

ENGINEERING FARM KNOWLEDGE FOR A
SEAMLESS DECISION SUPPORT SYSTEM

BY

HARBANS LAL

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1989

UNIVERSITY OF FLORIDA



3 1262 08552 3453

Copyright 1989
by
Harbans Lal

Dedicated to
the Farming Community of the World

ACKNOWLEDGEMENTS

I would like to express my deep hearted gratitude to Dr. R. M. Peart, Graduate Research Professor of Agricultural Engineering, for serving as a chairman for my dissertation work. The efforts of Dr. W. D. Shoup of the Department of Agricultural Engineering in formalizing the project and serving as cochairman on my supervisory committee are highly appreciated. The financial support for the studies and the dissertation was provided by International Benchmark Sites for Agrotechnology Transfer (IBSNAT). Dr. J. W. Jones, also from Agricultural Engineering Department, played a key role in arranging this support and helping me in structuring and conceptualizing different aspects of the dissertation. I specially thank him for his efforts and guidance. I would also like to thank Dr. Peter Hildebrand from the Food and Resource Economics Department and Dr. D. Dankel from the Department of Computer and Information Sciences, who provided a unique blend of expertise in farming systems and knowledge-based systems needed for the dissertation.

Mr. Jerry Davis and his wife from Santa Rosa County in North Florida provided the details about their farm setting for operational verification of the dissertation work. They

deserve my special thanks for sparing their time and valuable information. I also thank all participants of the miniconference on Knowledge-based Decision Systems using Logic Programming who spared time from their busy schedule to come to Gainesville and help us evaluate and qualify FARMSYS.

The patience and moral support of my wife Chitra, who kept my morale high all through the study period, deserves special appreciation. I would also like to thank my three daughters Bhawana, Monika and Prerna who found their dad busy with his studies whenever they wanted him to play with them.

I would also like to register my thanks for Dr. P. N. Sharma, my friend and my old time college and professional colleague, who inspired me to return to school after a professional career of 13 years.

Last but not the least, I would like to acknowledge the efforts of my parents, brothers and sisters whose hard work constituted the foundation work for my higher studies and ultimately this dissertation.

TABLE OF CONTENTS

		pages
ACKNOWLEDGEMENTS.....		iv
ABSTRACT.....		viii
 CHAPTER		
I	INTRODUCTION.....	1
	Justification.....	1
	Objectives.....	4
II	REVIEW OF LITERATURE.....	7
	Conventional Decision-aid Tools.....	7
	Machinery Management Applications.....	9
	Limitations.....	14
	Knowledge Engineering Concepts and Applications	16
	Knowledge Classification.....	17
	Knowledge-Based Decision Systems.....	18
	Applications.....	27
III	SYSTEM DESIGN AND IMPLEMENTATION.....	33
	Object-Oriented Simulation.....	37
	Farm as a System.....	40
	Knowledge Representation.....	46
	Model Description.....	47
	Simulation in PROLOG.....	53
	Expert Simulation Analysis.....	65
	Operations Analysis Report.....	69
	Machinery Usage Report.....	83
	Accumulated Work Report.....	90
	Knowledge-Based Yield Estimation.....	93
	Crop Yield Knowledge-Bases.....	95
	Expert Search System.....	97
	An Intelligent Information Management System...	102
	Characteristics.....	108
	Components and their Functions.....	109
	Knowledge-Bases Utilized and Generated.....	114

	pages
IV TESTING AND EVALUATION.....	130
Professional Qualification.....	131
Structured Response Analysis.....	142
Non-Structured Responses.....	144
Role of Mini-Conference in Evaluating Knowledge-based Systems.....	145
Operational Level Verification.....	146
Description of the Test Farm.....	148
Setting up Farm Knowledge for Simulation Runs.....	148
Operations Simulation Reports.....	153
Expert Analysis System.....	161
Test Runs with FARMSYS.....	172
Yield Estimation System.....	180
Integrated Decision System.....	182
V CONCLUSIONS AND RECOMMENDATIONS.....	186
APPENDICES	
I KNOWLEDGE-BASES.....	192
II MINI-CONFERENCE FOR FARMSYS QUALIFICATION.....	198
REFERENCES.....	212
BIOGRAPHICAL SKETCH.....	225

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

ENGINEERING FARM KNOWLEDGE FOR A
SEAMLESS DECISION SUPPORT SYSTEM

by

HARBANS LAL

December 1989

Chairman: Dr. R. M. Peart
Major Department: Agricultural Engineering

Farm operational knowledge including dynamic processes as well as heuristics for expert planning and management for machinery, labor and other resources has been represented and manipulated utilizing knowledge engineering concepts of artificial intelligence (AI) in logic programming (PROLOG). The new approach permitted integration of simulation, expert systems and databases under one single environment, thus leading to a seamless decision support system. It also enabled imparting some of the "intelligent" functions of simulation processes, generally performed by human experts, to computers, thus facilitating their usage by non-experts.

