CIMTEM (computer-based interactive model of troubleshooting in electronics manufacturing) . She validated her model in two ways. First, she compared model-generated troubleshooting actions and strategies to those of an expert; the CIMTEM was approximately 80 to 90% successful in matching the experts' activities. Second, bringing her system into the plant, on-line PCB troubleshooting activities were again compared to her model's output resulting in approximately 90% agreement. Finally, plant operations personnel not trained in troubleshooting, used CIMTEM as a on-line tutor. The development of accurate on-line tutors is an important contribution in electronics assembly as the expected turnover is high and there is a desire to have all personnel cross trained.

Steve Krosner is a retired IBM employee who completed his Ph.D. on a part-time at Georgia Tech. His thesis uses GT-FMS with a configuration based on electronics assembly data from an IBM manufacturing facility. His research proposes a model-based, hierarchical design for the cell-level supervisory controller; this design constitutes a theory of the type of control, display, and information requirements that effective human supervisory control of an FMS might require. To evaluate the effectiveness of his design, he ran an experiment that compared his proposed design to a conventional operator interface.

Major James E. Armstrong completed his Ph.D. in the area of group decision making. Using GT-FMS configured as a multi-cell flexible manufacturing facility, his research examines multi-operator decision making based on two different organizational structures: a hierarchical team with a supervisor and two cell controllers, and a heterarchical team of three operators who share supervisory responsibility for three cells. His experimental investigation identified strengths and weaknesses of each organizational structure and his subseqent models support design of the semantics of intelligent, context-sensitive operator displays and aids.

Theses

Armstrong, J. E. (1990). Distributed decision making in command-and-control of complex dynamic systems. Ph.D. thesis, Center for Human-Machine Systems Research,

Appendix

Copies of Publications Produced with Support of this Grant

School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Krosner, S. P. (1992). Using an extension of Rasmussen's abstraction hierarchy as a framework for design of a supervisory control system of a coomplex dynamic system. Ph.D. thesis, Center for Human-Machine Systems Research, School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Hettenbach, D. (1989). An investigation of decision making in supervisory control of a flexible manufacturing system, Technical Report CHMSR 89-2, M.S. thesis, Center for Human-Machine Systems Research, School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Benson, C. (1989).The use of single-page, direct manipulation interfaces in real time supervisory control systems. Technical Report CHMSR 89-3, M.S. thesis, Center for Human-Machine Systems Research, School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Cohen, S. (1990). A model of troubleshooting in electronics assembly manufacturing. M.S. thesis, Center for Human-Machine Systems Research, School of Industrial & Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Publications

Mitchell, C. M. Supervisory control: Human information processing in manufacturing systems. A. P. Sage (editor), *Concise Encyclopedia of Information Processing in Systems and Organizations*, Pergamon Press, Great Britain, 439-448, 1990.

Mitchell, C. M., Govindaraj, T., Armstrong, J.E., Benson, C. R., and Hettenbach. Human supervisory control of predominantly automated manufacturing processes: Conceptual issuses and empirical investigations. C. T. Leondes (ed.) *Control and Dynamic Systems, Volume 46: Manufacturing and Automation Systems: Techniques and Technologies (Part 2 of 5),* Academic Press, Inc., San Diego, CA., 255-306, 1991.

Cohen, S. M., Govindaraj, T., Mitchell, C.M. Analysis and aiding the human operator in electronics assembly. M. Helander and M. Nagamachi (Eds.) *Design for Manufacturability : A systems approach to concurrent engineering and ergonomics*, Taylor & Francis Ltd., London, 361-376, 1992.

Hettenbach, D. A., Mitchell, C. M., and Govindaraj, T. Decision making in supervisory control of a flexible manufacturing system. *Information and Decision Technologies*, Vol. 17, 1991, 255-278.

Benson, C., Govindaraj, T., Mitchell, C. M. and Krosner, S.P. Effectiveness of direct manipulation interaction in the supervisory control of flexible manufacturing systems. *Information and Decision Technologies*, Volume 18, No. 1, 1992, 33-53.

Notes

As background information, it might be helpful for you to know that we work closely with several other programs: 1) CIMS (Computer Integrated Manufacturing Systems) graduate

Appendix

Copies of Publications Produced with Support of this Grant

certificate program--most of our masters students also receive CIMS graduate certificates; 2) Material Handling Research Center (MHRC)--there are many overlapping interests with students and faculty affiliated with MHRC though our programs are separate; and 3) Manufacturing Research Center--this is an Institute wide effort that is just getting started and we hope to work closely with it as its research agenda takes form. IBM is actively involved in all three of these Georgia Tech efforts.

Our research is conducted at a much smaller scale and focused in a specialized area. We are interested in human-computer interaction in the control of predominantly automated manufacturing processes. This area of research is unique at Georgia Tech and separate from the various other manufacturing research and educational entities. Our focus is system design that specifically addresses the advantages and problems of human operators responsible for the productivity and safety of real-time manufacturing processes. Our research is both theoretical and empirical; we almost always collect human performance data in order to evaluate experimental system designs.

IBM's grant has greatly advanced our research. Without your support, our project would have involved fewer people and produced fewer research results. The grant also facilitated numerous technical interchanges with IBM personnel.

Appendix

Copies of Publications Produced with Support of this Grant

1990 Pergamon Press UK

Gardner H 1985 *The Mind's New Science*. Basic Books, New York
Lehman M M, Stenning V, Turski W M 1984 Another look at software design methodology. *ACM SIGSOFT Software Eng. Notes* 9(2), 38–53
Newell A, Card S K 1985 The prospect for psychological science in human–computer interaction. *Hum. Comput. Interact.* 1, 209–42
Newell A, Simon H A 1972 *Human Problem Solving*. Prentice-Hall, Englewood Cliffs, New Jersey
Norman D A, Draper S W (eds) 1986 *User Centered System Design*. Erlbaum, Hillsdale, New Jersey
Pressman R S 1987 *Software Engineering, A Practitioner's Approach*, 2nd edn. McGraw-Hill, New York
Shneiderman B 1987 *Designing the User Interface: Strategies for Effective Human–Computer Interaction*. Addison-Wesley, Reading, Massachusetts
Springer S P, Deutch G 1981 *Left Brain, Right Brain*. Freeman, San Francisco
Vick C R, Ramamoorthy C V (eds.) 1984 *Handbook of Software Engineering*. Van Nostrand Reinhold, New York

B. I. Blum
[Johns Hopkins University, Laurel, Maryland, USA]

Supervisory Control: Philosophical Considerations in Manufacturing Systems

Most research into manufacturing automation in general, and flexible manufacturing system (FMS) scheduling and control in particular, focuses on the derivation of fully automated control and scheduling techniques; for example, optimal or heuristic analytic models or knowedge-based systems. An alternative and more realistic paradigm to "lights out" automation is presented in this article. The alternative paradigm—supervisory control of manufacturing processes—entails the design of control and scheduling systems that explicitly integrate human decision makers with the underlying automation. Supervisory control is a design philosophy that explicitly addresses the roles and functions of both human and automatic components of the control process. Supervisory control systems make use of capabilities and compensate for the limitations of both human decision makers and automatic components. More specifically, supervisory control designs the human–computer interaction in order to augment and extend the human's role and decision-making effectiveness. Neither the goal nor the unintended side-effects of supervisory control are to automate the human decision maker out of the system or to reduce the human's role to a set of undesirable or ineffective tasks.

This article describes some of the limitations of automated control systems in manufacturing, in particular why full automation is not possible. It also reviews some of the limitations in the typical use of emerging computer technology to provide decision support to the human decision maker. With this discussion as background, research in supervisory control of flexible manufacturing systems conducted with GT-FMS (Georgia Tech-flexible manufacturing system) is summarized. GT-FMS is a real-time, interactive simulator that can be configured to represent actual or planned multicell and multiworkstation FMS installations. GT-FMS research includes the design and evaluation of an operator function model for FMS cell-level supervisory control; design and evaluation of an "intelligent" operator workstation; and the evaluation of hierarchical versus heterarchical managerial structures to coordinate multiperson, multicell FMSs.

1. Background

The debate on US competitiveness and productivity has focused attention on manufacturing and manufacturing innovation (Scott and Lodge 1985, Jaikumar 1986, Krugman and Hatsopoulos 1987, Cohen and Zysman 1988). One interesting conclusion is that the difficulty in manufacturing arises from deficiencies not so much in machines and technology, but ". . . in organizations and the use of people in production" (Cohen and Zysman 1988 p. 1111). This article addresses one facet of the issue: the role of people in the control of increasingly automated manufacturing environments. It provides the background for understanding the choices in automated scheduling and control of a flexible manufacturing system (i.e., analytic versus knowledge-based), and offers an alternative view that proposes the use of experienced human operators to interact with the scheduling and control system and to fine-tune it as needed. The latter view, called supervisory control, explicitly addresses the utilization of people in the manufacturing process and identifies the human decision maker as a critical component in the planning and control process. Although supervisory control does not require additional or different machines or technologies, it does require the rethinking of the role of people in manufacturing systems. An understanding of the philosophy and meaning of supervisory control permits the utilization of expensive and valuable human resources and allows the definition of operator functions that complement existing automated functions. The definition and well-defined engineering specification of the human functions in system control provide a necessary context for the related information-processing issues, including types and mechanisms for decision support, design of operator workstations, and human factors and ergonomics of display screens and operator interaction. Although this article examines supervisory control issues in the context of scheduling and control of flexible manufacturing systems, many of the ideas and some of the research results have applicability to more general manufacturing control processes.

2. *Limitations of "Full" Automation in Manufacturing Control*

2.1 The "Lights Out Factory"

Although one oft-expressed intention of factory automation is the drastic reduction or total elimination of the human workforce on the shop floor, (e.g., the "lights out factory"), it is much more likely that increased implementation of automation will lead to changes in the numbers and skills of workers on the shop floor, rather than the elimination of people (Jaikumar 1986, Rasmussen 1986). Thus, the factory of the future will include human decision makers on the shop floor, but the roles and scopes of responsibilities of these individuals are likely to change drastically as the implementation of automation progresses (Young and Rossi 1988).

The reason why human decision makers must remain an integral part of the system is not hard to discover. Automation technologies often result in considerable system down time. Shaiken (1985a,b) explains this phenomenon concisely (Shaiken 1985a p. 18):

> Reducing human input often means instituting complex technologies that are prone to trouble. The drive to eliminate uncertainties arising from human influence only winds up creating mechanical and electronic uncertainties. Thus, despite the vision of total automation, workers must in the end play critical roles in operating as well as unjamming and repairing, computer-based production systems.

The necessity of integrating human decision makers into the manufacturing process is particularly important in process control. The size, costs and risks associated with malfunctions in control systems make reliable control a necessary condition for successful operation (Chambers and Nagel 1985, Rasmussen 1986). The complexity of the system and the resulting inability of software to cope with all possible future events imply that human decision makers provide an essential backup for the computer-based control system (e.g., Young and Rossi 1988).

It is unlikely that the limitations of full automation will be corrected in the near future. For example, scheduling and control systems based on analytic models of the process contain inherent limitations. The academic community involved in manufacturing, material handling and scheduling research has repeatedly found that sophisticated mathematical models of production and control require unrealistic assumptions about the manufacturing process and its parameters. Examples of such assumptions include deterministic processing and routing times, or workers who are assumed to perform at the same speed and possess the same skill levels (e.g., Johnson 1988). When implemented in actual systems, models based on assumptions that are not met in the application may fail to provide the mathematical optimality promised by the basic research. Such prominent researchers as Buzacott and Yao (1986) predict that mathematical models will never reflect the range of behaviors and uncertainties of real systems, and that the most that can be expected is that an analytical model can address a small, but hopefully important, subset of issues. Further clouding the prospects of analytic research in scheduling and control is concern that, because the real problems in manufacturing automation are so complex, academic research will address pseudoreal problems and pay only lip-service to real manufacturing applications (Ho 1987).

The use of artificial intelligence (AI) techniques in manufacturing automation has been proposed to remediate the gaps left by analytical models (e.g., Bourne and Fox 1984, Fox and Smith 1984, Miller 1985, Smith *et al.* 1986). In a case study of a turbine engine job shop, it was found that 80–90% of a scheduler's time is spent identifying constraints not typically reflected in analytic models. ISIS, an AI system using constraint-based reasoning to find satisfactory, as opposed to optimal, schedules, was developed as an attempt to cope with the range of analytic and informal constraints found in actual systems (Smith *et al.* 1986). Yet experimental AI systems have not provided the flexibility and adaptability initially expected (e.g., Smith *et al.* 1986, Young and Rossi 1988). Human decision makers remain an integral part of such systems (Wright and Bourne 1988, Young and Rossi 1988). The general consensus is that it will be a long time, if ever, before systems based on AI techniques can perform better than trained operators in unanticipated or novel situations (Chambers and Nagel 1985, Rasmussen 1986).

Most manufacturing control research acknowledges both the limitations of the predominant tools for automated control and the inevitable and intrinsic role of humans in the manufacturing process (Young and Rossi 1988, Cohen and Zysman 1988). Either implicitly or explicitly, an autonomous manufacturing system utilizing either an analytical model (e.g., Jones and Maxwell 1986, Jaikumar 1986) or AI techniques (e.g., Fox and Smith 1984, Miller 1985, Astrom 1985, Astrom *et al.* 1986, Smith *et al.* 1986) assumes the presence of human operators who monitor the automation and correct and fine-tune the process when necessary. While acknowledging the presence of human decision makers, few researchers in operations research or AI attempt to address explicitly the engineering and design of manufacturing control systems that integrate automation with the humans who are responsible for overseeing the effectiveness of system operation. The study of human–machine interaction in complex dynamic systems, a related area of engineering, addresses this issue directly.

2.2 Supervisory Control Systems

Systems in which humans primarily monitor automated control processes are called supervisory control systems (Sheridan and Johansen 1976). The role of humans in supervisory control systems is to compensate for the limitations of the automation and to provide flexible response in novel situations. Large scale systems whose control depends upon both autonomous and human subsystems are not unusual in the broader context of complex high-risk military, space and industrial systems (Rasmussen and Goodstein 1987). For example, it has long been acknowledged in the design of space system control, both on the ground and in space, that the human in such systems provides a necessary and integral part of successful system operation (Cohen and Erickson 1985). There is little reason to believe that manufacturing systems will be different. To the contrary, a number of researchers point out that there is one distinction between US and Japanese factories in the type of labor force. Although Japanese factories have fewer people, their skill levels and scopes of responsibility are often broader than those of their US counterparts (Shaiken 1985b, Jaikumar 1986, Cohen and Zysmann 1988).

It is insufficient, however, merely to make a commitment to a skilled workforce; effective systems and good engineering design require precise specification of the role of the human decision maker in automated manufacturing systems and integration of the human component into the overall system specification. Experience and research from existing supervisory control systems may provide some direction in manufacturing systems. Thus one objective of this article is to illustrate the principles of supervisory control in the context of manufacturing systems, specifically the control and scheduling of flexible manufacturing systems.

2.3 *Supervisory Control Paradigm for FMS Control*

As applied to manufacturing control, particularly to control and scheduling of a flexible manufacturing cell, supervisory control is proposed as a conceptual framework for organizing the design of the FMS (Ammons *et al.* 1988). An FMS is a network of versatile workstations connected by a flexible material handling system. The FMS workstations are capable of performing many different operations of an associated process; for example, machining, assembly or fabrication. There is minimum changeover time between operations, and the material handling system is capable of executing any desired job routing (Ammons *et al.* 1985). FMS is a philosophy of automation rather than a specific type of system design, and as such it is a good vehicle to illustrate the philosophy and concepts of supervisory control.

The supervisory control paradigm for FMS proposes a control system design that successfully integrates the resources of analytical models, AI and human supervisory controllers (Ammons *et al.* 1988). The integration utilizes the capabilities and compensates for the limitations of each component. Analytic models form the foundation of the automatic scheduling and control system. Given this level of background automation, knowledge-based systems are designed and implemented to compensate for the known limitations of the mathematical models (e.g., unrealistic model assumptions). Finally, an operator interface to the control system provides the human decison maker with information and controls with which to monitor and fine-tune the system in response to unanticipated or changing system conditions. This philosophy is depicted in Fig. 1.

There are three basic tenets of the supervisory control paradigm. The first is that FMS control systems should be designed and engineered with an explicit understanding of the position and role of the human operator responsible for the system. The second tenet is a corollary to the first: the design process should represent the human functions with as much precision and detail as the specification of system software and hardware. This representation requires the development of a detailed, dynamic model of operator functions, extending over the range of possible system states (Mitchell and Miller 1986, Mitchell 1987). Finally, given a model of operator functions, the supervisory control paradigm requires integration of the automatic parts of the control system into an integrated workstation through which the human supervisor can monitor the process, tune the parameters and compensate effectively for the deficiencies of the control automation.

The last point is important. It requires the designers of FMS control systems to design explicitly the human functions into the system and focuses the design process on enabling the system supervisor responsible for safe and effective operation to control the system effectively. A control system is supervisory only if the human supervisor has the information, decision tools and controls necessary to ensure effective and safe system operation when the limits of automated control are reached. An ineffective human operator, (e.g., someone who is bored, someone who has been given tasks that are not compatible with human capabilities, or someone who lacks the proper decision support information or tools), destroys the effectiveness of the

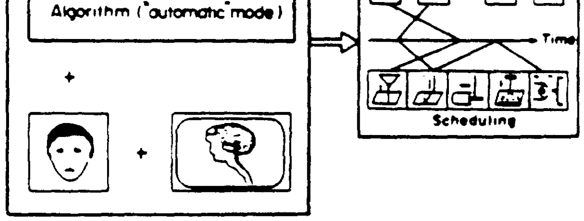

Figure 1
The supervisory control model

supervisory control design and reduces the system architecture to one of full automation plus a peripheral human who may occasionally interfere with the process.

The FMS supervisory control paradigm is a radical departure from the current emphasis in manufacturing automation design. Frequently, the human role in automated systems is defined as an afterthought. The human is included in the control process to compensate for the times or events that the automation handles inadequately or not at all. From an engineering perspective, the human's role is not designed, it is ad hoc, and often evolves over time as inadequacies arise in the automatic control system.

Similarly, human interfaces to such systems are also not designed. Typically, information displays provided to real-time decision makers are "data dumps" where a programmer unfamiliar with the domain or the operator's tasks designs information displays that show all data collected in the system at the level at which the data are collected—frequently the lowest level possible (Rasmussen 1986, Mitchell and Saisi 1987, Rasmussen and Goodstein 1987). It is the responsibility of the human supervisor to sift through the available data, aggregating and integrating as necessary. Likewise, user controls are often awkward; they are typically concatenations of low-level commands that sometimes leave the human supervisor in the position of tricking the system into performing the necessary functions.

Control system design explicitly incorporating the functions of the human supervisor requires not only an intention but also a rigorous specification of how humans are to be utilized in the system. Tools and techniques for effective supervisory control design are not widely available. Designers are often faced with a situation in which there is more user-interface technology than design knowledge about how to use the technology. Moreover, conventional wisdom and intuition do not necessarily result in useful applications. Section 3 reviews some of the problems associated with the design of operator workstations for complex systems, particularly the problems caused by the increasingly available automated decision aids. Given this background, GT-FMS (Georgia Tech-flexible manufacturing system) is described, as are several research programs using GT-FMS. The GT-FMS research program includes the design, implementation and empirical evaluation of supervisory control systems for FMS cell-level scheduling and control.

3. Use of Computers in Decision Support: Decision Making or Decision Aiding?

Advances in computer technology and AI provide new computational tools that greatly expand the potential to support decision making in the supervisory control of complex work environments (Woods 1986a). The most frequent use of this technology, however, is often inconsistent with human skills. ". . . the primary design focus is to use computational technology to produce a stand-alone machine expert that offers some form of problem solution . . . [Thus], the interface design process focuses on features to help the user *accept* the machine solution" (Woods 1986b p. 87). Woods notes that the primary issue in such systems is user acceptance of the proposed solution and that system designers will go so far as to suggest that the system should provide the user with placebo-like interaction (e.g., allow the user to report facts considered important, even though they are not used by the system) in order to facilitate user acceptance of the machine's recommendations.

Woods (1986b) identifies three problems with such systems. First, when the machine gives only its solution to a problem, the decision maker may not have the authority to override machine output in practice as well as in theory. Since the only practical options are to accept or reject system output, there is a great danger of what Woods calls the responsibility/ authority double-bind in which the user always either rejects or accepts the machine solution. The former discards the enhancements that intelligent decision support may add to overall system effectiveness; the latter abrogates the responsibility and purpose of the human decision maker in the system. The second problem is that it is not clear whether people are skilled at discriminating correct from incorrect machine solutions. The effectiveness of human decision makers in system control may depend on intimate involvement in the decision process rather than simply on evaluation of the decision product. Research in other supervisory control domains shows that there is an optimum level of control system automation beyond which a human cannot effectively make the transition from the role of a relatively passive monitor to that of an active system controller (Bergeron 1981). Woods identifies the potential loss of cognitive skill as the third problem. Humans are retained in systems to compensate for the limitations of automation. A user who depends almost exclusively on the recommendations of the machine expert may be ill-prepared for the occasions when the machine expert fails and his/her skill is essential to safe and effective system operation.

Recent research provides experimental data which demonstrate problems with "decision making" decision support. In a series of experiments at Georgia Tech, advice-giving systems consistently failed to improve overall system performance (Knaeuper and Morris 1984, Zinser 1986, Resnick *et al.* 1987, Zinser and Henneman 1988). The primary reason for the failure of these systems to enhance performance is that system users either did not ask for or did not take the advice. In one instance in which the machine-based system automatically recommended the next operator procedure, a pilot study showed that in order to dispel user animosity, the aid had to be "toned down" (Knaueper and Morris 1984). In the other two studies, although advice was free, subjects rarely asked for it; neither system had an implicit or explicit penalty for requesting advice.

These results raise interesting questions about the efficacy and style of decision support. In all three experiments, the aid explicitly gave advice, provided reminders and generally gave the impression that it was omniscient with regard to the task. Yet the human–computer interaction and related system performance did not suggest that advice-giving enhanced system effectiveness.

There is other research suggesting that decision support may not always be fruitless. Another Georgia Tech experiment used a computer to provide dynamically adapted system-status information. Information content and form was based on a domain-specific model of the human–machine interaction that tailored and grouped displayed information based on the current system state and current operator functions. This resulted in improved system performance across a variety of measures and did not have any user acceptance problems (Mitchell and Saisi 1987).

The differences between these sets of experiments provide insight into the more general issue of aiding. The experiment in which decision support had a positive effect used the computer to aid the user's decision-making process. The model of human–machine interaction was embedded into the workstation and provided system information at various levels of abstraction, with both type and level of abstraction estimated using a model of operator function and information about current system state. The workstation provided an initial view into the controlled system based on a "best guess" about the user's needs. Additional information, however, was always available at the user's request and the decision process always remained under the user's control.

Decision support systems that aid the user in the process of reaching a decision, rather than making or recommending a solution, are proposed as an alternative to the typical decision-aiding paradigm (Woods 1986b, Mitchell and Saisi 1987, Rasmussen and Goodstein 1987, Vicente 1987, Rubin *et al.* 1988). The basic principle that underlies a decision-aiding design is that automation and machine intelligence should enhance or extend human decision-making capabilities, not replace the decision maker (Woods 1986b).

In a recent article on decision support in the supervisory control of high-risk industrial systems, Rasmussen and Goodstein summarize this position succinctly (Rasmussen and Goodstein 1987 p. 663).

> Rather than continuing their efforts to make the preplanning (i.e., automation) of responses and countermeasures more and more complete and thus restrict the operator's own initiative, designers should take advantage of modern information technology to make available to operators their own conceptual model and their processing resources so as to allow the operators to function as their extended arm in coping with the plant. Such an interactive decision-making activity would thus benefit from this simultaneous availability of the design basis, up-to-date knowledge of the plant status, and accumulated operational experience.

Current research programs attempting to develop electronic or computer-based associates explicitly address the design of decision-aiding systems. The pilot's associate project is a research effort that addresses the operational issues of decision support in real-time decision-making environments (Chambers and Nagel 1985). The intent of this program is to produce a support system architecture that enhances human abilities, overcomes human limitations and complements individual human preferences.

A similar effort for a space satellite control-room application, OFMspert (Rubin *et al.* 1988), uses a blackboard architecture to infer operator intentions based on a normative model of operator function. Although OFMspert has been quite successful at inferring operator intentions for a laboratory task (Jones 1988), the next step in the development of an operator's associate—determination of the style and substance of interaction—is very difficult. Given a representation of operator intentions, OFMspert must interact with the user, providing information and/or assistance. The implementation and evaluation of such systems are essential (Bushman 1988).

Human–computer interaction, levels of automation and control of system initiative are unresolved research questions in manufacturing. In many ways manufacturing is a more difficult domain than typical supervisory control systems. In other system, (e.g., airplane cockpits), the system already exists and automation can be incrementally implemented in conjuction with existing pilot functions. In many manufacturing applications, such as FMS scheduling, there is not an "operator's job" to automate. The "factory of the future" and FMS are concepts waiting for system design specification to make them realities.

The Georgia Tech research program is one attempt to explore the essential features of this problem. GT-FMS was built as a domain in which to explore design possibilities for supervisory control of FMS scheduling. GT-FMS is a simulator that can be configured to represent many FMS systems. It is designed to be interactive and to facilitate the exploration of human–machine interaction issues in FMS control such as level of automation, supervisory control architecture and decision support system strategies. Section 4 summarizes the main features of GT-FMS together with recent and ongoing research conducted within the GT-FMS domain.

4. GT-FMS: A Domain for Research in Supervisory Control of FMS Scheduling

GT-FMS is a domain created to examine a range of research issues related to human–computer interaction and decision support in scheduling and control of FMSs. GT-FMS is an interactive, real-time simulator of a potentially multicell, multiworkstation FMS. GT-FMS is a real-time rather than a discrete-event simulation. Time flows proportionally to real time and a human decision maker can interact with GT-FMS in a manner similar to that of a scheduler or expeditor on the shop floor. GT-FMS was designed to provide a workbench or laboratory in which human interaction with FMS scheduling and control can be observed, controlled and empirically evaluated given proposed decision aids and definitions of human functions.

GT-FMS is written in C and runs in the Unix operating system environment. The basic simulator consists of more than 10 000 lines of source code. Increasingly sophisticated operator workstations add to this core system. A single-cell version also runs on a PC AT. The simulator has been configured with data from several real manufacturing systems and with both machining and electronics assembly data. Details about GT-FMS and research performed with it are given in the following subsections.

4.1 Structure of GT-FMS

Although flexible in configuration, GT-FMS makes several assumptions about system configuration and limitations. GT-FMS can have several cells, each with its own WIP and workstations. Workstations are uniquely configurable, each workstation with its own set of manufacturing operations. For example, in GT-FMS it is possible for two or more workstations to do the same task but at different levels of efficiency. Cell WIPs have a finite capacity; default is twenty. There is a flexible material handling system that can carry out any desired routing within and between cells. Workstations have the capacity to hold two parts, one in progress and one in a single item buffer. Parts automatically return to the WIP between visits to workstations. Work cells share a common input buffer. Parts arrive at the input buffer with a due date; part type designates the set and sequence of operations that must be completed before the due date.

Currently, there are three versions of GT-FMS based on actual data. One version uses data supplied by Motoren und Turbinen Union GmbH (MTU), a West German diesel engine manufacturer. The MTU version configures GT-FMS as a one-cell system with four identical machining centers and two load/unload stations. The MTU GT-FMS also includes two batch processes that require parts of one type to accumulate for processes performed outside the FMS cell (Dunkler 1986, Dunkler *et al.* 1988).

Another version of GT-FMS is based on data supplied by Lockheed–Georgia. It too is a machining operation with identical workstations and load/unload positions. This version is being used to examine the effectiveness of weighted operations priority due date scheduling.

The third version of GT-FMS uses IBM electronics assembly data. This version is again one cell; it has eight machines, two single in-line package (SIP) inserters, three dual in-line package (DIP) inserters and three robots whose primary job is to insert modules but which have the capability to insert SIPs and DIPs, although with less efficiency than the dedicated SIP or DIP insertion machines (Krosner *et al.* 1987).

There is also a multicell version of GT-FMS. Although not based on actual data, this version is constructed to examine multicell, multioperator interaction in FMS scheduling and control. The hierarchical version of the multicell GT-FMS consists of two cells (each with an operator) and a supervisor that coordinates the cells to meet overall system goals. The heterarchical version consists of three cells, each cell containing fewer machines than the two-cell system, with a cell operator for each cell and no decision maker who is designated as the supervisor (Armstrong and Mitchell 1986). Empirical research examines the effectiveness of the hierarchical and heterarchical team structures for different levels of system load and communication delays (Armstrong 1988).

Research with GT-FMS is both theoretical and empirical. Several of the completed and ongoing studies are summarized below.

4.2 Operator Function Model for GT-FMS

One of the original piecesof research with GT-FMS was the development of a model of proposed operator functions for FMS cell-level scheduling and control. The model defined two major operator functions. First, the operator monitors item movement within the cell to ensure that parts within the cell are processed in a timely manner; that is, on or before the due date. Furthermore, if an item looks as if it will not finish on time, the operator intervenes to minimize the amount of time by which a part is late. The second operator responsibility is to carefully monitor the relationship between the input buffer and the FMS cell. The operator monitors both current cell and input buffer contents with two goals in mind: cell contents are closely watched to ensure that inventory carrying costs are within reasonable bounds; input buffer contents are monitored to ensure that parts pulled into the cell by the automatic scheduling and control system are those that require immediate processing and whose processes can be performed within the cell given current cell status (e.g., the status of workstations that can perform the required operations).

These two operator functions may be called schedule management and inventory management depicted in Figs. 2 and 3, respectively. (Events in cell schedule management include machine failures, WIP arrivals, due-date changes for parts contained in the

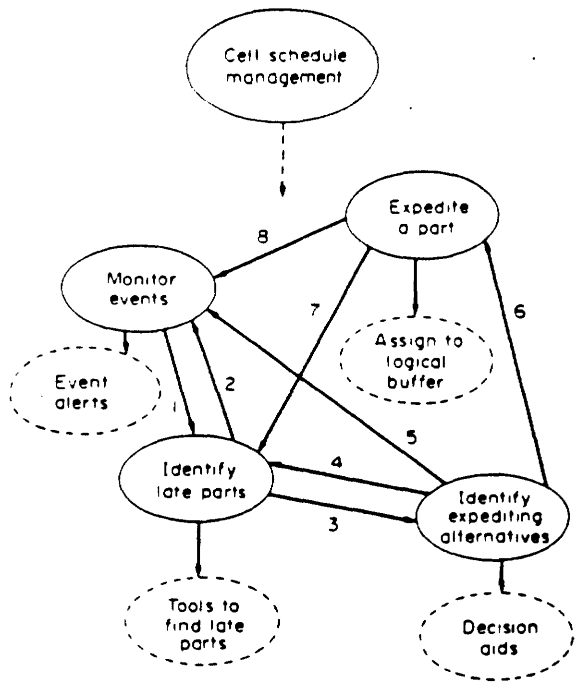

Figure 2
Operator function model for cell schedule management: 1, critical event occurs; 2, no late parts currently contained in cell; 3, one or more late parts found; 4, decision not to expedite late part with more late parts to consider; 5, decision not to expedite late part with no other late parts to consider; 6, decision to expedite a late part; 7, completion of expediting action; 8, end of task

cell, and schedule preemption by operator.) An operator function model was used to describe these functions more fully in the context of a dynamic manufacturing environment (Ammons *et al.* 1988). The plausibility of the model, together with an implementation of a specific set of operator interfaces and controls, was developed and empirically tested with the MTU version of GT-FMS. The experiment is described in the following subsections.

4.3 Supervisory Control of a Flexible Machining Center

As indicated above, this research used the GT-FMS version configured with MTU diesel engine data. Operator scheduling and control commands were based on the proposed model of FMS operator function. Operator commands included "expedite a part," "move a part" and "alter WIP setpoint." The "expedite" command was defined to allow the human operator to carry out both the inventory and cell schedule management subfuhctions. Using the "expedite" command, the human supervisor preempts the automatic scheduling system and logically routes a part to a specified destination. The destination is either one of the six machines if the expedited part is currently in the WIP, or to the WIP if the expedited part is currently in either the arrival buffer or another temporary system buffer. If a part is expedited to a machine, it will be the next part processed, preempting the part currently waiting in the machine's buffer. If a part is expedited from one of the buffers to the WIP, the part is immediately transported to the WIP. The "free" command is available to cancel a pending "expedite" command for a machining center. This notion of expediting as a limited-horizon schedule preemption is one result of the formal modelling process. Expediting may be implemented in many ways; typically it is performed in an ad hoc manner that creates two permanent classes of parts—those that are expedited and those that are not. The latter interpretation of expediting may have adverse impacts on underlying optimization routines. The operator expedite command with a more limited horizon provides operator control in the context of a local problem.

The human supervisor can move a part from a broken machine back to the WIP using the "move" command. This command returns the part to the WIP and places it back within the control of the automatic scheduling system.

The "alter WIP" command allows the operator to alter the WIP setpoint from a default value of fourteen to some other level between zero and twenty. Thus, this command serves as an inventory management command.

The operator workstation consists of a single CRT where system status information can be obtained. The primary means of decision support in this system is a decision aid displaying a rank-ordered list of parts most likely, given current system state, to finish processing late. Called the Rush page, this display page, together with a cell-status page, provides the primary information about the system.

An experimental evaluation showed that the supervisory controller of this FMS cell consistently controlled the FMS cell more effectively than either the "first come first served" or "shortest processing time" dispatch rules operating in a fully automatic manner. Data summarizing these experimental results are given in Fig. 4. Detailed results can be found in Dunkler (1986) and Dunkler *et al.* (1988).

4.4 Workstation Enhancement

The GT-FMS using MTU data was augmented with an operator workstation that uses menu commands and windows to access system data (Tipton 1987). It was thought that a more user-friendly workstation would enhance system performance (Krosner *et al.* 1987, Tipton 1987). Recently collected experimental data showed no improvement, however.

In parallel with the human factors enhancements to the workstation, a more sophisticated, model-based

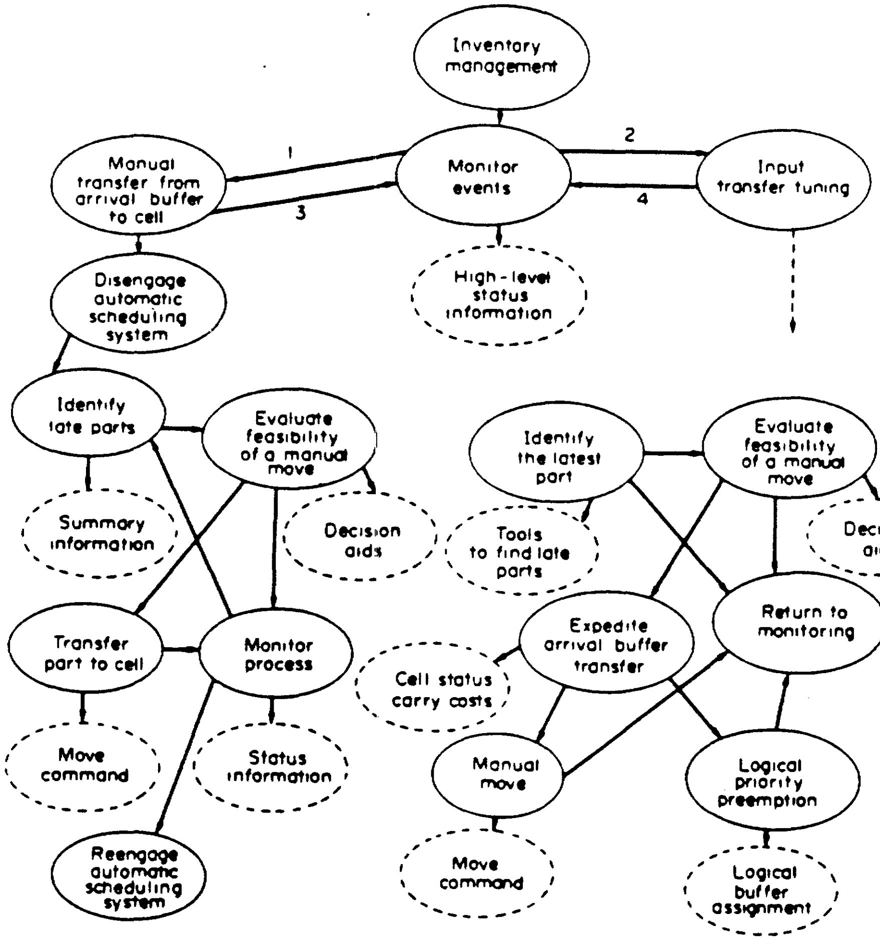

Figure 3
Operator function model for inventory management subfunction: 1, events including workstation failures or a large number of late parts contained in arrival buffer; 2, events including WIP departures, part completions and arrival buffer arrivals; 3, operator reengages automatic scheduling and controlling system and manual transfer subfunction is completed; 4, input transfer tuning is completed when the operator either physically moves, logically prioritizes a part in the arrival buffer or decides that preempting the automated schedule is not feasible

workstation is being designed for the electronics assembly version of GT-FMS. The intent of this project is to develop a model of human decision making that can provide the supervisory controller of the FMS with the correct information, at the appropriate level of abstraction, and in a timely manner. This model uses the operator function model (Mitchell 1987, Ammons *et al.* 1988) to structure information and Rasmussen's abstraction hierarchy (Rasmussen 1986) to guide the semantic representation of the information. When completed, the effectiveness of this workstation will be evaluated empirically.

4.5 Multioperator, Multicell Systems

The multicell, multioperator GT-FMS examines the effectiveness of two organizational structures: a hierarchical structure with two subordinates and a supervisor, and a heterarchical structure with three relatively autonomous cell controllers who coordinate voluntarily to achieve system goals. The multicell GT-FMS was enhanced to include the notion of batches, that is, a collection of parts due out of the system at the same time. Communication and coordination must occur among individual operators in order to meet not only part-due date at the cell level but also batch-due date at the overall system level. Figs. 5 and 6 shows the two organizational structures for this multicell, multioperator GT-FMS configuration.

Experiments are being conducted with the two- and three-cell systems in order to construct models of the command, control and communication processes for the two structures (Armstrong 1988). These models

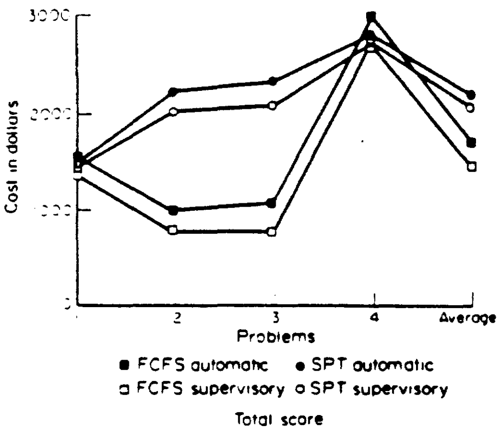

Figure 4
Results of supervisory control experiments

will give some insight into the multioperator decision process. Such models are a necessary prerequisite to understanding team performance and coordination and will provide insight for the design of teams that include both human and computer-based decision makers.

5. Summary

This article proposes supervisory control as an alternative to the goal of full automation in manufacturing processes. With supervisory control, the goal is to design into the control process human override functions that utilize human skills and enhance human effectiveness and overall system performance. A problem, however, in the design of human–machine interaction in complex, highly automated systems is the issue of decision support; in particular, it is important to distinguish between decision making and decision aiding. It is suggested here that decision support in the form of aiding the decision process is much more effective.

