Proceedings of the 2nd International Conference on Simulation in Manufacturing

1986

Proceedings of the

2nd International Conference on

SIMULATION in MANUFACTURING

24·26 June 1986
Chicago, USA

Edited by Mr. J. E. Lenz

SIM-2

An international event organised and sponsored by IFS (Conferences) Ltd, Kempston, Bedford, UK
in conjunction with Cahners Exposition Group, USA

Proceedings of the
2nd International Conference on
Simulation in Manufacturing

IFS (Conferences) Ltd
35-39 High Street
Kempston
Bedford MK42 7BT
UK
Tel: Bedford (0234) 853605
Telex: 825489

Cahners Exposition Group
Cahners Plaza
1350 East Touhy Avenue
PO Box 5060
Des Plaines
IL 60018
USA
Tel: (3112) 299-9311
Telex: 82882

AUTHOR INDEX

CONTENTS

New Technologies in Simulation

Simulation Role in Design

Simulation in Control Systems

Simulation on the Shop-Floor

Simulation in Decision Support for Management

Process Evaluation using Simulation

An intelligent simulation environment for the design and operation of FMS

K. E. Wichmann
Technical University of Denmark, Denmark

ABSTRACT

In recent years computer simulation has been widely accepted as a necessesary tool to succeed with the task: to design a FMS system, that will actually operate the way it is designed to do. However, so far this very complex task has been made even more complex, because of the complexity of just using the various simulation tools existing on the market today. A user friendly simulation environment, is software that will reduce the complexity of this design task. In the paper an EXpertsystem for MAnufacturing system Simulation, "XMAS" is described. This will allow the user to - in a few hours - model a proposed or an existing FMS, without having to spend hundreds of hours to learn a simulation language. XMAS will integrate the simulation software MAST with a knowledge based system, which has embedded the knowledge of the FMS modeling expert. XMAS is being written in the logic programming language PROLOG.

INTRODUCTION.

"...To use simulation correctly and intelligently, the practitioner is required to have expertise in .... probability, statistics, design of experiments, modeling, computer programming and a simulation language. This translates to about 720 hours of formal classroom instruction plus another 1440 hours of outside study ... just to get the basic tools. In order to really become proficient, the practitioner must then go out and gain real world, practical experience (hopefully under the tutelage of an expert).

The goal for the development of expert simulation systems is to make it possible for engineers, scientists and managers to do simulation studies correctly and easily without such elaborate training."

Robert E. Shannon, et al.
Simulation Jnl., June 1985

This statement indicates not only what type of simulation tools, that has to be developed, but also why so many engineers is failing or avoiding to use simulation as a major tool in the design of Flexible Manufacturing Systems, FMS. Even though computer simulation in recent years indeed has become widely accepted as a necessary tool for the design and planning of complex systems such as FMS's, and also has been used in many cases, it is not yet a commonly used tool by engineers and planners in industry. Experts in both simulation and FMS are very few. Consequently many questions regarding the behavior of an existing or proposed system, which simulation could have answered, has remained unanswered until the day of installation of a several million $ system, often finding that it do not operate the way it was supposed to do!

In some cases, simulation studies has been performed using external consultants or "internal consultants" e.g. from the computer department. Here the problem is how to bridge over the gap between modeling specialists with a background in computer science and the engineers, managers and other people engaged in the company to design and operate a FMS system: How can the manufacturing people communicate their specific knowledge, understanding and paradigms, uncertainties and questions to the modeling specialist, who in turn may have no manufacturing expertise and experience and certainly no deeper insight in the specific situation of the company; How will the modeling specialist interpret the sayings of the manufacturing people; he is going to built his model conduct his experiments, interpret the results of the simulation, in correspondence with his own understanding; and when he communicates this back to the people in the company, how will they interpret the modeling specialist, and so on so forth. This process not only carries a risk for misinterpretation, it also tends to be very costly and slow; a fact that keeps other companies away from simulation.

These problems has contributed to the total set of problems, that has led to the creation of numerous FMS installations, that are not operating the way their designers had hoped. One might say "hoped", as the design specifications for the system maybe were week and insufficient: as regards the system concept, its machines, its materials handling, its flow-rates between machines, its computer-based communication and software control and support, its operating strategies....