A decision support system (FARMSYS) with four components, namely, object-oriented simulation, logic-based expert system,

knowledge-based yield estimation system and intelligent information system, has been designed, constructed and evaluated. A team of experts evaluated professional qualifications of FARMSYS and found its structure excellent with a comprehensive framework for a variety of applications. All its components functioned correctly, intelligently, and in harmony with one another, using knowledge from an operational farm in north Florida.

The Operations Simulator successfully simulated field operations for complete cropping season and responded correctly to scheduled work hours and to the withdrawal of machinery and laborer. The Expert Analysis component successfully imitated the behavior of machinery management expert(s). Its recommendations and other support were effective, and demonstrated the possibility of reducing the machinery and labor force from the test farm without significantly affecting the timeliness of the majority of farm operations. The Yield Estimation system responded to different work hours during critical operations showing their effects on crop yields and profits. The Information Management System expedited the development and updating process of the farm knowledge-bases to carry out different tests with FARMSYS, thus demonstrating its successful functioning as an integrated decision support system.

PROLOG facilitated simulating field operations in an object-oriented manner, a new AI method of programming. Its

inferencing capabilities have been utilized to incorporate several expert systems within and outside the simulation and other components of FARMSYS. The object-oriented approach allowed for the modelling of aspects of the system that are typically ignored or difficult to model when using conventional approaches.

CHAPTER I INTRODUCTION

Justification

The stochastic nature of weather and complexity of other endogenous and exogenous factors responsible for crop production makes the job of the farmer much more complex and challenging in comparison to industrial production systems.

Machinery and labor constitute a major component of capital outlay for crop production in an operational farm system. They control the operational behavior of the farm systems. Farmers are generally faced with the problem of selecting appropriate machinery (size and mix), scheduling the correct amount of labor, deciding acreage for each crop along with its cropping calendar, and selecting methods for various operations (amount of tillage, number of cultivations, combining pesticide applications with other operations, etc.).

A number of farm-scale models have been developed to facilitate machinery and labor planning and management decision-making process by farmers under conflicting, complex and multiple choice situations. These models have been classified into six categories: linear programming models, simulation models, discount cash flow models, computerized budgeting models, machinery replacement models and machinery

cost and utilization records (Nelson and Bowers, 1968). Among all these categories, the simulation models capture the dynamic behavior of operational farm systems (Peart and Barrett, 1979). The ability of these models to duplicate the real world system depends upon their formulation and incorporation of factors responsible for their control in as realistic manner as possible (Elderen, 1981; Elderen, 1987).

The conventional approach to simulation is characterized as algorithmic or procedural. In this approach an algorithm or a procedure is first defined to solve the problem at hand. A computer program is then written using one of the procedural languages such as FORTRAN, BASIC or PASCAL to implement the algorithm. One of the major limitation of this approach of simulation is its inability to model intelligent behavior of the system. Often simple approximations are used instead. The path of activities that a customer takes in a service simulation (for example a simulation of the shop) is modelled by probabilities determined by prior observations and sampling, rather than by modelling the decision mechanism of the customer (O'Keefe and Roach, 1987).

In addition, the simulation does not provide solutions to the problems. It mainly shows the values of a set of variables over a period of time given certain assumptions. The interpretation of these values and drawing inferences from them lies beyond the scope and intent of the simulation program itself. Simulation cannot guide the user through the

decision-making process of recommending appropriate managerial actions which could improve the productivity of the system. This is completely left to the user's capabilities and initiative.

The results from simulation are often complex and remain within an exclusive domain of the experts for their interpretation. The scarcity and the cost of expert advice for output interpretation creates a serious bottleneck and can even nullify the advantages of simulation as a management and/or planning tool (Khuri et al., 1988).

Knowledge-based decision systems are integrated packages which combine the capability of expert systems and simulation, including necessary data bases (Peart et al., 1985; Peart et al., 1988). They can intelligently apply analytical tools to deal with real world problems. They are developed using modern Artificial Intelligence (AI) languages such as PROLOG, LISP, SmallTalk or specialized development programs called shells. They have the capability of handling qualitative knowledge in addition to quantitative and procedural knowledge. The knowledge representation techniques of these languages can be utilized to maintain farm and regional knowledge separate from the computer code of the model, thus making them versatile and flexible.