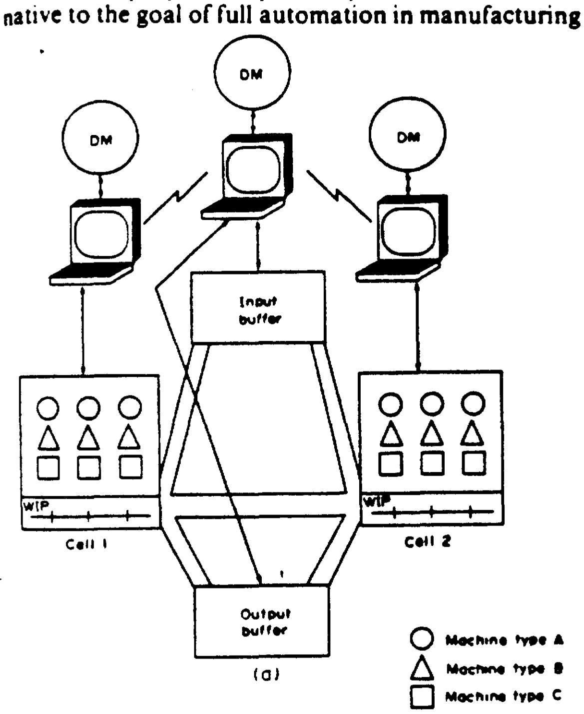

Figure 5
The hierarchical "team." DM, decision maker

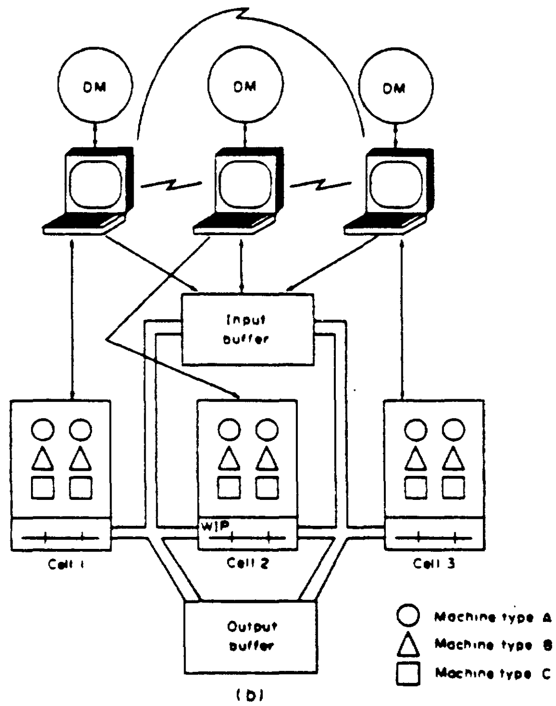

Figure 5
The heterarchical "team." DM, decision maker

Bibliography

Ammons J C, Govindaraj T, Mitchell C M 1988 A supervisory control paradigm for real time control of flexible manufacturing systems. *Ann. OR* **15**, 313–35

Ammons J C, Lofgren C B, McGinnis L F 1985 A large scale loading problem. *Ann. OR* **2**, 319–32

Armstrong J 1988 Evaluation and model of effects of heterarchical versus hierarchical organizational structures in team comand-and-control. Ph.D. Thesis. School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia

Armstrong J M, Mitchell C M 1986 Organizational performance in supervisory control of a flexible manufacturing system. *Proc. 1986 IEEE Int. Conf. Systems Man and Cybernetics*. IEEE Press, New York, pp. 1437–41

Astrom K J 1985 Process control—Past, present and future. *IEEE Control Syst. Mag.* August, 3–10
Astrom K J, Anton J J, Arzen K E 1986 Expert control. *Automatica* **22**(3), 227–86
Bergeron H P 1981 Single pilot IFR autopilot complexity/benefit tradeoff study. *J. Aircraft* **18**(9)
Bourne D A, Fox M S 1984 Autonomous manufacturing: Automating the job-shop. *IEEE Comput.* 17(9), 76–86
Bushman J B 1988 Identification of an operator's associate's model for cooperative supervisory control situations, Ph.D. Thesis. School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia
Buzacott J A, Yao D D 1986 Flexible manufacturing systems: A review of analytic models. *Manage. Sci.* **32**(7), 890–905
Chambers A B, Nagel D C 1985 Pilots of the future: Human or computer. *Commun. ACM* **28**(11), 1187–99
Cohen A, Erickson J D 1985 Future uses of machine intelligence and robotics for the space station and the implication for the U.S. economy. *IEEE J. Robot. Automat.* **1**, 117–23
Cohen S S, Zysman J 1988 Manufacturing innovation and American industrial competitiveness. *Science* **239**, 1110–15
Dunkler O 1986 *Human supervised scheduling and control in flexible manufacturing systems*, M.Sc. Thesis. School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia
Dunkler O, Mitchell C M, Govindaraj T, Ammons J C 1988 The effectiveness of supervisory control strategies in scheduling flexible manufacturing systems. *IEEE Trans. Syst., Man, Cybern.* **18**(2), 323–37
Fox M S, Smith S F 1984 ISIS: A knowledge-based system for factory scheduling. *Expert Syst.* **1**(1), 25–49
Ho Y C 1987 Basic research, manufacturing automation, and putting the cart before the horse. *IEEE Trans. Automatic Control* **32**(12), 1043
Jaikumar R 1986 Postindustrial manufacturing. *Harv. Bus. Rev.* **64**(Nov–Dec), 69–76
Johnson R V 1988 Optimally balancing large assembly lines with 'FABLE'. *Manage. Sci.* **34**(2), 240–53
Jones C V, Maxwell W L 1986 A system for manufacturing scheduling with interactive computer graphics. *IEEE Trans.* **18**(3), 298–303
Jones P M 1988 Constructing and validating a model-based operator's associate for supervisory control, M.Sc. Thesis. Center for Human–Machine Systems Research, CHMSR 88-1, Georgia Institute of Technology, Atlanta, Georgia
Knaeuper A, Morris N M 1984 A model-based approach for online aiding and training in process control. *Proc. IEEE Int. Conf. Cybernetics and Society*. IEEE Press, New York, pp. 173–77
Krosner S P, Mitchell C M, Tipton B C 1987 The role of the human operator in control of an FMS with random operation times. *Proc. IEEE Int. Conf. Systems, Man and Cybernetics*. IEEE Press, New York, pp. 671–74
Krugman P, Hatsopoulos G 1987 The problem of U.S. competitiveness in manufacturing. *N. Engl. Econ. Rev.* January/February, 18
Miller R K 1985 Artificial intelligence: A tool for manufacturing. *Manuf. Eng.* April, 56–62
Mitchell C M 1987 GT-MSOCC: A domain for research on human–computer interaction and decision aiding making in supervisory control systems. *IEEE Trans. Syst., Man Cybern.* 17(4), 553–72
Mitchell C M, Miller R A 1986 A discrete control model of operator function: A methodology for information display design. *IEEE Trans. Syst., Man Cybern.* **16**(3), 343–57
Mitchell C M, Saisi D S 1987 Use of model-based qualitative icons and adaptive windows in workstations for supervisory control. *IEEE Trans. Syst., Man Cybern.* **16**(4), 573–93
Rasmussen J 1986 *Information Processing and Human–Machine Interaction*. North-Holland, New York
Rasmussen J, Goodstein L P 1987 Decision support in supervisory control of high risk industrial systems. *Automatica* **23**(5), 663–67
Resnick D E, Mitchell C M, Govindaraj T 1987 An embedded computer-based training system for rhino robot operators. *IEEE Control Syst. Mag.* 7(3), 3–8
Rubin K S, Jones P J, Mitchell C M 1988 OFMspert: Inference of operator intentions in supervisory control using a blackboard architecture. *IEEE Trans. Syst., Man Cybern.* **18**(4), 618–37
Scott B R, Lodge G (eds.) 1985 *U.S. Competitiveness in the World Economy*. Harvard Business School Press, Boston, Massachusetts
Shaiken H 1985a The automated factory: The view from the shop floor. *Technol. Rev.* **88**(1), 16–24
Shaiken H 1985b *Work Transformed: Automation and Labor in the Computer Age*. Holt Rinehart, New York
Sheridan T B, Johansen G (eds.) 1976 *Monitoring Behavior and Supervisory Control*. Plenum, New York
Smith S F, Fox M S, Ow P S 1986 Constructing and maintaining detailed production plans: Investigations into the development of the knowledge-based factory scheduling system. *AI Mag.* 7(4), 45–61
Tipton B C 1987 *The Importance of Human–Computer Interfaces used in Human Supervisory Control of Flexible Manufacturing Systems*, CIMS Project and Center for Man–Machine Systems Technical Report, Georgia Institute of Technology, Atlanta, Georgia
Vicente K J 1987 *The Role of Information Technology in Emergency Management: Expert System or Cognitive Instrument?* Riso National Laboratory, Roskilde, Denmark
Woods D D 1986a Cognitive technologies: The design of joint human–machine cognitive systems. *AI Mag.* **3**, 86–92
Woods D D 1986b Paradigms for intelligent decision support. In: Hollnagel E, Mancini G, Woods D (eds.) *Intelligent Decision Support in Process Environments*. Springer–Verlag, New York, pp. 153–74
Wright P K, Bourne D A 1988 *Manufacturing Intelligence*. Addison–Wesley, New York
Young R E, Rossi M A 1988 Toward knowledge-based control of flexible manufacturing systems. *IEEE Trans.* **20**(1), 36–43
Zinser K 1986 *Modeling and Aiding Human Decision Making in Operating and Supervising Dynamic Systems*, Center for Man–Machine Systems Technical Report 86-4. School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, Georgia
Zinser K, Henneman R L 1988 Development and evaluation of a model of human performance in a large-scale system. *IEEE Trans. Syst., Man Cybern.* (in press)

C. M. Mitchell
[Georgia Institute of Technology, Atlanta, Georgia, USA]

gathering head, using a rotary motion, scoops the coal inward onto the gathering pan. A conveyor belt is behind the gathering head and moves the coal to the rear of the machine.

- A Conveyor Subsystem: The conveyor extends from the gathering head to the rear of the machine. An adjustable position conveyor boom forms the end of the conveyor system. It can move from right to left as well as up or down. Coal is dumped from the conveyor boom onto a haulage unit behind the CM.
- A Stabilization Jack: This hydraulic jack provides a stabilizing force to counter-balance the cutting force.

The continuous mining machine has ten tram control commands: slow/fast speed forward, slow/fast speed reverse, pivot left/right, turn left/right forward, and turn left/right reverse. These are open-loop commands. Execution of any of these commands can be terminated by either a stop command (implying the tram control loop is closed at a higher level where the sensory information is processed), or by a condition that some maximum time has expired (a safety time-out condition associated with this command).

The U. S. Bureau of Mines has been implementing a computer control system testbed [Sh 90]. This testbed is a distributed network linking the continuous mining machine, various sensor systems (length and angle measuring systems and a gyro, see figure 1), and an operator console which are all nodes on the network. This testbed can generally be referred to as BOM/NET [Sh 90].

Human Supervisory Control of Predominantly Automated Manufacturing Processes: Conceptual Issues and Empirical Investigations

Christine M. Mitchell, T. Govindaraj,
James E. Armstrong, Charlene R. Benson and Dean A. Hettenbach
Center for Human-Machine Systems Research, School of Industrial and Systems Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332-0205, USA

Background

People: The Problem and the Solutions

The debate on American competitiveness and productivity has focused attention on manufacturing and manufacturing innovation (Cohen and Zysman, 1988; Krugman and Hatsopoulos, 1987; Jaikumar, 1986; Scott and Lodge, 1985). One interesting conclusion is that the difficulty in manufacturing is not deficiencies so much in machines and technology, but "...in organizations and the use of people in production (p. 1111, Cohen and Zysman, 1988)." Most research on manufacturing automation focuses on the derivation of fully automated control and scheduling techniques, e.g., optimal or heuristic analytic models or knowledge-based systems. Although one often-expressed intention of factory automation is the drastic reduction or total elimination of the human workforce on the shop floor, (e.g., the 'lights out factory', e.g., Jaikumar, 1986), it is much more likely that increased implementation of automation will lead to changes in the numbers and skills of workers on the shop floor, rather than the elimination of people (Jaikumar, 1986; Rasmussen, 1986). Thus, the factory of the future will include human decision makers on the shop floor, but the roles and scopes of responsibilities of these individuals are likely to change drastically as the implementation of automation progresses (Young and Rossi, 1988).

Manufacturing Research

Most manufacturing control research acknowledges both the limitations of the predominant tools for automated control and the inevitable and intrinsic role of humans in the manufacturing process (Young and Rossi, 1988; Cohen and Zysman, 1988). Either implicitly or explicitly, an

CONTROL AND DYNAMIC SYSTEMS, VOL. 46

autonomous manufacturing system utilizing either an analytical model (e.g., Jones & Maxwell, 1986; Jaikumar, 1986) or artificial intelligence techniques (e.g., Fox and Smith, 1984; Smith et al., 1986; Miller, 1985; Astrom, 1985; Astrom et al., 1986) assumes the presence of human operators who monitor the automation, correct and fine-tune the process when necessary. Recently, in contrast to expert systems and 'deep' AI scheduling, the American Association for Artificial Intelligence (AAAI) Special Interest Group in Manufacturing (SIGMAN) workshop proposed interactive scheduling. Interactive scheduling is based on the belief that "...fully automated schedulers are not as desirable as interactive schedulers. The point here is that the man and machine bring complementary skills to the scheduling task, and that both are necessary to produce high quality schedulers (Kempf et al., 1991, p. 37)."

Human Supervisory Control In Manufacturing Systems

While acknowledging the presence of human decision makers, few researchers address explicitly the engineering and design of manufacturing control systems that integrate automation with the humans who are responsible for overseeing the effectiveness of system operation. Supervisory control is an alternative and more realistic paradigm to 'lights out' automation. Supervisory control of a manufacturing process entails the design of control and scheduling systems that explicitly integrate human decision makers into the underlying automation. Supervisory control is a design philosophy that explicitly addresses the roles and functions of both human and automatic components of the control process. Supervisory control systems make use of capabilities and compensate for the limitations of both human decision makers and automatic components. More specifically, supervisory control *designs* the human-computer interaction in order to augment and to extend the human's role and decision making effectiveness. Neither the goal nor the unintended side-effects of supervisory control are to automate the human decision maker out of the system nor to reduce the human's role to a set of undesirable or ineffective tasks.

Though supervisory control does not require additional or different machines or technologies, it does require rethinking of the role of people in manufacturing systems. An understanding of the philosophy and meaning of supervisory control permits utilization of expensive and valuable human resources and allows definition of operator functions that complement existing automated functions. The definition and well-defined engineering specification of the human functions in system control provide a necessary context for the related information processing issues including types and mechanisms for decision support, design of operator workstations, and human factors and ergonomics of display screens and operator interaction.

A previous paper (Mitchell, 1991) summarized several research efforts in human supervisory control for manufacturing systems. These research projects were conducted at Georgia Tech's Center for Human-Machine System Engineering. The projects included a description of GT-FMS (Georgia Tech's Flexible Manufacturing System (Ammons et al., 1988; Dunkler et al., 1988)--a real-time interactive flexible manufacturing system simulation, an operator function model (OFM) that proposed a role for the human operator in coordinating FMS cell-level control (Ammons et al., 1988), and initial research that provided experimental data to support the hypothesis that cell scheduling which includes a human supervisory controller results in significant improvement in overall system performance when compared to a fully automated scheduling and control system (Dunkler, 1986; Dunkler et al., 1988).

This paper is a sequel to Mitchell (1991). After a brief summary of the GT-FMS simulation, this paper reports on more recent research in human supervisory control in manufacturing. In particular, this paper describes three research efforts that blend theoretical and conceptual notions in human supervisory control with related experiments to evaluate their effectiveness in the GT-FMS domain. The first study examines the effectiveness of direct manipulation and other sophisticated human-computer interface technology on both training and trained performance of a manufacturing cell supervisor. The second study examines decision making of a system manager rather than a lower level FMS cell supervisor; the research sought to model decision making strategies of the system manager of GT-FMS and to investigate differences between the cell level supervisor and the system manager. Finally, we present the results of a study examining the effect of organizational structure on teams of operators responsible for a multi-cell FMS system.

Before examining the individual research efforts, we present a brief summary of the GT-FMS structure together with the ways it was reconfigured to meet the research needs. Furthermore, we note that although this paper examines supervisory control issues in the context of FMS scheduling and control, many of the ideas and some of the research results have applicability to more general manufacturing control processes.

GT-FMS Structure

The Georgia Tech Flexible Manufacturing System (GT-FMS) is a high fidelity simulation developed to provide an environment for studying human supervisory control designs for FMSs (Ammons et al., 1988). The GT-FMS environment is a real time interactive simulation of a multi-cell, multi-workstation flexible manufacturing system. A human decision-maker can interact with GT-FMS in a manner similar to that of a scheduler or expeditor on the shop floor. GT-FMS further provides.

> ... a controlled laboratory environment in which to implement and evaluate the supervisory control perspective for FMS scheduling. It facilitates research and validation in a framework of realistic manufacturing conditions, including human interaction with the scheduling and control system. (Dunkler et al., 1988, p. 225)

A variety of FMSs can be configured with GT-FMS due to its modular structure. Common to all configurations is an arrival buffer, where all parts reside when first entering the system, an FMS cell containing a central location for each cell's work-in-process (WIP) and the workstations, and an output buffer to which all completed parts proceed before exiting the system. GT-FMS can be one- or multi-celled; and each machine workstation within the cells can be configured uniquely, with its own set of capabilities.

Currently, there are two versions of GT-FMS based on data from existing manufacturing facilities. One version of GT-FMS uses IBM electronics assembly data. This version represents one cell, containing eight machines--two single in-line package (SIPs) inserters, three dual in-line package (DIPs) inserters, and three robots whose primary job is to insert modules but which have the capability to insert SIPs and DIPs although with less efficiency than the dedicated SIP or DIP insertion machines (Krosner et al., 1987). The Benson study, described in this paper, uses the electronic assembly configuration (Benson, 1989; Benson et al., 1991).

Also consisting of one cell, the second GT-FMS version uses data supplied by Motoren und Turbinen Union GmbH (MTU), a West German diesel engine manufacturer. The MTU version configures GT-FMS as a one cell system with four identical machining centers and two load/unload stations. The MTU GT-FMS also includes two batch processes that require parts of one type to accumulate for processes performed outside the FMS cell (Dunkler, 1986; Dunkler et al., 1988). The Hettenbach study, described in this paper, uses the diesel engine configuration (Hettenbach, 1989; Hettenbach et al., 1991).

Finally, there is a multi-cell version of GT-FMS. Although not based on actual data, this version of GT-FMS was constructed to examine multi-cell, multi-operator interaction in FMS scheduling and control. The Armstrong study, described in this paper, uses the multi-cell version to examine the effectiveness of hierarchical versus heterarchical team structures for different levels of system load and communication delays (Armstrong, 1990).The hierarchical version of the multi-cell GT-FMS consists of two cells (each with an operator) and a supervisor that coordinates the cells to meet overall system goals. The heterarchical version consists of three cells, each cell containing fewer machines than the two-cell system, with a cell operator for each cell; the heterarchical system does not have a decision maker who is designated as the supervisor (Armstrong and Mitchell, 1986; Armstrong, 1990).

Direct Manipulation Technology: Assessment of its Utility for FMS Operators[1]

Background

As computer hardware and system software costs are decreasing, system designers have access to an array of human-computer interaction devices such as mice, touch panels, voice input and output, and high fidelity graphics and windowing packages. One such human-computer interaction technology is that of direct manipulation--the representation of objects graphically and manipulation of those objects via pointing devices. Previous research involving human-computer interaction has been primarily in the context of computer programming, text editing or word processing applications; for examples, see *ACM-SIGCHI*, 1984, 1989. However, user interaction tasks in these applications differ significantly from those in human supervisory control tasks. Most programming or text editing tasks are not time-contingent. Supervisory system controllers are faced with opportunities that change over time. Once an opportunity passes, it can never be recovered. Secondly, the consequences of errors are very serious in the supervisory control domain. The consequences of errors for programmers or editors result only in a decrease in productivity. The programmer can always undo or redo an action and simply recompile the code. This will result only in the loss of time. In supervisory control systems, errors may be catastrophic and expensive (e.g., airline crashes, Three Mile Island, or eight hours of downtime in a $10 million FMS). Thus, the interfaces to such systems becomes increasingly important.

This research was designed to demonstrate and evaluate the effectiveness of an operator interface using direct manipulation interaction in FMS supervisory control. In particular, the effectiveness of two different FMS operator workstations--a conventional operator workstation with overlapping windows and a keyboard versus a direct manipulation workstation design was explored. The experiment used the Georgia Tech-Flexible Manufacturing System (GT-FMS).

GT-FMS Configuration and Conventional Operator Interface

The system configuration of GT-FMS used in this experiment is shown in Figure 1. It is comprised of eight insertion workstations, three buffers and a transportation system.

1. Benson, 1989; Benson et al., 1991.

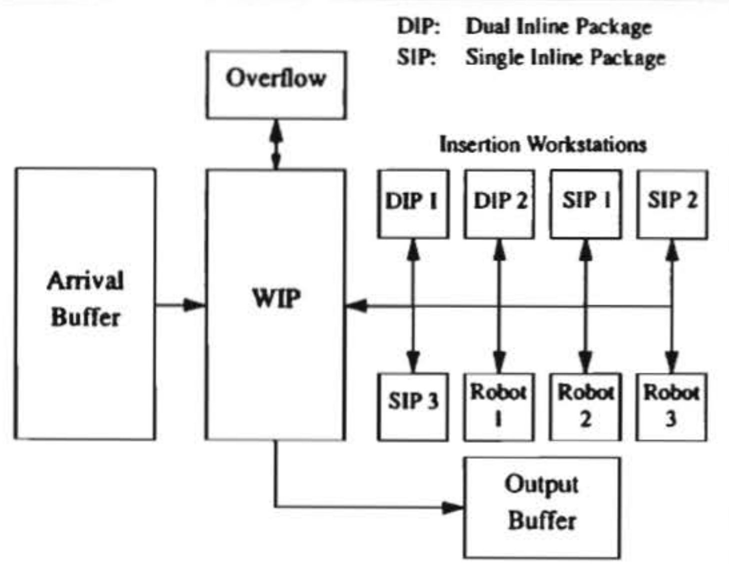

Figure 1. System Cofiguration of GT-FMS

Parts accumulate outside the FMS cell at the arrival buffer, which is an unlimited capacity buffer. An arriving part is stored in the arrival buffer until it is dispatched to the GT- FMS cell work-in-process (WIP) buffer. Within the cell, parts are stored in WIP, which has a finite capacity of 20 parts in this configuration. Parts wait in WIP for an available insertion workstation to perform the next required operation. If the WIP is full when a part arrives to WIP, the part is automatically routed to the overflow buffer. This buffer, too, has unlimited capacity. GT-FMS also has a material handling system capable of performing all routings shown by arrows in Figure 1.

As system supervisor and controller, the operator's major, though not necessarily complementary, goals of FMS cell scheduling and control are: 1) to minimize the cost associated with part completions that occur past the due date; and 2) to minimize the cost associated with cell inventory.

Operator controls include 'expediting' a part to give it priority over the automatic scheduler, removing a part from a failed machine, and increasing or decreasing the number of parts buff-

ered in the cells working in process (WIP) inventory. Operator performance was evaluated by a session score, a weighted linear combination of late parts and work-in-process inventory.

The conventional operator workstation for GT-FMS uses a basic *cell status page*. From this page, the operator can request additional information in the form of windows which cover part or all of the screen. Using this workstation, the operator can also execute control actions, such as expediting a part, removing a part from a broken workstation, reversing an expedition, or resetting the cell's minimum WIP level. The basic cell status page is shown in Figure 2.

From the basic cell display, the operator can monitor part movement, the status of the workstations, the WIP level, and the simulation time. All possible operator display and command options are listed across the top of the cell status page. The operator selects the desired command by moving the selection highlight bar via the cursor keys on the right side of the keyboard. The line directly below the command option line is reserved for information and error messages. It currently reads, "Real time operation resumed. Ready for input." Any parts that are late or projected to be late appear in red on this screen only. Parts currently on schedule are displayed in blue.

All physical locations to which parts may be moved or expedited are referenced with the function keys located on the left side of the keyboard. The function key associated with each location, e.g., the WIP and all workstations, are displayed in the title bar of their windows. For example, to reference WIP when expediting a part from the arrival buffer, the operator presses F2.

The WIP window is displayed on the left of the screen. For each part in WIP, the part tag and its next required operation are shown in the WIP window. The parts are listed according to their arrival times to the WIP buffer. For example, in Figure 2, f22 is the first part listed and, therefore, the oldest part in WIP. It will be the first part scheduled on a workstation that can perform a SIP insertion. The single digit number following the next required operation of each part numbers the parts to help the operator remember how many parts are currently in WIP. In Figure 2, there are fourteen parts in WIP.

The operator may gather more information about the system status or execute commands via the command line. The "Parts" command is a request for a list of all parts currently available in the system, including all parts in the arrival, overflow, in-transit and output buffers, as well as all parts shown on the basic cell status page. The "Bufr" command invokes a window which lists all parts currently residing in the arrival buffer. When the operator selects the "Rush" com-

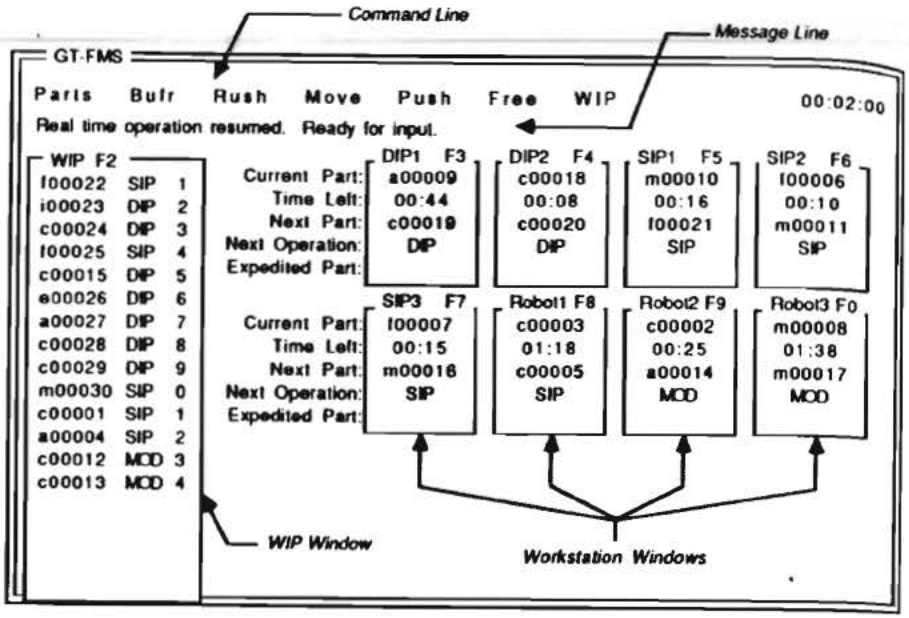

Figure 2. Basic Cell Status Display

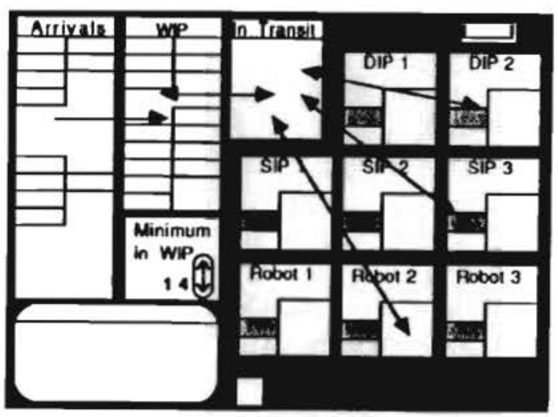

Figure 3. Part Movement

mand, a window containing a list of all parts that are currently late or projected to be late is shown on the lower half of the screen. If an operator wishes to move a part from a failed workstation, s/he may invoke the "Move" command. A small window will appear prompting the operator to type in the part tag. The "Push" command allows the operator to expedite a part. The operator will be prompted to type in the part tag and then will be prompted to press the function key associated with the workstation to which the part is to be expedited. The "Free" command allows the operator to "unexpedite" a part to a given workstation. Finally, the "WIP" command allows the operator to reset the minimum number of parts to be held in the work-in-process buffer.

The type of interface described above forces the operator to switch between pages of displays in order to retrieve all relevant information. It also employs the use of most of the keys on the keyboard, including the function and cursor keys. The cursor keys on the far right of the keyboard are used to manipulate the command highlight bar as well as scrolling the information on the arrival buffer and rush page displays. The basic alpha-numeric keys in the center of the keyboard are used to input part numbers and desired WIP levels. The function keys are used to designate locations associated with workstation numbers and WIP. Consequently, the operator's performance not only depends on her/his typing ability, but s/he is forced to move her/his hands from the alpha-numeric keys to the cursor and function keys and the "mental workload" associated with control tasks is increased since the operator must remember when to use the different sets of keys. The times associated with these transitions may slow the operator's execution of desired actions.

Direct Manipulation Operator Interface

Figure 3 depicts the proposed direct manipulation workstation configuration. The workstation consists of a single screen with the mouse being the single mode of operator input. The elimination of the keyboard eliminates the dependence of the operator's performance on her/his typing skills and experience. The contents, appearance and placement of the windows are dictated by a description of the major functions to be performed by the operator and a set of basic interface design principles.

Window Locations

The operator can access windows containing lists of the parts found in the arrival, overflow, in-transit and WIP buffers, as well as windows which represent each of the eight workstations. Since the logical flow of the parts is from arrival buffer to WIP buffer to workstation, the windows are placed in this order from left to right. The in-transit buffer window is placed between the WIP and the workstations since parts most frequently travel between these two locations. Also, since the primary part movement occurs between the arrival buffer, in-transit buffer, WIP buffer and workstations, these windows do not overlap. Figure 3 illustrates the flow of part movement on the display.

Because it is undesirable to have parts in the overflow buffer the window representing the overflow buffer is available only when there are parts residing there. When the overflow buffer is empty, this window is not available. When the operator opens the overflow buffer window, it appears in front of the arrival buffer window. The operator can view *either* the arrival buffer or the overflow buffer. This design was chosen primarily because there is no relation between the two windows, e.g., the operator cannot expedite a part from the arrival buffer to the overflow buffer, or vice versa. From either location, a part is expedited to the WIP buffer. So, it is not imperative that the operator view both windows simultaneously.

The operator can close the arrival, overflow, in-transit or part information windows completely to unclutter the screen. To reopen the arrival, overflow or in-transit buffer window, the operator clicks on an icon corresponding to each window. Figure 4 shows the icons corresponding to the arrival, overflow and in transit buffer windows.

The incorporation of a single screen display differs significantly from the conventional display described earlier. It eliminates the operator's need to search for and retain information between changing screens and the redundancy of displaying the same information on separate screens.

Part Representation

Parts are displayed in the windows labeled for their current location. All parts in the arrival buffer, overflow buffer, WIP buffer and robot workstations are displayed with the first letter of their next required operation in parenthesis following the part tag. This allows the operator to quickly recognize valid machines for part expedition.

The most important feature of the part field is its background color, which indicates its current status--late, projected late, or on schedule. This design feature was incorporated to aid the supervisor in her/his cell management function. The background color of the part field rather than the foreground was changed to eliminate white space and better attract the operator's attention. The colors alerts the operator to late or projected late parts. If the part's due date has already passed, the background will be red, the color most often used in alert situations to easily attract the operator's attention. If the part is projected to be late, the background of the part field will be yellow, cautioning the operator to take action before the part's due date passes. If the part is

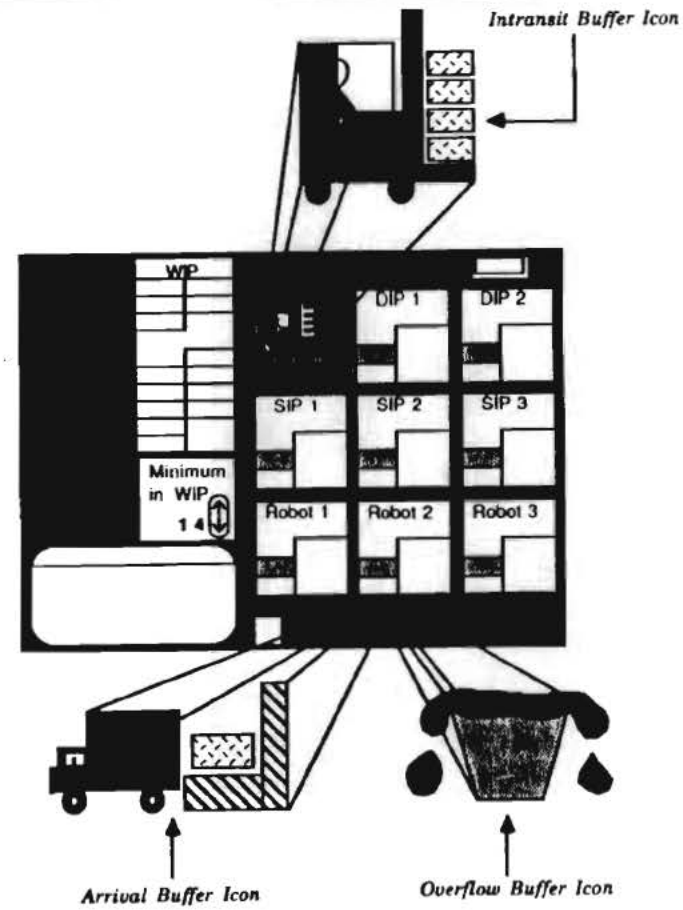

Figure 4. Icon Representations for Arrival, Overflow and In-transit Buffer Windows

on schedule, it's background will be white. If the operator expedites a part in WIP, the background will change from red, yellow or white to green. This reminds the operator that s/he has already taken steps to push that part through the system. The use of these different colors in indicating part status was incorporated to improve the conventional design by providing the operator with the additional information of whether the part is already late or just projected to be late.

Workstation Windows

The workstation windows are displayed on the right side of the screen. Unlike the conventional display, the workstations are grouped by type. Both DIP workstations are displayed in one line, the three SIPs in a second line, and the robots in a third. Colors were also used to group the machines by types. Soft, neutral colors were used so as not to distract the user from the more important alert colors associated with part status.

Each workstation window displays the part currently in the insertion position and its remaining processing time. Because the robots can perform all operations, each part displayed in any of the robot workstations is followed by the first letter of its current required operation. The part in the insertion position is shown in the center of the workstation window. To the left of the part in the insertion position, in the lower corner, is the part in the workstation buffer. Just above the part in the workstation buffer is a gray rectangle reserved for a part that is expedited to the workstation. If a part is expedited to a particular workstation, its part tag is displayed, and, if the workstation is a robot, its next required operation will also appear in this rectangle. The part's background will be green to correspond to the green background of the part in WIP, indicating that the operator has taken action to push the part through the system. The expedited part is displayed above the part in the workstation's buffer because it will be placed in the insertion position before the part in the workstation's buffer. The part placement in this design differs from the conventional design, where the expedited part is listed below both the part in the insertion position and the part in the workstation buffer, to better show the priority of the parts to be processed, since they occupy two different physical locations at the workstation. The part currently in the insertion position is separated from the parts to be processed. The part scheduled to be processed next, will move from the left of the workstation window to the right of the window, conserving the direction of part movement. If a part is expedited to a workstation, it is displayed above the part in the workstation buffer, simulating an ordered list. The highest priority item in the list, in this case, the expedited part, is the item at the top of the list. In the conventional design, there is also a field for each part's current required operation in all the workstation windows even though the DIP and SIP machines are dedicated to one operation. To eliminate redundancy, the current required operation for each part is not included in the DIP and SIP workstation windows in the direct manipulation workstation configuration.

Arrival and Overflow Buffer Windows

The arrival and overflow buffer windows occupy the same space on the screen. There is an icon resembling a truck backing up to a loading dock in the lower center of the screen (Figure 4). Any time the operator wishes to view the arrival buffer, s/he moves the cursor to this icon and clicks. This icon is always accessible and the arrival buffer can be viewed at any time. However, the overflow buffer can only be viewed when there are parts actually residing in the overflow buffer. Any time there are parts in the overflow buffer, an icon resembling an overflowing bucket will appear in the lower center of the screen next to the arrival buffer icon (Figure 4). The appearance of this icon alerts the operator that parts have been placed in the overflow buffer.

WIP Buffer Window

The WIP buffer window is always visible and contains a list of the parts currently residing in WIP (Figure 4). The operator can always tell how many parts are in WIP. The conventional display's WIP window is often covered by other windows. For example, when the rush page is displayed, the bottom portion of the WIP window is covered. The operator cannot see how many parts are presently in WIP. This may influence her/his decision of whether or not to expedite a part from the arrival buffer to WIP. In the proposed workstation configuration, the entire WIP window is always visible.

The supervisor can expedite parts from the WIP buffer to any workstation that does not already have an expedited part. Once expedited, a part will remain in WIP (with a green background) until the machine to which it is expedited completes the processing of its current part. The part is then placed in the in-transit system and transported to the machine. The machine remains idle until the expedited part arrives.

In-transit Buffer Window

The in-transit buffer window works much like the arrival and overflow buffer windows. This window can be opened and closed at the operator's discretion. When closed, an icon resembling a forklift truck will appear in the space to the right of the WIP window (Figure 4). Since the parts are being transported to a specific location, the part tag is followed by an arrow pointing the direction in which the part is traveling. If the part is being transported to the WIP, the part tag will be followed by a black arrow pointing to the left, since the WIP window is located to the left of the in-transit buffer window. If the part is traveling to a machine, a colored arrow pointing to the right will follow the part tag. The color of the arrow corresponds to the type of machine to which the part is being transported. While the operator cannot move or expedite parts that are currently in-transit, s/he may need to know which parts are traveling to specific locations.

Cursor Shape and Current Activity

The shape of the cursor reflects the current activity which the operator can perform. If the cursor is shaped like a question mark, the operator may retrieve additional information about parts on the screen. If the cursor is shaped like a clamp, the operator can "pick" parts up and move them to different locations. This method of using the cursor to reflect the current activity was adapted from Macintosh applications, e.g. MacDraw and MacPaint. In both applications, the cursor reflects the activity in which the operator is currently engaged.

The cursor (activity) options - a question mark to retrieve additional information and a clamp to move or expedite parts - are displayed on the bottom center of the screen. The operator simply clicks on the shape of the cursor corresponding to the activity s/he wishes to perform. The following sections describe in detail the activities of retrieving additional part information and expediting and moving parts to different locations.

Monitoring and Retrieving Additional Information

The operator can monitor the dynamic system states simply by observing the parts moving from place to place and watching for parts with red or yellow backgrounds. However, if the operator needs more information about any part listed on the display, s/he has access to any part's due date, projected time to finish and remaining operations. When the cursor is in the shape of a question mark, the operator can retrieve this additional part information. To change the cursor into a question mark, the operator moves the current cursor to the icon representing the question mark in the lower center of the screen. S/he then clicks the mouse and the cursor changes to a question mark. The question mark was chosen because this symbol represents the availability of information in many international airports and sights of travel (Marcus, 1987) and is a common symbol used in computer systems that allows the user to request "help" or additional information. After the cursor has been changed into a question mark, the operator can click on any part on the screen and the part information window will appear in the lower left corner of the screen.

The part tag appears in the title bar of this window. The first line of information in the part information window is the part's due date. If the due date has already passed, the time due is highlighted in red, corresponding to the part's background in the other windows of the screen. The second line is the part's projected completion time. If the part's due date has already passed, obviously, the projected completion time is past the part's due date. If this is the case, this time field is also highlighted in red. If, however, the part's due date has not passed, but the part is projected to finish after the due date, the projected completion time field is highlighted in yellow to correspond to the part's background in the other windows of the display. The third line is an ordered list of the part's remaining operations. This may influence the operator's decision on which parts to expedite if two parts are late but one part has five operations remaining while the other has only one operation remaining. The operator may also want to expedite parts that have one or two operations remaining so that the parts leave the system, clearing out space in the WIP buffer. The last line of the information window is reserved to indicate if the part is expedited to a machine, and if so, which machine. If the part is expedited, this message appears in green to provide consistency with the part's background in the other windows of the display. The operator has access to additional part information at all times. This feature was incorporated to aid the operator in making decisions for cell management and inventory management.

Expediting and Moving Parts

The primary way for the operator to minimize the late time associated with specific parts is to override the automatic scheduling system by expediting parts through the system. Before expediting a part, the operator must change the cursor into a clamp. The clamp symbolizes the action of picking up an object and dropping it in another location. To change the cursor into a clamp, the operator must move the cursor to and click on the clamp icon located in the lower center of the screen next to the question mark icon. Now the operator can expedite a part from the arrival and overflow buffers to WIP and from the WIP buffer to any of the insertion workstations. S/he can also move any parts located on broken machines back to the WIP buffer. To expedite a part from the arrival or overflow buffer to WIP, the operator simply moves the clamp cursor into the part rectangle in the arrival or overflow buffers and presses the mouse button. The part's rectangle will be inverted and the cursor will change into a clamp holding a small part, shown in Figure 5. As long as the operator holds down the mouse button, the clamp will be "holding" onto that part. The operator can then drag the part into the WIP buffer window and release the mouse button. The part will immediately be placed in-transit and the cursor will return to the open clamp representation. If the operator releases the mouse in a location to which the part cannot be legally moved, e.g., from the arrival buffer to a machine, the part will remain in its original location and the cursor will return to the open clamp representation.

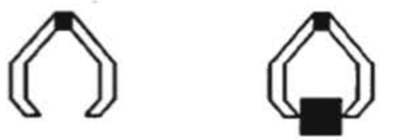

Figure 5. Cursor Representations for Expediting and Moving Parts

When an operator is attempting to expedite a part from the WIP buffer to a workstation, s/he must consider the workstation's capabilities. S/he cannot expedite a part to a DIP machine which requires a SIP operation next. Because all workstations cannot perform all operations, another feature was incorporated into the display design to direct the operator when expediting a part to a workstation. After "picking up" the desired part in WIP, as the operator drags the cursor through the workstation windows, the gray expedite field in the workstation window will turn green if that workstation can perform the part's next required operation, signaling the operator that "dropping" the part in this workstation window is a valid expedition. Once the part is dropped in a workstation window, the cursor will return to the open clamp representation and the part tag will appear in the workstation's expedite position and will be highlighted in green in the WIP buffer window.

Should the operator decide that s/he has expedited a part to a workstation in error, or decides to expedite another part to that machine, s/he can free the workstation and "unexpedite" the associated part. The operator simply moves the clamp cursor to the expedite position in the workstation window and "picks up" the part by pressing and holding the mouse button. S/he then drags the part into the WIP window and "drops" the part there by releasing the mouse button. The part's background will no longer be green, the workstation's expedite position will be empty and the cursor will return to the open clamp representation.

Moving a part from a workstation's insertion position or buffer back to the WIP buffer can only be executed if that workstation has broken down or is being repaired. The operator will know that the workstation has broken down or is being repaired by the large red icon that is displayed over the workstation window (Figure 6).

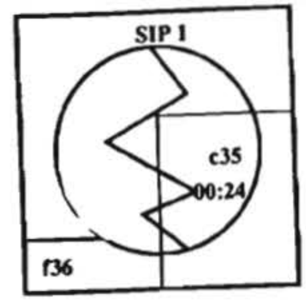

Figure 6. A Failed Workstation

Parts are moved from a broken workstation back to WIP the same way as they are unexpedited. The operator moves the open clamp cursor over the part in either the insertion position or the buffer and presses and holds down the mouse button. S/he then drags the part into the WIP win- dow and releases the mouse button. The part immediately goes in-transit to the WIP and the cursor returns to the open clamp representation.

Manipulating parts on the screen with the mouse eliminates the need to have the operator type commands, part numbers and destinations. The operator never needs to focus her/his attention on anything other than the screen, since s/he does not have to search for keys or correct typo- graphical errors. All part expeditions and movements are executed in a consistent manner. Sim- ilarly, if the operator makes an error or changes her/his mind, actions to reverse previous actions are executed in the same manner in which the original action was executed.

Monitoring and Adjusting the Minimum WIP Level

The operator can only exert one other type of control over GT-FMS. S/he can adjust the mini- mum number of parts held in the WIP buffer. At the beginning of each experimental session, this number is arbitrarily set to fourteen. The operator may wish to lower this number so that s/he can have more control over which parts come into the FMS cell and to keep the number of parts low so that s/he may more closely monitor that parts that are in the cell. S/he may wish to raise the minimum number of parts in WIP in order to increase throughput and ensure that parts will be pulled in early enough to meet their pending due dates. Thus, a trade-off is involved. No matter which strategy the operator chooses, s/he can adjust the minimum number of parts in WIP via the window located just below the WIP buffer window entitled "Minimum Parts in WIP."

The current minimum is displayed just below the title. The operator can adjust this number by using the control arrows to the right of the minimum number displayed. When s/he moves the cursor into the control box containing the up and down arrows, the cursor will automatically change into crosshairs. S/he may move the cursor onto one of the arrows and increase or decrease the minimum number in WIP by one. If s/he presses the mouse button down while the cursor is positioned over the up arrow, the new minimum number would be fifteen. Similarly, if the operator presses the mouse button down while the cursor is positioned over the down arrow, the new minimum number of parts in WIP would be thirteen. This type of control action is consistent with other applications on the Macintosh.

Evaluation

An experiment was conducted to evaluate the effectiveness of the two operator workstations. Session score and the number of 'expedite' operator movements were primary measures of performance. The experimental results are summarized below; additional details can be found in Benson (1989) and Benson et al. (1991).

The subjects' primary goal was to minimize the total cost associated with operating GT-FMS. When the effect of condition on total score was analyzed, the mean cost for the conventional interface condition ($1051.07) was significantly higher than the direct manipulation interface mean cost ($956.63). Figure 7 depicts the mean total scores for each of the ten sessions.

For the overall score and a session by session score, subjects using the direct manipulation interface scored significantly better, i.e., lower cost, than subjects using the conventional interface.

The number of times subjects expedited parts was also recorded to evaluate whether users of one interface exercised more control over the system than users of the other interface. The number of times that an expedite command was issued was significantly greater for subjects using the direct manipulation interface. Figure 8 shows the means for the number of parts subjects expedited in each session.