INTELLIGENT SIMULATION ENVIRONMENTS

Development of simulation languages.

As described by Shannon (13) simulation software has evolved from being programs written in general purpose languages such as FORTRAN (and many still are), to special purpose simulation languages (GPSS, SIMULA, SIMSCRIPT....), that was further developed in a third phase, to a fourth phase (1971-80) with the introduction of new languages such as SLAM, GASP IV, and others. At the end of this period, attention began to shift from just the power of the language to a concern for easier model development and execution. Oren and Zeigler pointed out several weaknesses in current languages and proposed that Simulation languages should separate the logically distinct stages of a) modeling, b) experimenting and c) output analysis into separate activities. Secondly, simulation environments should be created to take advantage of current computer capabilities for database management, graphics and program verification. (Oren and Zeigler 1979 (14) ref. in (13)).

"The current phase is one in which the development of simulation software is in a significant transition period. The emphasis is upon ease of use and providing an "Integrated simulation environment" rather than simply more powerful languages." Henriksen (12) says, "An integrated simulation environment is a collection of software tools for designing, writing and verifying simulation programs; ... preparing model input data; analyzing model output data; and designing and carrying out experiments with models." (Henriksen (12) ref.in (13)).

As Shannon continues (13): "This is precisely the goal of AI (Artificial Intelligence) based Expert Simulation Systems, with the added goal of embedding within the software as much of the expertise as possible."

A dedicated FMS simulation language

MAST (Manufacturing Systems design Tool) (John Lenz, (2),(8)) can be classified as a 4.th generation special purpose simulation tool. Together with SPAR (System Planning for Aggregate Requirements) and BEAM (Background and Enhanced Animation for MAST) this software provides a simulation environment, that has some of the characteristics of an integrated simulation environment. It is relatively easy to use in the sense that within its capability no programming is needed. The modeling process is datadriven, and the system prompts the user for much of the input data. (Figure 1). To plan the simulation experiments, the user, however, must be capable to edit a datastructure, - not writing programs but still manipulate numbers corresponding to the type of resources and problems the user has in mind. (Figure 2). With BEAM the user can draw a background - a model of the layout of the FMS system being studied - and see the dynamic movements of carts, parts, machines working in a color graphic animation.

Being dedicated to the study (simulation) of FMS it has specific capabilities that bridges the gap somewhat between the computer and the user. It has the capability to simulate all types of production equipment in discrete event simulation. The user can use his own terms for the resources being part of the FMS (e.g. machines, AGV's, parts, conveyors..) and events and activities that take place in the FMS (transports, processes...) and the status the resources can have (e.g. a machine could be busy, blocked, idle, shuttling, down...). The output of the simulation uses the same terms. A special feature which currently makes MAST quite unique, is its capability of modeling software control algorithms, and in particular the way it is being done: The user can select appropriate algorithms from a library of more than 30 decision rules typically used in FMS, including rules for part introduction and sequencing, station and operation selection, traffic control in the FMS, and in-process storage control.

Expertise required. With SPAR, MAST and BEAM you've got the simulation tool. However, one might still be reluctant to use simulation, because one still need expertise in using the tool, and expertise and understanding of the principles, possibilities, perils, and technological components in FMS. Even though the time required to train a user in how to use the software is but a few days, the experience and knowledge required to design a FMS system is a lot more difficult to get. This expertise is rare and not "in house" in most companies. FMS and Computer Integrated Manufacturing is not well understood, theoretical descriptions of these systems is still very incomplete; It seems, that fundamental theories and methods for planning, scheduling and control which have been applied for conventional manufacturing systems, do not apply very good for FMS and Computer Integrated Manufacturing.

The FMS designer with the goal to design a system, that will operate the way it is designed, can hence be in a double squeeze: from using a simulation language and from understanding the very complex nature of FMS.