The inferencing capability of AI languages can allow for the modelling of aspects of the system that are typically ignored or are difficult to model when using conventional

approach: for instance, adaptive behavior, where the activity attempted next is determined by some perception of the present state of the system (O'Keefe and Roach, 1987). Simple decision rules (for instance, always join the shortest queue), frequently inadequately captured in conventional simulation technique, can be better apprehended using AI techniques. It can also help incorporate expert systems as an integral component of the simulation to facilitate analysis and interpretation of their results.

In recent years there have been some attempts in using the above approach for developing models for an operational farm decision support system. However, there is no reference in the literature to an agricultural expert decision system with simulation and databases that has been developed using a single programming environment.

Objectives

The general object is to represent and manipulate farm operational knowledge, including dynamic processes as well as heuristics for expert planning and management for machinery, labor and other resources utilizing knowledge engineering concepts of artificial intelligence.

The specific objectives are as follows:

- 1) to construct a semantic representation of farm operational knowledge for crop production systems;

2) to manipulate farm operational knowledge for simulating field operations for a complete cropping calendar using object-oriented approach;

3) to design and implement an expert analysis system to analyze the simulation results in the context of the farm knowledge to study labor and machinery utilization, and to develop appropriate recommendations for efficient planning and management of farm operations;

4) to develop a mechanism of utilizing crop growth simulation models and/or knowledge of regional crop experts for predicting crop yields in the context of farm knowledge and simulated dates of operations and

5) to integrate the simulation and expert knowledge into a decision-aid tool with an interactive user interface, and to verify its professional qualifications and operational capabilities as an expert planning tool for multi-crop production systems.

The remainder of this dissertation is organized as four more chapters. The review of literature in Chapter II deals with the decision-aid tools developed utilizing algorithmic and procedural approaches, and concepts and applications of knowledge engineering in the field of agriculture.

The design and implementation of the decision system consisting of an object-oriented field operations simulation, an expert analysis system of the simulation reports, a knowledge-based crop yield estimation and an intelligent

information system are discussed in chapter III. This chapter also describes briefly how PROgramming in LOGic (PROLOG) has been utilized for writing the simulation, the first application of logic-based simulation in agriculture. Chapter IV reports the procedure and results of the evaluation and testing used to qualify FARMSYS as an integrated decision system for multi-crop production. The professional qualification of the system was carried out by letting a team of experts evaluate and critique the system, and the operational verification of its different components was conducted utilizing real farm data from north Florida.

Finally, Chapter V presents conclusions and the recommendations for future work for farm-scale knowledge-based systems in general and FARMSYS in particular.

CHAPTER II REVIEW OF LITERATURE

Conventional Decision-aid Tools

The crop production process can be described as a system of n-stage decision problems. These n-stages are differentiated by weather, the state of the crops and/or fields and availability of inputs, labor and machinery. At each decision-making moment the farmer selects some person and machine to perform certain operation(s) on fields and/or crops. They (the field and/or crop) have to be ready with appropriate soil moisture, ripeness, etc. to receive the services. The operation(s) on the fields are performed with available or selected labor and machines. The process transforms the current stage of the system into the next one with a new set of characteristics for the crop(s) and the field(s). This new state then becomes the next challenge for the farmer to make a decision and this process goes on until all the tasks of the production process are terminated.

The stochastic nature of the weather and complexity of other factors responsible for crop production, both endogenous and exogenous, makes the job of the farmer much more complex and challenging in comparison to other industrial production

systems. Farmers are always concerned about appropriate selection, efficient utilization and management of the resources at their disposal in order to have better control over the production process at different problem stages.

Agricultural scientists have been developing computer models to facilitate the decision-making process by farmers under conflicting, complex and multiple-choice situations (Jones et al., 1988; Tsai et al., 1987; Chen and McClendon, 1985). These models, both analytical and simulation, can be broadly classified into 1) plant-scale models, 2) field-scale models and 3) farm-scale models.

The plant-scale models study the behavior of individual plant and extrapolate the results for the whole field. Examples of such models are the crop growth simulation models (Jones et al., 1988) designed to study the physiological development and growth of plants. These models have been commonly used to better understand the development process of the plant. However, in recent years there have been some attempts to use them as decision-aid tools (Boggess and Amerling, 1983; Boote and Jones, 1986; Hoogenboom et al., 1987; Meyer and Curry, 1986).