For all ten sessions, the mean number of expedites for the conventional interface is lower than the mean number of expedites for the direct manipulation interface. The results from the individual t-tests for each session indicate that in six of the ten sessions the mean number of parts

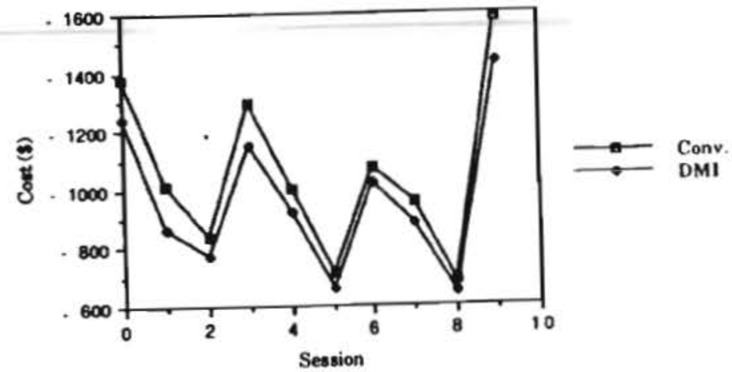

Figure 7. Average Total Score by Session

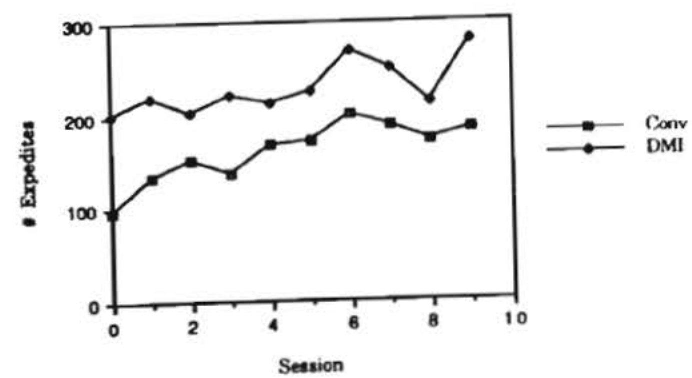

Figure 8. Average Number of Expedites by Session

expedited by subjects using the conventional interface was significantly lower than the mean number of parts expedited by subjects using the direct manipulation interface.

Summary

As a high fidelity simulation, GT-FMS provides insight as to how a direct manipulation interface might improve operator performance in a real-time supervisory control system. The overall goal in controlling the GT-FMS system is to minimize the cost associated with completing parts past their assigned due dates and the average amount of inventory located within the FMS cell. The conventional interface to GT-FMS presents overlapping window displays and employs the use of the entire keyboard, alpha-numeric, cursor and function keys, as the means of operator input. This research indicates that such a system restricts the freedom of operator interaction and control over the system. A direct manipulation interface to GT-FMS is more likely to increase the amount of operator intervention and increase the control the operator exerts over the system. Results indicate that for minimizing the components of lateness and inventory costs as a total, operators using the direct manipulation interface performed better than did subjects using the conventional interface configuration.

Experimental results apply to systems beyond the specific GT-FMS environment. Results from the GT-FMS simulation provide strong support that operator performance can be greatly influenced by the user interface configuration. In the wider area of supervisory control, the use of direct manipulation and the principles that were used to develop the GT-FMS interface for this research may provide a superior methodology over conventional interface design.

Decision Making of a Systems Manager of a Flexible Manufacturing System[2]

Background

For an FMS to be effective, the functions of the human supervisor must be defined and the limitations of the human taken into account. Recognizing previous GT-FMS research which supports actively integrating the human into the FMS control structure, an understanding of the appropriate level of control or role for the human is critical. Ammons et al. (1988) address the role of the human supervisor by proposing a realistic supervisory control paradigm for the cell-level FMSs.

In an FMS control system, the appropriate control system model and accompanying decision support are dependent upon the role of the human. Defining an appropriate role and subsequent

2. Hettenbach, 1989; Hettenbach et al., 1991.

responsibilities for the human as a function of the FMS objectives may thus be a critical step in designing the FMS control structure. Previous GT-FMS research addresses the role of the human supervisor as a cell-level controller (Ammons et al., 1988). As FMS control systems continue to develop, however, the degree of human interaction with these control systems will also evolve. Control systems may become more "intelligent," and the human role may shift the emphasis to elements of planning and management, moving the operator from the role of cell-level controller to system manager.

Human Operator as a System Manager

In this investigation, the role of the humans focuses on longer-term goals of the manufacturing system. This role reflects a level in an FMS control hierarchy that is "higher" than the item-movement level (i.e., the level investigated in Dunkler's (Dunkler, 1986; Dunkler et al., 1988) and Benson's research (Benson, 1989)). The research explores the human's response to this role together with the decision making process.

The supervisor in this investigation takes an aggregate view of the flexible manufacturing system. This approach is explored for several reasons. Typically, FMSs have some degree of human interaction as part of their control system, but this interaction is not always well defined, and it often occurs on an ad hoc basis. Previous research (Dunkler et al., 1988) indicated that FMS performance can be improved if the *ad hoc* nature of the human control actions is removed. However, no universal "best" definition exists for the design of human interaction, with hierarchical control models and actual FMS installations allowing varying degrees of human intervention throughout all control levels.

By modeling the supervisor from an aggregate view, additional insight can be obtained toward creating a supervisory environment which potentially takes better advantage of the human's judgment and decision-making skills. In this role, "systems manager," rather than cell "supervisor," may better describe the human control functions involved. Designing human intervention at a higher level removes the systems manager from minute-to-minute contingencies and allows the systems manager to focus on meeting the long-term objectives of the FMS. Furthermore, while a computer may handle minute-to-minute decisions in some FMS installations, a computer is not ultimately responsible for the performance of actual flexible manufacturing systems. Thus, human monitoring of aggregate FMS performance data in practice is virtually guaranteed.

For a systems manager to effectively control the FMS, s/he must be allowed to initiate certain control actions and be provided with clear goals against which to measure these control actions. Overall profitability of the FMS is likely to be an overriding concern for a systems manager. As such, the systems manager must initiate control actions which positively affect the profitability of the FMS.

An FMS is likely to process a wide variety of parts simultaneously, with parts belonging to many different customer orders. System cost performance is affected by completing customer orders on time, and by completing enough orders so that the cost of production per part is sufficiently low. Processing priorities of the parts within an FMS can significantly affect the operation of the FMS and thus have an impact on profits. By modifying the processing priorities within the FMS, the systems manager can emphasize a specific method of operation which provides the greatest profit potential for a specific period of time.

This investigation does not seek to prove or disprove a specific theory or hypothesis. Rather, it seeks to gain further insight into human decision making within an FMS environment. Decision processes of humans in an FMS systems manager's role are analyzed with the goals of better understanding and defining the human's role in an FMS control structure, improving feedback mechanisms of FMS control loops, and uncovering decision-making parameters used by human decision-makers in an FMS environment. As a systems manager, the human is placed in the FMS control structure on an aggregate level. This control level provides the systems manager with the opportunity to enhance FMS performance by modifying the part scheduling algorithm or expediting specific groups, or orders, of parts.

GT-FMS Configured for Systems Management

For this study GT-FMS was configured to simulate a machining center for diesel engine cylinder heads. In this configuration, GT-FMS had four identical machining centers and two load/unload stations. The data used in this experiment were based on a system installed by a West German diesel engine manufacturer, Motoren und Turbinen Union GmbH (MTU). The data were aggregated and scaled to provide realistic yet experimentally meaningful behavior. A complete description of MTU and the data from the MTU system can be found in Dunkler (1986) and Dunkler et al. (1988). Figure 9 depicts the GT-FMS process flow for this research.

Parts, Part Scheduling, and Orders of Parts

The systems manager is free to modify the computer-based part scheduling algorithm (cell scheduler) in response to trends noted in overall cell performance. In GT-FMS, the automatic

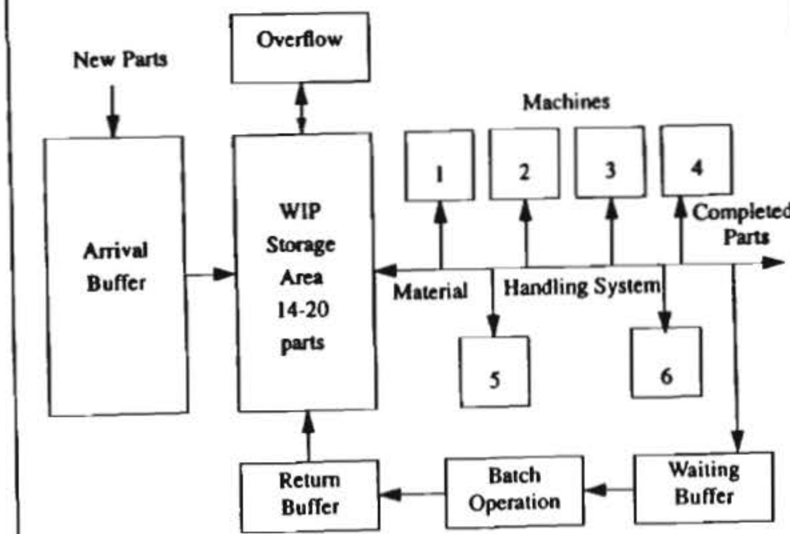

Figure 9. Process Flow in MTU GT-FMS Cofiguration

cell scheduler determines which part to place on an available machine. The scheduler examines all parts in cell inventory and then selects an appropriate part. To make this selection, the base GT-FMS scheduling automation was modified to use an algorithm called the weighted operation priority index (WOPI) (Han and McGinnis, 1986; Eagle, 1987). The WOPI algorithm is a weighted linear combination of the shortest-processing-time (SPT) and the earliest due date (EDD) scheduling algorithms. The systems manager modifies the WOPI algorithm by changing alpha, a weighting factor which orients the algorithm towards either due date or SPT scheduling. The systems manager decides when to change alpha and the magnitude of the change. In this experiment, alpha may assume values between 0.0 and 1.0, inclusive, in increments of 0.1.

For this study, GT-FMS was augmented to generate and track *orders* of parts rather than individual parts. All parts in a given order are of the same type and each part in an order has the same due date. The system performance measures reflect this enhancement. The score for each session has two components. The first component reflects completed parts and the second late parts. The session score is the difference between these two components. The completed parts component is the total number of parts completed during a session multiplied by the estimated profit realized from each part. The estimated profit per part is assumed to be $50.

A penalty cost is assessed for completing some or all of the parts in an order after the order's due date. The late parts component of the session score is the sum of the penalty costs assessed during a session. A penalty cost is calculated for each completed order which contains late parts. At the end of a session, penalty costs for incomplete orders are also included. The penalty cost is the product of the number of parts in an order (the order size), the amount of time (calculated in minutes) past the due date at which the last part in the order is completed, and a two dollar per part per minute late penalty on each late order.

Operator Controls and Displays

The operator workstation consisted of an Apple Macintosh II with a color monitor. GT-FMS was implemented in C. The operator workstation consisted of three display options listed horizontally across the top of the screen: "Windows," "Summary," and "Scheduler" as shown in the top of Figure 10. The "Windows" option provides another menu which allows a choice between the penalty cost and throughput display or the order status display.

Figure 10 illustrates the penalty cost and throughput display. The display shows two graphs: one for penalty costs associated with late parts and one for total parts completed (or throughput). These graphs are updated (i.e., a new point is plotted) every three seconds.

The throughput graph is also dynamic and averaged over the last minute of operation. This graph (in Figure 10, the curve with oscillations) provides a measure of how fast the FMS is producing parts.

The order status display provides the subjects with key summary information for each order currently in GT-FMS. Figure 11 illustrates this display. Completed orders are not displayed. The order status display is not dynamic.

For each unfinished order, the order status display lists performance statistics for the session, including the order number, the order due date, the size of the order, the number of completed parts, any penalty costs, and a summary of each unfinished part in the order.

Figure 11 also illustrates the summary window (entitled "Summary") within the order status display. The summary is not automatically shown when the order status display is chosen but

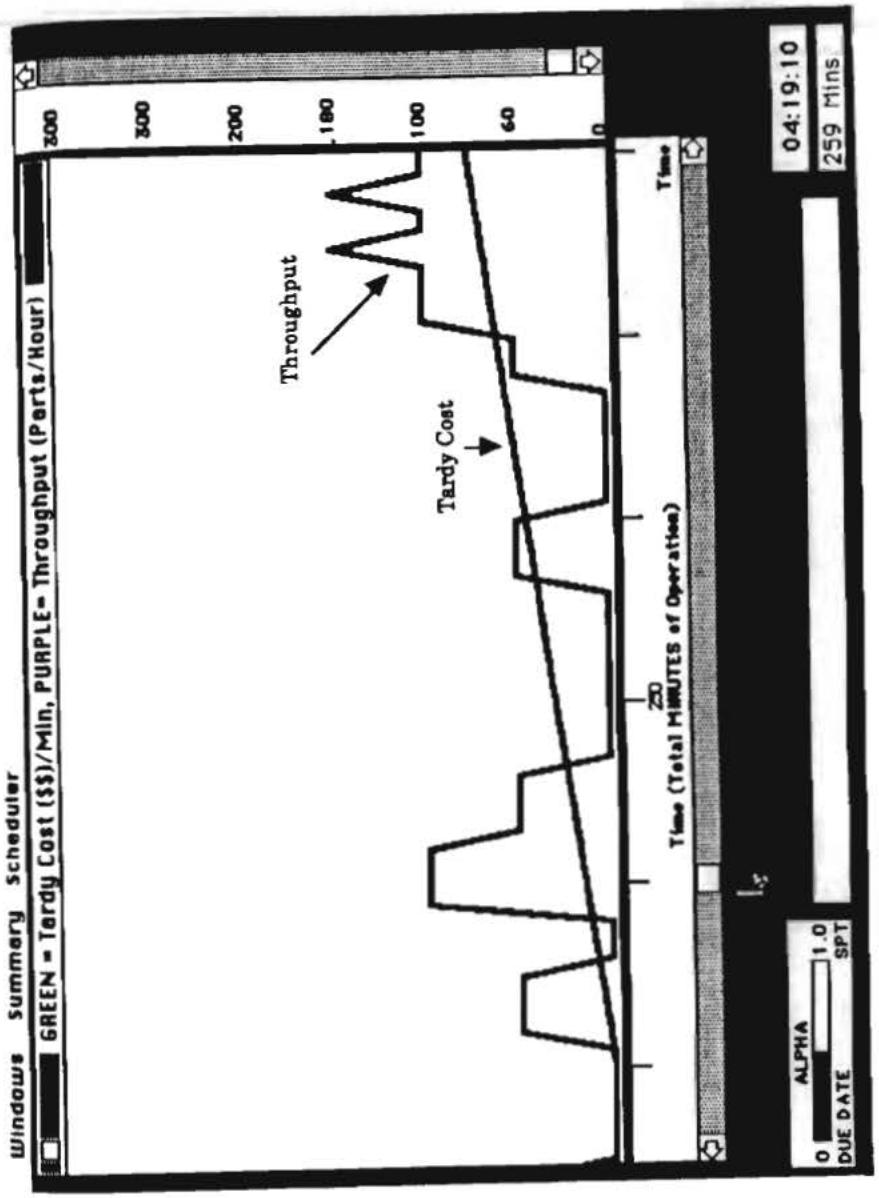

Figure 10. Penalty Cost and Throughput Display

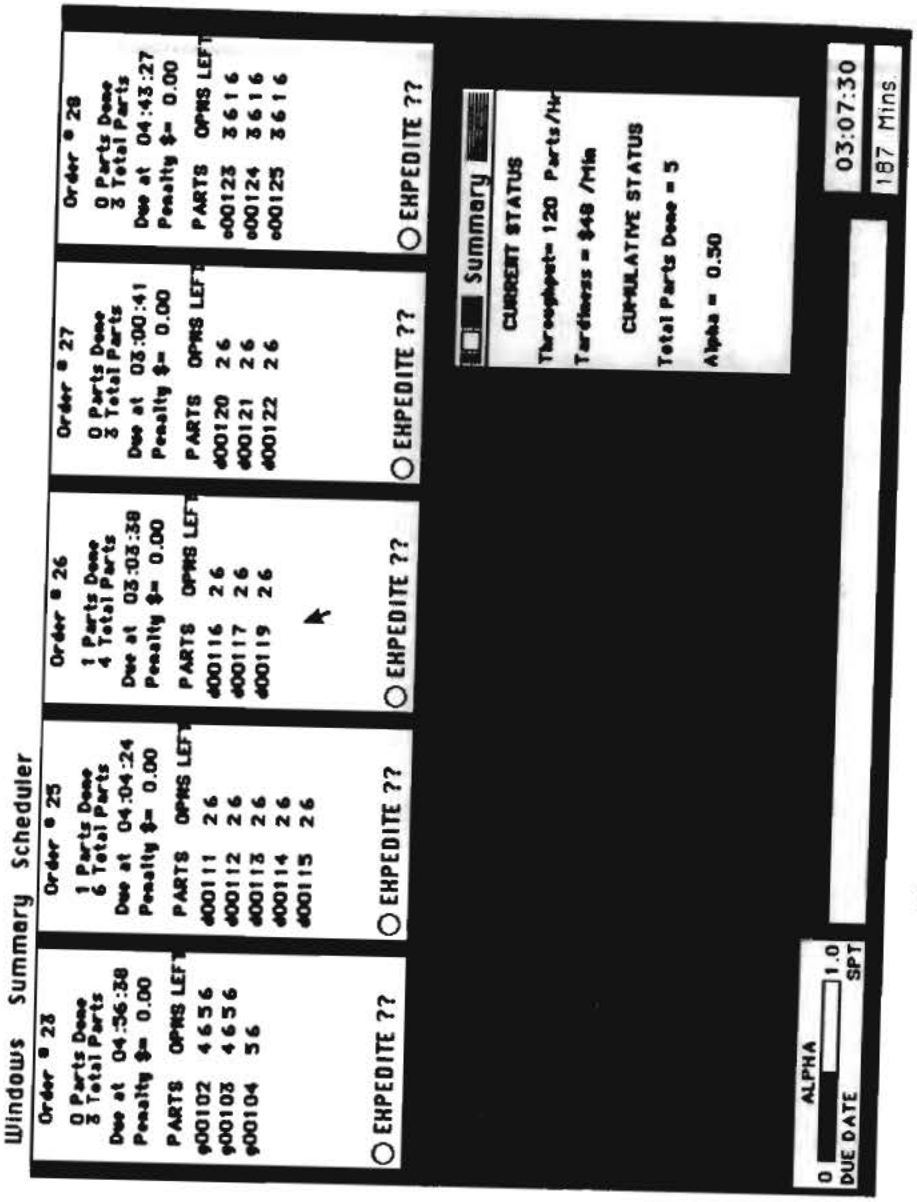

Figure 11. Summary Display with Order Status Display

must be selected from the list of display choices located at the top of the CRT screen (i.e., "Windows," "Summary," and "Scheduler").

The summary lists several performance statistics for the session. These include the total number of parts completed thus far and the current average penalty costs for late parts (tardiness). The summary window is dynamic. As a session progresses, the summary display updates every three seconds as long as this window is on the screen.

The *WOPI Weighting Factor* window displays current value of alpha and the resulting priority of the system scheduler, either SPT or due date, is shown as a bar graph on the alpha display. The filled portion of the bar graph corresponds to the current value of alpha. Additional detail for each of the displays can be found in Hettenbach (1989).

GT-FMS allows the systems manager two types of interventions to increase system profits. The first type enables the subjects to alter the weighting factor, alpha, used by the automatic system scheduler. To modify the value of alpha, the subject first selects the scheduler display with the mouse, then positions the mouse over the oval next to her/his choice and "clicks" (i.e., depresses and releases the control button on the mouse). The new value of alpha is then highlighted and the alpha window (i.e., the bar graph) is also updated with the new value. In addition, the two graphs in the penalty cost and throughput display are marked to record the change. These marks appear as parallel vertical bars on both the tardy graph and throughput display. The markings on the graph provide a way for the systems manager to review the history of each change in alpha during a session.

The second type of intervention enables a systems manager to expedite an order. If an order is expedited, the system scheduler places a priority on completing parts from the expedited order. When a machine becomes available, the system scheduler examines parts in the expedited order first. If a part in the expedited order can be processed on the available machine, then this part is placed on the machine without examining any of the other parts in any of the other orders.

To expedite an order, the subject first selects the order status display, then positions the mouse over the empty oval next to the word "EXPEDITE?" and "clicks." A red highlight bar, enclosing the order number on the order's display, then appears and the empty oval is darkened, indicating that the order is expedited. Subjects can "unexpedite" an order by positioning the mouse over a darkened oval and "clicking." The system scheduler will once again give parts from all orders equal priority as before. Only one order can be expedited at a time, but a systems manager can expedite or unexpedite an order at any time.

Empirical Investigation

An experiment was conducted to explore the FMS system manager's decision processes. Eight students participated in this investigation. These students were all volunteer graduate students from the Computer Integrated Manufacturing Systems (CIMS) program at the Georgia Institute of Technology. Subjects engaged in a total of eleven 60-minute sessions. The first two sessions were training sessions. Session score was used to assess performance

The exploratory nature of this research, coupled with the research's focus on *how* decisions are made (as opposed to the outcome of decisions), indicates that appropriate performance measures are not obvious. An appropriate method needs to be sufficiently generic so that it can be applied to GT-FMS and yet provide an acceptable level of detail to embody the intricacies of the subjects' decision processes.

To fulfill these requirements, it was determined that modeling the subjects' decisions and strategies with the Rasmussen's decision ladder might provide the necessary structure for the research goals (Rasmussen, 1984; Rasmussen, 1986; Rasmussen and Goodstein, 1987). Rasmussen's decision ladder is ".. independent of the specific system and its immediate control requirements," (Rasmussen, 1984, p. 142) so it provided the generalizability necessary for use with the relatively restrictive environment of GT-FMS.

Figure 12 illustrates the basic decision ladder developed by Rasmussen. By beginning at the lower left (i.e., "Activation") of this ladder, and proceeding through each circle (state of knowledge) and rectangle (data processing activity), following the bold arrows, each step of a control decision is addressed. The ladder thus provides a "schematic map of the sequence of information processes involved in a control decision" (Rasmussen, 1984, p. 144).

The data processing activities in the decision ladder are mental reasoning processes which lead directly to the states of knowledge. Applied to GT-FMS, for example, the "Observe" data processing activity might involve scanning a particular display but focusing only on information considered important or relevant. Thus, a systems manager might scan the order status display of Figure 12 and focus on only one aspect of this display. Other information is provided, but the systems manager mentally sorts and places specific priorities on this information concentrating only on the data that s/he determines is "important." This data processing might then lead to a "Set of Observations" state of knowledge which could include "next required process-

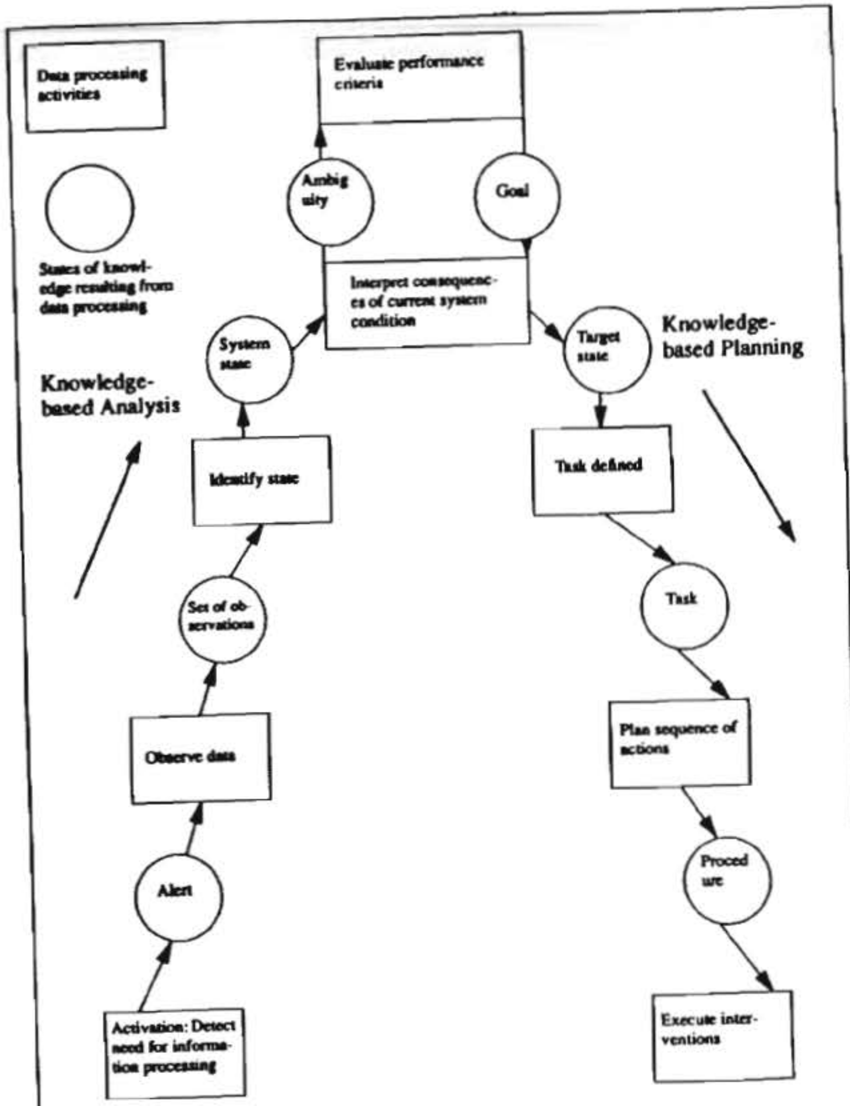

Figure 12. Basic Rasmussen Decision Ladder (Rasmussen, 1986)

ing operations." Thus, the next required processing operations would be the "Set of Observations" that resulted from the systems manager's mental sorting. Information other than next required processing operations is provided, but it is evidently not considered important.

This "Set of Observations" state of knowledge would then be included in the "Identify" data processing activity as the systems manager proceeded along the decision ladder. In this activity, the systems manager might attempt to answer such questions as "what's unusual about the current set of next required processing operations?" or, "what's the underlying reason for the current set of next required processing operations?" Based on the answers to such questions, the systems manager might then define the current system state. This system state would be the systems manager's interpretation of current conditions in the FMS. For instance, continuing the above example, the systems manager could define a system state as "machines 1, 2, 3 and 4 have failed," or "most of the next required processing operations require a small amount of machine time" depending upon the results of his or her "Identify" data processing activity.

This process of alternating data processing activities with resulting states of knowledge continues through the decision ladder in Figure 12 for each control decision through the "Execute" data processing activity, which involves coordinating the desired control actions formulated by the systems manager. For GT-FMS, the available control actions, as previously discussed, are expediting orders of parts or modifying the scheduling algorithm.

If the states of knowledge used by systems managers in controlling GT-FMS can be identified, then their control decisions can be mapped to a decision ladder and evaluated. These states of knowledge might reveal various aspects of the control decisions such as: what information the systems managers considered, what system states did they define, what goals did they develop, what target states did they attempt to achieve, or what strategies did they employ? These aspects of the control decisions might then support recommendations concerning the systems manager's role which could enhance the control function of flexible manufacturing systems.

Identifying the states of knowledge of control decisions and mapping these to Rasmussen's decision ladder was done using verbal protocols from each subject during each data-collecting session. The usefulness of verbal protocols for analyzing decision processes is well documented; see for example Ericsson and Simon (1984). Subjects were required to "talk aloud" and describe their interventions as they occurred. These protocols were completely free-form, with the exception that the subjects were asked to include a description and an intent, as a minimum,

for each control action. Several subjects went significantly beyond this minimum during their interventions.

Decision Ladder Models of FMS System Managers

The verbal protocols were prepared by converting recordings obtained during the experimental sessions into a printed text for each subject. The next step was to use the transcripts from the verbal protocols and the computer data files to construct, for every subject, a decision ladder for each allowed control action. The data log was used to indicate what information was displayed to the subject at the time of the intervention. The verbal protocols and computer output files for every subject were then examined one session at a time. For a given intervention (i.e., either expedite or scheduler modification), each intent stated by a subject was reviewed. Coupled with the computer output data, distinct consistencies among the stated intents from the protocols were recorded. When all of the sessions for a subject were completed, consistencies across sessions were then evaluated. As situations were repeated throughout a session, and across several sessions, the decision processes were broken down and mapped onto the decision ladder. This mapping was done separately for each strategy.

Discussion of the results of the decision ladder analysis is organized by the type of control decision (e.g., either expedite or scheduler modification) and follows the outline of data processing activities, from "Activate" to "Execute," in Rasmussen's decision ladder.

Expedite Decision

For the expedite decision, subjects sought to define a state of knowledge based on the most recent information available for each order. The order status display was the predominant choice of the subjects to provide this information. Six of the subjects used this display exclusively. Two of the subjects reviewed the penalty cost and throughput display regularly, yet, their expedite decisions were also based solely on existing conditions and not on performance history information.

The subjects' approaches were consistent. They distilled system information from the order status display into a state of knowledge described as the current system state. The current system state, once determined by the subject in response to the latest update of the order status display, was the basis for the remainder of the decision process.

Although the subjects generally used the same display for defining the current system state, the information extracted from this display varied significantly among the subjects. For instance, a few subjects used the existence of an expedited order to define a system state. Thus, one of the "Set of Observations" states of knowledge used by these subjects consisted of the presence of an expedited order. Some subjects characterized a system state by the presence of orders which had only the load/unload operation remaining, while others used the elapsed simulation time to define the system state. The elapsed simulation time was used to determine a system state described as "end of session was near." This system state changed the expedite decision for some subjects for a short period of time (this is discussed in more detail later for individual subjects).

Even though the information obtained from the order status display varied, some consistent patterns emerged. For example, all of the subjects used the order due date information to define a system state of either "tight" or "loose" due dates. Most subjects also evaluated whether operations that required long processing times (i.e., "long" operations), particularly operation 5, or operations that required small processing times (i.e., "short" processing operations) were characteristic of the current system state. As the sessions progressed, this evaluation of processing operations became more frequent as subjects concluded that operation 5 was difficult to complete on time. In addition, most subjects defined a system state which had at least one order already late or projected to be late.

Rasmussen (1984, 1986) noted in his evaluation of verbal protocols from a power plant control room that the entire decision-making process described by his decision ladder was only used when the operators were faced with new or unfamiliar situations. In most cases, the operators developed a "sequence for special situations by chaining subroutines of general applicability and using solutions from prior experience... leading to a great repertoire of short-cuts and by-passes in the decision process..." (Rasmussen, 1984, p. 144). Rasmussen describes these subroutines as chained states of knowledge, where the subjects move from one state of knowledge directly to a task or procedure, bypassing portions of the decision process. Chained states of knowledge result in consistent actions based on well-defined system states. However, for GT-FMS, these chained states of knowledge were rare. Subjects generally proceeded through all data processing stages of the decision model, evaluating each system state against the system goals, even when faced with seemingly familiar situations (possible causes of this behavior are discussed later).

Figure 13 illustrates an example of a decision strategy for the expedite control action which uses a chained state of knowledge. In this decision sequence, the subject defined a system state, then, proceeded immediately to a task definition. The task, "expedite the order with one operation remaining," is an immediate consequence of the system state definition. Figure 14 illus-

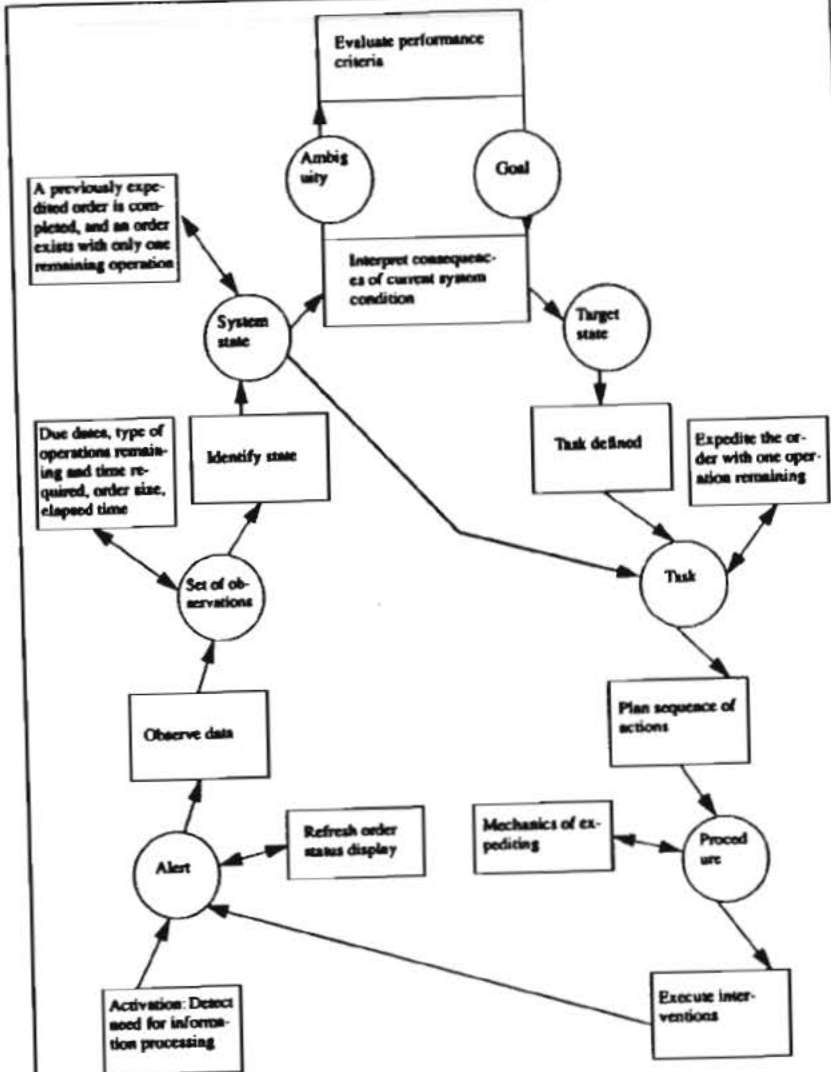

Figure 13. Strategy for Expedite Control Decision, Chained States of Knowledge

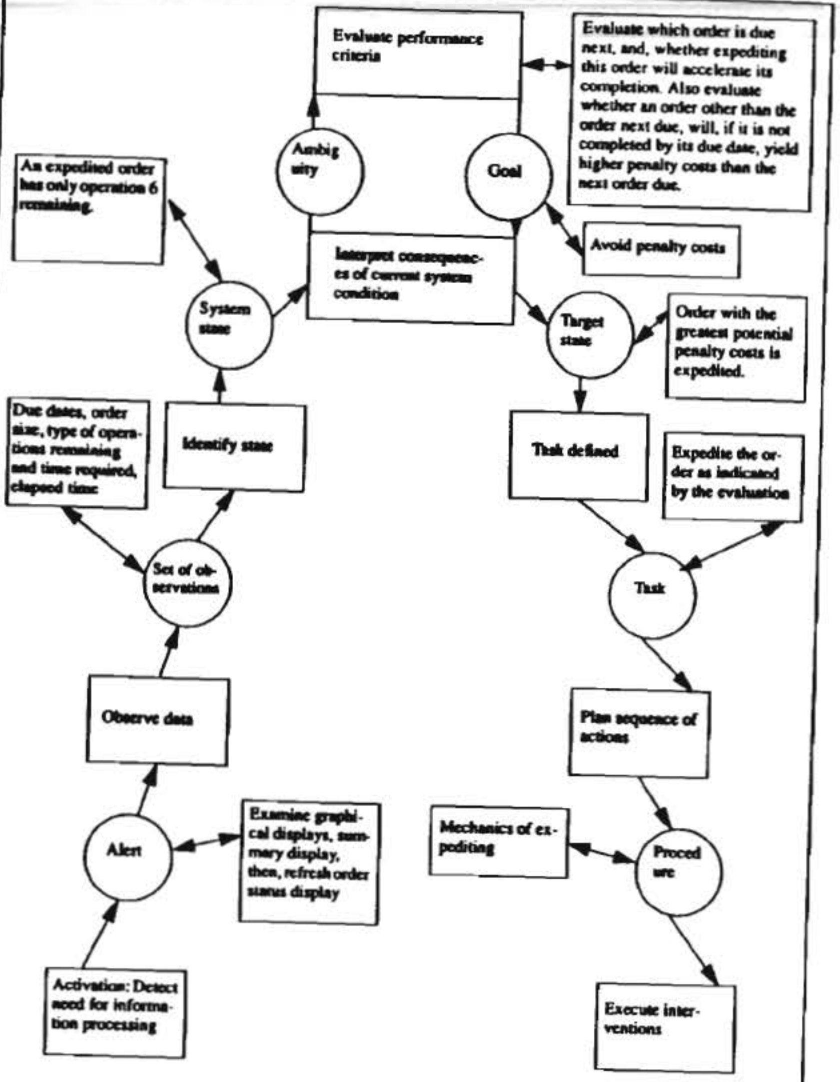

Figure 14. Strategy for Expedite Control Decision

trates a typical expedite decision ladder without chained states of knowledge. In this decision ladder, tasks are defined and implemented as a result of system performance evaluations, goals, and target states.

Although the subjects generally proceeded through the entire decision ladder, some exceptions to this pattern emerged. Figure 15 displays the expedite decision ladder for subject 2. As indicated by Figure 15, in most cases subject 2 did not proceed through the complete evaluation/goal steps of the decision model, but rather reacted, according to chained states of knowledge, to the system states she defined. In addition, the system state defined as "end of session is near" caused several subjects to by-pass the evaluation/goal sequence and to immediately modify their interventions. These subjects stopped evaluating late orders and repeatedly expedited parts remaining with operations that had short processing times once this system state was defined.

Since subjects used the entire realm of data processing activities for expedite interventions, they repeatedly evaluated system performance based on the system states they identified. Again, some consistencies emerged from these evaluations. For example, most subjects evaluated whether any of the orders currently in the FMS would be late. Likewise, determining which of the current orders, if completed past their due date, would yield the highest penalty cost was also common. Subjects focused primarily on order size in making this judgment. Most subjects also evaluated which of the orders had the earliest due date.

Although some system performance evaluations were common, most of the evaluations were unique to each subject. However, even these unique evaluations were consistent in that they evaluated a very detailed level of system performance. For example, subject 4 determined whether recent incoming orders would be delayed at the load/unload station by leaving an order expedited, or, whether the expedited order could still be completed sooner if it remained expedited. If an order was already late, subject 5 evaluated whether another order, with only a small processing time remaining, should be expedited before the late order was done. The detailed evaluations of system performance reveal a high degree of confidence among the subjects in their ability to precisely determine and predict the state of the system.

These detailed evaluations may have been partially responsible for the infrequent occurrences of chained states of knowledge. Rather than react to a system state that appeared "familiar," the subjects attempted to gain a more thorough knowledge of system performance.

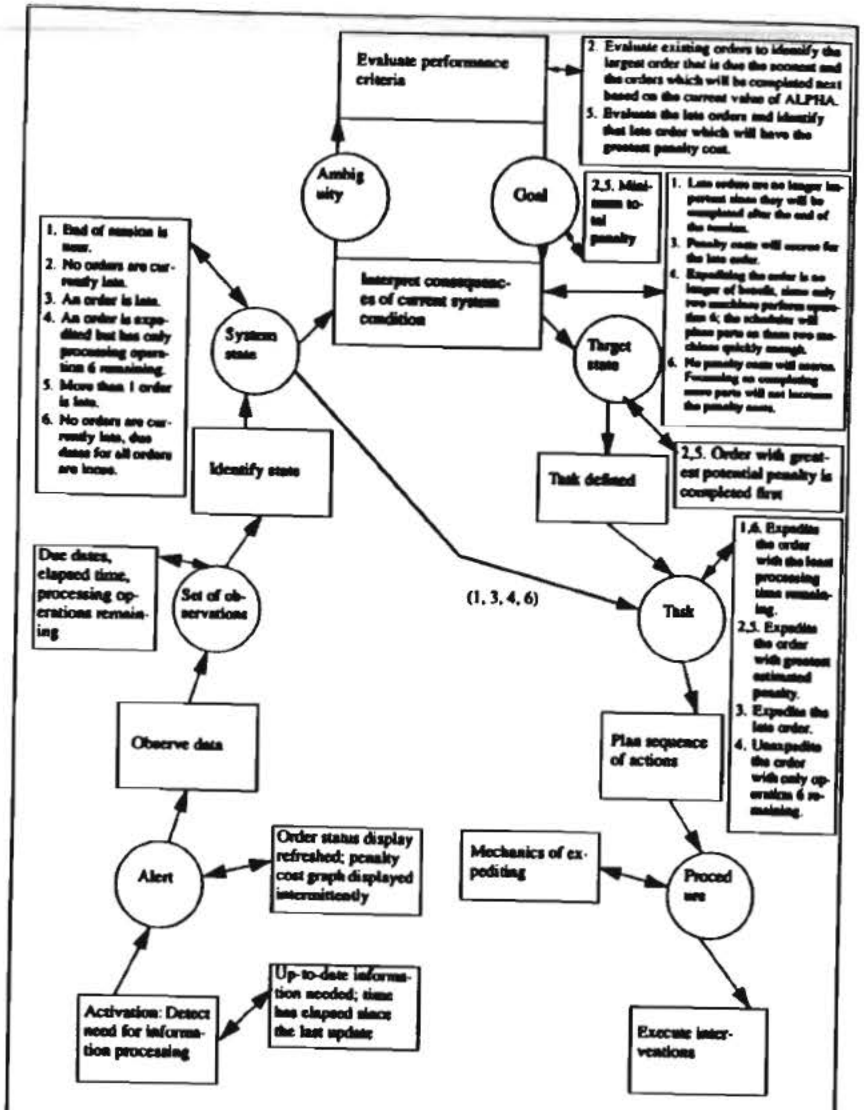

Figure 15. Expedite Decision Ladder, Chained States of Knowledge Dominant

The repeated system performance evaluations also required an on-going comparison of these assessments against system goals. Even though the overall system goal was given to each subject (i.e., maximum profits) during the training, the subjects' system goals were not identical. Subjects formulated their own goals. For example, subject 4's stated goal was to minimize the number of late orders, whereas subjects 5, 6, and 8 identified "avoiding all penalty costs" as their goal. These differences in system goals resulted in varying strategies of operation.

Even though the subjects' goals varied, the primary focus for all subjects involved penalty costs. Completing parts, or throughput, was definitely a secondary goal for the subjects. Half of the subjects did identify "maximizing throughput" as a goal, and this goal influenced the expedite interventions in some cases for these subjects. However, this goal was always secondary to goals concerning penalty costs.

The target states defined by the subjects resulted directly from their evaluations and goals. The target states consistently involved an assessment of processing priority, i.e., deciding which types of orders or which specific order should be processed next, given the system goals, performance and current state. In addition, the target state consistently involved only the current system orders. Subjects did not anticipate the arrival of certain orders or existence of certain conditions when using the expedite intervention.

Like the subjects' system performance evaluations, target states were frequently very detailed rather than general. For example, subject 6, in response to certain system conditions, defined a processing sequence that assumed "long operations occupy the machines while short operations are in the system, and thus prevent processing of the shorter operations." Subject 5 defined as a target state that "operation 6's from nearly completed orders are finished prior to the operation 6's from new orders."

Expediting an order was almost always the task which resulted from proceeding through the data processing activities for the expedite intervention. In fact, most subjects always had an order expedited. However, in response to certain system conditions and performance evaluations, several subjects re-initiated the expedite decision process and maintained the status quo rather than expedite an order. These subjects by-passed the "Formulate" and "Execute" data processing activities and the "Procedure" state of knowledge, proceeding directly to the "Activation" data processing activity in the decision ladder. The decision process was re-initiated by defining a new system state based on more recent information. For subjects 3 and 4, unlike their peers, the status quo was usually no orders expedited. These subjects had a much more limited set of system states and resultant evaluations which concluded with expediting an order.

For all subjects, the decision analysis for the expedite intervention was continuous throughout each session. Once a task was defined and a procedure (if any) implemented, the subjects seemed to instantaneously proceed to the "Alert" state of knowledge - updating the main order status screen and thereby re-initiating the entire decision process. Subjects 2 and 8 also checked their performance history on the penalty cost and throughput display - for varying lengths of time - prior to refreshing the order status display. Overall, the subjects varied considerably in their speed of processing. Processing time was, in all cases, dependent on the significance of the changes in system state since the last expedite decision sequence. More changes generally implied an increased processing time.