THE GOAL FOR THE XMAS SYSTEM

With the XMAS Expert system, it is our goal to create an "Intelligent Simulation Environment" that is dedicated (domain specific) to the design of and operational planning in Flexible Manufacturing Systems. Hence, the general goals for integrated expert simulation environments referred to earlier applies here; and the expertise to be embedded in the simulation software is the expertise required to use the simulation software and understanding characteristics of FMS. XMAS is however not intended to replace expertise. The intention is to provide expert advice and consistency wherever possible.

The goal is to develop a system, that REDUCES THE COMPLEXITY of the design process, REDUCES THE RISKS in FMS design, and increases the probability of designing a FMS, that operates the way it was modeled, simulated and finally decided upon. XMAS should allow the user to EASILY and QUICKLY (not weeks and days, but hours and minutes) model a proposed or an existing FMS, without having to spend hundreds of hours to learn a new modeling language. The user will not need to know the form and internal representation of the model, and he will not need to know the format and content of the datastructures, which is manipulated in the simulation runs.

The XMAS system will provide the user with menus to assist him in setting up different simulation experiments, and select types and forms of output he may desire. The user will interact with the system in a dialog: queries from the system will help the user in supplying necessary data, and the user can ask the system for explanations, why it does like it does, why it asks like it asks, what conclusions it has reached, how it did it, etc. Some of the input, as the layout of the FMS will be input graphically, the user interactively moving icons around on the screen. XMAS will integrate the 4.th generation simulation tool, SPAR, MAST and BEAM with a knowledge based system, which captures the knowledge of the FMS simulation modeling expert, the creator of thesse three programs. XMAS is being written in the logic programming language PROLOG.

Integrating FORTRAN and PROLOG

The approach taken is to create an intelligent frontend and backend to existing 4.th generation simulation software. With this strategy, we combine the advantages of different languages into one simulation environment: FORTRAN's efficiency in number crushing needed for the simulation run time, and PROLOG's flexibility needed for the varying symbol manipulation in the knowledge based system. Also, using MAST for this application, not only a tested, validated model is being provided, but also experience from several studies of existing and proposed FMS systems. MAST has been chosen for other reasons as well: the correct use of the three programs SPAR, MAST and BEAM reflects a design methodology for FMS, that allows the user to get deep understanding of the system being studied; the capability of the simulator itself as mentioned above; and the experience and knowledge of the creator of the system, John Lenz. This is the knowledge to be captured and embedded in the expert systems knowledge base.

PROLOG. The essence of knowledge based systems is the representation of knowledge, inference about knowledge and selection techniques of problem solving procedures. PROLOG has been chosen for several reasons too. One is the event of a new developed PROLOG compiler that runs very fast on a personal computer. (The PROLOG system is developed in Denmark, and is marketed in USA as "Turbo PROLOG") it is very cheap, yet still very powerful. It is not hard to interface PROLOG to the FORTRAN based simulation environment; it is relatively easy to work with; it facilitates a very nice user interface with pop-up-windows, menus, mous, etc. But the most important reason is of course the nature of the language:

PROLOG is very suitable to address problems that are primarily symbol manipulation. PROLOG is logic programming: the programmer does not tell the computer exactly how to do (as in traditional procedural languages), but he tells the computer what it shall do. Hence, the program does not contain instructions that the computer executes sequentially, but a description of the problem, that it shall find a solution for. This description of problems consists of a description of the objects, facts and relations being part of the problem. This way problems can be represented in terms of rules rather than detailed algorithms. The organization of this rules typically encompass conditional if-then statements. Problem solving is achieved by search through the rules and selection of relevant rules based upon goal driven activities. As pointed out by Adelsberger et al. (15), "Rule based systems ... currently constitute the best available means for representing the problem solving approach of human experts..." As new inputs enter the database, the behavior of the system changes, providing a datadriven (goal driven) program execution. Further more, new rules can easily be added to account for new situations; a very important feature because an expertsystem must be frequently updated.

HOW TO USE THE EXPERT SIMULATION ENVIRONMENT, XMAS

On figure 3 is a diagram of the total system concept. To the left is illustrated the flow of data in the simulation environment, and to the right we have the flow of data in the Expert System Environment. In the following we will describe step by step how the system can be used.