The field-scale models study the behavior of an individual field in terms of its adaptability to different cropping systems for its environmental and soil characteristics. These models generally do not consider the operational constraints (machinery and labor requirements) to

implement the candidate or the selected cropping systems (Tsai et al., 1987).

The farm-scale models capture the behavior of the farm as a whole. They consider the operational constraints of the farm and can help farmers make strategic planning and/or tactical management decisions. Both quantitative techniques such as linear programming and simulation, and analytical mathematical models have been used to develop such models.

All three types of models, discussed above, are important in developing decision-aid tools for a farm setting. They should be considered as complementary rather than competitor or substitutes for one another. The plant-based models should serve as the basic tools for developing field-scale models. The field scale models can then serve as a starting point for farm-scale models.

Machinery Management Applications

The machinery and labor constitute a major component of capital outlay in an operational farm system. In addition to climatic and soil factors on which farmer has little or no direct control, the machinery and labor control the operational behavior of a farm system. Therefore, a large number of farm-scale models have been developed to deal with machinery and labor management aspects of farming operations. These models serve as decision-aid tools for selecting an appropriate mix for the machinery set, scheduling various

field operations, and estimating the cost of owning and operating these machinery sets for a given farm setting. The six categories of these types of models, namely linear programming, simulation models, discount cash flow models, computerized budgeting models, machinery replacement models and machinery cost and utilization records (Nelson and Bowers, 1968), can be broadly divided into two groups: 1) the analytical models and 2) the simulation models.

Analytical models are designed mainly for selecting and pricing the machinery operations based upon the given cropping calendars, cost of timeliness and other related factors. Most of these models employ similar algorithms in selecting and scheduling the machinery for different operations. They first estimate available times (hours) to complete each operation based upon available workable days and number of work hours per working day. The models then estimate actual times (hours) needed for each operation based upon machinery capacity. The actual working times are added to the beginning times of different operations to determine their completion times.

The actual start and finish times for critical operations such as planting and harvesting are utilized to estimate timeliness cost (Hetz & Esmay, undated; Burrows and Siemens, 1974; Ozkan and Edwards, 1986a; Sowell et al., 1988; Gui and Siemens, 1986; Siemens et al., 1988). Some researchers have utilized this algorithm in a reverse order to estimate the

machinery size to complete the operation within optimum time period (Edwards and Michael, 1980; Ozkan et al., 1984; Chen, 1986; Siemens et al., 1988).

In all these applications, the machinery and labor time, and timeliness of operations are converted in economic terms to estimate the net returns from crop production activity by using different machinery sets. In addition, some researchers have developed software for estimating machinery capacities and their cost of operations with no consideration to timeliness factors (Ozkan and Edwards, 1986a; Lal and Frerie, 1984).

There is a wide variation in application of simulation techniques for machinery management related problems. It ranges from simulating one single operation for one crop to simulating complete growing season with mono-crop or multi-crop. The single operation models have mainly been developed for harvesting operations. Glunz (1985) and Thai and Wilson (1988) used discrete event simulation approach (Pritsker and Pegden, 1979) for simulating harvesting operations for citrus and peaches, respectively. On the other hand, Miles and Tsai (1987) and Labiadh and Frisby (1987) used a discrete process-oriented approach for simulating grain and hay harvesting operations.

The models for a complete cropping season for a farm setting have been developed using the activity network technique (Link and Bockhop, 1964; Link, 1967; Von Bargen and

Peart, 1969), linear programming technique (Nagarajan et al., 1985; Bender, 1984; Candler, 1968; Waund and Mundell, 1968; Geyer et al., 1963), simulation approach (Tsai et al., 1987; Chen and McClendon, 1985; Smith, 1985) and database management concept (Gauthier and Kok, 1988).

Activity network analysis

In this approach the complete production system is represented as a set of activities and events which are then analyzed using standard operations research techniques such as PERT (Project Evaluation and Review Technique) or CPM (Critical Path Method) as explained by Link (1967).

The preference for one activity over another in case of a conflicting situations and overlapping of timings for different activities are some of the factors for which it is difficult to account in this approach of simulation.

Linear programming approach

In this approach the n-stages of crop production are represented by n-periods during which available time is allocated to different activities to attain an optimum allocation strategy. In this case the problem is solved using a procedure (simplex algorithm) which optimizes the solution by considering all the periods simultaneously. But the stages in crop production are sequential, dynamic and interdependent. This means that decisions at each stage would depend upon what

was achieved during earlier stage(s). In addition, the basic assumptions such as 1) additivity of resources and activities, 2) linearity of objective function, 3) divisibility of resources and activities, and 4) single valued expectations of the linear program restrict its capability in modelling whole farm systems accurately. In addition, the linear programming technique has to use an average weather year and other values of operational coefficients. It cannot account for nonlinear effects of bad weather, for example.