Modifying the Scheduler

In terms of the variety of system states defined, performance evaluations, goals and target states, the scheduler modification intervention was much less complex than the expedite intervention. Subjects modified the scheduler (by changing the value of the weighting factor, alpha) much less frequently than they expedited orders, so this decision process itself was initiated less frequently.

As with the expedite intervention, information was primarily obtained from the order status display. Although all subjects viewed the cost summary display intermittently, they did not base their interventions on information from this display. Subjects generally proceeded through the alert stage of the decision process once, for both the expedite decision and the scheduler decision, but then proceeded through the remainder of the process separately for each type of intervention.

Due date status was the one type of information obtained from the order status display that was consistent among the subjects. Again, as with the expedite intervention, most subjects defined a system state based on their assessment of due dates as either "loose" or "tight."

Other than due dates, however, subjects used the order status display to obtain a variety of information. Based on this varying information, many different system states were defined. For example, a few of the subjects defined the beginning of the session as a system state, using the elapsed simulation time information, and modified the weighting factor immediately from its default value. Other subjects focused on the type of operations currently in the system while some associated due dates with the type of operations (e.g., "the long operations have tight due

dates") in defining their system state. Also as with the expedite intervention, several subjects used the elapsed simulation time to define a system state as "the end of session is near" and modified the scheduler to reflect this information.

Unlike the expedite intervention, the subjects modified the scheduler using chained states of knowledge and fewer evaluations of system performance versus goals when deciding to modify the scheduler. For example, Figure 16 illustrates a chained state of knowledge for the scheduler modification decision of subject 3. While subject 3 did interpret the consequences of the current system state, the evaluation/goal and target state steps were by-passed. No projections of system performance, based on the current defined system state were attempted, and a target state relative to system goals was not defined. All of these chained states of knowledge involved associating low values of alpha with "tight" due dates and higher values of alpha with "loose" due dates. These associations either resulted in immediate changes of alpha or a re-initiation of the entire scheduler modification decision process if alpha was already at the desired value.

Even though chained states of knowledge were more common for the scheduler intervention, some evaluations did occur frequently. For example, all of the subjects evaluated the impact of raising alpha to 1.0 or 0.9 on system throughput, and most of the subjects evaluated whether they should test the sensitivity of the system to changes in the weighting factor, alpha. Also, subjects' evaluations, as with the expedite intervention, were generally very detailed. For example, subject 2 evaluated whether raising alpha to prevent parts with long operation time from being loaded onto the machines would increase penalty costs by making the parts with long operations late or would increase throughput by completing additional parts.

When evaluations of system performance were made, the subjects once again had to incorporate system goals to guide their interventions. These goals, consistent with the expedite intervention, were not identical to the overall goal presented in the training and varied among the subjects. For instance, three of the subjects identified "high" throughput as a goal, while other subjects identified "avoiding penalty costs" as a system goal.

Post-evaluation interpretations and target states focused on processing priority, but, unlike the expedite intervention, this focus was on processing priority to specific groups or classes of parts. The most common focus was on a processing priority based on due dates and processing times.

The task defined was always either to change alpha, based on an evaluation or on chained states of knowledge, or to maintain the status quo if a system state was not identified which triggered

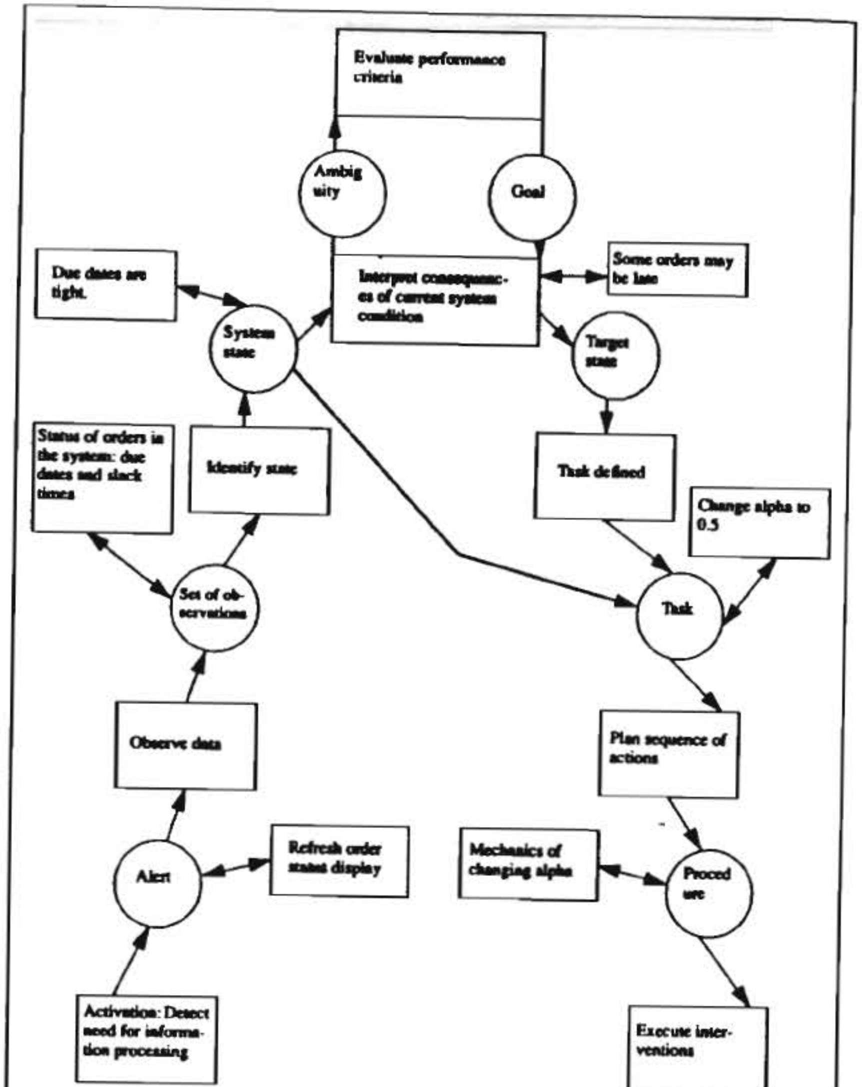

Figure 16. Scheduler Decision Ladder, Chained States of Knowledge

a scheduler intervention. Subjects generally mirrored the expedite intervention in that they went from the procedure state of knowledge directly to an alert condition and re-initiated the decision sequence by updating the order status display.

Overall, modifications to the schèduler occurred much less frequently than expediting, and most subjects changed alpha less frequently as they completed more sessions. Generally, these changes involved placing alpha at values of 0.0 and 0.1 when a low value was indicated or placing alpha at 0.9 or 1.0 when a high value was indicated. The subjects' detailed evaluations of system performance revealed a knowledge of the scheduling algorithm and its impact on the FMS, and the subjects reinforced this knowledge by continuing to test the sensitivity of the system to changes in alpha.

Summary

The goal of this research was to gain further insight into human decision making within an FMS environment. In the environment simulated by GT-FMS, human systems managers are allowed to intervene in the FMS in two ways: expediting an order, which gives an individual order processing priority, or modifying the scheduling algorithm, which gives groups of parts processing priority. Evaluating the decisions made by the systems managers in implementing these two interventions has provided some insight for FMS supervisory control systems.

First, the subjects in this experiment interacted with the FMS on a very detailed level. Even though their role in this simulation placed them at a higher level of control, providing them primarily with summary information and performance history, the subjects evidently needed and wanted more detailed information on system performance. Performance history and trend information were not factors in their decisions. Further, even though they were not provided with detailed system status information, they were still able to make very detailed evaluations of system performance and incorporate these evaluations into effective control strategies. Chained states of knowledge, where the human reacts to certain standard system states, were not as common as might have been expected based on the limits of the subjects' interventions and Rasmussen's (1984, 1986) results. The subjects continued to prefer thorough evaluations, incorporating as much evidence as they could obtain, versus reacting to standard system states based on the summary information. They also seemed more concerned with the situation at hand, and how they could best influence this situation, rather than incorporating or trading-off their current decisions as part of a long term performance strategy. Evidently, while humans may be effective as part of a higher or aggregate level FMS control system, they still prefer having access to detailed knowledge of lower level system components.

In addition, the subjects were able to understand and control the scheduling algorithm, even though, again, they were primarily given system summary information. They continued to test the sensitivity of the scheduler when they felt this testing did not conflict with current system goals. This testing occurred throughout the sessions, indicating that the subjects were continuing to learn more about the dynamics of the scheduler as the sessions progressed. Still, in 10 total sessions, the subjects seemed to use the scheduling modification intervention effectively. Additional sessions, or, in the case of an actual FMS system, perhaps months of training, would probably increase the systems manager's understanding of scheduling dynamics even more. In actual FMS installations, the tendency may be to exclude the human from the operation of the scheduler, yet the results of this research indicate that this may not be the best approach.

Flexible manufacturing is a philosophy which can greatly enhance the overall productivity of small-lot or batch manufacturers. The increasing complexity of the control systems required by modern FMSs has resulted in numerous research efforts which address control structure design. While no single design is "optimal" for every, or even most, FMS installations, effective control system designs are often characterized by a hierarchy of several information-sharing control levels which incorporate many manufacturing decision-making functions.

Since human judgment is critical to manufacturing decision-making, human intervention is a component of many control system designs. The implementation of human intervention, however, varies significantly among FMS control structures, and often occurs on an ad hoc basis. Previous supervisory control research has indicated that overall FMS performance can be enhanced if the ad hoc nature of human intervention is removed. Thus, human intervention is likely to remain as an important aspect of FMS control policies.

As FMS control systems become more "intelligent," the role of the human in the control structure will also evolve. An FMS may become more of a tool for the human who controls it, with the human responsible for achieving system goals. Thus, defining an appropriate role for the human as a function of the FMS objectives may be critical in the design of the FMS control structure. This role should both respect the limitations of the human and exploit the human's inherent skills. Knowledge of the decision processes used by humans in an FMS environment can help define this role.

This investigation evaluates the decision processes of humans in an FMS environment. The experimental results support making humans an integral part of the manufacturing control process, since an intricate knowledge of the system state and system sensitivity were crucial to human decision-making in GT-FMS. Further, even when the goals for FMS operation were specified in terms of a single aggregate measure, viz. the overall profit, humans in this experiment used detailed system status information, rather than summary information or system performance history, as the basis of their control decisions.

Distributed Decision Making in Complex Systems[3]

Introduction

The control of complex, dynamic systems often requires multiple human operators to be integrated into a command-and-control system (Athans, 1987; Tenney and Sandell, 1981). Thus, an organization is required to provide a way by which operators, working on different parts of the total control task, may coordinate their activities to produce a unified team effort.

The design of command-and-control systems is a particularly difficult problem. First, the decision agents, both human decision makers, computer-based algorithms, and decision aids, are often geographically dispersed due to environmental and survivability reasons. Second, the combination of advances in sensor technology and increases in information transmission capabilities, can generate much more data about the controlled system and its operating environment than the control system can process into useful information.

The nature of the controlled system itself causes difficulties. The controlled system usually consists of multiple subsystems which have access to different information. These subsystems are making their own local decisions but they must work together to accomplish a system-wide goal. Therefore, each subsystem may be operating with limited knowledge about the remainder of the system.

In short, designing the organizational structure of a team supported by complex decision support systems is a distributed decision making problem. Thus, better designs for command-and-control systems depend on understanding distributed decision making.

This research investigated the design of distributed decision making architectures and organizational forms for the control of complex, dynamic systems. In particular, the research addressed three significant factors affecting the performance and behavior of teams of decision makers in distributed command-and-control systems:

(1) Organizational structure of the team.

3. Armstrong, 1990; Armstrong and Mitchell, 1986.

(2) System load.

(3) Time delays in automated status reporting and computer-based, message communications.

Developing appropriate organizational structures and decision aids for command-and-control systems must be based on an understanding of the nature and dynamics of distributed decision making. Both classical organization theory (Kickert, 1980; Levis, 1984) and traditional human-machine systems research (Kelley, 1968) have used a single decision maker paradigm as the conceptual basis for understanding and designing command-and-control systems. Actual command-and-control systems, however, typically have multiple decision makers who function in a distributed environment. The decision makers normally interact via computer and communications networks. Successful system control performance depends, in large measure, on the effectiveness and efficiency of the interactions among the decision makers. Establishing who should communicate what information, to whom, and when are critical design issues. Consequently, it is important to investigate specifically the effects of a distributed environment and of the structure of the decision making team on overall system performance.

This research investigated team performance in distributed decision making, in particular how teams, organized in different structures, accomplished coordination and made decisions in a command-and-control environment. An experimental domain, C2-GT-FMS (Command-and-Control of Georgia Tech Flexible Manufacturing System) was constructed to support the investigation. C2-GT-FMS is a real-time interactive simulation of a multi-operator, multi-cell flexible manufacturing system. To investigate structurally induced differences in team performance, two very different organizational forms, hierarchy and a heterarchy, were applied to the problem of integrating FMS (Flexible Manufacturing System) cells into one multi-cell system.

C2-GT-FMS

C2-GT-FMS (Command-and-Control of GT-FMS) is based on GT-FMS (Ammons et al., 1988; Dunkler et al., 1988; Armstrong and Mitchell, 1986) with extensions that include properties necessary to investigate distributed decision making in a simulated command-and-control (C2) environment.

C2-GT-FMS modeled the salient features of many real systems that C2 systems control:

(1) large-scale, distributed system;

(2) discrete-event, dynamic system;

(3) stochastically arriving tasks with time pressure whose relative importance may change;

(4) stochastically occurring events that degrade the availability of resources;

(5) reconfigurable system by reassigning tasks or redistributing resources among subsystems.

C2-GT-FMS simulated a multi-cell flexible manufacturing system. The function of C2-GT-FMS was to fabricate armor plates for assembly onto Army combat vehicles such as the Abrams Tank or Bradley Fighting Vehicle. The primary task of the two C2-GT-FMS command-and-control systems was to supervise the movement and processing of batches of items through the fabrication process in response to changing priorities to meet specified due dates and times. Operators in C2-GT-FMS communicated via a computer-based message system.

Hierarchical versus Heterarchical Organizational Structures

To regulate C2-GT-FMS, two command-and-control systems were developed based on two different organizational structures. One structure, C2-GT-FMS1, represents the typical structure for a hierarchical decision making team. It consists of a supervisor and two cell operator subordinates. The other structure, C2-GT-FMS2, represents a configuration for a typical heterarchical team. It consists of three cell operators who functioned as a team without an explicit supervisory structure. Figure 17a and 17b depict the two systems.

C2-GT-FMS1: The Hierarchy

The hierarchical system is a two-cell FMS (Figure 17a). It consists of a supervisor responsible for overall system (i.e., batch performance) and two cell operators who are responsible for cell level performance. In the hierarchical system each cell consists of nine machines, in three banks of three types. The supervisor coordinates cell resources (i.e., repair robots) and system tasks (i.e., batches of parts with due dates) to maximize system performance. Cell operators can request the supervisor to assist with cell problems by transferring resources or system tasks.

C2-GT-FMS2: The Heterarchy

The heterarchical system is a three-cell FMS. It also consists of 18 machines, but machines are equally divided among three cells instead of two cells. This means that each cell in the heterarchy has three banks of machines with two rather than three machines of each type. This structure is illustrated in Figure 17b. This three-cell system has three cell operators who manage their own cell and cooperate with the other two cell operators to enhance overall system performance.

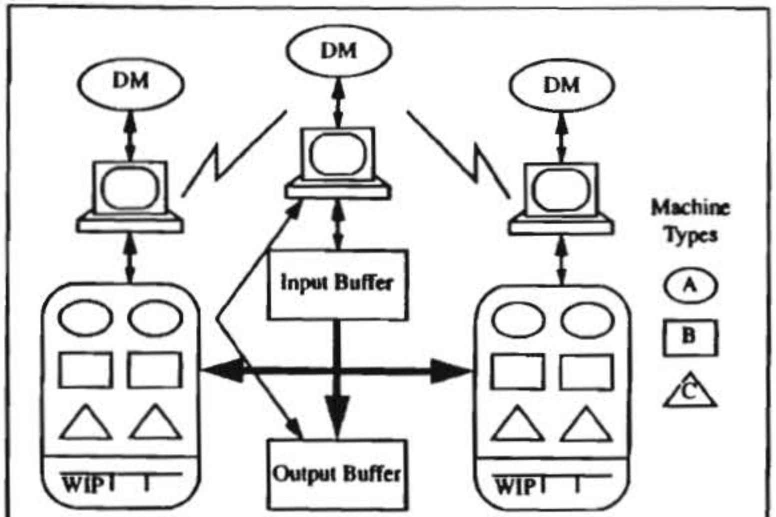

Figure 17a. C2-GT-FMS1 Hierarchical System Structure

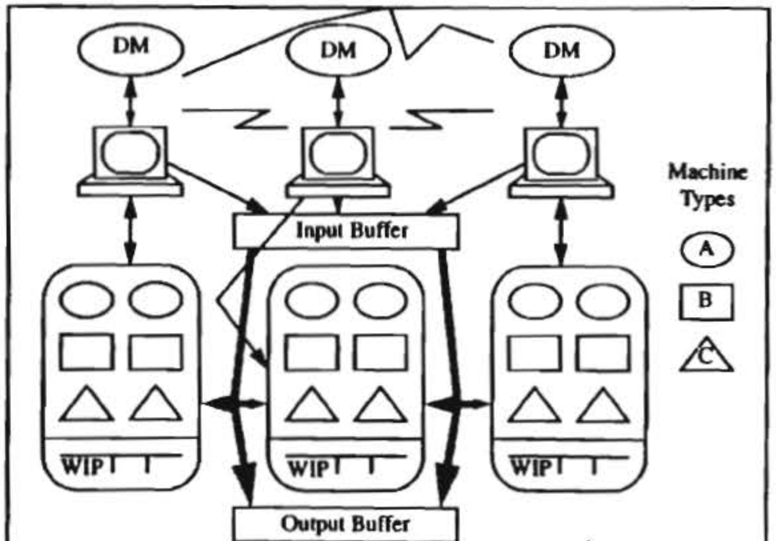

Figure 17b. C2-GT-FMS2 Heterarchical System Structure

Both the hierarchical and the heterarchical systems process four part types. Parts typically need to visit multiple machines and machines of different types. Thus, machine failures sometimes require, to ensure timely part completion, the transfer of parts or robots to another cell. The next section explains the role of the operator in each system.

The goals and functions for decision makers in the two different organizational structures are carefully designed to insure that both C2-GT-FMS1 and C2-GT-FMS2 have the same operational capabilities. The team or organizational goal for both structures is to process batches, sets of parts (similar to 'orders' in the Hettenbach study), through the system to maximize the readiness of each batch. The concept of readiness motivates the kind of real-time trade-off decision making that occurs in actual operational situations.

Batch readiness is based on the proportion of late parts in the batch: the more parts that are late, the lower the readiness. Also, the metric for readiness has a time penalty for the average time late of the overdue parts in a batch: the higher the average time late, the greater the time penalty and the lower the readiness. Batch readiness is computed using the size of each batch (the number of parts in the batch), and the average time late of the expected or actual number of late parts.

To determine team performance in terms of batch readiness, a separate readiness rating was calculated for each batch, both completed and uncompleted batches. Then, an average readiness for all batches is computed. Average session batch readiness is used as a measure of team performance. Detailed descriptions and computations for batch readiness is given in Armstrong (1990).

Experiment

In the experimental investigation, eight three-person teams operated two command-and-control structures in different operating environments. The hypothesis examined was that the cooperative, i.e., heterarchical, structure can perform better in more difficult operating environments while the hierarchical structure is more efficient in lower levels of environmental complexity.

The independent variables selected for this experiment were structure (the distributed decision making architecture of the team), load (the number of tasks arriving per unit time to the team), and delay (both a time delay for information exchange between machines and a time delay for message communications between decision makers).

There was a range of dependent measures: (1) average batch readiness score; (2) number of parts completed on-time; (3) number of parts completed late; (4) percentage of robot utilization; (5) number of messages sent; (6) number of part transfers; (7) number of robot redistributions; (8) number of message errors; (9) number of part transfer errors; and (10) number of robot redistribution errors. The first four measures assess how well the team performed the primary task of controlling C2-GT-FMS. The next three measures explain how performance was attained. The last three measures concern the errors that were made in coordinating the control of C2-GT-FMS.

Results

Results obtained from analyses of the dependent variables were divided into three categories. The first category examined the performance or output of the teams in the two structures. The next category studied the coordination process by which team performance was achieved. Third, errors in that process were analyzed. This chapter only describes the high level results. Detailed analyses for all measures can be found in Armstrong (1990).

Organizational structure significantly affected team performance. Results indicated that the hierarchy's performance was characterized by better speed (moving more parts through the system) while the heterarchy's performance was characterized by better precision (prioritizing the movement of parts through the system so that fewer parts were late).

Based on the four significant interactions involving structure with load or delay, there was much evidence to support the belief that designers must consider the limitations and abilities of different structure types to cope with load and delay conditions. For example, as these results confirmed, a relatively short delay can overload the information processing ability of the heterarchical structure. The heterarchy reached an information processing limit before the hierarchy since heterarchical operators had to process information from two levels: the local, individual level and the system-wide, team level.

The data overwhelmingly showed that load was a highly significant factor which affected the performance of both structures but in different ways. The hierarchy had difficulty when load was high because a "bottleneck" formed at the system level. The bottleneck occurred at the supervisor's position. Team-level situations rapidly queued up. Supervisors could not finish processing all the system-wide information to recognize a particular team situation and then decide, based on an overall evaluation of each cell, what actions to direct, before another team situation developed.

High load caused a different phenomena in the heterarchy. When load was high, team-level situations were often ignored by the heterarchy due to a "tunnel vision" effect. During high load, operators became too focused on local information, which was changing rapidly, and ignored system-level information. Thus, operators in the heterarchy became over-involved in individual-level tasks to the detriment of team-level tasks.

Summary

This research investigated how teams, organized in different structures, accomplished coordination and made decisions in a command-and-control environment. Experimental results suggested the need for flexible, reconfigurable command-and-control structures based on the characteristics of the decision making environment.

This research is important in several respects. First it is one of very few empirical attempts to address the critical issue of multiple decision makers in a complex, high-fidelity command-and-control situation. Second, the computer-based nature of the communication system provides important insight on the effects of these message systems as vehicles for communication in distributed command-and-control systems. Third, the experiment investigation provided important insights into the nature of distributed decision making and the related decision support needs of teams of distributed decision makers.

Conclusion

This chapter summarized the results of three recently conducted research programs addressing human supervisory control in predominantly automated manufacturing process. The three approaches provide complementary perspectives. The first project, (Benson, 1989), explored the application of state-of-the-art human-computer technology to enhance the operator interface for an FMS cell level controller. The results support the careful exploration of emerging technology to enhance the efficiency and effectiveness of human supervisory controllers in manufacturing processes.

The second project, Hettenbach (1989), was more speculative in nature and examined the decision processes for FMS *system managers*, as opposed to cell supervisors. The results suggest the need for more extensive research examining the decision making levels and processes of human operators in predominantly automated manufacturing systems.

The third project, Armstrong (1990), the most extensive study to date, examined the role of *teams* of people coordinating multi-operator, multi-cell, distributed manufacturing processes.

The results suggest careful examination of the organizational structure, specifically the need for flexibility, in the design of distributed and complex systems.

Taken together, the three research efforts in this chapter together with the previous work carried out in the context of GT-FMS (e.g., Mitchell, 1991), provide some insight into the design of manufacturing supervisory control systems. Conceptually, human supervisory control provides an effective alternative to the 'lights out manufacturing' and suggests productive paths for continuing research, development, and implementation.

References

Ammons, J. C., C. B. Lofgren, and L. F. McGinnis (1985). "A Large Scale Loading Problem," *Annals of OR*, vol. 2, 319-332.

Ammons, J. C., T. Govindaraj, and C. M. Mitchell (1988). "A Supervisory Control Paradigm for Real Time Control of Flexible Manufacturing Systems," *Annals of OR*, 313-335.

Armstrong, J.M. and C. M. Mitchell (1986). "Organizational Performance in Supervisory Control of a Flexible Manufacturing System," *Proceedings of 1986 IEEE International Conference on Syst., Man. & Cybern.*, October, 1437-1441.

Armstrong, J. E. (1990). Distributed Decision Making for Command-and-Control of Complex Dynamic Systems. Ph.D. Thesis, Center for Human-Machine Systems Research, Georgia Institute of Technology, Atlanta, GA, Technical Report 90-1.

Astrom, K. J. (1985). "Process Control-Past, Present and Future," *IEEE Control System Magazine*, August, 3-10.

Astrom, K. J., J. J. Anton, and K. E. Arzen (1986). "Expert Control," *Automatica*, Vol. 22, No. 3, May, 277-286.

Athans, M. (1987). "Command and Control (C2) Theory: A Challenge to Control Science," *IEEE Transactions on Automatic Control*, vol. AC-32, no. 4, Apr.

Benson, C. (1989). The Use of Single-Page, Direct-Manipulation Interfaces in Real-Time Supervisory Control Systems. M. S. Thesis, Center for Human-Machine Systems Research, Georgia Institute of Technology, Atlanta, GA, Rep. 89-3.

Benson, C. R., T. Govindaraj, C. M. Mitchell, and S. P. Krosner (1991). "Effectiveness of Direct Manipulation Interaction in the Supervisory Control of Flexible Manufacturing Systems," *Information and Decision Technology*, in press.

Cohen, S. S. and J. Zysman (1988). "Manufacturing Innovation and American Industrial Competitiveness," *Science*, Vol. 239, 1110-1115.

Dunkler, O. (1986). Human Supervised Scheduling and Control in Flexible Manufacturing Systems, M. S. Thesis, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA.

Dunkler, O., C. M. Mitchell, T. Govindaraj, and J. C. Ammons, (1988). "The Effectiveness of Supervisory Control Strategies in Scheduling Flexible Manufacturing Systems," *IEEE Trans. Syst., Man, and Cybern.*, 223-237.

Eagle, B. (1987). "The Design and Application of the Weighted Operation Priority Index Dispatching Rule for Discrete-Interval Human Supervisory Control of a Flexible Manufacturing System," Proposal to the Faculty and CIMS Committee, School of Industrial and Systems Engineering, Georgia Institute of Technology, June.

Ericsson, K. A. and H. A. Simon (1984). *Protocol Analysis, Verbal Reports as Data*, MIT Press.

Fox, M. S., and S. F. Smith (1984), "ISIS: A Knowledge-Based System for Factory Scheduling," *Expert Systems*, Vol. 1, No. 1,25-49, July.

Han, M. H., and L. F. McGinnis (1986). "Due Dates in Manufacturing Shops: Unweighted Case," Working Paper, Material Handling Research Center, Georgia Institute of Technology, November.

Hettenbach, D. A. (1989). An Investigation of Decision Making in Supervisory Control of a Flexible Manufacturing System. Technical Report 89-2, Center for Human-Machine Systems Research, Georgia Institute of Technology, March.

Hettenbach, D. A., C. M. Mitchell, and T. Govindaraj (1991). Decision making in supervisory control of a flexible manufacturing system *Information and Decision Technology*, in press.

Jaikumar, R. (1986). Postindustrial manufacturing. *Harvard Business Review*, November-December, 69-76.

Jones, C. V. and W. L. Maxwell (1986). "A System for Manufacturing Scheduling with Interactive Computer Graphics," *IIE Transactions*, September, 298-303.

Kelley, C. R. (1968). *Manual and Automatic Control: A Theory of Manual Control and Its Application to Manual and to Automatic Systems*. John Wiley & Sons, Inc., New York, 1968.

Kempf, K., C. Le Pape, S. F. Smith, and B. R. Fox (1991). "Issues in the Design of AI-Based Schedulers: A workshop report. *AI Magazine*, January, 11-5, 37-46.

Kickert, W. J. M. (1980). *Organization of Decision Making: A Systems Theoretical Approach*, North Holland, New York.

Krosner, S. P., C. M. Mitchell, B. C. Tipton (1987)."The Role of the Human Operator in Control of an FMS with Random Operation Times," *Proc. of 1987 IEEE International Conference on Systems, Man and Cybernetics*, Alexandria, VA., October, 671-674.

Krugman, P. and G. Hatsopoulos, (1987). "The Problem of U.S. competitiveness in Manufacturing," *N. Engl. Econ. Rev.*, January/February, 18.

Levis, A. H. (1984). "Information Processing and Decision Making Organizations: A Mathematical Description," *Large Scale Systems*, North Holland, vol. 7, 151-163.

Marcus, Aaron (1987). "Graphic Design for Computer Graphics," in Baecker, R.M. and Buxton, W. A. S., *Readings in Human Computer Interaction: A Multidisciplinary Approach*, Los Altos, CA: Morgan Kaufmann, 320-332.

Miller, R. K. (1985). "Artificial Intelligence: A Tool for Manufacturing," *Manufacturing Engineering*, April, 56-62.

Mitchell, C. M. (1991). Supervisory control: Philosophical considerations in manufacturing systems. *Concise Encyclopedia of Information Processing in Systems and Organizations*, Andrew P. Sage (ed.), Pergamon Press, 439-448.

Rasmussen, J. (1984). "Strategies of State Identification and Diagnosis in Supervisory Control Tasks, and Design of Computer Based Support Systems," *Advances in Man-Machine Systems Research*, Rouse, W. B., (ed.), JAI Press, Inc., Vol. 1., 139-194.

Rasmussen, J. (1986). *Information Processing and Human-Machine Interaction*, New York: North-Holland.

Rasmussen, J., and L. P. Goodstein (1987). "Decision Support in Supervisory Control of High Risk Industrial Systems, *Automatica*, Vol. 23, No. 5, 663-671.

Scott, B. R., and G. Lodge (eds.) (1985). *U.S. Competitiveness in the World Economy*, Boston: Harvard Business School Press.

SIGCHI (1984 through 1991). *Proceedings of the Association of Computing Machinery (ACM) Special Interest Group on Human-Computer Interaction (SIGCHI)*.

Smith, S.F., M.S. Fox, and P. S. Ow (1986). "Constructing and Maintaining Detailed Production Plans: Investigations into the Development of the Knowledge-Based Factory Scheduling System," *AI Magazine*, vol. 7, no. 4, Fall, 45-61.

Tenney, R.R. and N. R. Sandell (1981a). "Structures for Distributed Decisionmaking," *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-11, no. 8, 517-526, Aug.

Tenney, R.R. and N. R. Sandell (1981b). "Strategies for Distributed Decisionmaking," *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-11, no. 8, 527-538, Aug.

Young, R. E. and M. A. Rossi (1988). "Toward Knowledge-based Control of Flexible Manufacturing Systems," *IIE Transactions*, vol. 20, no. 1, March, 36-43.

TECHNIQUES FOR THE INTEGRATION OF MANUFACTURING SYSTEMS

G. Harhalakis, C.P. Lin, & L. Mark

Systems Research Center
University of Maryland
College Park, Maryland 20742

1 INTRODUCTION

Current research in the area of manufacturing systems is quite intensive in dealing with product and process design, production planning, and job execution. However, the design of such systems has been traditionally made in a functional fashion that emphasized "local" solutions, using closed and self-contained architectures. This, together with the use of heterogeneous databases and incompatible computer operating systems have led to "islands of automation" (figure 1) of various engineering application systems. Naturally, these systems suffer from data inconsistencies and lack of control of functional interactions between them.

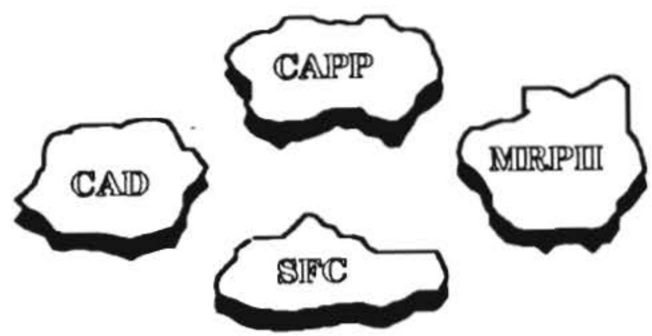

Figure 1: Islands of Automation

Current and future trends for the use of computers in manufacturing include the control and the integration of information flow for production operations into a computer-controlled factory management system. Various research

CONTROL AND DYNAMIC SYSTEMS VOL. 46

Design for Manufacturability

The modes and designs of product assembly units are many and varied. In each case, whether dealing with automated assembly, or those involving human workers, it is desirable to optimize the design in order to maximize productivity. This results in increased operator efficiency and well being through the creation of acceptable workloads, interesting and meaningful work tasks, and ambient work environments.

Design for Manufacturability deals with these parameters in-depth and proposes a systems approach to the design-manufacturing continuum. This aggregates the cost and benefits of labour, material, machines, and ergonomics through design analysis. A wide range of topics is covered, including ease of manual/automatic assembly; biomechanical, cognitive, and perceptual workload; task allocation, job satisfaction; socio-technical systems design; computer support; and design evaluation. The important issue of design, productivity, and employment in Developing Countries is also discussed.

This book will be essential reading for human factors engineers, manufacturing researchers, product design professionals, and for all those involved with interactions between product and process.

Martin Helander is Professor at the Department of Industrial Engineering, State University of New York at Buffalo, USA. He is a widely published author and is presently the USA Editor for the journal, *Ergonomics*.

Mitsuo Nagamachi is Professor at the Department of Industrial Engineering, Hiroshima University, Japan

ISBN 0-7484-0009-5

9 780748 400096

Taylor & Francis
London • Washington DC

Design for Manufacturability

M. Helander/M. Nagamachi

Taylor & Francis

Design for Manufacturability

A Systems Approach to Concurrent Engineering and Ergonomics

Edited by
M. Helander & M. Nagamachi
with
The International Ergonomics Association

Taylor & Francis

To Jens Rasmussen

Design for Manufacturability
A Systems Approach to Concurrent Engineering and Ergonomics

Edited by
M. Helander
Department of Industrial Engineering
State University of New York at Buffalo
and
M. Nagamachi
Department of Industrial Engineering
Hiroshima University, Japan

Published in collaboration with the
International Ergonomics Association

Taylor & Francis
London • Washington, DC
1992

The future development of integrated manufacturing enterprises requires a cross-disciplinary understanding of both engineering knowledge and social aspects with the organization.

References

Alting, L. and Zhang, H., 1990, Trends in the development of the future integrated manufacturing systems, ASME Manufacturing International 90, Atlanta, Georgia, USA, March.

Ammons, J.C. and Govindaraj, T., 1988, Decision models for aiding FMS scheduling and control, *IEEE Transactions on Man, Systems and Cybernetics*, **18**(5), 744–56.

Aoki, M., 1988, A new paradigm of work organization: the Japanese experience. WP 36, World Institute for Development Economics Research of the United Nations University.

Drucker, P.F., 1988, The coming of the new organization, *Harvard Business Review*, **January-February**, 45–53.

Goldratt, E.M. and Cox, J., 1984, The Race, North River Press.

Gupta, D. and Buzacott, J.A., 1989, A framework for understanding flexibility of manufacturing systems, *Journal of Manufacturing Systems*, **8**(2), 89–97.

Hartland-Swann, J.R., 1988, MRP or JIT: which is best, Ingersoll Engineers, UK.

ICAM, 1981, *Integrated Computer-Aided Manufacturing Volume V*, Ohio: Wright Patterson Air Force Base.

Kaplan, R.S., 1989, Management accounting for advanced technological environments, *Science*, **245**, 22–6.

Kaplan, R.S., 1990, The four-stage model of cost systems design, *Management Accounting*, **February**, 22–6.

PROMODELPC; User's Manual, Production Modeling Corporation of Utah, 1834 South State Orem, UT 84058.

Rasmussen, J., 1990, Use of field studies for design of workstations for integrated manufacturing systems, in International Conference on Human Factors in Design for Manufacturability and Process Planning. IEA, 1990.

Rasmussen, J., Pejtersen, A.M. and Schmidt, K., 1990, Taxonomy for cognitive work analysis, Risø National Laboratory, RISØ-M-2871, MOHAWK Workshop, May 1990.

Savage, C.M., 1988, Fifth Generation Management for Fifth Generation Technology, SME Blue Book Series, Dearborn, MI: Society of Manufacturing Engineers.

Yoshikawa, H., 1988, Social aspects of the factory of the future, Round Table at the 38th CIRP General Assembly.

Chapter 23

Analysis and aiding the human operator in electronics assembly

Sally M. Cohen, Christine M. Mitchell and T. Govindaraj

Center for Human-Machine Systems Research,
School of Industrial and Systems Engineering,
Georgia Institute of Technology, Atlanta, Georgia 30332-0205, USA

Abstract. This chapter describes the use of the operator function model (OFM) to represent troubleshooting of printed circuit board assemblies. The model was derived from empirical data based on observations and concurrent protocols of troubleshooters. The 'raw' data were converted into cases. Based on analysis of the troubleshooting cases, an operator function model of troubleshooting was proposed. Model validation entails comparison of model predicted troubleshooting with actual operator troubleshooting. The model is potentially used for understanding the troubleshooting process and providing the knowledge required by an on-line troubleshooting decision support system or a tutoring system.

Introduction

The role of the human operator in the supervisory control of advanced manufacturing processes is uncertain. Automation technology for electronics assembly changes the role of the human operator from direct manual intervention to monitoring and fault management of a predominantly automated process. The new manufacturing philosophy includes just-in-time (JIT) inventory management, five-sigma quality, and immediate attention to problems in the assembly process. These goals require operators to have flexible decision-making responsibility and effective decision aids. Unfortunately, the rapid innovations in manufacturing technology leave open the questions of what precisely the operators will do, what decisions they will have to make, and what decision support is needed.

This chapter describes an on-going study of human troubleshooting in a printed circuit board assembly process. Our interest is modelling the role of the plant floor operators and, based on this model, to design effective decision support structures. Before describing the plant itself together with the data and proposed model, some comments are needed about the research approach and model structure.

The research methods

Research in advanced manufacturing systems is badly needed, but is accompanied by the question of how to go about undertaking it. There are several research efforts focused on the role of the human operator in advanced manufacturing. Using a flexible manufacturing system (FMS) simulator (GT-FMS), Georgia Tech's Center for Human-Machine Systems Research explored a variety of issues related to the control of an FMS process: effectiveness of supervisory control (Dunkler *et al.*, 1988), effectiveness of the human as systems manager rather than controller (Hettenbach *et al.*, 1991), the use of direct manipulation for FMS operator work stations (Benson *et al.*, 1992), organization of teams of FMS operators in hierarchies versus heterarchies (Armstrong, 1990), and model-based design of the operator's work station (Krosner *et al.*, 1990). The problem with this research was that in attempting to generalize it or apply it to actual manufacturing environments, there was a great deal of uncertainty about how or what results transferred. Although GT-FMS is a flexible laboratory environment and can be configured to resemble actual or planned facilities, it contains assumptions that are not met in actual applications. Many manufacturing design practitioners felt that these assumptions constituted major flaws in the generalizability of the GT-FMS research. Moreover, it was not clear how to 'fix' GT-FMS or create a new laboratory domain that could overcome the limitations.

Our concern about generalizabilty of laboratory research is echoed by other researchers (e.g., Klein *et al.*, in press; Woods, in press). Researchers interested in naturalistic decision-making question the extent to which large portions of laboratory research generalized at all (e.g., Klein *et al.*, in press). In an extensive programme of research on human problem-solving in fault diagnosis tasks, Rouse and his colleagues (e.g., Rouse and Hunt, 1984) conclude that human problem-solving tends to be context-dominated.

Given the problems with generalizability of laboratory research, our group decided to attempt to use an actual manufacturing facility to gather data and to formulate models and decision aids for human operators. Using case study methods and building models based on data from actual operations, the hope is that we will be able to generalize and that models and methods derived from the specific, but real, will have more generalizability than those derived from generalized, but 'unreal', laboratory tasks. This process itself is clearly a research question. One goal is to learn if and how we can transfer domain-specific insights, models, and aids to domain-general insights, i.e., generalize the application-specific results to a class of similar applications.

Models

As engineers and designers we use models to understand and, given that understanding, to design – to design machines, systems, work for operations personnel, information and control systems, and, perhaps, decision aids. Models of human operators in complex systems are mechanisms to organize our knowledge about what the operator should or does know and how (s)he structures that knowledge to make decisions or solve problems. It is closely tied to definitions of mental models proposed by Rouse and Morris (1986) and Rasmussen (1986). Rouse and Morris define mental models as the mechanisms whereby humans are able to generate descriptions of system purpose and form, explanations of system functioning and observed system states, and predictions of future states. Rasmussen notes that for the purpose of system design it is not necessary to have detailed models of the actual mental processes or structures used by the operator; rather higher level structural models of the mental activities operators use will suffice. Such models might be considered engineering as opposed to psychological models.

One engineering model of operators in complex systems is the operator function model (OFM) developed by Mitchell (1987) and used to describe and prescribe operator activities in supervisory control. The operator function model (OFM) provides a flexible framework for representing operator activities in the context of complex systems. The OFM is a representation of how an operator might decompose and coordinate activities to meet system objectives and ensure system safety. OFMs represent the interrelations between dynamic system states or operator knowledge and operator activities.

The OFM is a network in which nodes represent operator activities. Activities are structured hierarchically, representing primary operator control functions or purpose at the highest level and individual control actions at the lowest. Actions can be both physical (e.g., an information query or system control command) or cognitive (e.g., information gathering, information processing, and decision-making). The OFM network is heterarchic, that is, at the same level, there may be several activities that, given system state, are undertaken concurrently. The heterarchy accounts for the coordination and concurrent nature of operator activities as well as the operator's dynamic focus of attention. The operator function model is a prescriptive model that specifies non-deterministically a set of plausible operator activities given current system state and recent operator actions. As such, it provides a structure to represent knowledge about the system and operator activities, and a mechanism to define expectations of operator activities given current system state.

The OFM is certainly not the only representation that could be used. The data-driven nature of a case study, however, makes the OFM a useful candidate. It provides a means to structure and organize observed behaviour, permitting the modeller to hierarchically abstract low level actions into meaningful higher groupings.

Background

NCR recently built a state-of-the-art electronics assembly plant for printed circuit boards (PCB) used in computerized sales terminals. This plant is located in metropolitan Atlanta and Georgia Tech faculty and graduate students have worked closely with NCR to help plan, design and operate the facility. Our group's (Center for Human–Machine Systems Research) particular interest is helping to model the role of the plant floor operators and, based on this model, to design effective decision support structures. Georgia Tech graduate students attended the NCR two week training programme for the newly hired plant operators. The plant came on-line on January 1, 1990, and graduate students participated in plant floor operations, and gathered data on the type, frequency, and process of plant floor decisions. Potential model structures include an operator function model and the abstraction hierarchy (Rasmussen, 1986). Potential aids based on this model include model-based displays (intelligent displays controlled by a software implementation of the operator model) and a case-based reasoning system to assist with fault management in the configuration of the surface mount technology (SMT) line.

This study proceeded with a series of steps. First, as indicated above, graduate students trained with new plant floor operators. Second, we developed a representation of the assembly and troubleshooting processes. Third, we identified places in the process where data are or could be collected. Next, we identified areas of human decision-making that were both important and tractable for modelling. Finally, after extensive observation and data collection in the PCB troubleshooting area, a model that combines case-based reasoning and operator functions for troubleshooting was proposed. Below, we briefly describe the initial steps in the study and conclude with a detailed description of the modelling work.