The technical design and operation of an FMS involves problems in three separate areas (Lenz and Wichmann, (4)): Aggregate or capacity planning, effects upon performance due to integration, and operational strategies. "Capacity planning is needed to establish feasible target levels of utilization. These are then used as a basis for measurement of the integration effects. These are the "costs" in efficiency due to integrated operation of all components. The third area is in operational strategies which include the study of batch sizing, scheduling algorithms and other control algortihms." (4).

STEP I

Before modeling the system and simulate it, the objectives and the requirements for the manufacturing system should be absolute clear. Data must also be collected from different departments in the company describing the selected parts (with pallets and fixtures) to be manufactured, the production requirements of the different parts, the kind of manufacturing processes and manufacturing equipment to be used in the system, processing times, and routing sheets (the machines required to make the part and their sequence). Also some concepts / ideas of how the layout of the manufacturing system could look like must be sketched.

Future plans with XMAS includes a subsystem, that could help in clarifying and defining production objectives, reassuring that no objectives that are contradictions to one another is being chosen. Also some future interface with group technology and process planning software is being considered. SPAR already has an option for tooling configuration for FMS, a process plan and tooling specifications given from CNC programs.

STEP II : Capacity analysis with XMAS

The system is invoked from the Expertsystem environment (ESE). A menu will guide the user and help/ explanations will be available. The SPAR program is invoked. The purpose of using SPAR is to set up TARGETS for the manufacturing system capacity and performance: production equipment utilizations, work-in-process levels, pallet cycle times and all necessary data to perform the **capacity analysis**. (6). Input of data is given by responding to questions in a menu. SPAR is - the input data given - automatically producing the capacity analysis, e.g.: It calculates the minimum number of machines needed and feasible production levels, it computes the expected system utilization, the required number of pallets and checks the feasibility of the system by detecting and reporting insufficient machine or transporter capacity.

The datafiles including the system description and targets is read by the PROLOG system, and the user can by consulting the knowledgebase via the inference engine get expert advice for evaluation of the system and target levels. The user can change the targets by add or retract production equipment, change the planning horizon, change the routing and other parameters affecting the capacity of the system. SPAR is not doing any simulation. It's calculations are pure mathematics.

The STEP-II Knowledge base (Kdb-II)

The knowledge in the Kdb-II will be design guidelines for FMS design. For example, there exist a "rule" that no system should be designed with the target that machines should be used more than about 80-85 % of available time. Time for maintenance and break downs must be taken into account in the design. Likewise other "rules" for the materials handling and transportation systems, pallet cycles and in-process storage is to be implemented. The user should be able to get reasons and explanations for this rules. Also the user should be able to get advice in how to configure his actual system in order to get it feasible and set up realistic targets.

STEP III : Setting up simulation goals with XMAS

When completed with feasible target levels for capacity the next step will be to plan and run the simulation. The inputfile for MAST is a data structure containing several data cards that all together describe the system and the operating conditions to be simulated. A piece of this data structure for a sample system is shown in figure 2. However the user does not need to care about this datastructure, as it is being read by the PROLOG system, and represented on his screen - not as numbers and figures, but english sentences.

Integration effects Depending on the nature of the system to be studied, various simulation goals can be tested. One important goal is to detect the effects the integration of machines, materials handling systems and computer control has on production, capacity utilization and other critical parameters of the system. This integration effects is one of the main reasons for simulating at all. The capacity planning of the system would be easy if not for integration effects. The reasons for these in each case, however, is very different: sometimes it's due to too much work-in-progress (too many pallets introduced into the system), sometimes lack of transporter capacity, sometime imbalances in the workload for the different stations in the system resulting in queue build up, congestion or blockage, and several other factors or combination of factors. To detect the integration effects and the reasons for them is one of the issues in FMS design, that requires expertise and experience from several studies. An expertise, that is going to be embedded in the rule base as much as possible.

Selecting options from a menu, the user determines what type of parameters to look at: Decision rules (i.e. control algorithms for the FMS control), Sensitivity analysis, and so forth. Via the inference engine the user can consult the knowledge base. The system helps the user to choose the appropriate decision rules, control the production schedule by running some parts as batches, change the part mix, change the . mber of pallets, give machines different priorities. Also the user might want to try the systems sensitivity to possible machine breakdowns, maintenance time and other types of reliability studies.