Preliminary training and system description

The NCR PCB assembly system is depicted in Figure 23.1. At this time, printed circuit boards are manually inserted into a screen printer; subsequently small and large components are inserted automatically by chip shooters. At this point, there is visual inspection, manual insertion of components that the automatic insertion equipment cannot handle, and correction of observed problems. Then boards pass through a reflow oven, through-hole components are inserted manually and the board proceeds to the wave solder process. The assembly process concludes with the board being cleaned, sheared, and manually touched up, if necessary. Testing and repair comprise the final stage of the process; boards

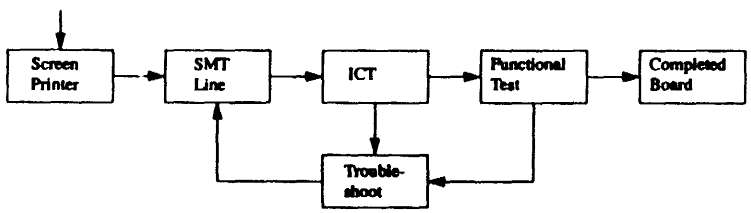

Figure 23.1 Printed circuit board assembly.

are not shipped unless they successfully pass all tests. At this time there are two tests, an in-circuit test (ICT), and if the board passes the ICT test, a hot mock-up test concludes testing and the board is ready for shipment.

The plant initially attempts to produce more than one hundred successfully tested boards each day. There are approximately twenty operators who staff the plant over a twelve hour day, six days a week. Operators are trained in a State of Georgia Soldering School for one week and then rotate through various positions. NCR's goal is to have all operations personnel cross-trained. The Georgia Tech graduate students participating in this project attended the week-long soldering school, NCR's in-house training programme, and have rotated through a number of plant floor operations (i.e., on-the-job (OJT) training).

Decision making and data collection

The plant is so new that much of the process decision-making proceeds by trial and error, and many problems are one-of-a-kind (or at least the first-of-a-kind). For example, one shift encountered a large number of boards with misplaced components; the cause was the newly installed operating system in the small part chip shooter.

Data are collected at almost every position in the process, but are predominantly logged as historical process or defect data, rather than data that can quickly be used as feedback to modify the production process. The lack of immediate use of fault data is primarily due to the recency of the installation rather than the intent of the process designers.

Several months of observation suggested that, at this point in the plant's evolution, the testing and troubleshooting process for finished boards (at the in-circuit test (ICT) stage), is an important process and one which is stable enough to permit data collection, analysis, and modelling. As a result, the remainder of this paper describes the troubleshooting process itself, a case data base built by observing experienced troubleshooters diagnose faults on individual boards, and a proposed model representing the troubleshooter's decision-making process.

Troubleshooting in electronics assembly

Once a PCB is populated with components, soldered, cleaned, and sheared, the board is mounted on an in-circuit test machine (ICT) which checks individual components as well as the connections between components for proper functioning. If the board passes the ICT test, it moves to functional testing, the last step in the production process. Our research focuses on the troubleshooting process when the ICT fails a tested board.

Printed circuit board troubleshooting data

If a board fails the in-circuit test, the ICT machine produces a ticket listing the detected failure(s). The operator uses the ticket (Figure 23.2 depicts a sample

These are the board faults.
TA 7052 MAX Processor
Tues Feb 06 07:10:13 1990

---

c58 has failed
22C1 SMT
Measured: 0.015772u
Nominal: 0.010000u
High Limit: 0.015000u
Low Limit: 5000.0p
Capacitance in Farads

Figure 23.2 Sample ICT ticket.

ICT ticket) as initial data for the troubleshooting process. Although the ICT ticket provides useful information about the cause of failure, this information may not be a direct indication of the source of the problem. Thus, sometimes, troubleshooting, given the ICT ticket, is straightforward; at other times, however, it is much more complex, requiring detailed search and troubleshooting strategies and knowledge.

The case study portion of this research entailed observation and data collection for approximately 300 PCB troubleshooting incidents. A scheme was developed to code each troubleshooting incident as a 'case'. Each case contains the ICT ticket information, the operator's suspicions as to the location of the defect, the physical activities performed to find the fault, the activities performed to repair the fault, the defect code (the code with which NCR track defects), and aspects of the production process that may have contributed to the defect (e.g., machine malfunction such as part placement errors, power outages, new supplier, etc.). Illustrative samples of the initial case data are given in Figure 23.3.

Case 1
43091: A short is listed with U30 as a common device.
ICT Failure Information:
Reading: short
Value w.r.t. nominal:
Nodes: 558, 627
Common Devices: u30

Case 2
39551: c12 failed.
ICT Failure Information:
Reading: c12
Value w.r.t. nominal:
Nodes:
Common devices:

Figure 23.3 Cases for PCB board defects giving PCB board number and ICT error.

Through observation of the troubleshooting process and analysis of the case data, patterns in the troubleshooting process emerge. Recurring defects and operator search strategies become discernible. A model is proposed that structures these defect patterns, operator troubleshooting activities, and fault diagnosis.

An operator function model of the PCB troubleshooting process

The proposed operator function model represents the process of identifying defects on a PCB given a variety of ICT failure ticket readings and board symptoms. The operator function model's nodes represent classes of troubleshooting activities; the arcs define enabling conditions for the next step in the diagnostic process.

The top level OFM is depicted in Figure 23.4. The primary functions, normally occurring sequentially, are (a) assess the ticket, (b) based on current information, invoke a process in a troubleshooting category until a failure is identified, (c) diagnose the failure, and (d) execute a repair activity. The troubleshooting activity continues until a failure is identified.

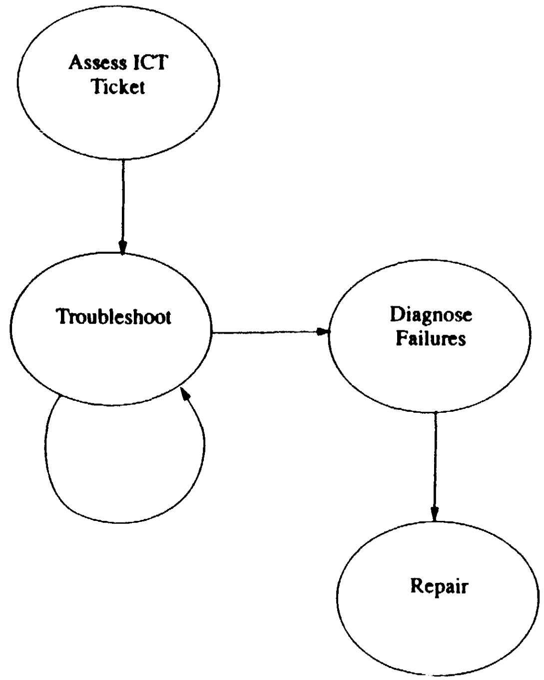

Figure 23.4 Top level OFM for PCB troubleshooting.

The model structures categories of PCB troubleshooting and their corresponding enabling conditions are depicted in Figure 23.5. The primary categories are (1) recognize a known problem, (2) detect a common symptom, (3) detect a temporary trend, and (4) execute a standard operating procedure and/or apply a rule of thumb. The first three categories characterize the operator's ability to recognize a fault from fault data. Standard operating procedures and rules of thumb represent the operator's general knowledge of system properties and troubleshooting strategies that are used for unfamiliar failures.

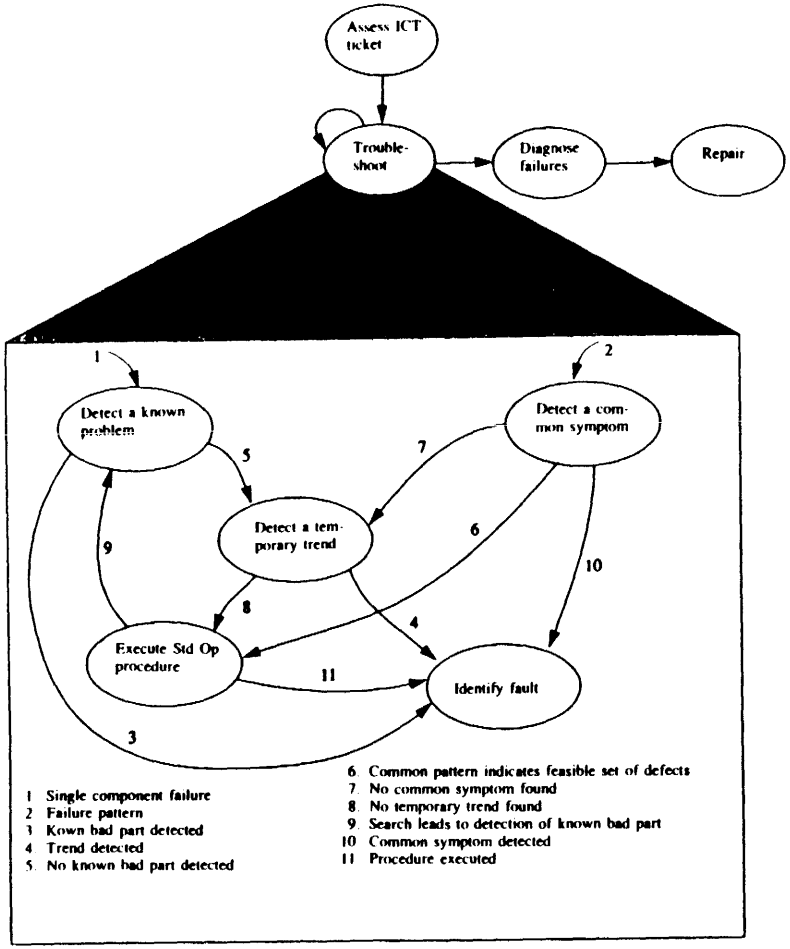

Figure 23.5 OFM decomposition of troubleshoot function into lower level activities.

Known problems

Known problems are PCB defects which can be detected directly from the ICT ticket reading. A sample list of Known problems is shown in Figure 23.6. Detection of the fault in the Known problem category is based solely on the information in the ticket reading. The operator assesses the ICT failure ticket to determine if there is a single failure (e.g., one resistor, one capacitor, one ICT, etc.) and if it matches any of the list of recurring single component failures. If the failure matches a component on the Known problems list, the search terminates.

a. If c12 fails, automatically replace it. It is a known bad part. Replace the part.
b. 74fF244D is a known bad part. U56 and u65 both have this part number. Follow the lifted leg procedure to find the bad part(s) and replace it(them).
c. U37 tends to have lifted leads. Use a dental tool to detect lifted legs. Solder the legs.
d. U30 (max chip) is prone to solder bridges and unsoldered pins. Visually inspect the part. If there are bridges, remove the excess solder.
f. r254 is often damaged. Visually inspect the part. If it is damaged, replace the part.
g. If I1 and I2 fails, the ticket is bogus. Re-run the board.
h. u51 is a known bad part. Replace it.

Figure 23.6 Sample list of Known problems.

The list of components which are considered Known problems changes over time. As the manufacturing process is constantly modified and transformed, different components become problematic. Therefore, the list continuously evolves.

Common symptoms

Common symptoms, like Known problems, are detected directly from an ICT ticket reading. While Known problems represent single component failures, Common symptoms are groups of components or ticket readings which characterize the defect. The operator assesses the ICT failure ticket to determine if there is a commonly recurring pattern. A pattern is two or more component failures, short readings, or open readings. If the reading matches a pattern in the list of Common symptoms, search concludes.

The list of Common symptoms also changes as the manufacturing process changes. The list is updated as new patterns recur and old ones are no longer seen. Figure 23.7 shows specific examples of Common symptoms: a faulty reading of r19 and r24 that indicates u56 (which is connected to both) is defective; a reading of a short between nodes 661 and 625 that indicates c108 must be moved away from a via; and a reading with any combination of resistors such as r228, r239, and r279 that indicate u51 is defective.

A. The following resistors are connected to U56 and U65:

r19
r21
r24
r240
r241
r248

If the ticket indicates any one of these resistors, or some combination of them, then the fault can be a bad U56, a bad U65, or a bad u56 and u65. First, ohm out the resistor to verify the ticket reading. Then follow the lifting legs procedure to find which one(s) is(are) bad. It does not matter if U56 or U65 is tested first.

B. The following is a list of known nail shorts:

354/608 - c85
388/608 - c93
697/698 - c76
661/625 - c108
298/296 - c185
345/608 - c171
307/608 - c206
211/625 - c87

The operator moves the part indicated so a greater percentage is on the pads.

C. Opens between

691 and 692
694 and 695

indicate that r317 and r318 are the wrong part.

Note: This symptom appears to be a temporary trend

D. Resistors listed on page 9B1 (r228, r239, r279) of the schematics and on lines DRQ0- DRQ3 are connected to u51. If any of these resistors has failed with a low reading, u51 should be replaced.

Figure 23.7 Sample list of Common symptoms.

Temporary trends

Temporary trends represent the operator's ability to remember failures seen recently which are not consistent enough to become Known problems or Common symptoms. Thus, Temporary trends are failures stored in short-term memory; whereas, Known problems and Common symptoms are failures stored in long-term memory. Like Known problems and Common symptoms, Temporary trends can also be recognized from ICT ticket readings. The operator assesses the ticket to determine if (s)he has seen that reading recently. If (s)he recalls seeing this defect once before, search terminates. An example of a Temporary trend is a sporadic placement problem. A hand-placement operator misplaces crystals on several boards. The troubleshooter finds a board with y2 reversed. After 30 minutes (s)he sees a ticket with y2 and remembers to inspect for a reversed component.

If the pattern recurs (i.e., the operator sees the ticket reading more than once a day), the ticket reading becomes listed as a Known problem or Common symptom. Determining whether a reading is a Temporary trend is subjective. The same ticket reading may appear frequently, yet the defect may be different every time. Therefore, the operator must determine if a recurring problem truly exists.

Standard operating procedures

Standard operating procedures are generalized search routines of troubleshooting search tests. If the operator does not find a Known problem, Common symptom, or Temporary trend, (s)he performs a Standard operating procedure. Standard operating procedures are not board specific; they can be used for any PCB board design. Depending on the ICT ticket reading, the operator selects one of a repertoire of search tests. S(he) will proceed through the tests until the fault is detected. Figure 23.8 depicts the six standard operating procedures for this application.

In addition, Figure 23.8 shows the decomposition of the standard operating procedure for resistors. Inspecting the model, we see that the operator first performs visual inspection. If the operator sees no defect, (s)he determines the next test by examining the measured resistance reading. Although the ticket indicates the actual resistance reading, the operator notes only the deviation from the nominal value. The operator performs a different sequence of troubleshooting search tests depending on the resistance measurement. If the reading is slightly lower than the nominal value, the operator suspects a defective IC connected to the resistor. The operator first 'ohms out' the resistor with a multimeter to verify that the ticket reading is accurate. If the multimeter reading differs from the ICT reading, the operator re-runs the board through ICT. If, however, the multimeter reading is low as well, then the operator checks the schematics to generate a list of all IC's connected to the component. If any of these IC's are Known problems, the operator follows the strategy for a known problematic component. Checking for a Known problem is part of this Standard operating procedure because a known problematic IC cannot be detected solely from the ICT ticket. If none of the connected IC's is a Known problem, the operator tests each IC by lifting legs until (s)he finds the faulty component. If the resistor measures slighty higher or much lower than nominal on the ICT tester, the operator suspects the reading to be bogus. However, the operator visually inspects the component before drawing this conclusion. If there is no visible error, the operator re-runs the ICT test. Before re-running the board, the operator may re-flow the component or via near the component to improve the connection of the board with the tester. If the ICT reading for a resistor is much higher than nominal, the operator suspects some sort of open connection. (S)he ohms out the resistor using a multimeter to verify the ticket reading. If the multimeter reading indicates something other than the nominal value, the operator re-runs the board. If the multimeter reading indicates that the resistor is a good component, the troubleshooter checks continuity along the traces by the resistor

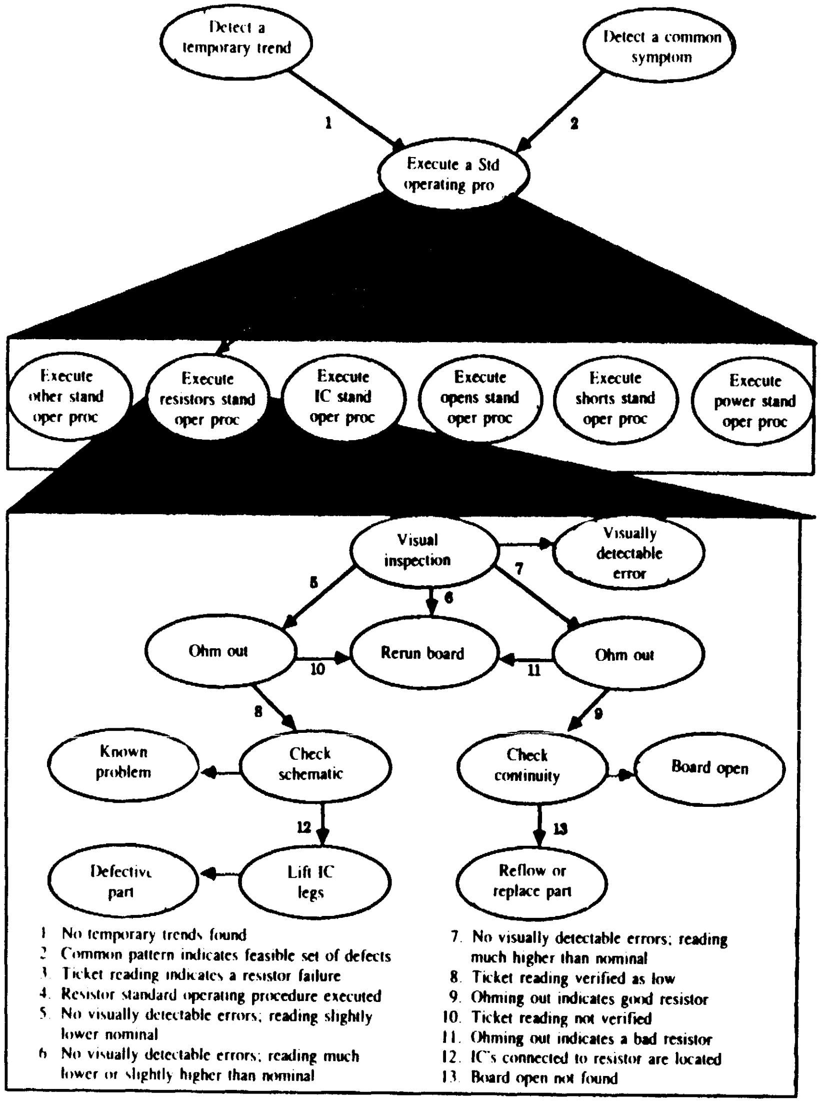

Figure 23.8 OEM standard operating procedure for a resistor.

in search of a board open. If no open is found, the operator may re-flow the component or a nearby via, or replace the component.

Detailed models for the remaining standard operating procedures are given in Cohen (1990).

Rules of thumb

Rules of thumb represent the operator's general knowledge of electronics assembly and fault diagnosis tests. Rules of thumb are shown in Figure 23.9. The operator uses these rules to determine the order in which to perform diagnostic tests. The operator often uses Rules of thumb without articulating the knowledge being utilized, thus these rules are modelled as part of the Standard operating procedures. Examples of Standard operating procedures (SPO) using Rules of thumb (ROT) include the following: if a short has common devices, inspect those devices first (SOP) since they are usually the source of the problem (ROT); if a resistor fails at a reading slightly lower than its nominal value, check IC's which are connected to it (SOP), for defective IC's may load the resistor down (ROT).

Rules of thumb

I. Discrete components (capacitors, resistor, diodes, etc.) rarely fail.
II. Visually inspect components before performing any diagnostic tests. Placement errors occur often and are easiest to spot.
III. QFPs are more susceptible to solder bridges.
IV. Tester connections are sometimes poor. It may be helpful to ohm out resistors and connections to verify ticket readings.
V. On a short, common devices are generally the source of the problem.
VI. It is easier to replace gullwing leaded parts than QFPs. If an IC is likely to have an internal defect, check gullwing parts first.
VII. If an open is listed on a ticket with one other component, there is usually only one defect.
VIII. A plugged node may cause a resistor to fail at a high reading or an open to be called out on a ticket.
IX. It is important to remember the order of ICT tests. If a board fails in analogue test, it never gets to digital testing. Therefore, the tester might indicate that a resistor failed even when the defect involves an open connection between components.
X. If a part is bad, it may cause other components to read incorrectly as well.
XI. A reading of a resistor which is slightly higher or much lower than nominal may be bogus.
XII. Checking continuity is a time-consuming procedure. Powering-up is also time-consuming and may damage the board. These procedures should be performed only if the error was not detected by visual inspection.

Figure 23.9 Rules of thumb.

Application of the model

Figure 23.10 contains the fault detection cases from Figure 23.3 interpreted with the proposed model. The model was used to 'parse' or explain the observed operator activities, actions and conclusions. The current model successfully accounts for 80–90 per cent of the observed PCB troubleshooting cases.

Case 1

43091: A short is listed with U30 as a common device. U30 is known to have bridges (SOP 8, Known Problem d), so it is visually inspected. A short is found and the excess solder is removed (Action 8). The defect is attributed to the screen printer (Cause 2).

INPUTS:
  Reading: short
  Value w.r.t. nominal:
    Notes: 558, 627
  Common Devices: u30
PROCESS:
  Invoke Known Problems:
  Invoke Common Symptoms: no match
  Invoke SOP: SOP8
    Check: for common devices
    Check: list of common devices for known problems
    Check: inspect u30
OUTPUT: bridge on u30

Case 2

39551: c12 failed. This is a known bad part (Known Problem a). The operator replaces (Action 1) it and attributes the problem to the vendor (Cause 3).

INPUTS:
  Reading: c12
  Value w.r.t. nominal:
    Nodes:
  Common Devices:
PROCESS:
  Invoke Known Problems: match for Known Problem
  Invoke Common Symptoms:
  Invoke SOP:
    Check:
    Check:
    Check:
OUTPUT: replace c12

Figure 23.10 Model-based fault diagnosis for PCB board defects.

A computerized version of this model is being developed. The computer-based model will take ICT ticket information as input, determine appropriate search strategies, and diagnose a feasible set of possible faults. Model validation will compare the model output to operator output for a new set of PCB fault diagnosis cases.

Conclusion

This research is interesting in many ways. First, it models human behaviour in an actual manufacturing environment. The OFM model of PCB troubleshooting was evolved to account for observed field study behaviour; it was fairly successful in allowing the modellers to structure observed behaviour. Given its structure, the model was then used to prescribe successfully fault diagnosis activity in a range of cases.

Future use of the model and the insights gained from it include applications in operator training and decision aiding. As with any model, the modelling process organizes knowledge and the process through which knowledge is applied. The OFM for the PCB troubleshooting can help a novice troubleshooter navigate the learning curve to become an expert. Similarly, the model may provide the knowledge or intelligence for an on-line troubleshooting aid. Currently, symptom-cause and fault diagnosis knowledge is very informal making it difficult and lengthy to cross-train operators. By organizing and presenting symptom-cause pairs and troubleshooting strategies in context-appropriate ways, operator effectiveness may be enhanced and operator training time reduced significantly.

References

Armstrong, J.E., 1990, Distributed decision making in command-and-control of complex dynamic systems, Ph.D. Thesis, Center for Human-Machine Systems Research, Georgia Institute of Technology, CHMSR 90-1.

Benson, C.R., Govindaraj, T., Mitchell, C.M. and Krosner, S.P., 1989, Effectiveness of direct manipulation interaction in the supervisory control of FMS part movement, in *Proceedings of the 1989 IEEE International Conference on Systems, Man, and Cybernetics, Cambridge, MA, November, 1989*, pp. 947–52.

Benson, C., Govindaraj, T., Mitchell, C.M. and Krosner, S.P., 1992, Effectiveness of direct manipulation interaction in the supervisory control of flexible manufacturing systems, *Information and Decision Technologies*, **18**(1), 33–53.

Cohen, S.M., 1990, A model of troubleshooting in electronics assembly manufacturing, M.S. Thesis, Center for Human-Machine Systems Research, Georgia Institute of Technology, CHMSR 90-3.

Dunkler, O., Mitchell, C.M., Govindaraj, T. and Ammons, J.C., 1988, The effectiveness of supervisory control strategies in flexible manufacturing systems, *IEEE Transactions on Systems, Man and Cybernetics*, **SMC-18**(2), 223–37.

Hettenbach, D.A., Mitchell, C.M. and Govindaraj, T., 1989, Decision making in supervisory control of a flexible manufacturing system, in *Proceedings of the 1989 IEEE International Conference on Systems, Man, and Cybernetics, Cambridge, MA, November, 1989*, pp. 953–8.

Hettenbach, D.A., Mitchell, C.M. and Govindaraj, T., 1991, Decision making in supervisory control of a flexible manufacturing system, *Information and Decision Technologies*, **17**(4), 255–78.

Klein, G., Calderwood, R. and Orasanu, J. (Eds), in press, *Decision Making in Natural Settings*, Norwood, NJ: Ablex.

Krosner, S.P., Mitchell, C.M. and Govindaraj, T., 1989, Design of FMS operator decision support system using the Rasmussen abstraction hierarchy, in *Proceedings of the 1989 IEEE International Conference on Systems, Man, and Cybernetics, Cambridge, MA, November, 1989*, pp. 959–64.

Mitchell, C.M., 1987, GT-MSOCC: a research domain for modelling human-computer interaction and aiding decision making in supervisory control systems, *IEEE Transactions on Systems, Man, and Cybernetics*, **SMC-17**(4), 553–72.

Rasmussen, J., 1986, *Information Processing and Human-Machine Interaction*, New York: North Holland.
Rouse, W.B. and Hunt, R.M., 1984, Human problem solving in fault diagnosis tasks, in Rouse, W.B. (Ed.) *Advances in Man-Machine Systems Research. Volume I*, pp. 195–222, Greenwich, CT: JAI Press.
Rouse, W.B. and Morris, N.M., 1986, On looking into the black box: Prospects and limits in the search for mental models, *Psychological Bulletin*, **100**, 349–63.
Woods, D.D., in press, Research methods for the study of cognition outside the experimental psychological laboratory, in Klein, G., Calderwood, R. and Orasanu, K. (Eds) *Decision Making in Natural Settings*, Norwood, NJ: Ablex.

Chapter 24
Intelligent computer–human interaction in real-time, multi-tasking process control and monitoring systems

Wayne Zachary, Joan Ryder, Lorna Ross and Monica Zubritzky Weiland

CHI Systems Incorporated, Spring House, PA

Abstract. The human operator in automated manufacturing systems must share attention among competing task demands and deal with real-time problem data. This real-time multi-tasking (RTMT) computer user can benefit greatly from intelligent support from the work station through which the system monitoring/control occurs. This paper describes research to develop an RTMT intelligent interface that supports the human decision-maker by applying knowledge of the task domain and of the system user's decision-making process. A novel modelling framework for human operators in RTMT environments, called COGNET, is introduced. An intelligent computer interface for a sample RTMT domain (a distributed sensor monitoring problem) is then developed, based on a COGNET model of that domain. The interface incorporates the COGNET model of the user as a way of understanding, reasoning about, and ultimately anticipating and supporting user goals and actions.

Introduction

The human role in automated manufacturing systems is increasingly that of monitoring and controlling large-scale real-time processes via a computer work station (see Berger *et al.*, 1989; Sanderson, 1989). The human opertor in these domains must share attention among competing task demands and deal with real-time problem data. This real-time multi-tasking (RTMT) computer user can benefit greatly from intelligent support from the work station through which the system monitoring/control occurs. RTMT environments include many of the most challenging problem domains humans face, including aircraft (and other vehicle) cockpits, nuclear power control rooms, air traffic control, hospital operating rooms, satellite and telecommunication network control, and weapons systems operation, in addition to automated manufacturing environments. One major way of supporting the development of more effective human–computer interfaces for automated manufacturing systems is to develop and use models of the problem-solving strategies employed by their human opertors. This position has been advocated by Rasmussen (1986), Zachary (1985, 1988), and

Decision-making in supervisory control of a flexible manufacturing system

D.A. Hettenbach*, C.M. Mitchell and T. Govindaraj

Center for Human–Machine Systems Research, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205, USA

Received February 1990
Revised August 1990

Human intervention is a component of many flexible manufacturing system (FMS) control structure designs. As FMS control systems become more 'intelligent', the role of the human in the control structure will also evolve. An FMS may become more of a tool for the human who controls it, with the human responsible for achieving system goals. Thus, defining an appropriate role for the human as a function of the FMS objectives may be critical in the design of the FMS control structure. Knowledge of the decision processes used by humans in an FMS environment can help define this role.

This paper evaluates the decision processes of humans in an FMS environment. GT-FMS, a real-time simulator of an FMS, was implemented with data from an actual FMS installation. An experiment was conducted in which humans interacted with the control system of GT-FMS from an aggregate level. The humans described each of their control actions, and their decision processes were evaluated by mapping these descriptions onto a Rasmussen's [22] model of human decision-making.

The experimental results support making humans an integral part of the FMS control process, since an intricate knowledge of the system state and system sensitivity were crucial to human decision-making in GT-FMS. Human subjects in this experiment used detailed status information, rather than system performance history, as the basis of their control decisions and were inconsistent in defining their goals.

1. Background

Flexible manufacturing is an approach to manufacturing that is primarily used for the production of similar (but not necessarily identical) items in low volumes. A flexible manufacturing system (FMS) is flexible because it can easily adapt to demand and design changes [4].

The scope of an FMS may vary by application. However, a typical FMS usually includes the following components [7, p. 891]:

(1) A set of machines or work stations that have some degree of flexibility, in particular they do not require significant set-up time or change-over time between successive jobs.

(2) A material handling system that is automated and flexible, i.e. it permits jobs to move between any pair of machines so that any job routing can be followed.

(3) A network of supervisory computers and microprocessors.

(4) Storage, locally at the work stations, and/or centrally at the system level.

Development and implementation of flexible manufacturing systems has generally paralleled the development of manufacturing automation, but automation is not a prerequisite for using the FMS approach. Flexible manufacturing may be considered a 'manufacturing philosophy', that is 'based on the concept of effectively controlling material flow through a network of versatile production stations using an efficient and versatile material handling and storage system' [4].

1.1. FMS control strategies

For an FMS to function effectively, the control decisions inherent in each component of the FMS must be linked together to form a control system. As FMS technology continues to advance, these control systems have become increasingly complex. This complexity implies that the FMS control function has also advanced, and evolved into an operations management function requiring varying degrees of automatic control [19]. To address these complexities, an FMS may be controlled from a systems perspective, taking

* Currently with NCR Corporation, Retail Systems Division, Manufacturing Operations, Atlanta, Georgia.

North-Holland
Information and Decision Technologies 17 (1991) 255–278

into account the characteristics of each component of the FMS.

The effectiveness of a systems perspective for FMS control is well documented, and several approaches to this type of control have been attempted. In one approach, the control requirements for an automated manufacturing system are defined as a three-level hierarchy consisting of [6]:

(1) Pre-release planning.
(2) Input or release control.
(3) Operation control.

Gershwin et al. [13] use a multi-level systems perspective for control system design. Their design defines a time-scale dependent hierarchy of manufacturing system control. This hierarchy consists of machine-level control, cell-level control and factory-level control. Hutchinson [17], Ranky [20], Kimemia and Gershwin [18], and Akella et al. [1] provide additional examples.

The numerous research efforts addressing FMS control strategies indicate that the systems approach to FMS control may be characterized by a hierarchy of several information-sharing control levels. This design is effective because it allows the FMS to respond to a wide spectrum of contingencies. Since flexible manufacturing systems are inherently dynamic, this type of system response may be critical. However, the systems approach also requires that the FMS control system incorporate many manufacturing decision-making functions. Some of these functions may not be part of an on-line control structure in traditional manufacturing applications.

1.2. Human interaction in FMS control

To address some of the contingencies as well as the dynamics of FMSs, many hierarchical control strategies employ a manual override (or manual control mode) allowing human intervention if the FMS cannot respond on its own. Hutchinson [17] states that 'in fact, most FMS systems require a high degree of human intervention because of failures in both hardware and software, human error, maintenance, and changes in operating environment' [17, p. 288]. Kimemia and Gershwin [18, p. 354] suggest, concerning an FMS control policy, that 'it is important that this policy employ feedback so as to respond to failure and to allow human operators (who can deal with a wider range of situations than envisioned by system planners) to override control decisions on rare occasions'. In addition, in the FMS Handbook [8], the short-term tasks of the FMS line supervisor are described as work order scheduling and dispatching, tool management and reaction to system failures. The handbook describes medium-term tasks of the supervisor as dividing production into batches, maximizing machine utilization, and responding to disturbances in production planning/material availability. Although much research has focused on enabling an FMS to be as 'operatorless' as possible, human intervention continues to play a significant role in modern FMSs.

Existing research addressing this human intervention, or supervisory control, has discussed several issues. In one study, supervisory control is characterized by 'intermittent monitoring and control by the human, with monitoring predominating' [14, p.6]. This study concludes that 'there is . . . a pressing need for design principles formulated and tested in situations analogous to those which will be found in the automated manufacturing plant' [14, p. 15].

The Georgia Tech Flexible Manufacturing System (GT-FMS) is a research domain that was developed to provide an environment for studying supervisory control design principles for FMSs [3]. The GT-FMS environment consists of a real time interactive simulation of a multi-cell, multi-workstation flexible manufacturing system. GT-FMS further provides

> . . . a controlled laboratory environment in which to implement and evaluate the supervisory control perspective for FMS scheduling. It facilitates research and validation in a framework of realistic manufacturing conditions, including human interaction with the scheduling and control system [10, p. 225].

GT-FMS has been used as an effective domain for several research initiatives. Dunkler [9, 10] configured GT–FMS to simulate a machining center for diesel engine cylinder heads. The human's role in the control system was to monitor and fine-tune an FMS that used two scheduling dispatch rules: First-Come-First-Served (FCFS) and Shortest Processing Time

(SPT). Dunkler found a significant difference between the performance of GT-FMS under fully automatic control versus supervisory control. He concluded that if a human supervisor is part of an FMS control system, and allowed certain types of interventions, then due date and inventory performance of GT-FMS could be improved [10]. Dunkler's results support strongly the idea of actively integrating humans into operational controls of automated manufacturing environments [10]. Dunkler further concluded that a 'preferred' dispatching rule existed for a particular state of an FMS. He also suggested that one human supervisor activity might be to select, from among various dispatching rules, the rule that is preferred based on the current state of the FMS.

Additional GT-FMS research has explored the effect of interface design on a manufacturing supervisor's performance [5]. In this research the basic configuration of GT-FMS used for Dunkler's experiments was slightly modified: the GT-FMS is a circuit card assembly system rather than a flexible machining system. This research compares the supervisor's performance with an icon-based, direct manipulation display interface versus a more conventional display. The research concluded that a direct-manipulation display can enhance the supervisor's performance for some measures of performance.

For an FMS to be effective, the functions of the human supervisor must be defined and the limitations of the human taken into account. Recognizing previous GT-FMS research which supports actively integrating the human into the FMS control structure, an understanding of the appropriate level of control or role for the human is critical. Ammons et al. [2, 3] address the role of the human supervisor by proposing a realistic supervisory control paradigm for FMSs. This proposal defines levels of automation within an FMS and an operator function model for the item-movement (lowest) level of automation.

In an FMS control system the appropriate control system model and accompanying decision support are dependent upon the role of the human. Defining an appropriate role and subsequent responsibilities for the human as a function of the FMS objectives may thus be a critical step in designing the FMS control structure. As FMS control systems continue to develop, the degree of human interaction with these control systems will also evolve. Control systems may become more 'intelligent', yet human intervention is likely to remain an important aspect of FMS control. The eventual role of the human may 'shift the emphasis to elements of planning and commitment' [25, p. 238].

1.3. Human operator as FMS manager

In this investigation the role of the human focuses on longer-term goals of the manufacturing system. This role reflects a level in an FMS control hierarchy that is 'higher' than the item-movement level (i.e. the level investigated in Dunkler's [9, 10] and Benson's [5] research. The investigation evaluates the human's response to this role.

The supervisor in this investigation takes an aggregate view of the flexible manufacturing system. This approach is explored for several reasons. Typically, FMSs have some degree of human interaction as part of their control system, but this interaction is not always well defined, and it often occurs on an ad hoc basis. Previous research [10] indicated that FMS performance can be improved if the ad hoc nature of the human control actions is removed. However, no universal 'best' definition exists for the design of human interaction, with hierarchical control models and actual FMS installations allowing varying degrees of human intervention throughout all control levels. By modeling the supervisor from an aggregate view, additional insight can be obtained toward creating a supervisory environment that potentially takes better advantage of the human's judgment and decision-making skills. In this role, 'systems manager', rather than 'supervisor', may better describe the human control functions involved. Designing human intervention at a higher level removes the systems manager from minute-to-minute contingencies and allows him/her to focus on meeting the long-term objectives of the FMS. Furthermore, while a computer may handle minute-to-minute decisions in some FMS installations, a computer is not ultimately responsible for the performance of actual flexible manufacturing systems. Thus, human monitoring of aggregate FMS performance data in practice is virtually guaranteed.

For a systems manager to effectively control the FMS, s/he must be allowed to initiate certain control actions and be provided with clear goals against which to measure these control actions. Overall profitability of the FMS is likely to be an overriding concern for a systems manager. As such, the systems manager must initiate control actions that positively affect the profitability of the FMS.

An FMS is likely to process a wide variety of parts simultaneously, with parts belonging to many different customer orders. System cost performance is affected by completing customer orders on time, and by completing enough orders so that the cost of production per part is sufficiently low. Processing priorities of the parts within an FMS can significantly affect the operation of the FMS and thus have an impact on profits. By modifying the processing priorities within the FMS, the systems manager can emphasize a specific method of operation that provides the greatest profit potential for a specific period of time.

This investigation does not seek to prove or disprove a specific theory or hypothesis. Rather, it seeks to gain further insight into human decision-making within an FMS environment. Decision processes of humans in an FMS systems manager's role are analyzed with the goals of better understanding and defining the human's role in an FMS control structure, improving feedback mechanisms of FMS control loops, and uncovering decision-making parameters used by human decision-makers in an FMS environment. As a systems manager, the human is placed in the FMS control structure on aggregate level. This control level provides the systems manager with the opportunity to enhance FMS performance by modifying the part scheduling algorithm or expediting specific groups, or orders, of parts.

2. GT-FMS configured for a systems manager

An experiment was conducted in which humans interacted with the control system of GT-FMS from an aggregate level. The human operator's goal was to increase profits from the FMS by completing groups or orders of parts on time and by keeping system throughput high. This goal was achieved through the control mechanisms available to the humans: modifying the FMS scheduling algorithm and expediting orders of parts (orders are discussed in more detail later). The humans verbally described each of their control actions, and their decision processes were evaluated by mapping these descriptions onto the Rasmussen [22] model of human decision-making. With the exception of the modifications described below, GT-FMS was configured identically to Dunkler's experiments [10].

2.1. Parts and part scheduling

The systems manager is free to modify the computer-based part scheduling algorithm (cell scheduler) in response to trends noted in overall cell performance. In GT-FMS, the automatic cell scheduler determines which part to place on an available machine. The scheduler examines all parts in the cell inventory and then selects an appropriate part. To make this selection, the base GT-FMS scheduling automation was modified to use an algorithm called the weighted operation priority index (WOPI) [11, 15]. The WOPI algorithm is a weighted linear combination of the shortest-processing-time (SPT) and the earliest due date (EDD) scheduling algorithms. The systems manager modifies the WOPI algorithm by changing alpha, a weighting factor that focuses the algorithm towards either due date or SPT scheduling. The systems manager decides when to change alpha and the magnitude of the change. In this experiment, alpha may assume values between 0.0 and 1.0, inclusive, in increments of 0.1.

The part data for GT-FMS are configured as in the Dunkler experiment [10]. Thus, twenty part types representing the twenty different types of cylinder heads at MTU[1] are grouped so that seventeen of these part types are represented by seven part groups. These part groups maintain approximately the same workload in GT-FMS as in the MTU FMS (a complete description of part types can be found in Dunkler et al. [10].

[1] Motoren und Turbinen Union, a German company that manufactures diesel engines.

2.2. Orders

GT-FMS was augmented in this investigation to generate and track *orders* of parts rather than individual parts. The system performance measures reflect this enhancement. Each order's size is randomly generated and uniformly distributed between three and six parts. All parts in a given order are of the same type (e.g. type b, c, d, etc.). The part types are generated as in previous GT-FMS research [10], using the MTU data. Thus, the mix of part types over all orders reflects actual demand data from the MTU system.

Each order in GT-FMS is identified by an order number. This number represents the sequential ordering of the order within the simulation. For example, order number 14 is the fourteenth order to arrive at the FMS for processing. Each part in an order has the same due date, called the order due date, and this due date is also generated in the same way as part due dates in previous GT-FMS research.

In the GT-FMS environment designed for this research, no distinction is made between an 'arrival buffer' inventory area and a work-in-process (WIP) area; only one common inventory area exists. The size of this inventory area is essentially unlimited; however, the maximum number of orders, of any size, which can be in the FMS is ten. New orders are generated as the inventory area becomes depleted, but only one new order is generated at a time.

The order structure of the parts in GT-FMS is invisible to the cell scheduler. As machines become available for part processing, the cell scheduler processes all orders in the cell inventory area simultaneously, i.e. when a machine becomes available, a part from the inventory area is selected from among all parts currently in the inventory, irrespective of the order number of the part.

2.3. Performance scores and goals

The goal for the systems manager is to achieve the highest profit possible for each experimental session. A penalty is assessed for completing parts past their due dates, and a profit per part is accumulated for each part completed during a session. This scoring mechanism was incorporated into the GT-FMS model for several reasons. First, the profit function parallels the trade-offs inherent to the role of the systems manager and characteristic of the WOPI scheduling algorithm, namely throughput and timeliness of production. The scoring also provides the systems manager with a method of feedback for performance in each session. Session scores are displayed immediately after each session. Subjects are free to track their own performance from session to session, but performance tracking over sessions is not automatically provided as part of the GT-FMS configuration.

The score for each session has two components. The first component of the score is the completed parts component and the second is the late parts component. The session score is the difference between these two components.

The completed parts component is the total number of parts completed during a session multiplied by the estimated profit realized from each part. The estimated profit per part is assumed to be \$50.

A penalty cost is assessed for completing some or all of the parts in an order after the order's due date. The late parts component of the session score is the sum of the penalty costs assessed during a session. A penalty cost is calculated for each completed order that contains late parts. At the end of a session, penalty costs for incomplete orders are also included. The penalty cost is the product of the number of parts in an order (the order size), the amount of time (calculated in minutes) past the due date at which the last part in the order is completed, and a two dollar per part per minute late penalty on each late order.

2.4. Operator workstation: Hardware

The hardware consisted of an Apple Macintosh II with a color monitor. GT-FMS was implemented in C. The displays and user interface controls used Apple's Toolbox windowing and display routines.

2.5. Operator workstation: Displays

The operator workstation consisted of three display options listed horizontally across the top of the screen: 'Windows', 'Summary', and 'Scheduler' as shown in the top of Fig. 1. The 'Windows' option provides another menu, which allows a choice between the penalty cost and throughput display or the order status display. All displays use standard pulldown menus from Apple's Toolbox windowing utilities package. A description of the displays available for monitoring GT-FMS in the investigation follows.

2.5.1. Penalty cost and throughput display

Figure 1 illustrates the penalty cost and throughput display. The display shows two graphs: one for penalty costs associated with late parts and one for total parts completed (or throughput). These graphs are updated (i.e. a new point is plotted) every 3 seconds.