Given the system description and user-responds to queries from the inference engine, the input file is being consistently modified by the expert system to reflect the simulation goals being decided upon. The inference engine match conditions and constraints given by the user with facts and rules in the knowledgebase. It provides help, suggestions, displays reasoning and explanations. Hence the user no longer needs to edit a datastructure. Now he only needs to describe the problem and the datastructure is automatically being consistently defined by the rules in the system.

THE STEP-III knowledge base (Kdb-III)

The knowledge contained in the Kdb-III is of two types: one is for the internal editing of the inputfile: for example, "if there is given a conveyor in the system description, certain decision rules must be applied, inserted automatically in the inputfile, and explained to the user". The other type of knowledge is on what to look for and what to check when studying a FMS system: the simulation goals. These can be of different types described by Shannon et.al 1985, (3):

- Evaluation of a proposed system against specific criterions such as target levels for capacity utilization, production, pallet cycles etc.
- Comparison of different systems : various system layouts with different types or

numbers of machines, different materials handling or transportation systems, or comparison of different proposed operating policies or procedures for the FMS control or scheduling.

- Prediction of or estimating the systems performance under some projected set of conditions. It could be conditions with regards to the manual intervention of operators in the system.
- Sensitivity analysis such as breakdowns of machines, or transporters
- Optimization, that is determining which combination of factor levels will produce the best overall response of the system. For example determine the number of pallets in the system giving maximum system utilization.
- Functional relations, For example determine the relationship between the work-in-proces level and the available capacity.
- Transient behavior. looking for specified transient behavior such as bottlenecks, excessive queue buildup, utilization imbalances, etc.

Given a type of GOAL, the system will set up the experiment(s) for the user, prepare the inputfile and select an output format for the results. In some cases, when having decided upon a specific simulation goal, not only one simulation run is necessary, but a series of simulation runs. After each run, the resulting data of interest will be written into a file, and after the last run of the series, the data will all be collected and presented in an appropriate way, graphically, statistically, ... XMAS will automatically initiate the simulation run, save the data, automatically change input parameters, run the simulation again, save the data, change parameters ...., etc.

STEP IV: Analyze the simulation results with XMAS

After having run the simulation(s), the actual system performance will be automatically matched with the targets for the system and goals for the simulation. XMAS will contain knowledge on how to evaluate the observations, that is experimental judgemental knowledge, that only experienced experts usually is in possession of.

XMAS will automatically produce windows, statistics, graphics, reports, which highlight the goals selected by the user in STEP III. There should be a library of formulas with variables that on request by the user or automatically (according to the choice of simulation goal) is instantiated with the actual values. The user can consult the knowledge base for help and explanations. In effect the XMAS system is a diagnostic system in this step.

THE STEP IV Knowledge base Kdb-IV

The knowledge in this knowledgebase will consist of rules, that makes the system follow up on the simulation goals. That is partly "rules" for reading the Kdb-III knowledge base and partly specific rules for interpreting and diagnosing an FMS system with regards to for example reasons for integration effects that reduce the capacity utilization or causes malfunctions in some other way. Then, the Kdb-IV will contain predicates and algorithms, that produces statistics and graphics and finally, it will contain more specific knowledge gradually being stored as the number of simulation experiments is growing and the knowledge about the specific system is increasing.

USING XMAS FOR OPERATIONAL PLANNING IN THE FMS

As the above described procedure goes for design of a new system, this section intends to describe the use of simulation when operating a real FMS system. The only way to plan and predict the operation in an FMS environment, we believe, is by applying some form of intelligent simulation. One main reason for this is the different events of integration effects occurring when the configuration of the system changes due to e.g. machine breakdowns, or introduction of another partmix into the system. Having a model of the system already built with relevant data about machines, parts, control algorithms etc. it is apparent, that this model can be used to decide about what to do when operating the FMS.