The throughput graph is also dynamic and averaged over the last minute of operation. This graph (in Fig. 1, the curve with oscillations) provides a measure of how fast the FMS is producing parts.

2.5.2. Order status display

The order status display provides the subjects with key summary information for each order currently in GT-FMS. Figure 2 illustrates this display. Completed orders are not displayed. The order status display is not dynamic.

For each unfinished order, the order status display lists performance statistics for the ses-

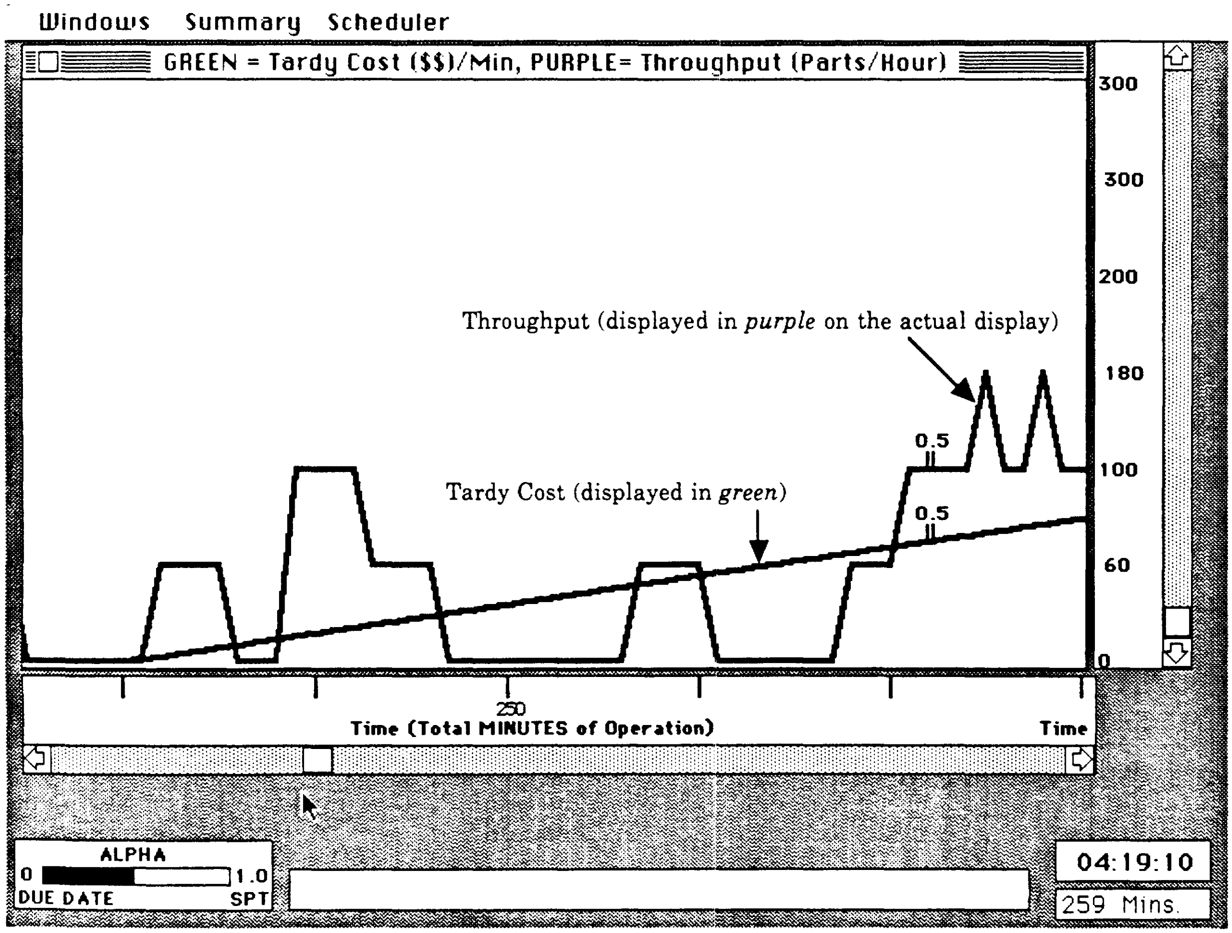

Fig. 1. Penalty cost and throughput display.

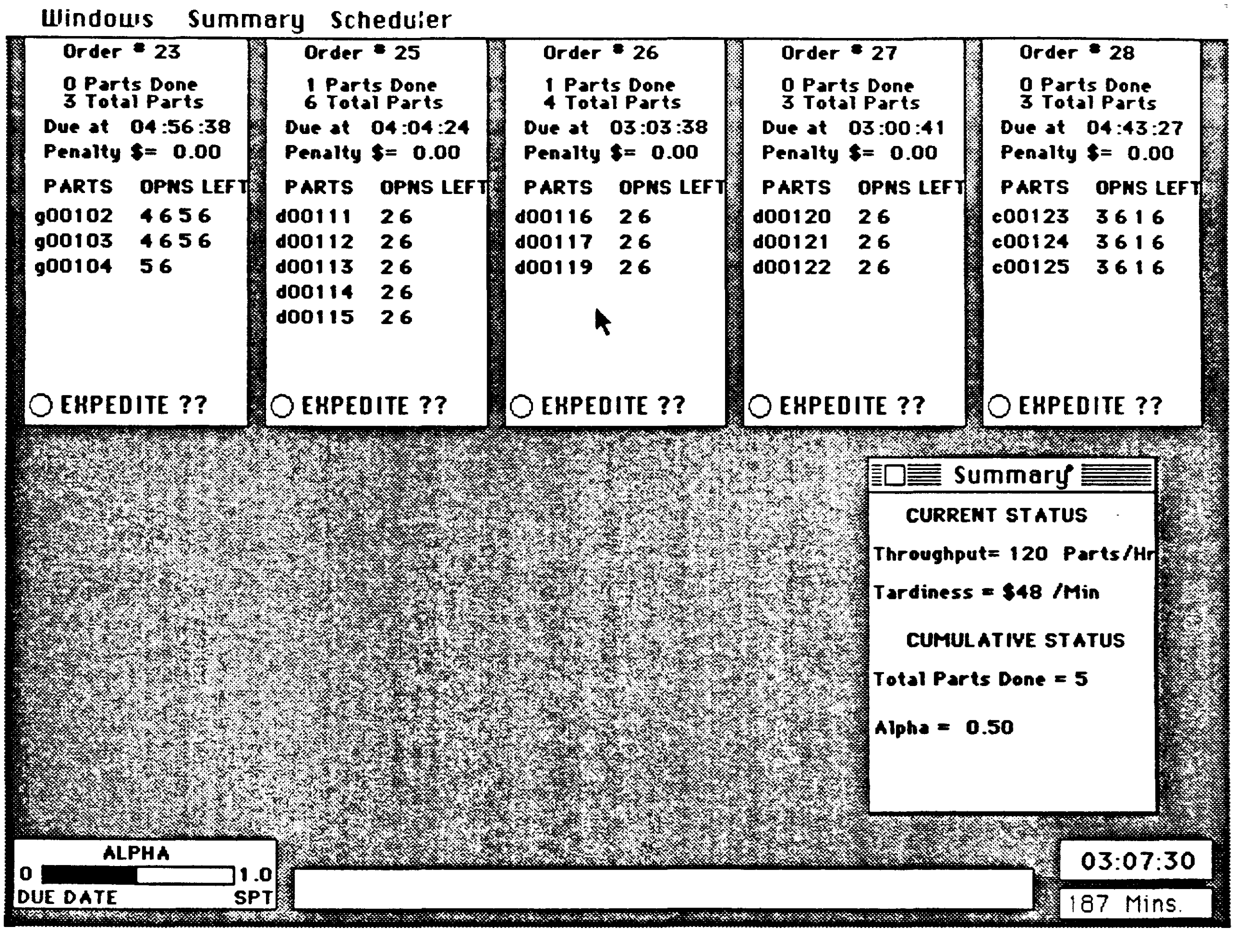

Fig. 2. Order status display.

sion, including the order number, the order due date, the size of the order, the number of completed parts, any penalty costs, and a summary of each unfinished part in the order.

2.5.3. *Summary status display*

Figure 2 also illustrates the summary window (entitled 'Summary') within the order status display. The summary is not automatically shown when the order status display is chosen but must be selected from the list of display choices located at the top of the CRT screen (i.e. 'Windows', 'Summary', and 'Scheduler').

The summary lists several performance statistics for the session. These include the total number of parts completed thus far and the current average penalty costs for late parts (tardiness). The summary window is dynamic. As a session progresses, the summary display updates every 3 seconds as long as this window is on the screen.

2.5.4. *WOPI weighting factor*

The current value of alpha and the resulting priority of the system scheduler, either SPT or due date, is shown as a bar graph on the alpha display. The filled portion of the bar graph corresponds to the current value of alpha. Additional detail for each of the displays can be found in Hettenbach [16].

2.6. *Operator workstation: Controls*

GT-FMS allows the systems manager two types of interventions to increase system profits. The first type enables the subjects to alter the weighting factor, alpha, used by the automatic system scheduler. To modify the value of alpha, the subject first selects the scheduler display with the mouse, then positions the mouse over the oval next to his or her choice and 'clicks' (i.e. depresses and releases the control button on the mouse). The new value of alpha is then highlighted and the alpha window (i.e. the bar graph) is also updated with the new value. The ovals used throughout the displays in this investigation are standard radio buttons from Apple's Toolbox windowing utilities package. In addition, the two graphs in the penalty cost and throughput display are marked to record the change. These marks appear as parallel vertical bars on both the tardy graph and throughput display. The markings on the graph provide a way for the systems manager to review the history of each change in alpha during a session.

The second type of intervention enables a systems manager to expedite an order. If an order is expedited, the system scheduler places a priority on completing parts from the expedited order. When a machine becomes available, the system scheduler examines parts in the expedited order first. If a part in the expedited order can be processed on the available machine, then this part is placed on the machine without examining any of the other parts in any of the other orders.

To expedite an order, the subject first selects the order status display, then positions the mouse over the empty oval next to the word 'EXPEDITE?' and 'clicks'. A red highlight bar, enclosing the order number on the order's display, then appears and the empty oval is darkened, indicating that the order is expedited. Subjects can 'unexpedite' an order by positioning the mouse over a darkened oval and 'clicking'. The system scheduler will once again give parts from all orders equal priority as before. Only one order can be expedited at a time, but a systems manager can expedite or unexpedite an order at any time.

3. Experiment

Eight students participated in this investigation. These students were all volunteer graduate students from the Computer Integrated Manufacturing Systems (CIMS) program at the Georgia Institute of Technology. Five of the students were male and three were female. Two of the students had some manufacturing experience. All of the subjects were paid five dollars per hour for their participation and were informed at the beginning of the investigation that the subject achieving the highest total score would receive a $25 prize.

The session scores were used to assess performance for the prize. The prize was awarded based upon the highest total score, which was the sum of eight session scores. These session scores were also used as a measure of the 'controllability' of the GT-FMS environment (this is discussed in more detail later).

Each of the subjects received a training manual at the beginning of the investigation. Subjects were free to reference this manual as they controlled GT-FMS but were not permitted to take the manual outside of the experiment room.

Subjects were also provided with a summary listing of process operation and machine data for their use during the experiment. In addition, subjects were free to take notes or make any calculations desired during the experiment. However, subjects were not permitted to use any other reference material during the investigation other than the materials provided to them.

Subjects engaged in a total of 11 sessions each. The first session did not involve controlling GT-FMS but consisted of a review of the training manual with the experimenter. GT-FMS did not run during this session. This session lasted about 60 minutes, and the subjects were free to ask any questions concerning the control or operation of GT-FMS. However, questions concerning specific strategies or approaches were not answered by the experimenter.

The remaining ten sessions lasted approximately 45 minutes each. The next two sessions were also training sessions and not used for data analyses. The subjects received verbal instructions during the training sessions which reinforced the training manual review questions. The training was as consistent as possible across the subjects.

All 11 sessions were conducted over a 3-week period, with most subjects participating in one session per day on consecutive days. During the sessions, the experimenter was present in the experimental room with the subject. The experimenter took notes on subject performance and answered questions concerning operational aspects of the system from the training manual. However, as during the training sessions, questions concerning specific strategies or approaches were not answered by the experimenter.

There were ten different simulation sessions, each characterized by a different initial system state or 'warm-up' period. Each subject was exposed to all ten sessions. The first two sessions were given in the same order for all subjects, and the order of the last eight sessions was randomly assigned. Each of the 45 minute sessions simulated 225 minutes of actual FMS operation. The initial system states used in the investigation were also based on a pilot study.

3.1. Measures of performance

Since this investigation seeks to better understand how humans respond to an FMS environment and their resulting decision processes, a method of evaluating the subjects' performance in terms of their decision processes was required. The exploratory nature of this research, coupled with the research's focus on *how* decisions are made (as opposed to the outcome of decisions), indicates that appropriate performance measures are not obvious. An appropriate method needs to be sufficiently generic so that it can be applied to GT-FMS and yet provide an acceptable level of detail to embody the intricacies of the subjects' decision processes.

To fulfill these requirements, it was determined that mapping the subjects' decisions and strategies with the decision ladder developed by Rasmussen might provide the necessary structure for the research goals [21–23]. Rasmussen's decision ladder is '. . . independent of the specific system and its immediate control requirements' [21, p. 142], so it provided the generalizability necessary for use with the relatively restrictive environment of GT-FMS.

Figure 3 illustrates the basic decision ladder developed by Rasmussen. By beginning at the lower left (i.e. 'Activation') of this ladder, and proceeding through each circle (state of knowledge) and rectangle (data processing activity), following the bold arrows, each step of a control decision is addressed. The ladder thus provides a 'schematic map of the sequence of information processes involved in a control decision' [21, p. 144].

The data processing activities in the decision ladder are mental reasoning processes that lead directly to the states of knowledge. Applied to GT-FMS, for example, the 'Observe' data processing activity might involve scanning a particular display but focusing only on information considered important or relevant. Thus, a systems manager might scan the order status display of Fig. 2 and focus on only one aspect of this display. Other information is provided, but the systems manager mentally sorts and places specific priorities on this information concentrating only on the data that s/he determines is 'important'. This data processing might then lead to a 'Set of Observations' state of knowledge which could include 'next required processing operations'. Thus, the next required processing operations would be the 'Set of Observations' that resulted from the systems manager's mental sorting. Information other than next required processing operations is provided, but it is evidently not considered important.

This 'Set of Observations' state of knowledge would then be included in the 'Identify' data processing activity as the systems manager proceeded along the decision ladder. In this activity, the systems manager might attempt to answer such questions as 'What's unusual about the current set of next required processing operations?' or, 'What's the underlying reason for the current set of next required processing operations?'. Based on the answers to such questions, the systems manager might then define the current system state. This system state would be the systems manager's interpretation of current conditions in the FMS. For instance, continuing the above example, the systems manager could define a system state as 'machines 1, 2, 3 and 4 have failed', or 'most of the next required processing operations require a small amount of machine time' depending upon the results of his or her 'Identify' data processing activity.

This process of alternating data processing activities with resulting states of knowledge continues through the decision ladder in Fig. 3 for

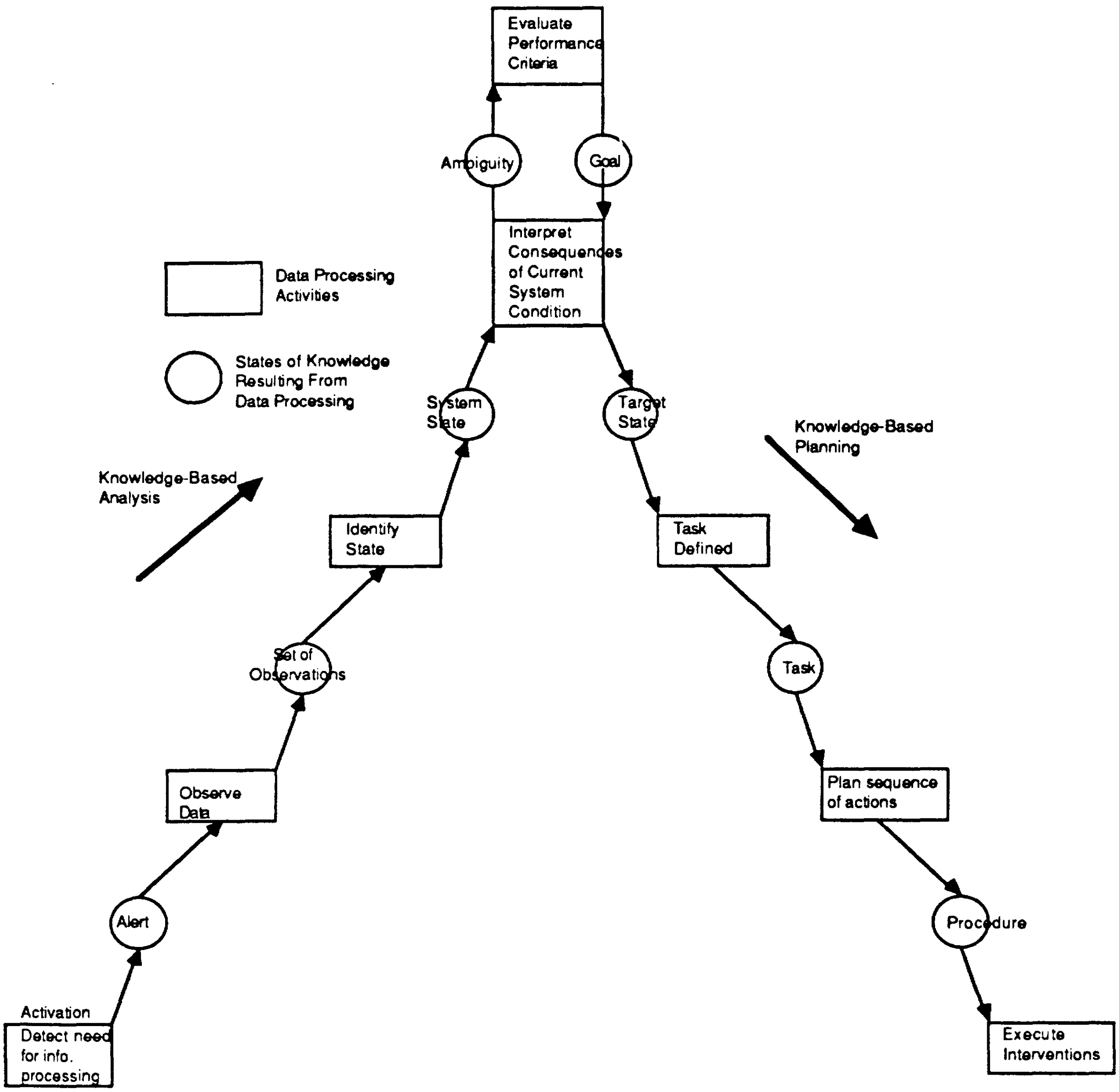

Fig. 3. Rasmussens's decision ladder.

each control decision through the 'Execute' data processing activity, which involves coordinating the desired control actions formulated by the systems manager. For GT-FMS, the available control actions, as previously discussed, are expediting orders of parts or modifying the scheduling algorithm.

If the states of knowledge used by systems managers in controlling GT-FMS can be identified, then their control decisions can be mapped to a decision ladder and evaluated. These states of knowledge might reveal various aspects of the control decisions such as: What information did the systems managers consider? What system states did they define? What goals did they develop? What target states did they attempt to achieve? What strategies did they employ? These aspects of the control decisions might then support recommendations concerning the systems manager's role, which could enhance the control function of flexible manufacturing systems.

Identifying the states of knowledge of control decisions and mapping these to Rasmussen's decision ladder was done using verbal protocols from each subject during each data-collecting session. The usefulness of verbal protocols for analyzing decision processes is well documented (see, for example, Ericsson and Simon [12]). Subjects were required to 'talk aloud' and describe their interventions as they occurred. These protocols were completely free-form, with the exception that the subjects were asked to include a description and an intent, as a minimum, in their descriptions. Several subjects went significantly beyond this minimum during their interventions.

Besides the verbal protocols, computer-compiled data files were also recorded during the sessions. These files tracked the subjects' interventions throughout each session, indicating what information was displayed and the elapsed session time when the control action occurred.

As discussed earlier, the experimenter was present in the experimental room with the subjects during each of their sessions. This provided the basis for general impressions of subject performance: Did the subjects appear rushed to make their decisions? How did the mechanics of controlling the system influence the subjects? The presence of the experimenter also ensured that the subjects effectively participated in the verbal protocols.

In addition to the verbal protocols, a paired t-test was used to evaluate the session scores for the subjects. This evaluation examined the 'controllability' of the GT-FMS environment.

4. Results and discussion

4.1. Statistical analysis

If the subjects could not influence system operation, then the system response to the subjects' interventions might be limited, potentially biasing the subjects' decisions. Thus, a paired t-test was performed to test whether, on average, the human-supervised system performed differently than the weighted operation priority index scheduling system running without human intervention. The fully automatic control used several values of alpha (0.1, 0.9 and 1.0). For each level of alpha, a paired t-test was performed comparing the data from 64 subject runs with the data of fully automatic scheduler runs. The subject data were normalized, and the normalization value, d, was calculated by subtracting the session score for the automatic scheduler from the score for each subject. The paired t-test provided 63 degrees of freedom for the error term, and was calculated with the hypothesis that $d = 0$ for each value of alpha in the scheduler.

The t-test indicated that for alpha values of 0.9 and 1.0, the subjects and the scheduler performed significantly differently, and for an alpha value of 0.1, the subjects and the scheduler performed about the same. Figures 4, 5 and 6 compare the high, low and average subject score to the scheduler with alpha values of 0.1, 0.9 and 1.0. Significantly, even for an alpha value of 0.1, the high subject score is always higher than the automatic scheduler score.

Although the subjects did not seem to clearly outperform the scheduler at an alpha level of 0.1, the data do indicate that the subjects could influence system operation. Thus, the subjects' decision processes were most likely not biased by an unresponsive system and provide a good basis for evaluation.

4.2. Decision ladders

The verbal protocols were prepared by converting recordings obtained during the experimental sessions into a printed text for each subject. The next step was to use the transcripts from the verbal protocols and the computer data files to construct, for every subject, a decision ladder for each allowed control action. The data log was used to indicate what information was displayed to the subject at the time of the intervention. The verbal protocols and computer output files for every subject were then examined one session at a time. For a given intervention (i.e. either expedite or scheduler modification), each intent stated by a subject was reviewed. Coupled with the computer output data, distinct

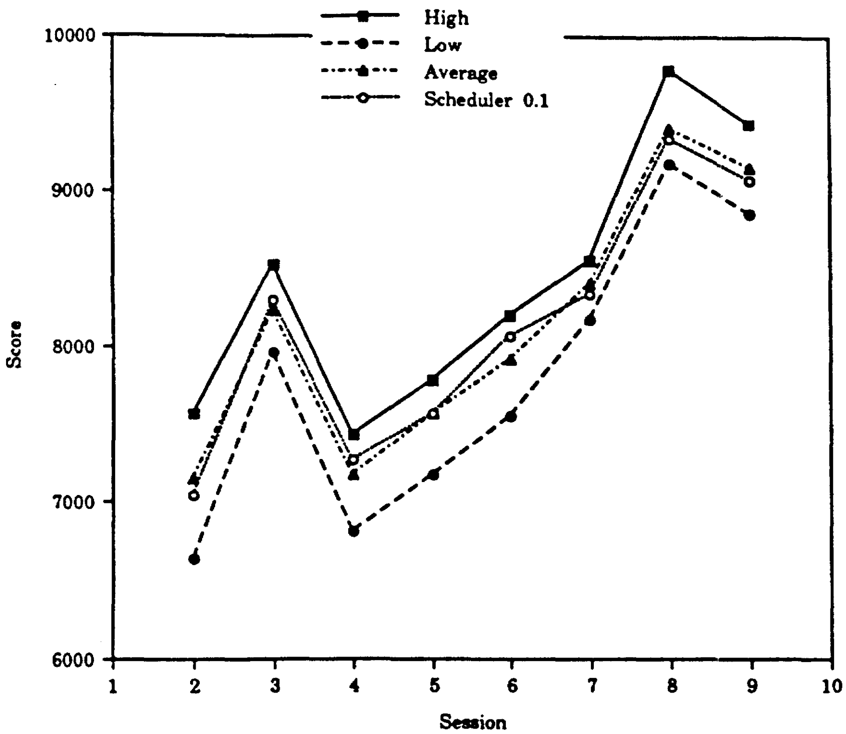

Fig. 4. High, average, and low subject scores vs. scheduler with alpha = 0.1.

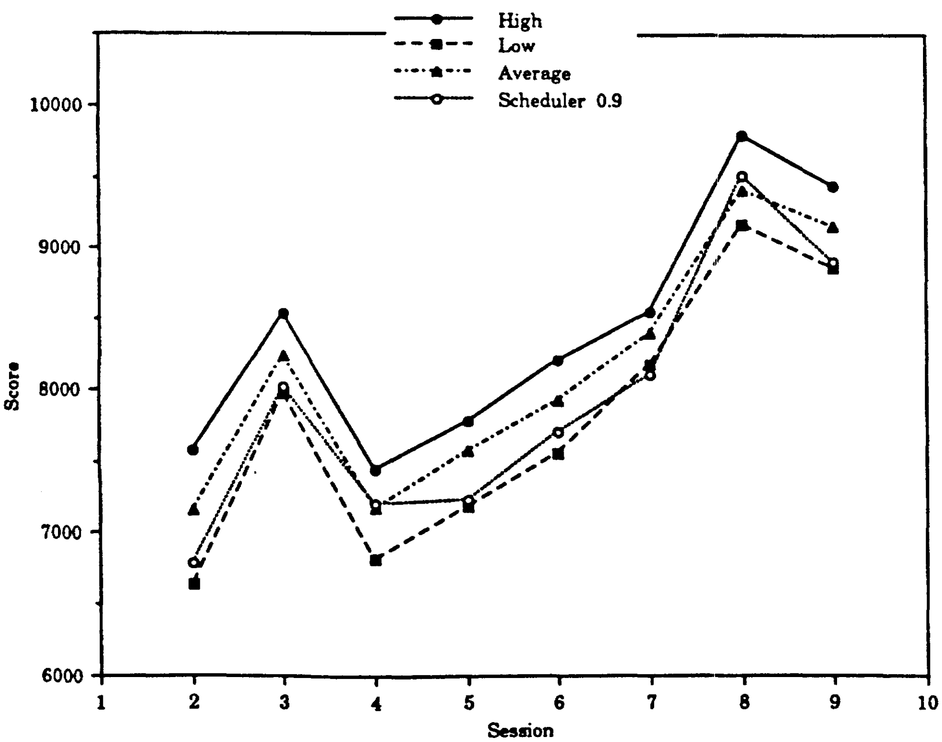

Fig. 5. High, average, and low subject scores vs. scheduler with alpha = 0.9.

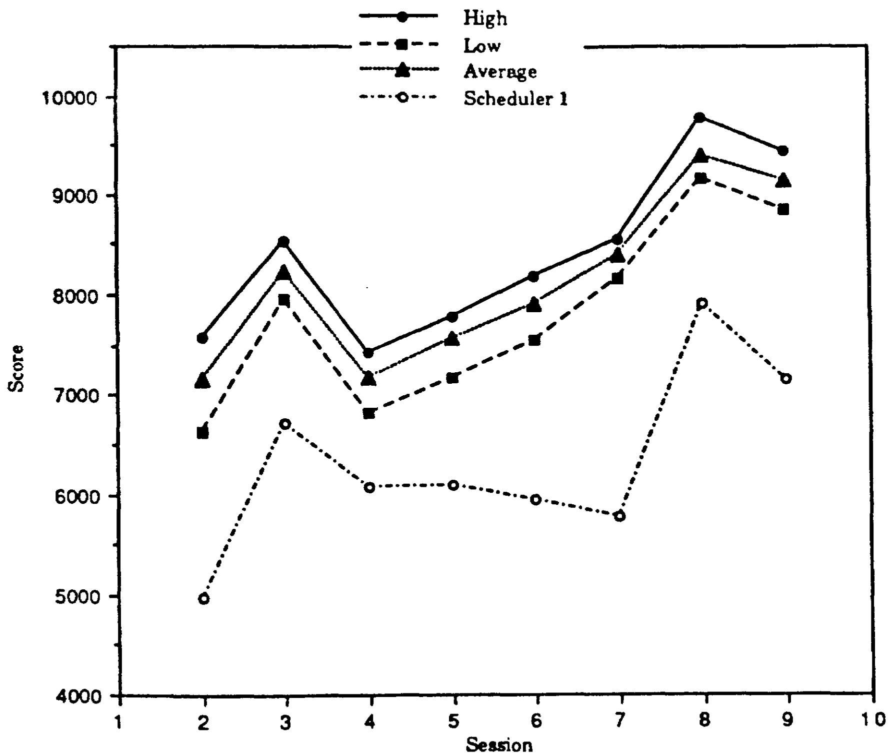

Fig. 6. High, average, and low subject scores vs. scheduler with alpha = 1.0.

consistencies among the stated intents from the protocols were recorded. When all of the sessions for a subject were completed, consistencies across sessions were then evaluated. As situations were repeated throughout a session, and across several sessions, the decision processes were broken down and mapped onto the decision ladder. This mapping was done separately for each strategy.

4.3. Results

Discussion of the results of the decision ladder analysis is organized by the type of control decision (e.g. either expedite or scheduler modification) and follows the outline of data processing activities, from 'Activate' to 'Execute', in Rasmussen's decision ladder.

4.4. Expedite decision

For the expedite decision, subjects sought to define a state of knowledge based on the most recent information available for each order. The order status display was the predominant choice of the subjects to provide this information. Six of the subjects used this display exclusively. Two of the subjects reviewed the penalty cost and throughput display regularly, yet their expedite decisions were also based solely on existing conditions and not on performance history information.

The subjects' approaches were consistent. They distilled system information from the order status display into a state of knowledge described as the current system state. The current system state, once determined by the subject in response to the latest update of the order status display, was the basis for the remainder of the decision process.

Although the subjects generally used the same display for defining the current system state, the information extracted from this display varied significantly among the subjects. For instance, a few subjects used the existence of an expedited order to define a system state. Thus, one of the

'Set of Observations' states of knowledge used by these subjects consisted of the presence of an expedited order. Some subjects characterized a system state by the presence of orders that had only the load/unload operation remaining, while others used the elapsed simulation time to define the system state. The elapsed simulation time was used to determine a system state described as 'end of session was near'. This system state changed the expedite decision for some subjects for a short period of time (this is discussed in more detail later for individual subjects).

Even though the information obtained from the order status display varied, some consistent patterns emerged. For example, all of the subjects used the order due date information to define a system state of either 'tight' or 'loose' due dates. Most subjects also evaluated whether operations that required long processing times (i.e. 'long' operations), particularly operation 5, or operations that required small processing times (i.e. 'short' processing operations) were characteristic of the current system state. As the sessions progressed, this evaluation of processing operations became more frequent as subjects concluded that operation 5 was difficult to complete on time. In addition, most subjects defined a system state that had at least one order already late or projected to be late.

Rasmussen [21, 22] noted in his evaluation of verbal protocols from a power plant control room that the entire decision-making process described by his decision ladder was only used when the operators were faced with new or unfamiliar situations. In most cases, the operators developed a 'sequence for special situations by chaining subroutines of general applicability and using solutions from prior experience... leading to a great repertoire of short-cuts and by-passes in the decision process...' [21, p. 144]. Rasmussen describes these subroutines as chained states of knowledge, where the subjects move from one state of knowledge directly to a task or procedure, bypassing portions of the decision process. Chained states of knowledge result in consistent actions based on well-defined system states. However, for GT-FMS these chained states of knowledge were rare. Subjects generally proceeded through all data processing stages of the decision model, evaluating each system state against the system goals, even when faced with seemingly familiar situations (possible causes of this behavior are discussed later).

Figure 7 illustrates an example of a decision strategy for the expedite control action which uses a chained state of knowledge. In this decision sequence, the subject defined a system state, then proceeded immediately to a task definition. The task 'expedite the order with one operation remaining', is an immediate consequence of the system state definition. Figure 8 illustrates a typical expedite decision ladder without chained states of knowledge. In this decision ladder, tasks are defined and implemented as a result of a system performance evaluations, goals, and target states.

Although the subjects generally proceeded through the entire decision ladder, some exceptions to this pattern emerged. Figure 9 displays the expedite decision ladder for subject 2. As indicated by Fig. 9, in most cases subject 2 did not proceed through the complete evaluation/goal steps of the decision model, but rather reacted, according to chained states of knowledge, to the system states she defined. In addition, the system state defined as 'end of session is near' caused several subjects to by-pass the evaluation/goal sequence and to immediately modify their interventions. These subjects stopped evaluating late orders and repeatedly expedited parts remaining with operations that had short processing times once this system state was defined.

Since subjects used the entire realm of data processing activities for expedite interventions, they repeatedly evaluated system performance based on the system states they identified. Again, some consistencies emerged from these evaluations. For example, most subjects evaluated whether any of the orders currently in the FMS would be late. Likewise, determining which of the current orders, if completed past their due date, would yield the highest penalty cost was also common. Subjects focused primarily on order size in making this judgment. Most subjects also evaluated which of the orders had the earliest due date.

Although some system performance evaluations were common, most of the evaluations

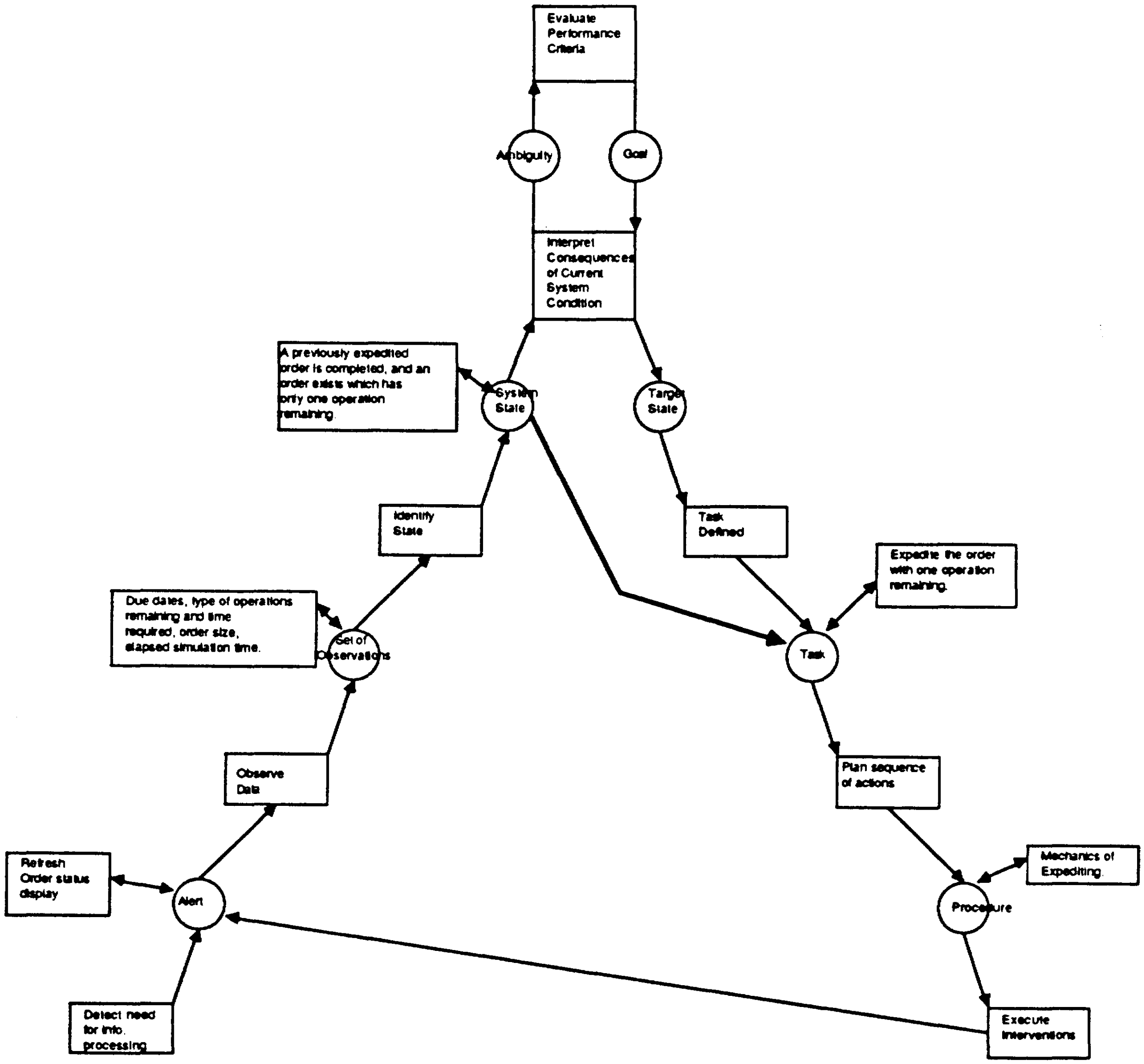

Fig. 7. Strategy for expedite control decision, chained states of knowledge.

were unique to each subject. However, even these unique evaluations were consistent in that they evaluated a very detailed level of system performance. For example, subject 4 determined whether recent incoming orders would be delayed at the load/unload station by leaving an order expedited, or, whether the expedited order could still be completed sooner if it remained expedited. If an order was already late, subject 5 evaluated whether another order, with only a small processing time remaining, should be expedited before the late order was done. The detailed evaluations of system performance reveal a high degree of confidence among the subjects in their ability to precisely determine and predict the state of the system.

These detailed evaluations may have been partially responsible for the infrequent occurrences of chained states of knowledge. Rather than react to a system state that appeared 'familiar',

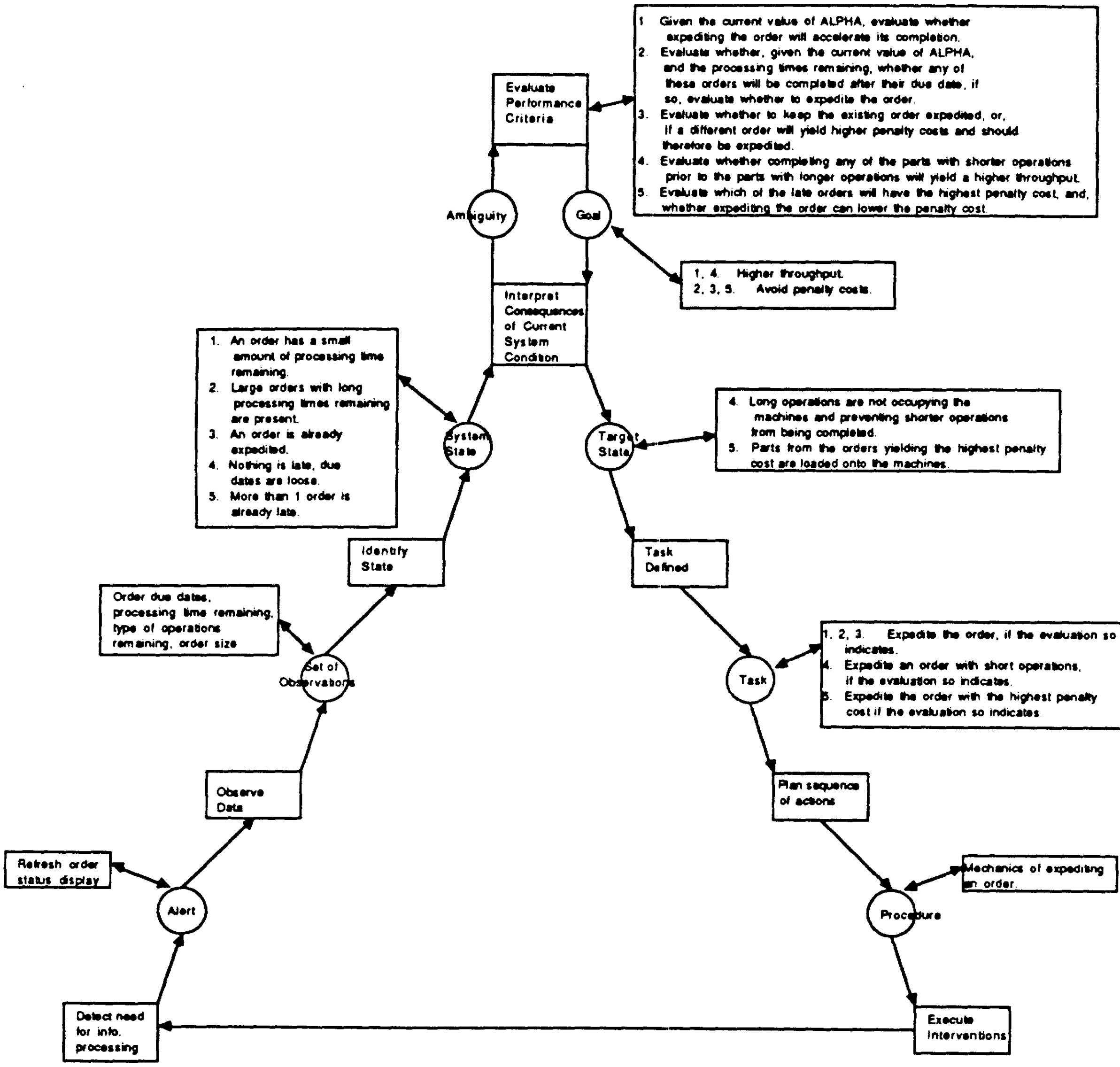

Fig. 8. Expedite decision ladder with no chained states of knowledge.

the subjects attempted to gain a more thorough knowledge of system performance.

The repeated system performance evaluations also required an on-going comparison of these assessments against system goals. Even though the overall system goal was given to each subject (i.e. maximum profits) during the training, the subjects' system goals were not identical. Subjects formulated their own goals. For example, subject 4's stated goal was to minimize the number of late orders, whereas subjects 5, 6, and 8 identified 'avoiding all penalty costs' as their goal. These differences in system goals resulted in varying strategies of operation.

Even though the subjects' goals varied, the primary focus for all subjects involved penalty costs. Completing parts, or throughput, was definitely a secondary goal for the subjects. Half of the subjects did identify 'maximizing throughput' as a goal, and this goal influenced the expedite

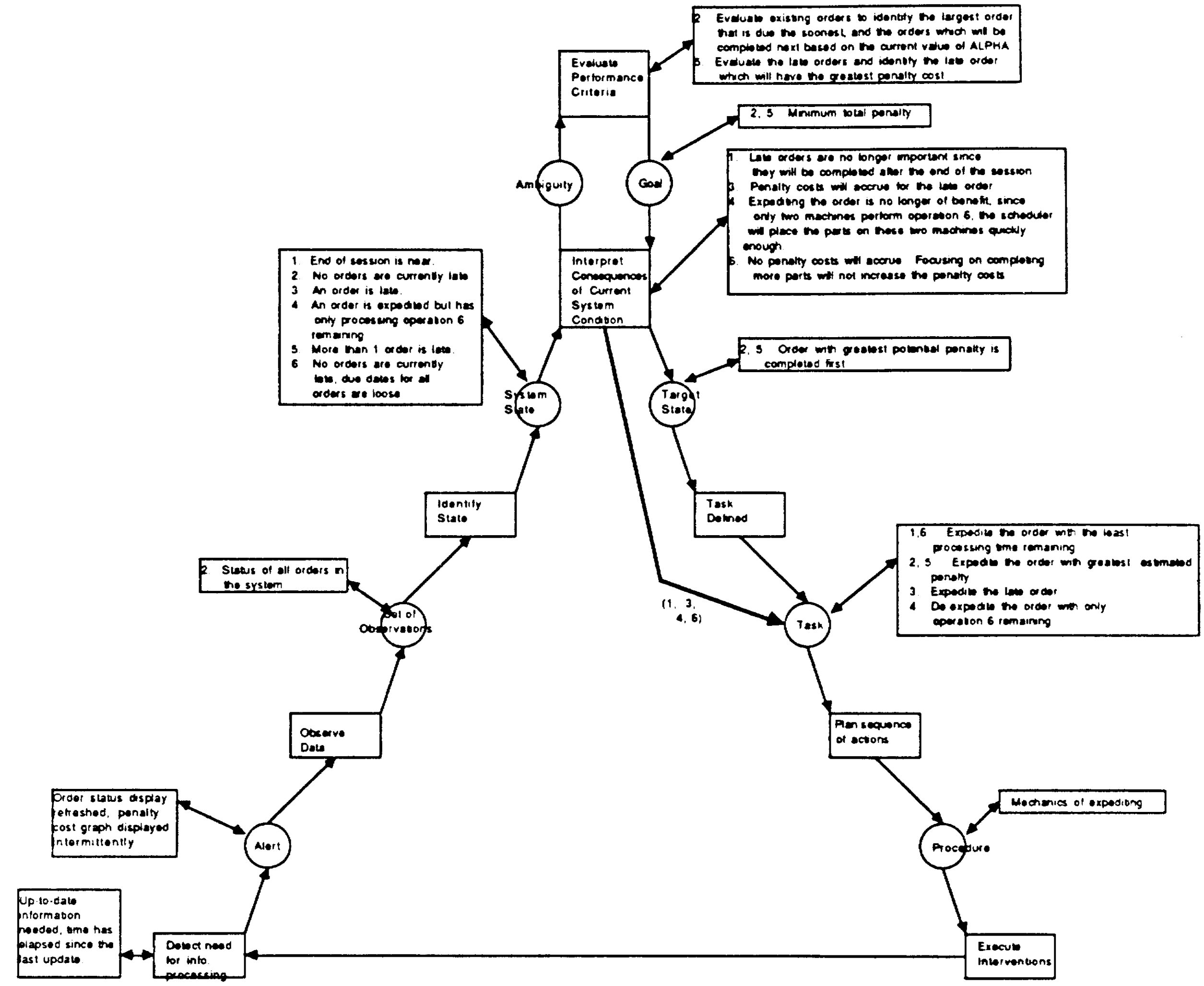

Fig. 9. Expedite decision ladder, chained states of knowledge dominant.

interventions in some cases for these subjects. However, this goal was always secondary to goals concerning penalty costs.

The target states defined by the subjects resulted directly from their evaluations and goals. The target states consistently involved an assessment of processing priority, i.e. deciding which types of orders or which specific order should be processed next, given the system goals, performance and current state. In addition, the target state consistently involved only the current system orders. Subjects did not anticipate the arrival of certain orders or existence of certain conditions when using the expedite intervention.

Like the subjects' system performance evaluations, target states were frequently very detailed rather than general. For example, subject 6, in response to certain system conditions, defined a processing sequence that assumed 'long operations occupy the machines while short operations are in the system, and thus prevent processing of the shorter operations'. Subject 5 defined as a target state that 'operation 6's from nearly completed orders are finished prior to the operation 6's from new orders'.

Expediting an order was almost always the task that resulted from proceeding through the data processing activities for the expedite intervention. In fact, most subjects always had an order expedited. However, in response to certain system conditions and performance evaluations, several subjects re-initiated the expedite decision process and maintained the status quo rather than expedite an order. These subjects by-passed the 'Formulate' and 'Execute' data processing activities and the 'Procedure' state of knowledge, proceeding directly to the 'Activation' data processing activity in the decision ladder. The decision process was re-initiated by defining a new system state based on more recent information. For subjects 3 and 4, unlike their peers, the status quo was usually no orders expedited. These subjects had a much more limited set of system states and resultant evaluations, which concluded with expediting an order.

For all subjects, the decision analysis for the expedite intervention was continuous throughout each session. Once a task was defined and a procedure (if any) implemented, the subjects seemed instantaneously to proceed to the 'Alert' state of knowledge – updating the main order status screen and thereby re-initiating the entire decision process. Subjects 2 and 8 also checked their performance history on the penalty cost and throughput display – for varying lengths of time – prior to refreshing the order status display. Overall, the subjects varied considerably in their speed of processing. Processing time was, in all cases, dependent on the significance of the changes in system state since the last expedite decision sequence. More changes generally implied an increased processing time.

4.5. *Modifying the scheduler*

In terms of the variety of system states defined, performances evaluations, goals and target states, the scheduler modification intervention was much less complex than the expedite intervention. Subjects modified the scheduler (by changing the value of the weighting factor, alpha) much less frequently than they expedited orders, so this decision process itself was initiated less frequently.

As with the expedite intervention, information was primarily obtained from the order status display. Although all subjects viewed the cost summary display intermittently, they did not base their interventions on information from this display. Subjects generally proceeded through the alert stage of the decision process once, for both the expedite decision and the scheduler decision, but then proceeded through the remainder of the process separately for each type of intervention.

Due date status was the one type of information obtained from the order status display that was consistent among the subjects. Again, as with the expedite intervention, most subjects defined a system state based on their assessment of due dates as either 'loose' or 'tight'.

Other than due dates, however, subjects used the order status display to obtain a variety of information. Based on this varying information, many different system states were defined. For example, a few of the subjects defined the beginning of the session as a system state, using the elapsed simulation time information, and modified the weighting factor immediately from its default value. Other subjects focused on the type of operations currently in the system while some associated due dates with the type of operations (e.g. 'the long operations have tight due dates') in defining their system state. Also, as with the expedite intervention, several subjects used the elapsed simulation time to define a system state as 'the end of session is near' and modified the scheduler to reflect this information.

Unlike the expedite intervention, the subjects modified the scheduler using chained states of knowledge and fewer evaluations of system performance versus goals when deciding to modify the scheduler. For example, Fig. 10 illustrates a chained state of knowledge for the scheduler modification decision of subject 3. While subject 3 did interpret the consequences of the current system state, the evaluation/goal and target state steps were by-passed. No projections of system performance, based on the current defined system state, were attempted and a target state relative to system goals was not defined. All of these chained states of knowledge involved associating low values of alpha with 'tight' due dates and higher values of alpha with 'loose' due dates. These associations either resulted in immediate changes of alpha or a re-initiation of the entire scheduler modification decision process if alpha was already at the desired value.

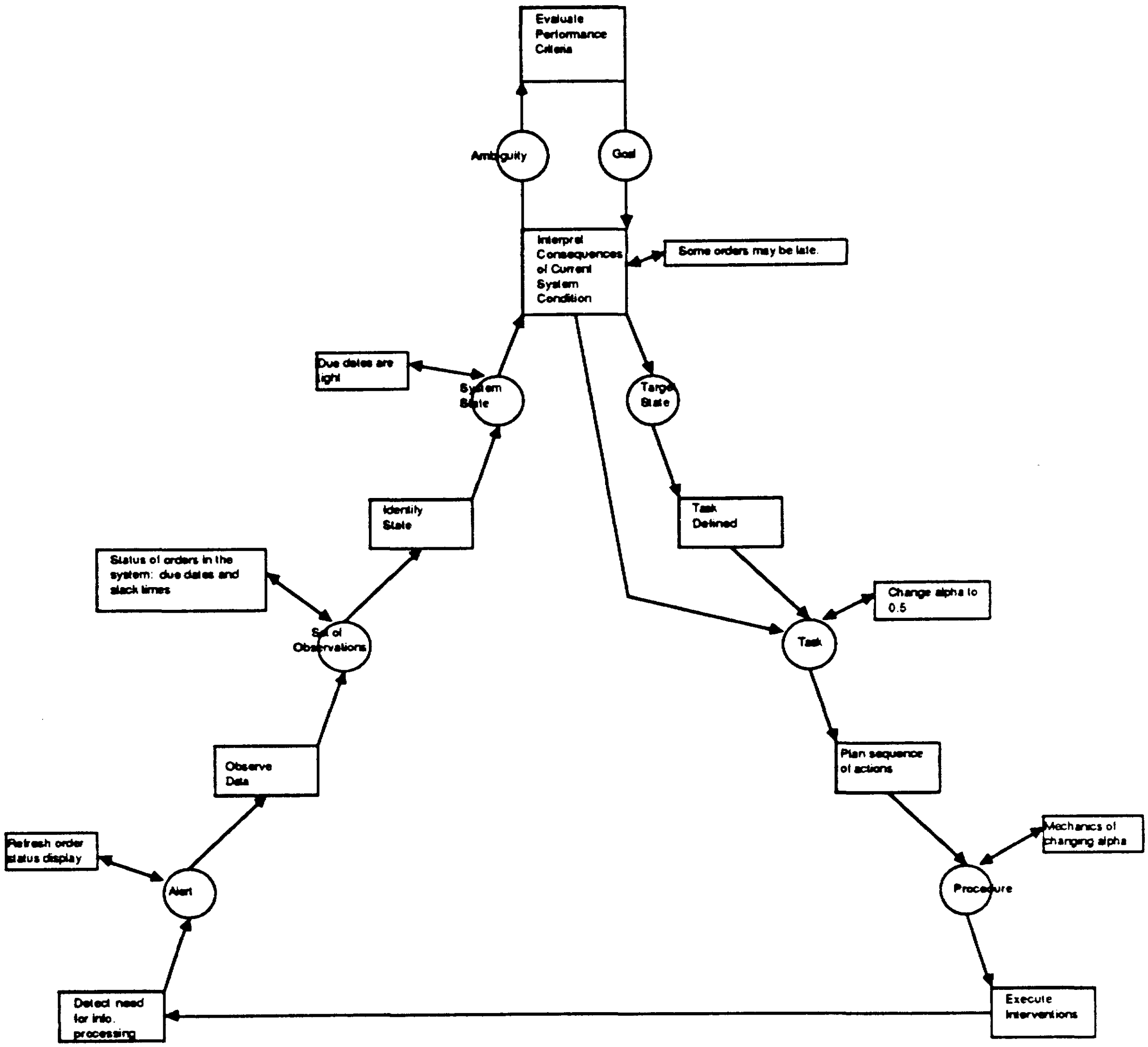

Fig. 10. Automatic scheduler decision ladder, chained states of knowledge.

Even though chained states of knowledge were more common for the scheduler intervention, some evaluations did occur frequently. For example, all of the subjects evaluated the impact of raising alpha to 1.0 or 0.9 on system throughput, and most of the subjects evaluated whether they should test the sensitivity of the system to changes in the weighting factor, alpha. Also, subjects' evaluations, as with the expedite intervention, were generally very detailed. For example, subject 2 evaluated whether raising alpha to prevent parts with a long operation time from being loaded onto the machines would increase penalty costs by making the parts with long operations late or would increase throughput by completing additional parts.

When evaluations of system performance were made, the subjects once again had to incorporate system goals to guide their interventions. These goals, consistent with the expedite intervention, were not identical to the overall goal presented in the training and varied among the subjects.

For instance, three of the subjects identified 'high' throughput as a goal, while other subjects identified 'avoiding penalty costs' as a system goal.

Post-evaluation interpretations and target states focused on processing priority, but, unlike the expedite intervention, this focus was on processing priority to specific groups or classes of parts. The most common focus was on a processing priority based on due dates and processing times.

The task defined was always either to change alpha, based on an evaluation or on chained states of knowledge, or to maintain the status quo if a system state was not identified which triggered a scheduler intervention. Subjects generally mirrored the expedite intervention in that they went from the procedure state of knowledge directly to an alert condition and re-initiated the decision sequence by updating the order status display.

Overall, modifications to the scheduler occurred much less frequently than expediting, and most subjects changed alpha less frequently as they completed more sessions. Generally, these changes involved placing alpha at values of 0.0 and 0.1 when a low value was indicated, or placing alpha at 0.9 or 1.0 when a high value was indicated. The subjects' detailed evaluations of system performance revealed a knowledge of the scheduling algorithm and its impact on the FMS, and the subjects reinforced this knowledge by continuing to test the sensitivity of the system to changes in alpha.

4.6. User interface and subjective assessment of subject performance

Table 1 illustrates some general subject performance statistics. The table indicates the number of interventions, over all data-collecting sessions, for each subject. Control interventions that were not included in a subject's decision ladders are displayed in Table 1 as a percentage of the total interventions. A control intervention was not included in a decision ladder if the verbal protocol was unclear or the strategy was unique or unclear. 'De-expedite' control interventions were infrequent and were also not included in the subject's decision ladders (de-expedite interventions are included in the number of interventions listed in Table 1, however).

4.7. Subjective evaluation

In addition to the analysis of the verbal protocols completed for each type of intervention, several general impressions of the subjects' performance were noted as the sessions progressed. For example, the subjects did not seem rushed or hurried in their decisions as they monitored the system. Even though, as previously mentioned, the processing time required for each decision varied among the subjects, all of the subjects seemed to have enough time to proceed through their decision sequences.

Although goals differed, the primary focus of each subject was penalty costs associated with late orders. Throughput was always secondary.

Table 1
Control decisions: Frequency and consistency

Subject	Interventions			% Not represented in decision ladders
	Expedite	Scheduler	Total	
1	112	72	184	16
2	253	107	360	23
3	79	25	104	9
4	62	15	77	18
5	170	46	216	25
6	156	23	179	7
7	173	31	204	17
8	202	93	295	14
AVG	151	52	202	16

While some subjects reversed this focus in response to a system state of 'end of session is near', this re-focus was, in almost all instances, much too close to the end of the session, usually with just a couple of minutes remaining, to have any significant impact.

While, as discussed earlier, the subjects' evaluations of system performance were usually very detailed, all of the subjects indicated that they thought they could have done even better if they had known more details concerning the system operation. Specifically, they sought to know which parts were on the machines, which parts were on the material handling system, which machines had failed, and an estimate of machine repair time. The subjects thus did not request more performance information, such as past scoring histories, score-to-date, etc. but rather more detailed status information. This response is not surprising given the type of status information used by the subjects to define system states.

4.8. Summary

The goal of this research was to gain further insight into human decision-making within an FMS environment. In the environment simulated by GT-FMS, human systems managers are allowed to intervene in the FMS in two ways: expediting an order, which gives an individual order processing priority, or modifying the scheduling algorithm, which gives groups of parts processing priority. Evaluating the decisions made by the systems managers in implementing these two interventions has provided some insight for FMS supervisory control systems.

First, the subjects in this experiment interacted with the FMS on a very detailed level. Even though their role in this simulation placed them at a higher level of control, providing them primarily with summary information and performance history, the subjects evidently needed and wanted more detailed information on system performance. Performance history and trend information were not factors in their decisions. Furthermore, even though they were not provided with detailed system status information, they were still able to make very detailed evaluations of system performance and incorporate these evaluations into effective control strategies. Chained states of knowledge, where the human reacts to certain, standard system states, were not as common as might have been expected based on the limits of the subjects' interventions and Rasmussen's [21, 22] results. The subjects continued to prefer thorough evaluations, incorporating as much evidence as they could obtain, versus reacting to standard system states based on the summary information. They also seemed more concerned with the situation at hand, and how they could best influence this situation, rather than incorporating or trading-off their current decisions as part of a long-term performance strategy. Evidently, while humans may be effective as part of a higher or aggregate level FMS control system, they still prefer having access to detailed knowledge of lower level system components.

In addition, the subjects were able to understand and control the scheduling algorithm, even though, again, they were primarily given system summary information. They continued to test the sensitivity of the scheduler when they felt this testing did not conflict with current system goals. This testing occurred throughout the sessions, indicating that the subjects were continuing to learn more about the dynamics of the scheduler as the sessions progressed. Still, in ten total sessions, the subjects seemed to use the scheduling modification intervention effectively. Additional sessions, or, in the case of an actual FMS system, perhaps months of training, would probably increase the systems manager's understanding of scheduling dynamics even more. In actual FMS installations, the tendency may be to exclude the human from the operation of the scheduler, yet the results of this research indicate that this may not be the best approach.

Even though the subjects were able to interact with the system and to understand the details of its operation, they were inconsistent in defining their goals. This is especially important, given the subject's emphasis on thorough evaluations, since these evaluations depend on goals to determine appropriate actions. Subjects defined their own goals, and then based their intervention strategies on them. Commonly, the system goal of 'maximum profits' given to the subjects as part of their training evolved into 'avoid penalty costs and disregard throughput' as the subjects monitored the FMS.

Based on the variances in goals and the focus of their interventions, the subjects evidently had difficulty focusing on both aspects of the profit function (i.e. throughput and penalty costs) and consistently placed a priority on penalty costs. However, a change in goals, which made throughput a priority, often occurred at the end of the sessions. While this change was too late to be useful, it does seem to indicate that the subject understood the session scoring mechanism, and, furthermore, that they felt that they did have control over throughput by using the expedite or scheduling change interventions.

As discussed earlier, a prize was awarded to the subject with the highest session score total for the eight data-collecting sessions. None of the subjects consistently outscored the others, and no subject scored consistently lower than the others. The subject with the highest overall score, however, was able to win the prize by significantly outscoring his peers in one of the sessions. In this session, he emphasized *both* throughput and penalty costs by using the expedite control to place a priority on short processing operations while keeping alpha low. Most subjects only made throughput a priority during a session if they concluded that none of the orders in the FMS would be completed past its due date. This strategy resulted in placing a priority on throughput for infrequent, brief periods of time which were too short to be useful.

This seemingly natural tendency to emphasize timeliness in manufacturing and the variability of goals present across the subjects in this experiment underscores the need for system goals to be constantly reinforced, as part of the control system itself, in FMS supervisory control. The results also seem to indicate that multiple system goals (e.g. high quality, low cost, high output, etc.) may be ranked in supervisory control systems. The supervisor may assign weights to the goals based on his or her own biases and control the system accordingly. Continued reinforcement of system goals may ensure more consistency of strategy among several supervisors and between designers and the humans who ultimately control the systems. Meeting due dates, while obviously important, is likely to be only one of several important aspects of performance in an FMS installation.

The subjects generally proceeded through the entire decision process, evaluating system performance against defined goals and formulating a target state, for each intervention. Chained states of knowledge were more common for the scheduler intervention, but overall most subjects evaluated each situation independently. The subjects' system performance evaluations were dependent on an intricate knowledge of the system state and often included a measure of the sensitivity of the system to a given intervention. The subjects further continued to test the sensitivity of the system throughout the sessions. The subjects' approach seems to contradict the philosophy of including human decision-makers in the control system solely under alarm conditions and supports making them an integral part of the manufacturing control process. Allowing human intervention under alarm conditions only might force the supervisor to forfeit the opportunity to track system sensitivity or make an evaluation based on an intricate knowledge of the system state, background knowledge that was crucial to the subjects' decisions in this investigation. Detailed system status information could be provided to a human who monitors a manufacturing system. However, it is doubtful in this experiment that if the subjects had only monitored, and not intervened in any way, that they would have attempted to gain the detailed knowledge of the system operation they achieved.

In addition, the subjects' approach supports the use of simulation techniques to help human supervisors of FMS control systems. Simulation, with appropriate feedback, allows the supervisor to test the sensitivity of the system to possible interventions, thus potentially improving the evaluation process of an intervention decision.

Since the subjects used current, detailed system status information, and not performance information or performance history, as the basis for their decisions, expanding the status information available to them may have improved their performance. For example, information concerning average flowtime by part type may have improved the accuracy of the subjects' evaluation of 'tight' or 'loose' due dates and the subsequent system state definition. This type of information can be contrasted in an actual FMS installation, for example with providing data such as machine utilization or overall tardiness performance. Thus, feedback mechanisms for supervisory control systems may actually enhance a human's performance by emphasizing the data types that were important to the subjects.

5. Conclusion

Flexible manufacturing is a philosophy which can greatly enhance the overall productivity of small-lot or batch manufacturers. As the development of computers and manufacturing automation has accelerated, FMS installations have become more common, more versatile, and more complex.

The increasing complexity of the control systems required by modern FMSs has resulted in numerous research efforts that address control structure design. While no single design is 'optimal' for every, or even most, FMS installations, effective control system designs are often characterized by a hierarchy of several information-sharing control levels that incorporate many manufacturing decision-making functions.

Since human judgment is critical to manufacturing decision-making, human intervention is a component of many control system designs. The implementation of human intervention, however, varies significantly among FMS control structures, and often occurs on an ad hoc basis. Previous supervisory control research has indicated that overall FMS performance can be enhanced if the ad hoc nature of human intervention is removed. Thus, human intervention is likely to remain as an important aspect of FMS control policies.

As FMS control systems become more 'intelligent', the role of the human in the control structure will also evolve. An FMS may become more of a tool for the human who controls it, with the human responsible for achieving system goals. Thus, defining an appropriate role for the human as a function of the FMS objectives may be critical in the design of the FMS control structure. This role should both respect the limitations of the human and exploit the human's inherent skills. Knowledge of the decision processes used by humans in an FMS environment can help define this role.

This investigation evaluates the decision processes of humans in an FMS environment. The experimental results support making humans an integral part of the manufacturing control process, since an intricate knowledge of the system state and system sensitivity were crucial to human decision-making in GT-FMS. Furthermore, even when the goals for FMS operation were specified in terms of a single aggregate measure, namely the overall profit, humans in this experiment used detailed system status information, rather than summary information or system performance history, as the basis of their control decisions.

Acknowledgement

This research was supported in part by an IBM department grant (Dr. Stan Belyeu, technical monitor).

References

[1] R. Akella, Y. Choong and S.B. Gershwin, Performance of hierarchical production scheduling policy, IEEE Trans. Components, Hybrids, and Manufacturing Technology CHMT-7 (3) (1984) 225–240.

[2] J.C. Ammons, T. Govindaraj and C.M. Mitchell, Decision models for aiding FMS scheduling and control, IEEE Trans. Systems, Man and Cybernetics 18 (5) (1988).

[3] J.C. Ammons, T. Govindaraj and C.M. Mitchell, A supervisory control paradigm for real-time control of flexible manufacturing systems, Ann. Op. Res. 15 (1988) 313–335.

[4] J.C. Ammons, and L.F. McGinnis, Flexible manufacturing systems: An overview for management, in: J.A. White (Ed.), The Production Handbook (Wiley, New York, 1985).

[5] C. Benson, The use of single-page, direct-manipulation interfaces in real-time supervisory control systems, M.S. Thesis, Center for Human-Machine Systems Research, Georgia Institute of Technology, Atlanta, GA, Report 89-3 (1989).

[6] J.A. Buzacott, Optimal operating rules for automated manufacturing systems, IEEE Trans. Automatic Control AC-27 (1) (1982), 80–89.

[7] J.A. Buzacott and D.D. Yao, Flexible manufacturing systems: A review of analytical models, Management Sci. 32 (7) (1986) 890–905.
[8] Draper Laboratory. Flexible Manufacturing Systems Handbook (Noyes, Park Ridge, N.J., 1984).
[9] O. Dunkler, Human aided control of a flexible manufacturing system, Technical Report 87-2, Center for Human-Machine Systems Research, Georgia Institute of Technology (1987).
[10] O. Dunkler, C.M. Mitchell, T. Govindaraj and J.C. Ammons, The effectiveness of supervisory control strategies in scheduling flexible manufacturing systems, IEEE Trans. Systems, Man, and Cybernetics 18 (2) (1988) 223–237.
[11] B. Eagle, The design and application of the weighted operation priority index dispatching rule for discrete-interval human supervisory control of a flexible manufacturing system, Proposal to the Faculty and CIMS Committee, School of Industrial and Systems Engineering, Georgia Institute of Technology (June 1987).
[12] K.A. Ericsson, and H.A. Simon, Protocol Analysis, Verbal Reports as Data (MIT Press, 1984).
[13] S.B. Gershwin, R.R. Hildebrant, R.R. Suri and S.K. Mitter, A control perspective on recent trends in manufacturing systems, IEEE Control Systems Magazine (April 1986) 3–14.
[14] J.S. Greenstein and R.A. Wysk, Human–computer interaction in automated manufacturing systems, Proceedings of Computer Aided Manufacturing-International's 9th Meeting of Members, Dallas, Texas (1980).
[15] M.H. Han and L.F. McGinnis, Due dates in manufacturing shops: Unweighted case, Working Paper, Material Handling Research Center, Georgia Institute of Technology (Nov. 1986).
[16] D.A. Hettenbach, An investigation of decision making in supervisory control of a flexible manufacturing system, Technical Report 89-2, Center for Human–Machine Systems Research, Georgia Institute of Technology (March 1989).
[17] G.K. Hutchinson, The control of flexible manufacturing systems: Required information and algorithm structures, IFAC Symposium on Information-Control Problems in Manufacturing Technology, Japan (1977) 285–292.
[18] J. Kimemia and S.B. Gershwin, An algorithm for the computer control of a flexible manufacturing system, IIE Trans. (Dec. 1983) 353–362.
[19] S.Y. Nof, A.B. Whinston and W.I. Bullers, Control and decision support in automatic manufacturing systems, AIIE Trans. 12 (2) (1980) 156–169.
[20] P.G. Ranky, FMS in CIM, Robotica 3 (1985) 205–213.
[21] J. Rasmussen, Strategies of state identification and diagnosis in supervisory control tasks, and design of computer based support systems, in: W.B. Rouse (Ed.), Advances in Man–Machine Systems Research, Vol. 1 (JAI Press, Inc., 1984) 139–194.
[22] J. Rasmussen, Information Processing and Human–Machine Interaction, New York: (North-Holland, New York, 1986).
[23] J. Rasmussen and L.P. Goodstein, Decision support in supervisory control of high risk industrial systems, Automatica 23 (5) (1987) 663–671.
[24] E.M. Roth, K.B. Bennett and D.D. Woods, Human interaction with an 'Intelligent' machine, Int. J. Man–Machine Studies 27 (1987) 1–47.
[25] W.B. Rouse, Intelligent decision support for advanced manufacturing systems, Manufacturing Rev. 1 (4) (1988) 236–243.

Effectiveness of direct manipulation interaction in the supervisory control of flexible manufacturing systems

Charlene R. Benson *, T. Govindaraj, Christine M. Mitchell and Stephen M. Krosner **

Center for Human–Machine Systems Research, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205, USA

Received May 1990
Revised August 1990

The use of direct manipulation interface technology may be a valuable design technique for operator interaction and control with real-time, supervisory control systems. To evaluate the effectiveness of a direct manipulation interface, two user interfaces to a simulation of a flexible manufacturing system were developed. One was a conventional, alphanumeric interface and the other was a direct manipulation, graphical interface. Five measures that reflected operator performance were analyzed. Subjects using the direct manipulation interface operated the system significantly better on three of the five measures.

1. Introduction

Over the past twenty years a new type of manufacturing system has emerged: the flexible manufacturing system (FMS). An FMS is a network of workstations, buffers and a material handling system that has recently been combined with computer control leading to many partially or totally automated FMS installations (e.g. [5, 9]). With its versatile workstations requiring minimum changeover time between different operations and a material handling system capable of executing any desired job routing [2], FMS has shown major advantages for medium variety/medium volume markets [1]. These advantages include quicker response times to market change, reduction in work-in-process inventories, shorter lead times, reduction in space requirements, and more efficient material handling and manufacturing control [1, 6, 16, 23].

The real-time control problem in FMS is to achieve changing production goals in spite of many complications. Those complications include limited resources, random machine failures, unavailability of production materials, item movement within the system with respect to a specific material handling configuration, and dynamic item release within the system. As a result of these components, FMS scheduling and control is a problem that must be solved repeatedly and rapidly [1].

The ultimate goal of some manufacturing systems designers is total automation. It has recently acquired the title of the 'lights out factory' [9]. This refers to a manufacturing system in which no human intervention is necessary. However, Shaiken [19] reported that FMS facilities experienced downtime as high as 60%. And because the complexity of the system often contributes to its unreliability, it is much more likely that increased implementation of automation will lead to changes in the numbers and skills of workers on the shop floor, rather than the elimination of people [9, 18]. The factory of the future will include a human decision-maker who monitors the system in real time and fine tunes the control process to adapt to the changing system state and production goals using problem-solving capabilities to enhance system performance [18].

Dunkler et al. [7] performed an experiment in which the performance of an automatically controlled flexible manufacturing system was compared with the performance of the system augmented by human supervision and intervention. Two different control algorithms were used, first-come-first-served (FCFS) and shortest processing time (SPT). The goal of improving overall system performance was achieved by meeting due date and minimizing inventory. The experimental results showed that with human supervision both due date and inventory performance of the FMS were improved. The results supported the idea of actively integrating humans into operational controls of automated manufacturing environments. Dunkler et al. [7] also mentioned the need to enhance the graphical interfaces used to control the FMS. It was suggested that pages or windows be used to integrate all of the information necessary to perform one or more of the operator's control tasks.

Although most analytic models and artificial intelligence models for FMS scheduling and control assume the presence of a human operator to monitor and supervise the system in real time [8, 10], most do nothing more than recommend the design of an 'appropriate' operator workstation [1]. There must be a more specific approach to the design of the operator workstation. Typically, information displays provided to real-time decision-makers are 'data dumps' where a programmer unfamiliar with the operator's tasks designs information displays that display all possible accessible data with the premise that it might be needed at some time – frequently at the lowest level possible [14, 17, 18]. This is an unacceptable approach to the design of the operator's workstation. After all, the representation of the system to the supervisor contributes to his/her understanding of system operations and functions, his/her own ideas about the control she can exert over the system, and how his/her control will effect the system [4, 15, 17]. Rasmussen [17] suggests that the operator views the complexity of the system, represented by the interface, as an objective feature. In other words, the operator accepts that the level of complexity s/he views through the system displays *is* the actual system complexity. Thus, a superior design can contribute to the enhancement of operator performance. Similarly, a poor design is likely to degrade performance.

Previous research involving human-computer interaction has been primarily in the context of computer programming, text editing or word processing applications (for examples, see [20]). However, user interaction tasks in these applications differ significantly from those in human supervisory control tasks. Most programming or text editing tasks are not time-contingent. Supervisory system controllers are faced with opportunities that change over time. Once an opportunity passes, it can never be recovered. Secondly, the consequences of errors are very serious in the supervisory control domain. The consequences of errors for programmers or editors result only in a decrease in productivity. The programmer can always undo or redo an action and simply recompile the code. This will result only in the loss of time. In supervisory control systems, errors may be catastrophic and expensive (e.g. airline crashes, Three Mile Island, or eight hours of downtime in a $10 million FMS). Thus, the interfaces to such systems becomes increasingly important.

As computer hardware and system software costs are decreasing, system designers have access to an array of human–computer interaction devices such as mice, touch panels, voice input and output, and high fidelity graphics and windowing packages. One such human–computer interaction technology is that of direct manipulation – the representation of objects graphically and manipulation of those objects via pointing devices. For a detailed discussion on the evolution of direct manipulation see Benson [3].

This paper describes research designed to demonstrate and evaluate the effectiveness of an operator interface using direct manipulation interaction in FMS supervisory control. In particular, the effectiveness of two different FMS operator workstations – a conventional operator workstation with overlapping windows and a keyboard versus a direct manipulation workstation design – was explored. The experiment used the Georgia Tech-Flexible Manufacturing System (GT-FMS), a high-fidelity simulator of a single-cell flexible manufacturing system [7].

2. GT-FMS: The system and a conventional operator workstation

Georgia Tech-Flexible Manufacturing System (GT-FMS) is a high-fidelity research domain created to examine a range of issues related to

* Currently with NCR Human Interface Center, Atlanta, Georgia.
** Currently with IBM, Atlanta, Georgia.

North-Holland
Information and Decision Technologies 18 (1992) 33–53

human–computer interaction and decision support in scheduling and control of flexible manufacturing systems. It facilitates research and validation in a framework of realistic manufacturing conditions, including competing goals, transient system characteristics, and human interaction with the scheduling and control system [1]. GT-FMS is an interactive, real-time simulator of a potentially multi-cell, multi-workstation flexible manufacturing system. A human decision-maker can interact with GT-FMS in a manner similar to that of a scheduler or expeditor on the shop floor. GT-FMS was designed to provide a workbench or laboratory in which human interaction with scheduling and control can be observed, controlled and empirically evaluated given proposed decision aids and definitions of human functions [12].

A variety of FMSs can be configured with GT-FMS owing to its modular structure. Common to all configurations is an arrival buffer, where all parts reside when first entering the system, an FMS cell containing a central location for each cell's work-in-process (WIP) and the workstations, and an output buffer to which all completed parts proceed before exiting the system. GT-FMS can be one-celled or multi-celled; and each machine workstation within the cells can be configured uniquely, with its own set of capabilities.

For this research, GT-FMS was configured as a circuit card assembly plant using data adapted from Wittrock [22], and modified to better facilitate the examination of real-time human interactive control of a flexible manufacturing facility. It represents the assembly of thirteen different configurations of printed circuit cards and is based on actual IBM facilities. For this system, it is possible for two or more workstations to perform the same task but at different levels of efficiency. Each machine contains physical spaces for two parts, one in-progress and one in a single item buffer. The other properties of GT-FMS – centralized WIP storage, to which parts travel between each operation and a flexible material handling system that can carry out any desired routing within the cell – are included in this configuration.

The system configuration of GT-FMS used in this experiment is shown in Fig. 1. It is comprised of eight insertion workstations, three buffers and a transportation system. Parts accumulate outside the FMS cell at the arrival buffer, which is an unlimited capacity buffer. An arriving part is stored in the arrival buffer until it is dispatched to the GT-FMS cell work-in-process (WIP) buffer. Within the cell, parts are stored in WIP, which has a finite capacity of 20 parts in this configuration. Parts wait in WIP for an available insertion workstation to perform the next required operation. If the WIP is full when a part arrives to WIP, the part is automatically routed to the overflow buffer. This buffer, too, has unlimited capacity. GT-FMS also has a material handling system capable of performing all routings shown by arrows in Fig. 1.

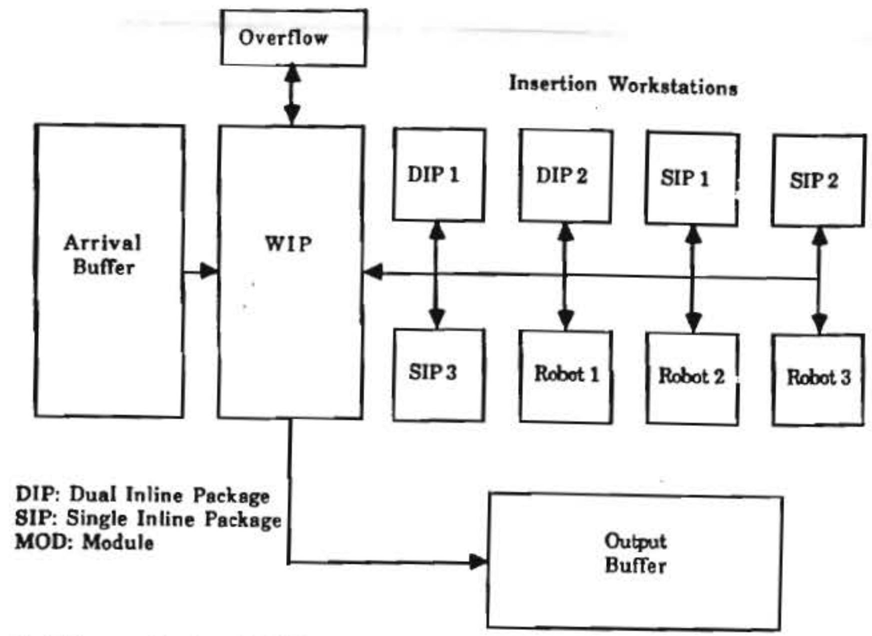

Fig. 1. System configuration of GT-FMS.

2.1. GT-FMS operator functions

The supervisor of the FMS has several responsibilities and the success with which s/he performs these responsibilities is measured and compared across the two interfaces. This section describes the operator goals, the functions the operator may execute and the criteria against which these functions are measured.

2.1.1. Operator goals

As system supervisor and controller, the operator's major, though not necessarily complementary, goals of FMS cell scheduling and control are: (1) to minimize the cost associated with part completions that occur past the due date, and (2) to minimize the cost associated with cell inventory. Assuming these goals, a model was developed that identified two major operator functions: (1) cell schedule management, an operator function whose purpose is to override or fine-tune the automated cell scheduling system when some part is unlikely, given the dynamics of the automated process, to meet its due date; and (2) inventory management, an operator function which consists of the control of cell contents by movement of parts from the arrival buffer into the FMS cell. Details of these two operator functions are given in Dunkler et al. [7]. The operator had several controls to modify the system and override the automatic scheduler.

2.1.2. Operator controls

The operator's highest priority in controlling the FMS is insuring that parts are completed on time. One common method used to compensate for a late or overdue part is to 'expedite' the part. In GT-FMS the operator can expedite a part by scheduling the first free workstation that can perform the part's next required operation. Thus, the operator must monitor and identify late parts, evaluate the feasibility of expediting the next operation for a late part, select a suitable workstation, and actually execute the expedite command to override or anticipate the automatic scheduling and control system. Parts in this configuration of GT-FMS can only be expedited from WIP to a workstation or from the arrival and overflow buffers to the WIP.

The operator also has the ability to move parts from failed workstations back to the WIP buffer so that other machines can perform their remaining operations. If a late or overdue part is sitting on a broken workstation, the operator can move this part so that it does not wait past its due date for the broken workstation to be repaired. The operator does not know how long a workstation will be down and may sometimes choose to leave parts that are on time on a broken workstation so that the other workstations can perform operations on parts that are late or overdue.

The operator's last option for compensating for late parts is to expedite parts from the arrival buffer to the cell's WIP. This control action, however, conflicts with the operator's second function – inventory management. Parts residing in the arrival buffer are not considered part of the cell's work-in-process. A cost is associated with each part that is held in the cell's work-in-process. Therefore, the operator tries to keep the work-in-process inventory as low as possible. S/he must decide when to sacrifice a low WIP level in order to meet a part's due date. If a cell's WIP buffer is full, it would be unwise to expedite a part from the arrival buffer to WIP since it would be placed into the overflow buffer, from which it cannot travel to and from workstations to be worked on. All parts must be returned to the WIP buffer befc re the operator or the automatic scheduling and control system can schedule workstations to perform their required operations. Not only are these parts considered part of the cell's work-in-process and a cost associated with their presence in the cell, but there is a longer transportation time associated with returning these parts to WIP and they can only be returned when there is space in the WIP buffer. It is, therefore, undesirable to have parts in the overflow buffer, especially if those parts are late or projected to be late.

The operator can also control a cell's WIP level by reducing or increasing the minimum number of parts in WIP. S/he can lower or raise this number according to the priority of the functions. If s/he lowers the minimum number of parts held in WIP, s/he will better be able to monitor the status of the parts currently in WIP and will leave available space in WIP in case s/he must expedite a part from the arrival buffer. However, lowering the minimum number of parts held in WIP may cause more parts held in the arrival buffer to be late and decrease system throughput. If there are more parts in WIP, there will likely be a greater mix of the operations required and workstation idle time may decrease. Thus, the operator must decide when to sacrifice high machine utilization and throughput for a lower work-in-process inventory level.

2.1.3. Score

Because these tradeoffs between expediting late parts and controlling the cell's WIP level exist, the operator's performance is evaluated using a combination of cost associated with each. A dollar value is multiplied by the total number of minutes that parts are completed past their due dates. Another dollar value is multiplied by the average number of parts held in WIP and overflow over the experimental session. These two costs are added to compute the overall cost of system operation. Since the operator's first priority is to compensate for late parts, the cost associated with late parts is greater than that of average work-in-process.

The operator interface allows the human supervisor to monitor and control activities that improve the overall performance of the FMS cell. The interface should aid monitoring and information retrieval. It should also allow the operator to successfully execute the controls of late part expedition and WIP management. The following section describes the operator workstation configuration implemented on a PC-AT. This workstation resembles many actual or planned operator interfaces in manufacturing systems [13].

2.2. Conventional GT-FMS workstation

The conventional operator workstation for GT-FMS uses a basic *cell status page*. From this page, the operator can request additional information in the form of windows that cover part or all of the screen. Using this workstation, the operator can also execute control actions, such as expediting a part, removing a part from a broken workstation, reversing an expedition, or resetting the cell's minimum WIP level. The basic cell status page is shown in Fig. 2.

From the basic cell display, the operator can monitor part movement, the status of the workstations, the WIP level, and the simulation time. All possible operator display and command options are listed across the top of the cell status page. The operator selects the desired command by moving the selection highlight bar via the cursor keys on the right-hand side of the keyboard. The line directly below the command option line is reserved for information and error messages. It currently reads: 'Real time operation resumed. Ready for input.' Any parts that are late or projected to be late appear in red on this screen only. Parts currently on schedule are displayed in blue.

All physical locations to which parts may be moved or expedited are referenced with the function keys located on the left-hand side of the keyboard. The function keys associated with each location, e.g. the WIP and all workstations, are displayed in the title bar of their windows. For example, to reference WIP when expediting a part from the arrival buffer, the operator presses F2.

The WIP window is displayed on the left of the screen. For each part in WIP, the part tag and its next required operation are shown in the WIP window. The parts are listed according to

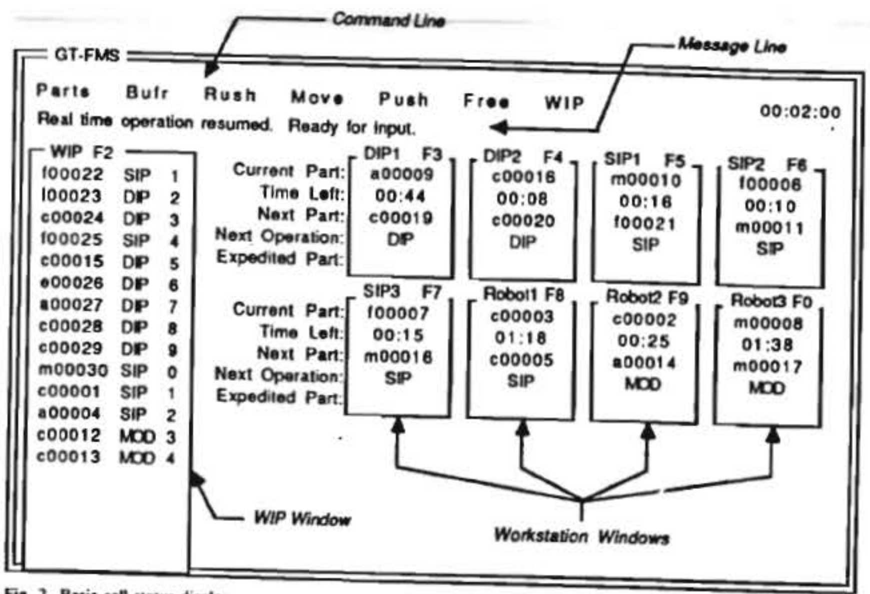

Fig. 2. Basic cell status display.

their arrival times to the WIP buffer. For example, in Fig. 2, f22 is the first part listed and, therefore, the oldest part in WIP. It will be the first part scheduled on a workstation that can perform a SIP insertion. The single digit number following the next required operation of each part numbers the parts to help the operator remember how many parts are currently in WIP. In Fig. 2, there are fourteen parts in WIP.