ADAPTIVE DESIGN. While in traditional manufacturing systems this type of decision making has been a manual or isolated computer aided planning / production control task, we will for FMS systems characterize it as being a design task, that is

Adaptive design. Therefor the same procedure as outlined above, can be applied with the only difference, that at this stage we do not need to build a new model. What we need to do is to make adequate changes in the existing model. As the real system changes we need to change the model in accordance with the real systems status. XMAS will provide a user interface, that as above will make the daily use of the system quite easy. FMS supervisors, planners, operators can each time a change, to be planned or due to malfunction, is occurring ask "what-if"-questions to the XMAS system before actually taking any action, that might have led to an even worse situation.

The adaptive design knowledge base

The knowledge bases described above is naturally supposed to be used to solve parts of the problems we address in adaptive design. But a company specific knowledge base could be created as operators and FMS supervisors gather more and more knowledge and experience from daily operation of the FMS. Hence the knowledgebase should be designed "open", so that the people involved with the operation of the FMS themselves can update the knowledgebase. Beyond the advantage of having this knowledge stored to support decision making, it would make the company less vulnerable if one operator leaves the company or is transferred to another job. The experience is not disappearing with the man but are being kept in the company's knowledgebase.

Further research and development should be done as regards the integration of this software with the computer systems existing elsewhere in the company. For example giving the expert system access to data from the datafiles produced by the real time FMS control system, or the part database from some CAD/CAM system, Group technology system etc.

CONCLUSION

The need for new types of simulation tools has been discussed. An example of an integrated, intelligent simulation environment has been described. The goal is to develop a system, that REDUCES THE COMPLEXITY of the design process, REDUCES THE RISKS in FMS design, and increases the probability of designing a FMS, that operates the way it was modeled, simulated and finally decided upon. The XMAS system will integrate the 4.th generation simulation tool, MAST, with a rule based system, which captures the knowledge of the FMS simulation modeling expert. XMAS is being written in the logic programming language PROLOG.

ACKNOWLEDGMENTS

I wish to thank John Lenz, who not only is providing the simulation software but also his experience and expertise and is actively participating in the project and the knowledge acquisition process. Also the students, Nils W.Lorenzen, Søren Jensen and Anders L.Kragh, for good discussions and initial implementations.

REFERENCES

(1) Wichmann, Knud Erik :"XMAS : Expertsystem for manufacturing system simulation. Integrated use of simulation software and expert systems as decision support in design and operation of FMS systems." The Institute of Production Management and Industrial Engineering, The technical University of Denmark 1985 (Work paper In Danish).

(2) Lenz, John : MAST, SPAR & BEAM, CMS research inc. 945 Bavarian Ct. Oshkosh. WI54901, USA.

(3) Shannon,R.E, R.Mayer, and H.Adelsberger, "Expert systems and Simulation", Simulation 44:6. pp.275-284, 1985

(4) Lenz,J. and Wichmann,K.E. "Expertsystem for FMS-design" in Proceedings of "Flexible Manufacturing Systems 86", Sponsored by SME and Manufacturing Engineering Magazine.

(5) PC-Prolog, Prolog Development Center, Købmagergade 26 E, 1150 Kbh.K, Denmark.

(6) Wichmann, K.E and Lenz, J.E. "Integration of a FORTRAN based simulation program and a PROLOG based expert system for intelligent manufacturing systems simulation". Invited paper for the conference on intelligent simulation Environments, SCS, San Diego, Cal. 1986.

(7) Proceedings of the first International Conference on Simulation in Manufacturing". IFS publ.1985

(8) Lenz, J.E. "What happens when you do'nt simulate" and "MAST: A as advanced as the FMS it studies" in ref. (7)

(9) E.Rich: "Artificial Intelligence", McGraw-hill book company, 1983

(10) Proceedings of the conference on "Intelligent Simulation Environments", January 1986, San Diego, Cal., Edited by P.Luker & H.Adelsberger, Simulation Series, vol.17 no1. 1986. The Society for Computer Simulation.

(11) Proceedings from the seminar "Expert Systems and Simulation Models", November 1985, Tucson, Arizona, organized by B.P.Zeigler, Univ.of Arizona and P.E.Baltes, SPE.

(12) Henriksen, J.O. "The integrated simulation Environment". Operations research vol.31 no.6, pp.1053-1073, 1983.

(13) Shannon, R.E. "Intelligent Simulation Environments", in ref.(10). pp.150-156.

(14) Oren T.I. and B.P.Zeigler, "Concepts for advanced Simulation Methodologies", Simulation Vol.32, Nr.3 pp.69-82, 1979.

(15) Adelsberger, H. et al. "Rule based object oriented simulation systems" in ref (10) pp.107-112.

```
System Planning for Aggregate Requirements

PART ID(10 CHAR) 2V709
PRODUCTION REQUIREMENT  35
PARTS PER PALLET  2

ENTER OPERATION DESCRIPTION(10 CHAR) AND OPERATION TIME
TYPE END TO TERMINATE OPERATION ROUTE

 1        OPERATION ID:  LD/ULD            TIME:   2.00
 2        OPERATION ID:  BUFFER            TIME:   0.10
 3        OPERATION ID:  15HS              TIME:  20.82
 4        OPERATION ID:  BUFFER            TIME:   0.10
 5        OPERATION ID:  LD/ULD            TIME:   2.00
 6        OPERATION ID:  BUFFER            TIME:   0.10
 7        OPERATION ID:  15HS              TIME:   9.42
 8        OPERATION ID:  BUFFER            TIME:   0.10
 9        OPERATION ID:  LD/ULD            TIME:   2.00

OPERATION NUMBER TO CORRECT :?
```

FIGURE 1.

```
ROUT,20,2,3,72,4,72,5,72*                                BA
ROUT,20,3,6,15,7,15*                                     C
;-----------------------------------------------------------
;
;   PART #TSRAM-1039 AND ROUTE
;
PART,21,1039,1,1,1,0,32,4*
ROUT,21,1,6,15,7,15*                                     C
ROUT,21,2,3,9.6,4,9.6,5,9.6*                             BA
ROUT,21,3,6,15,7,15*                                     C
;-----------------------------------------------------------
;
;   PART #TSRAM-1040 AND ROUTE
;
PART,22,1040,1,1,1,0,32,4*
ROUT,22,1,6,12.5,7,12.5*                                 C
ROUT,22,2,3,75.6,4,75.6,5,75.6*                          BA
ROUT,22,3,6,12.5,7,12.5*                                 C
;-----------------------------------------------------------
;
;   PART #DÆKSE-1045 AND ROUTE
;
PART,23,1045,1,1,1,0,12,4*
ROUT,23,1,6,5,7,5*                                       C
ROUT,23,2,3,96,4,96,5,96*                                BA
ROUT,23,3,6,5,7,5*                                       C
;-----------------------------------------------------------
;
;   PART #DÆKSE-1046 AND ROUTE
;
PART,24,1046,1,1,1,0,12,4*
ROUT,24,1,6,5,7,5*                                       C
ROUT,24,2,3,125,4,125,5,125*                             BA
ROUT,24,3,6,5,7,5*                                       C
;-----------------------------------------------------------
;
;   PART #DÆKSE-1047 AND ROUTE
;
PART,25,1047,1,1,1,0,12,4*
ROUT,25,1,6,5,7,5*                                       C
ROUT,25,2,3,108,4,108,5,108*                             BA
ROUT,25,3,6,12,7,12*                                     C
;-----------------------------------------------------------
;*****************   ***************************************
;-----------------------------------------------------------
;
;
;   STATION DESCRIPTION
;
STAT,1,1,1,0,1,1,1,1,2*                                  BB
STAT,2,1,1,0,1,1,1,3,5*                                  BB
STAT,3,1,1,0,1,1,1,7,8*                                  BA
 . .                                                     ..
```