The operator may gather more information about the system status or execute commands via the command line. The 'Parts' command is a request for a list of all parts currently available in the system, including all parts in the arrival, overflow, in-transit and output buffers, as well as parts shown on the basic cell status page. The 'Bufr' command invokes a window that lists all parts currently residing in the arrival buffer. When the operator selects the 'Rush' command, a window containing a list of all parts that are currently late or projected to be late is shown on the lower half of the screen. If an operator wishes to move a part from a failed workstation, s/he may invoke th: 'Move' command. A s window will appear prompting the operato type in the part tag. The 'Push' command al the operator to expedite a part. The oper will be prompted to type in the part tag and will be prompted to press the function key ciated with the workstation to which the pa to be expedited. The 'Free' command allows operator to 'unexpedite' a part to a given w station. Finally, the 'WIP' command allows operator to reset the minimum number of p to be held in the work-in-process buffer.

The type of interface described above fo the operator to switch between pages of disp to retrieve all relevant information. It also ploys the use of most of the keys on keyboard, including the function and cu keys. The cursor keys on the far right of keyboard are used to manipulate the comm highlight bar as well as scrolling the informa on the arrival buffer and rush page displays. basic alpha-numeric keys in the center of keyboard are used to input part numbers desired WIP levels. The function keys are used to designate locations associated with workstation numbers and WIP. Consequently, the operator's performance not only depends on her/his typing ability, but s/he is forced to move her/his hands from the alpha-numeric keys to the cursor and function keys and the 'mental workload' associated with control tasks is increased since the operator must remember when to use the different sets of keys. The times associated with these transitions may slow the operator's execution of desired actions.

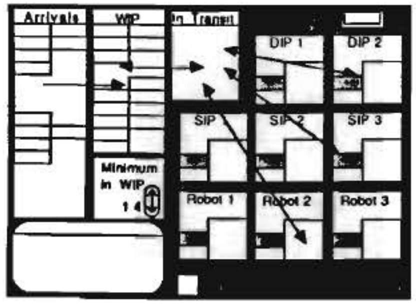

Fig. 3. Part movement.

3. A direct manipulation interface to GT-FMS

An alternative approach to designing a supervisory control interface is introduced here. The proposed interface is based on an explicit set of design principles and employs a high resolution windowing system and direct manipulation interaction techniques that completely eliminate the need for a keyboard. The following is a list of design principles and heuristics that were applied to the design of the proposed operator workstation interface for GT-FMS.

(1) The mouse was employed as the only operator input source.

(2) People read from left to right.

(3) The use of color reinforces important system states.

(4) The use of concrete metaphors reduces unnecessary information processing.

(5) All important, relative information can be viewed simultaneously, and the windows that hold the information do not overlap.

(6) The human–computer system controls should be consistent with the controls in the actual system or similar applications.

These design principles were used to develop a direct manipulation operator workstation for GT-FMS. Figure 3 depicts the proposed workstation configuration. The sections that follow discuss each feature and related design principles.

3.1. Window locations

The operator can access windows containing lists of the parts found in the arrival, overflow, in-transit and WIP buffers, as well as windows that represent each of the eight workstations. Since the logical flow of the parts is from arrival buffer to WIP buffer to workstation, the windows are placed in this order from left to right. The in-transit buffer window is placed between the WIP and the workstations since parts most frequently travel between these two locations. Also, since the primary part movement occurs between the arrival buffer, in-transit buffer, WIP buffer and workstations, these windows do not overlap. Figure 3 illustrates the flow of part movement on the display.

Because it is undesirable to have parts in the overflow buffer the window representing the overflow buffer is available only when there are parts residing there. When the overflow buffer is empty, this window is not available. When the operator opens the overflow buffer window, it appears in front of the arrival buffer window. The operator can view *either* the arrival buffer or the overflow buffer. This design was chosen primarily because there is no relation between the two windows, e.g. the operator cannot expedite a part from the arrival buffer to the overflow buffer, or vice versa. From either location, a part's expedition destination is the WIP buffer. So, it is not imperative that the operator view both windows simultaneously.

The operator can close the arrival, overflow, in-transit or part information windows completely to unclutter the screen. To reopen the arrival, overflow or in-transit buffer window, the operator clicks on an icon corresponding to each window. Figure 4 shows the icons corresponding

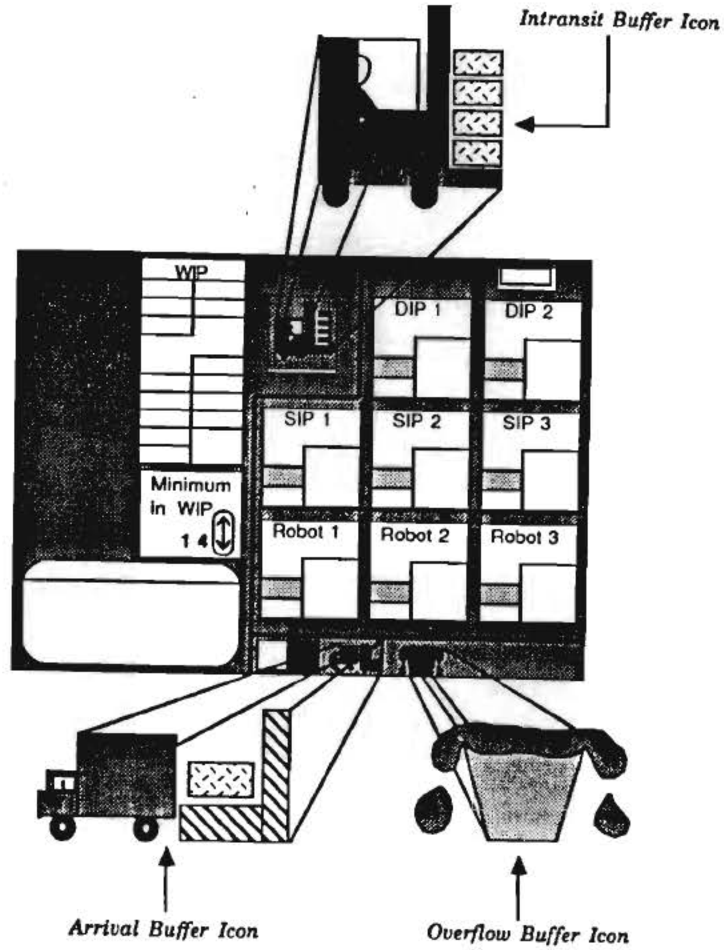

Fig. 4. Icon representations for arrival, overflow and in-transit buffer windows.

to the arrival, overflow and in transit buffer windows.

The incorporation of a single screen display differs significantly from the conventional display described earlier. It eliminates the operator's need to search for and retain information between changing screens and the redundancy of displaying the same information on separate screens.

3.2. Part representation

Parts are displayed in the windows labeled for their current location. All parts in the arrival buffer, overflow buffer, WIP buffer and robot workstations are displayed with the first letter of their next required operation in parentheses following the part tag. This allows the operator quickly to recognize valid machines for part expedition.

The most important feature of the part field is its background color, which indicates its current status – late, projected late, or on schedule. This design feature was incorporated to aid the supervisor in his/her cell management function. The background color of the part field rather than the foreground was changed to eliminate white space and better attract the operator's attention. The colors associated with the third principle listed in the previous section were employed to alert the operator to late or projected late parts. If the part's due date has already passed, the background will be red, the color most often used in alert situations to easily attract the operator's attention. If the part is projected to be late, the background of the part field will be yellow, cautioning the operator to take action before the part's due date passes. If the part is on schedule, it's background will be white. If the operator expedites a part in WIP, the background will change from red, yellow or white to green. This reminds the operator that s/he has already taken steps to push that part through the system. The use of these different colors in indicating part status was incorporated to improve the conventional design by providing the operator with the additional information of whether the part is already late or just projected to be late.

3.3. Workstation windows

The workstation windows are displayed on the right-hand side of the screen. Unlike the conventional display, the workstations are grouped by type. Both DIP workstations are displayed in one line, the three SIPs in a second line, and the robots in a third. Colors were also used to group the machines by types. Soft, neutral colors were used so as not to distract the user from the more important alert colors associated with part status.

Each workstation window includes a place for the part currently in the insertion position and its remaining processing time. Because the robots can perform all operations, each part displayed in any of the robot workstations is followed by the first letter of its current required operation. The part in the insertion position is shown in the center of the workstation window. To the left of the part in the insertion position, in the lower corner, is the part in the workstation buffer. Just above the part in the workstation buffer is a gray rectangle reserved for a part that is expedited to the workstation. If a part is expedited to a particular workstation, its part tag, and its next required operation, if the workstation is a robot, will appear in this rectangle. The part's background will be green to correspond to the green background of the part in WIP, indicating that the operator has taken action to push the part through the system. The expedited part is displayed above the part in the workstation's buffer because it will be placed in the insertion position before the part in the workstation's buffer. The part placement in the proposed design differs from the conventional design, where the expedited part is listed below both the part in the insertion position and the part in the workstation buffer, to better show the priority of the parts to be processed, since they occupy two different physical locations at the workstation. The part currently in the insertion position is separated from the parts to be processed. The part scheduled to be processed next will move from the left of the workstation window to the right of the window, conserving the direction of part movement. If a part is expedited to a workstation, it is displayed *above* the part in the workstation buffer, simulating an ordered list. The highest priority item in the list, in this case the expedited part, is the item at the top of the list. In the conventional design, there is also a field for each part's current required operation in all the workstation windows even though the DIP and SIP machines are dedicated to one operation. To eliminate redundancy, the current required operation for each part is not included in the DIP and SIP workstation windows in the direct manipulation workstation configuration.

3.4. Arrival and overflow buffer windows

The arrival and overflow buffer windows occupy the same space on the screen. There is an icon resembling a truck backing up to a loading

dock in the lower center of the screen (Fig. 4). Any time the operator wishes to view the arrival buffer, s/he moves the cursor to this icon and clicks. This icon is always accessible and the arrival buffer can viewed at any time. However, the overflow buffer can only be viewed when there are parts actually residing in the overflow buffer. Any time there are parts in the overflow buffer, an icon resembling a bucket overflowing with water will appear in the lower center of the screen next to the arrival buffer icon (Fig. 4). The appearance of this icon alerts the operator that parts have been placed in the overflow buffer.

3.5. WIP buffer window

The WIP buffer window is always visible and contains a list of the parts currently residing in WIP (Fig. 4). The operator can always tell how many parts are in WIP. The conventional display's WIP window is often covered by other windows. For example, when the rush page is displayed, the bottom portion of the WIP window is covered. The operator cannot see how many parts are presently in WIP. This may influence his/her decision whether or not to expedite a part from the arrival buffer to WIP. In the proposed workstation configuration, the entire WIP window is always visible.

The supervisor can expedite parts from the WIP buffer to any workstation that does not already have a part expedited to it. Once expedited, a part will remain in WIP (with a green background) until the machine to which it is expedited completes the processing of its current part. The part is then placed in-transit to be transported to the machine. The machine will remain idle until the expedited part arrives.

3.6. In-transit buffer window

The in-transit buffer window works much like the arrival and overflow buffer windows. This window can be opened and closed at the operator's discretion. When closed, an icon resembling a forklift truck will appear in the space to the right of the WIP window (Fig. 4).

Since the parts are being transported to a specific location, the part tag is followed by an arrow pointing the direction in which the part is traveling. If the part is being transported to the WIP, the part tag will be followed by a black arrow pointing to the left, since the WIP window is located to the left of the in-transit buffer window. If the part is traveling to a machine, a colored arrow pointing to the right will follow the part tag. The color of the arrow corresponds to the type of machine to which the part is being transported. While the operator cannot move or expedite parts that are currently in-transit, s/he may need to know which parts are traveling to specific locations.

3.7. Cursor shape and current activity

The shape of the cursor reflects the current activity which the operator can perform. If the cursor is shaped like a question mark, the operator may retrieve additional information about parts on the screen. If the cursor is shaped like a clamp, the operator can 'pick' parts up and move them to different locations. This method of using the cursor to reflect the current activity was adapted from Macintosh applications, e.g., MacDraw and MacPaint. In both applications, the cursor reflects the activity in which the operator is currently engaged. For example, if the operator wants to draw an object freehand, the cursor is in the shape of a pencil. If the operator wishes to spray paint an object, the cursor is shaped like a spray can. Both applications allow the user to switch between cursor shapes and the activities associated with each shape by displaying the group of cursor options and allowing the user to move the present cursor to a picture of the desired activity and press the mouse button. The same approach was applied to GT-FMS to lower information processing by introducing subtle cues to reinforce action choices.

The cursor (activity) options – a question mark to retrieve additional information and a clamp to move or expedite parts – are displayed on the bottom center of the screen. The operator simply clicks on the shape of the cursor corresponding to the activity s/he wishes to perform. The following sections describe in detail the activities of retrieving additional part information and expediting and moving parts to different locations.

3. Monitoring and retrieving additional formation

The operator can monitor the dynamic system ates simply by observing the parts moving from ace to place and watching for parts with red or :llow backgrounds. However, if the operator eeds more information about any part listed on ie display, s/he has access to any part's due ate, projected time to finish and remaining perations. When the cursor is in the shape of a uestion mark, the operator can retrieve this dditional part information. To change the cursor into a question mark, the operator moves the urrent cursor to the icon representing the question mark in the lower center of the screen. S/he hen clicks the mouse and the cursor changes to a question mark. The question mark was chosen because this symbol represents the availability of information in many international airports and sights of travel [11] and is a common symbol used in computer systems that allows the user to request 'help' or additional information. After the cursor has been changed into a question mark, the operator can click on any part on the screen and the part information window will appear in the lower left corner of the screen.

The part tag appears in the title bar of this window. The first line of information in the part information window is the part's due date. If the due date has already passed, the time due is highlighted in red, corresponding to the part's background in the other windows of the screen. The second line is the part's projected completion time. If the part's due date has already passed, obviously the projected completion time is past the part's due date. If this is the case, this time field is also highlighted in red. If, however, the part's due date has not passed, but the part is projected to finish after the due date, the projected completion time field is highlighted in yellow to correspond to the part's background in the other windows of the display. The third line is an ordered list of the part's remaining operations. This may influence the operator's decision on which parts to expedite if two parts are late but one part has five operations remaining while the other has only one operation remaining. The operator may also want to expedite parts that have one or two operations remaining so that the parts leave the system, clearing out space in the WIP buffer. The last line of the information window is reserved to indicate if the part is expedited to a machine, and if so, which machine. If the part is expedited, this message appears in green to provide consistency with the part's background in the other windows of the display. The operator has access to additional part information at all times. This feature was incorporated to aid the operator in making decisions for cell management and inventory management.

3.9. Expediting and moving parts

The primary way for the operator to minimize the late time associated with specific parts is to override the automatic scheduling system by expediting parts through the system. Before expediting a part, the operator must change the cursor into a clamp. The clamp symbolizes the action of picking up an object and dropping it in another location. To change the cursor into a clamp, the operator must move the cursor to and click on the clamp icon located in the lower center of the screen next to the question mark icon. Now the operator can expedite a part from the arrival and overflow buffers to WIP and from the WIP buffer to any of the insertion workstations. S/he can also move any parts located on broken machines back to the WIP buffer. To expedite a part from the arrival or overflow buffer to WIP, the operator simply moves the clamp cursor into the part rectangle in the arrival or overflow buffers and presses the mouse button. The part's rectangle will be inverted and the cursor will change into a clamp holding a small part, shown in Fig. 5. As long as the operator holds down the mouse button, the clamp will be 'holding' onto that part. The operator can then drag the part into the WIP buffer window and release the mouse button. The part will immediately be placed in-transit and the cursor will return to the open clamp representation. If the operator releases the mouse in a location to which the part cannot be legally moved, e.g. from the arrival buffer to a machine, the part will remain in its original location and the cursor will return to the open clamp representation.

Fig. 5. Cursor representations for expediting and moving parts.

When an operator is attempting to expedite a part from the WIP buffer to a workstation, s/he must consider the workstation's capabilities. S/he cannot expedite a part to a DIP machine which requires a SIP operation next. Because all workstations cannot perform all operations, another features was incorporated into the display design to direct the operator when expediting a part to a workstation. After 'picking up' the desired part in WIP, as the operator drags the cursor through the workstation windows, the gray expedite field in the workstation window will turn green if that workstation can perform the part's next required operation, signaling the operator that 'dropping' the part in this workstation window is a valid expedition. Once the part is dropped in a workstation window, the cursor will return to the open clamp representation and the part tag will appear in the workstation's expedite position and will be highlighted in green in the WIP buffer window.

Should the operator decide that s/he has expedited a part to a workstation in error, or decides to expedite another part to that machine, s/he can free the workstation and 'unexpedite' the associated part. The operator simply moves the clamp cursor to the expedite position in the workstation window and 'picks up' the part by pressing and holding the mouse button. S/he then drags the part into the WIP window and 'drops' the part there by releasing the mouse button. The part's background will no longer be green, the workstation's expedite position will be empty and the cursor will return to the open clamp representation.

Moving a part from a workstation's insertion position or buffer back to the WIP buffer can only be executed if that workstation has broken down or is being repaired. The operator will know that the workstation has broken down or is being repaired by the large red icon that is displayed over the workstation window (Fig. 6).

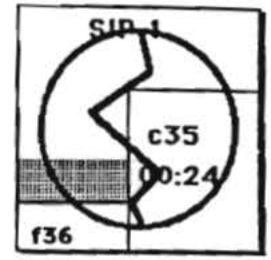

Fig. 6. A failed workstation.

Parts are moved from a broken workstation back to WIP the same way as they are unexpedited. The operator moves the open clamp cursor over the part in either the insertion position or the buffer and presses and holds down the mouse button. S/he then drags the part into the WIP window and releases the mouse button. The part immediately goes in-transit to the WIP and the cursor returns to the open clamp representation.

Manipulating parts on the screen with the mouse eliminates the need to have the operator type commands, part numbers and destinations. The operator never needs to focus his/her attention on anything other than the screen, since s/he does not have to search for keys or correct typographical errors. All part expeditions and movements are executed in a consistent manner. Similarly, if the operator makes an error or changes his/her mind, actions to reverse previous actions are executed in the same manner in which the original action was executed.

3.10. Monitoring and adjusting the minimum WIP level

The operator can only exert one other type of control over GT-FMS. S/he can adjust the minimum number of parts held in the WIP buffer. At the beginning of each experiment session, this number is arbitrarily set to fourteen. The operator may wish to lower this number so that s/he can have more control over which parts come into the FMS cell and to keep the number of parts low so that s/he may more closely monitor that parts that are in the cell. S/he may wish to raise the minimum number of parts in

WIP in order to increase throughput and ensure that parts will be pulled in early enough to meet their pending due dates. Thus, a tradeoff is involved. No matter which strategy the operator chooses, s/he can adjust the minimum number of parts in WIP via the window located just below the WIP buffer window entitled 'Minimum Parts in WIP'.

The current minimum is displayed just below the title. The operator can adjust this number by using the control arrows to the right of the minimum number displayed. When s/he moves the cursor into the control box containing the up and down arrows, the cursor will automatically change into crosshairs. S/he may move the cursor onto one of the arrows and increase or decrease the minimum number in WIP by one. If s/he presses the mouse button down while the cursor is positioned over the up arrow, the new minimum number would be fifteen. Similarly, if the operator presses the mouse button down while the cursor is positioned over the down arrow, the new minimum number of parts in WIP would be thirteen. This type of control action is consistent with other applications on the Macintosh. For example, the system control panel, accessed through the Apple menu, employs these arrow controls when a user wishes to reset the date and time. Therefore, this feature offers some consistency to operators who may have previously used a Macintosh system.

3.11. Summary of the direct manipulation workstation

The proposed GT-FMS workstation consists of a single screen with the mouse being the single mode of operator input. The elimination of the keyboard eliminates the dependence of the operator's performance on his/her typing skills and experience. The contents, appearance and placement of the windows are dictated by a description of the major functions to be performed by the operator and a set of basic interface design principles. The next section describes an experiment that compared the two GT-FMS interfaces and the statistical design used to evaluate the effects of interface style on operator performance and overall system effectiveness.

4. Method

4.1. Subjects

Twenty students from Georgia Institute of Technology, fifteen males and five females, participated in the experiment. All subjects were students enrolled in engineering graduate programs. The subjects were divided into two groups. The first group controlled GT-FMS using the conventional, multi-page interface. Nine of these ten subjects had used a personal computer prior to the experiment and six of the ten estimated that, on average, they used a personal computer at least twice a week. The second group controlled GT-FMS using the direct manipulation, single-page interface. In contrast, only six of these ten had used a Macintosh prior to the experiment and two subjects estimated that they used a Macintosh at least two times per week.

4.2. Experimental materials

Two sets of written instructions were used in the experiment. One set of instructions was given to the subjects who used the conventional interface, while the other set was given to the subjects who used the direct manipulation interface. Both sets of instructions include three sections of information. The first two sections address the physical automatic control structures in GT-FMS and are identical for both sets of instructions. The last section of each set of instructions describes the operator interface and explains detailed procedures for operating the system with either the conventional interface or the direct manipulation interface. Both sets of instructions include questions at the end of each section emphasizing the key parts in the preceding text and a brief summary of basic priorities and strategies that should be applied when controlling GT-FMS.

4.3. Experimental method

The subjects engaged in a total of 11 sessions each. The length of the first session was approximately 60 minutes; the remaining ten sessions

lasted 45 minutes. Sessions were run on consecutive week days with one session per day. Most subjects completed the experiment in 11 working days. The first session was a training session during which the subjects received and read the GT-FMS training manual for their particular workstation configuration. Students controlled GT-FMS for the other ten sessions. During the first of the ten remaining sessions, an experimenter was available to answer all subjects' questions about the structure and operations of GT-FMS.

There were ten different simulation sessions. The sessions were characterized by different initial system states. The first session began after the system ran independently for 20 minutes. The last nine sessions began after the system ran independently for 30 minutes to generate a wider part mix. The seeds used for the generation of part numbers, due date assignments, machine failures, machine repair times and time between machine failures were changed for each session to generate a completely different set of events for each session. All subjects were exposed to the same order of sessions.

4.4. Dependent measures

The first dependent measure was the session score which was computed as a sum of the next two dependent measures referred to as the component scores. One component score was the lateness cost, calculated at two dollars per minute that a finished part left the system past its due date and two dollars per minute that an unfinished part remained in the system past its due data at the end of a session. The second component score was inventory cost, calculated by assessing a cost of ten dollars per part for the average GT-FMS inventory during the session. This included the parts in the WIP and overflow buffers. Thus the session score was computed as score = \$2 × minutes past due + \$10 × average inventory level.

Besides session score and the two component costs of average inventory and lateness, two other measures were examined to determine the level of interaction supported by the different interfaces. The first was the number of operator expedite actions executed. This measure was examined to determine whether the users of one interface exercised more control over the system than users of the other interface and to indicate the ease with which the operator adapted to executing system commands.

The last measure was the number of errors made by each operator. Errors for GT-FMS were operator actions that were initiated but not successfully executed. For the conventional workstation configuration, errors include selecting a command option, such as 'Push', but never actually executing the expedite. This type of error is caused by typographical errors, incorrect part selection, incorrect machine selection, or the operator changing his/her mind after selecting this command option. The same types of errors are associated with the 'Move' and 'Free' commands. Similar errors in the direct manipulation workstation configuration would include 'picking up' a part and dropping it in an illegal location. These errors are associated with moving parts from broken machines, expediting parts and unexpediting parts.

4.5. Statistical analysis

The experiment compared the effect on overall system performance of the conventional versus direct manipulation operator workstation. There were two display conditions: the conventional interface and the direct manipulation interface. The independent variables considered were display condition and session. The experiment was designed and run as a two-factor, nested factorial design. There was one value for each dependent measure per cell, so the design was balanced in all cases.

In the experimental design, subjects were nested within condition, since each individual participated in only one of the two display conditions. There may be a condition × session interaction. No condition × subject interaction can exist, however, since subjects did not participate in both display conditions. Similarly, there can be no three-way condition × subject × session interaction.

Statistical analysis were performed using the General Linear Model (GLM) procedure of SAS statistical software [21]. The General Linear Model procedure was applied to the entire experiment considering all sessions. t-Tests were performed for each session allowing the experimenter to examine if operators using one interface did better for particular sessions than operators using the other interface. The following section presents the results of these analyses. It includes a discussion of performance measures and general subject reactions.

5. Results and discussion

The data collected in the experiment described in the previous section were analyzed to determine the effects of the independent variables (condition and session) on each of the performance measures. This section presents the results of the statistical analyses and includes a discussion of observed operator interaction, reactions and suggestions for the improvement of both the conventional and direct manipulation interfaces. For a more detailed analysis, see Benson [3].

5.1. *Statistical results*

The subject's primary goal was to minimize the total cost associated with operating GT-FMS. When the effect of condition on total score was analyzed, the mean cost for the conventional interface condition ($1051.07) was significantly higher than the direct manipulation interface mean cost ($956.63). All main effects were significant, while the higher order, condition × session interaction was not significant. Figure 7 represents the mean total scores for each of the ten sessions. Table 1 shows the ANOVA results for the total score.

To mitigate the possibility of a learning effect on overall performance, the first four sessions were excluded, one by one, and the ANOVA rerun for the remaining sessions. Each analysis had the same results: interface, condition and session were significant with subjects using the direct manipulation interface performing better than subjects using the conventional interface.

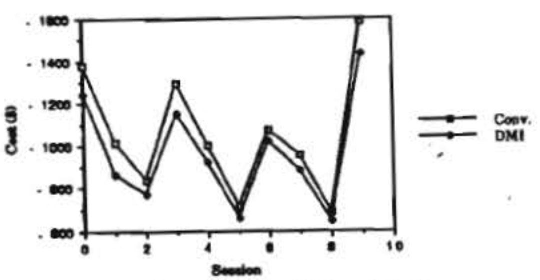

Fig. 7. Average total score by session. □, Conv; ◆, DMI.

Table 1
ANOVA results for total score

Source	df	SS	MS	F-value	PR > F
Condition	1	445904.02	445904.02	45.75	0.0001*
Session	9	13205243.72	1467249.30	150.53	0.0001*
Subject (condition)	18	1159365.74	64409.18	6.61	0.0001*
Condition × session	9	91163.70	10129.30	1.04	0.4112
Total (model)	37	14901676.74	402748.02		

* Denotes significance level <0.05.

The number of times subjects expedited parts was also recorded to evaluate whether users of one interface exercised more control over the system than users of the other interface. When the number of parts expedited was used as the dependent measure, condition had a significant effect on the average number of parts expedited for the direct manipulation interface condition (229.65) over the mean number of parts expedited for the conventional interface condition (160.85). As with all scoring metrics, the effects of session and subject (condition) were significant. The higher order effect of condition × session was not significant. Table 2 summarizes the analysis for the number of parts subjects expedited. Figure 8 shows the means for the number of parts subjects expedited in each session.

For all ten sessions, the mean number of expedites for the conventional interface is lower than the mean number of expedites for the direct manipulation interface. The results from the individual t-tests for each session indicate that in six of the ten sessions the mean number of parts expedited by subjects using the conventional interface was significantly lower than the mean number of parts expedited by subjects using the direct manipulation interface.

The other dependent measures of component part lateness, inventory costs and the number of incomplete moves, as well as the two measures presented in this section can be summarized as follows:

(1) Subjects using the direct manipulation interface achieved a significantly lower total cost associated with operating GT-FMS than did the subjects using the conventional interface.

(2) The direct manipulation interface better enabled subjects to compensate for cost associated with parts completed past their assigned due dates. Lateness cost was the more heavily weighted of the two components comprising the total score.

(3) Subjects in the conventional interface had a significantly lower cost associated with inventory levels than did subjects using the direct

Table 2
ANOVA results for number of expedites

Source	df	SS	MS	F-value	PR > F
Condition	1	236672.00	236672.00	170.83	0.0001*
Session	9	130525.70	14502.85	10.47	0.0001*
Subject (condition)	18	441996.90	24555.38	17.72	0.0001*
Condition × session	9	22804.60	2533.84	1.83	0.0666
Total (model)	37	831999.20	22486.46		

* Denotes significance level <0.05.

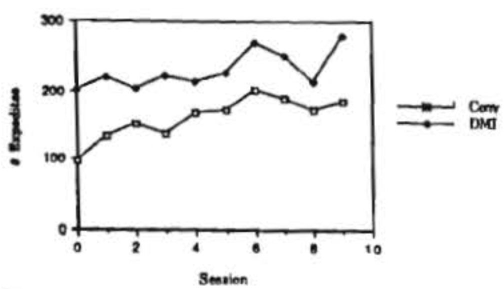

Fig. 8. Average number of expedites by session. □, Conv; ◆, DMI.

manipulation interface. However, low inventory levels are sacrificed when parts are expedited to compensate for costs associated with parts completed past their assigned due dates.

(4) Subjects using the direct manipulation interface executed a significantly higher number of expedites than did subjects using the conventional display condition.

(5) The number of incomplete actions made by subjects using the conventional interface was not significantly more than subjects using the direct manipulation display. One interpretation is that subjects using the direct manipulation display felt more comfortable exploring the system and were not afraid to attempt illegal moves.

This section has addressed the statistical significance of the subjects' performance. The following section addresses learning and offers an interpretation of the results obtained in the previous analyses.

5.2. Interpretation of results of the main effect

The results obtained from the statistical analyses indicate that interface condition is a major determinant of operator performance. Most important is the result that subjects using the direct manipulation interface achieved a significantly lower total cost associated with operating GT-FMS than did the subjects using the conventional interface. Subjects using the direct manipulation interface also outscored subjects using the conventional interface in compensating for costs associated with parts completed past their assigned due dates. Even though the subjects using the conventional interface achieved a lower average inventory score ($171.38) than the average inventory score ($183.49) for subjects using the direct manipulation interface, the difference in the means was quite small compared with the difference in the means of lateness performance – conventional ($879.69) and direct manipulation ($773.15). Subjects were told that the lateness component of total score was weighted more heavily than the inventory component.

Subjects using the conventional interface condition worked primarily from the rush page to address the task of expediting parts. Initially, the operators expedited only the first 5–7 parts listed on the rush page. After the third session, they began to expedite those parts on the rush page that were located in the arrival buffer and all the parts in the WIP buffer displayed in red. Because the rush page did not indicate which parts held in the WIP were currently expedited to workstations, subjects using the conventional workstation configuration often tried to expedite parts that were already expedited. Initially, even though a message appeared on the message line stating that the part was already expedited to a workstation, the subjects asked the experimenter why that part could not be expedited. Subjects did not have a sense of feedback from the rush page. Even though they had just taken an action, such as expediting a part to a workstation, the rush page did not change. If they chose the 'Rush' option to update the rush page, the part they had just expedited still possessed the same status on the rush page. There was no indication that the operator had taken action to compensate for the part's status. In some cases the operator would become preoccupied and continually re-select the 'Rush' option until the part moved to the associated workstation.

Initially, subjects using the conventional interface condition did not consider the number of parts currently in the WIP buffer before expediting parts from the arrival buffer. Eventually, they would notice that a part had been placed in the overflow buffer and were more cautious about checking the number of parts in WIP before expediting a part from the arrival buffer. In the conventional interface condition, if subjects wanted to know how many parts were in transit, they could use the list of inventory associated with the 'Bufr' option, but had no way of knowing whether the parts were in transit to the WIP buffer or to the workstations. In any case, the subjects using the conventional interface condition *never* considered the parts in transit with the exception of a part they had just expedited.

In contrast, some of the subjects using the direct manipulation interface began to consider the parts in transit, but only after four or five sessions controlling GT-FMS. Two of the ten subjects using the direct manipulation interface never considered the parts in transit, and one subject never opened the in-transit buffer window.

The subjects using the direct manipulation interface first expedited parts that were red and then focused attention on the parts displayed in yellow. Most subjects infrequently used the question mark to get further information about parts. They felt they could create a greater impact on the system performance by expediting *more* parts than by trying to expedite the latest parts or the parts that had fewer remaining operations and that using the question mark to get further part information took too much time. Results obtained by Dunkler et al. [7] also support that expediting parts is the most effective way for the operator to enhance system performance.

The results of this experiment also support this approach. Subjects in the direct manipulation interface condition expedited more parts than did the subjects using the conventional interface condition (an average of 229.65 versus 106.85). Note from Table 3 that the mean, minimum and maximum number of parts expedited in the direct manipulation interface is greater than the mean, minimum and maximum number of parts expedited in the conventional interface, except for Session 2 in which the maximums were equal. Since expediting more parts, especially from the arrival buffer to WIP, directly affected the work-in-process inventory, the results of the trade-off are evident. The mean difference in the lateness scores was greater than the mean difference in the inventory score. Since the lateness scores were weighted more heavily, the subjects using the direct manipulation condition succeeded in achieving a significantly lower total score than did the subjects using the conventional interface.

The number of moves initiated by operators but never completed was initially analyzed to attempt an evaluation of the number of errors associated with typographical errors made by subjects using the conventional interface. However, the percentage of errors made by subjects using the conventional interface, either typographical or otherwise, was not significantly greater than the errors of dropping parts in illegal locations by subjects using the direct manipulation interface. Subjects using the direct manipulation interface did indicate to the experimenter that they had tried illegal moves just to see if the system really would not allow them. None of the subjects using the conventional display expressed this type of exploration of the

Table 3
Individual session analysis for difference in number of expedites

Session	Condition	Mean	Std Dev.	Min.	Max.	t	Prob > $\|t\|$
0	Conv	96.8	49.63	53	182		
	DMI	201.8	50.06	133	294	−4.7103	0.0002*
1	Conv	133.5	73.25	41	243		
	DMI	220.3	41.71	165	302	−3.2564	0.0044*
2	Conv	151.8	68.15	48	282		
	DMI	202.4	51.67	105	282	−1.8709	0.0777
3	Conv	138.2	57.86	59	254		
	DMI	222.4	64.67	118	352	−3.0683	0.0066*
4	Conv	167.5	51.19	66	236		
	DMI	213.3	52.71	145	314	−1.9710	0.0643
5	Conv	173.5	47.35	80	264		
	DMI	225.6	73.35	141	387	−1.8871	0.0754
6	Conv	200.9	61.45	106	305		
	DMI	268.2	68.89	183	428	−2.3053	0.0333*
7	Conv	189.9	39.21	122	258		
	DMI	250.2	70.16	188	428	−2.3724	0.0290*
8	Conv	172.4	29.69	118	228		
	DMI	213.2	76.56	156	419	−1.5711	0.1336
9	Conv	184.0	67.15	101	345		
	DMI	279.1	88.71	161	465	−2.7029	0.0146*

* Denotes significance level <0.05.

system controls. In fact, the average percentage of incomplete actions was greater for the direct manipulation display condition. This could have possibly resulted from the subjects' interest in exploring the system capabilities.

The statistical analyses indicating that the effect of interface condition is a major determinant of operator performance may better be understood upon the consideration of the observations of subject interaction discussed in the previous section. The following section addresses the reactions of the subjects who participated in the experiment, questions asked relating to each interface condition, and suggestions subjects made to improve each interface condition.

5.3. *Subject reactions*

During the first two sessions, the subjects were introduced to GT-FMS. The subjects read a manual and the experimenter was present to answer any questions the subjects had. After the first two sessions, the subjects began to control GT-FMS. The questions asked by the subjects during this session were mostly specific to the different interfaces. All ten subjects asked the same question concerning the rush page and voiced concern on this particular feature. The question was: 'I just expedited a part to the WIP from the arrival buffer, but the rush page still says that the part is in the arrival buffer. Why?' The answer to this question is that the rush page is not a dynamic display page. When the subjects selected the rush page, it reflected a 'snapshot' of the system at that particular moment. Invariably, immediately after executing a move or an expedite, subjects reselected the rush command to update one part's location for some feedback even though they knew they had just moved it. In fact, when one su'ject was not moving or expediting parts, he moved the command highlight bar to the 'Rush' command option and *continuously* pressed the return key, updating the rush page. This subject executed the 'Rush' command an average of 960 times per session. The total average for all subjects executing the 'Rush' command option was 169 times per session.

Most of the subjects using the conventional interface display were also confused as to which command option, 'Move' or 'Push', was used to expedite parts. Most subjects knew that 'Push' was used to expedite parts from the WIP to a workstation, but they thought that 'Move' should be used to move a part from the arrival buffer to the WIP buffer. This was probably due to the fact that when parts were expedited from the WIP to a workstation, they did not immediately move to that workstation. On the other hand, when a part was expedited to WIP, it was immediately placed in transit. 'Move' was also confused with 'Free'. When a part was expedited to a workstation, subjects wanted to remove that part from the workstation, so they tried the 'Move' command option. One question most often asked about the direct manipulation interface was also frequently asked by subjects using the conventional interface display. The question concerned the utilization of idle workstations. Most subjects wanted to know how they could move parts from the arrival buffer to workstations that were currently idle. In both cases, the operators had to expedite parts from the arrival buffer to WIP. When a part arrived to the WIP buffer, if the workstation was still idle, the part was automatically dispatched to the available workstation.

Subjects using the direct manipulation interface who had never used a mouse before this experiment expressed some discomfort in manipulating the cursor via the mouse. However, by the end of the first session, all subjects indicated that they were comfortable with the mouse.

At the end of the experiment, subjects completed a questionnaire offering suggestions to improve the interface configuration with which they interacted. Those who used the conventional interface invariably requested that the rush page be updated dynamically. Some also indicated that displaying the number of parts currently in WIP at the top of the WIP window would help since the lower part of the WIP window is not visible when the rush page is displayed.

Users of the direct manipulation interface offered suggestions concerning the method of switching between the question mark and clamp cursors. One suggested that he be able to double-click, and no matter what the cursor's current position on the screen, it should change

to the alternative cursor. Another suggested using a two-button mouse, dedicating one button to switching between cursor types.

Other suggestions for improving the direct manipulation interface included indicating the status of expedited parts. Currently, when a part is expedited, its background is green. Subjects noted that if parts entered the WIP buffer that were already late, they would most likely want to unexpedite any parts that were only projected to be late and replace them with those parts already late. Subjects also indicated a desire to know how late parts were without having to click on each individual part. They generally wanted a priority of the parts that were already late and indication of the parts projected to be late that would become late within the next few minutes.

The initial reactions to the system indicated that subjects were slightly overwhelmed by the number of parts that were late and projected to be completed past their due dates, but as the subjects interacted more with the system, they began to relax and concentrate on developing strategies to control GT-FMS. Subjects in both display conditions said that they enjoyed participating in the experiment, but while subjects using the conventional interface were relieved that the experiment had ended, subjects using the direct manipulation interface expressed interest in returning to 'play' with GT-FMS after the experiment was over. All twenty subjects were extremely dependable and, with the exception of a subject who became ill, never missed a scheduled session. Subjects in both display conditions, however, felt that controlling this system would be a monotonous full-time job.

6. Summary

As a high fidelity simulation, GT-FMS provides insight as to how a direct manipulation interface might improve operator performance in a real-time supervisory control system. The overall goal in controlling the GT-FMS system is to minimize the cost associated with completing parts past their assigned due dates and the average amount of inventory located within the FMS cell. The conventional interface to GT-FMS presents overlapping window displays and employs the use of the entire keyboard, alpha-numeri cursor and function keys, as the means operator input. This research indicates that su a system restricts the freedom of operator inte action and control over the system. A dire manipulation interface to GT-FMS is more like to increase the amount of operator interventi and increase the control the operator exerts ov the system. Results indicate that for minimizi the components of lateness and inventory cos as a total, operators using the direct manipul tion interface performed better than did subjec using the conventional interface configuratio

Experimental results apply to systems beyo the specific GT-FMS environment. Results fro the GT-FMS simulation provide strong suppo that operator performance can be greatly i fluenced by the user interface configuration. the wider area of supervisory control, the use direct manipulation and the principles that we used to develop the GT-FMS interface for th research may provide a superior methodolo over conventional interface design.

Acknowledgement

This research was supported in part by an IB department grant (Dr. Stan Belyeu, technic monitor).

References

[1] J.C. Ammons, T. Govindaraj and C.M. Mitchell, De sion models for aiding FMS scheduling and contr IEEE Trans. Systems, Man and Cybernetics 18 ((1988) 744–756.

[2] J.C. Ammons, C.B. Lofgren and L.F. McGinnis, large scale loading problem, Ann. Op. Res. 2 (198 319–332.

[3] C.B. Benson, The use of single-page, direct manipul tion interfaces in real time supervisory control system Master's Thesis, Center for Human–Machine Syste Research, Georgia Institute of Technology, Atlan GA, Report 89-3 (1989).

[4] P.K. Buckley and J.B. Long, Effects of system a knowledge variables on a task component of 'telesho ping', in: P. Johnson and S. Cook (Eds.), People a Computers: Designing the Interface (Cambridge U versity Press, 1985) 76–91.

[5] J.A. Buzacott and D.D. Yao, Flexible manufacturi

systems: A review of analytic models, Management Sci. 32 (7) (1986) 890–905.
[6] N.H. Cook, Computer managed parts manufacture, Sci. Amer. 232 (1975) 23–29.
[7] O. Dunkler, C.M. Mitchell, T. Govindaraj and J.C. Ammons, The effectiveness of supervisory control strategies in scheduling flexible manufacturing systems, IEEE Trans. Systems, Man, and Cybernetics 18 (2) (1988) 223–237.
[8] M.S. Fox and S.F. Smith, ISIS: A knowledge-based system for factory scheduling, Expert Systems 1 (1) (1984) 25–49.
[9] R. Jaikumar, Postindustrial manufacturing, Harv. Bus. Rev. (Nov./Dec. 1986) 69–76.
[10] C.V. Jones and W.L. Maxwell, A system for manufacturing scheduling with interactive computer graphics, IIE Trans. (Sept. 1986) 298–303.
[11] Aaron Marcus, Graphic design for computer graphics, 1983, in: R.M. Baecker and W.A.S. Buxton (Eds.), Readings in Human Computer Interaction: A Multidisciplinary Approach (Morgan Kaufmann, Los Altos, CA, 1987) 320–332.
[12] C.M. Mitchell, Supervisory control: Human information processing in manufacturing systems, in: Andrew P. Sage (Ed.), Concise Encyclopedia of Information Processing in Systems and Organizations (Pergamon Press, 1988).
[13] C.M. Mitchell, Personal communications, 1989.
[14] C.M. Mitchell and D.L. Saisi, Use of model-based qualitative icons and adaptive windows in workstations for supervisory control systems, IEEE Trans. Systems, Man, and Cybernetics SMC-17 (4) (1987) 573–593.
[15] D.A. Norman, Cognitive engineering, in: D.A. Norman and S.W. Draper (Eds.), User Centered System Design New Perspectives on Human-Computer Interaction (Lawrence Erlbaum, 1986) 31–61.
[16] E.J. Phillips, FMS: Advantages, problems, opportunities, Mech. Eng. 107 (6) (1985) 33–43.
[17] J. Rasmussen, Skills, rules, and knowledge: Signals, signs and symbols, and other distinctions in human performance models, IEEE Trans. Systems, Man, and Cybernetics SMC-13 (1983) 257–266.
[18] J. Rasmussen, Information Processing and Human-Machine Interaction (North-Holland, New York, 1986).
[19] H. Shaiken, Work Transformed: Automation and Labor in the Computer Age (Holt, Rheinhart & Winston, New York, 1985).
[20] SIGCHI, Proceedings of the Association of Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction (SIGCHI) (1984, 1989).
[21] P.C. Spector, J.H. Goodnight, J.P. Sall and W.S. Sarle, SAS User's Guide: Statistics, Version 5 (SAS Institute Inc., Cary, NC, 1985).
[22] R.J. Wittrock, Scheduling algorithms for flexible flow lines, IBM J. Res. Development 29 (4) (1985) 401–412.
[23] J. Zygmont, Flexible manufacturing systems: Curing and cureall, High Technol. (Oct. 1986) 22–27.