FIGURE 2

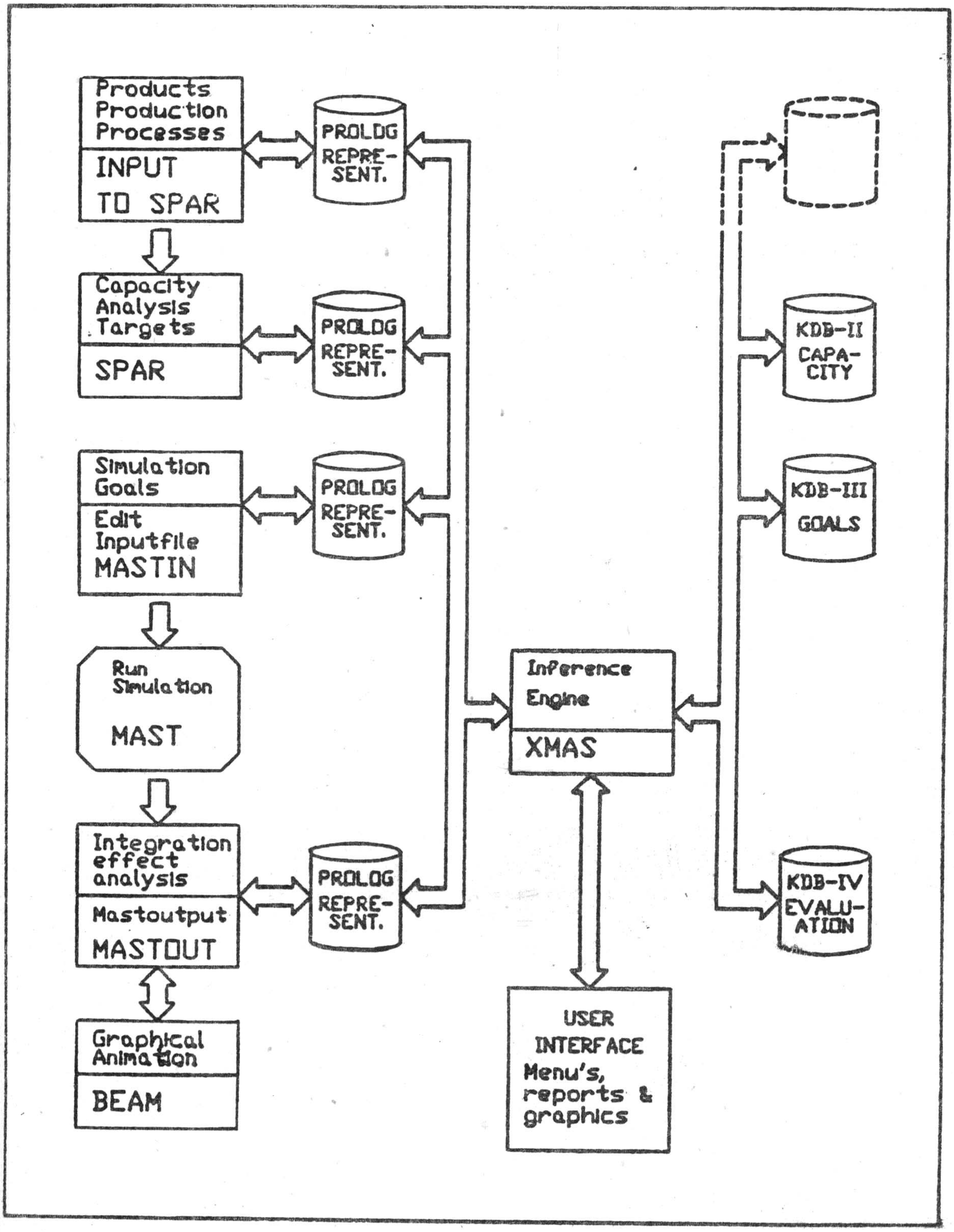
Products
Production
Processes
INPUT
TO SPAR
PROLOG
REPRE-
SENT.
Capacity
Analysis
Targets
SPAR
PROLOG
REPRE-
SENT.
Simulation
Goals
Edit
Inputfile
MASTIN
PROLOG
REPRE-
SENT.
Run
Simulation
MAST
Integration
effect
analysis
Mastoutput
MASTOUT
PROLOG
REPRE-
SENT.
Graphical
Animation
BEAM
Inference
Engine
XMAS
USER
INTERFACE
Menu's,
reports &
graphics
KDB-II
CAPA-
CITY
KDB-III
GOALS
KDB-IV
EVALU-
ATION

FIGURE 3.