Integer Programming (Brown, 1974) and integration of linear programming with simulation (Bender, 1984; Konaka, 1987) are some of the approaches which have been tried to overcome the limitations of the linear programming approach.

Simulation model

Simulation has been utilized to study both continuous and discrete systems. Continuous systems are characterized by smooth uninterrupted changes in their state variables. The rates of the state variables in such systems are represented by differential or difference equations. In case of discrete systems, the state of the system changes in distinct stages, such as queuing models (Pritsker, 1984).

There exists great a variation among different farm scale simulation models in terms of their approaches to development. Tsai et al., 1987 used a systems approach for optimizing multiple cropping systems employing a deterministic activity

network approach which combines simulation and optimization techniques. Their model can be characterized as a field-scale model. They assumed that the farm is well equipped with all necessary machinery for the crops considered and there are no constraints on scheduling of different field operations.

Chen and McClendon (1985) and Smith (1985) have developed simulation models for handling double crop systems (two crops in one year). Chen and McClendon's simulation model is designed for soybean and wheat as double cropping. On the other hand, Smith's model (CROP-PLAN) handles a variety of single crops and is developed to work on an electronic spread sheet package such as Lotus 1-2-3.

Limitations

To add to the general constraints of the conventional approach to simulation discussed in Chapter I, the farm-scale simulation models reviewed and discussed above have been found having the following limitations.

1. The farm-and-region specific knowledge is generally "hard wired" in their computer code. This makes them very location specific and inappropriate for other locations with a different type of farm setting.

2. They permit assigning of only one tractor to an implement. Farmers with more than one tractor at their disposal in real world situation can operate the same implement with different tractors based upon their

availability. They also have a set of priorities for assigning different tractors to different implements.

3. The available work hours for different operations are estimated using uniform rules for all operations based upon the soil moisture characteristics and/or historic workable days data (Bender, 1984; Rumsey et al., 1986). In most situations, farmers use heuristics for deciding daily working hours specific to each crop and each operation. For example, different amounts of rain make farmers stop different operations. Certain operations such as land preparation could be stopped even with slight rains. On the other hand, critical operations such as planting or baling dry hay could be continued till the heavy rains make it impossible for the farmer to work in the field.

4. They do not allow for irrigation days which make the field non-available for other operations. This results in an over-estimation of available time for plant protection operations.

5. They are generally designed with limited interfaces that require laborious entry of data files and study of many pages of detailed output. This restricts their ability to interact with the user as decision-aid tools. Even though it has been long recognized by machinery management experts themselves that the computer models with no support to teach or guide the user on how to utilize its results would find limited success (Eisgruber, 1968).

Knowledge Engineering Concepts and Applications

Knowledge engineering includes the task of acquiring knowledge, and designing and building knowledge-based systems. It deals with the representation and use of knowledge in electronic form to solve problems that normally require human intelligence or attention. Users can refer to knowledge-based systems in much the same way as they might refer to an expert, consultant and advisor. These systems intend to alleviate man-computer communication problems.

In knowledge-based systems (popularly known as expert systems) "knowledge" rather than "data" is the essential raw material to be processed. The data are "facts and figures from which a conclusion can be inferred." On the other hand, knowledge is "a clear and certain perception of some thing, the act, fact or state of knowing and understanding" (Webster, 1979, page 1007). In artificial intelligence knowledge-base is a body of facts, rules and heuristics which form the basis of knowledge systems. The information about other information in a knowledge-base or knowledge about knowledge is referred to as meta-knowledge (Williamson, 1985). Different individuals can derive different knowledge from the same data set depending upon their level of expertise and meta-knowledge.

Knowledge Classification

Knowledge has been classified differently by different authors. Addis characterizes knowledge in three groups: asserted knowledge, heuristic knowledge and mode of inference (Addis, 1985). Williamson (1985) classified it as 1) descriptive knowledge, 2) prescriptive knowledge and 3) heuristic knowledge.

The knowledge required to represent a dynamic system and to predict its operational behavior, as mentioned in Chapter I, can also be classified into three broad categories: 1) quantitative knowledge, 2) qualitative knowledge and 3) procedural knowledge.

Quantitative knowledge can be expressed in terms of numbers and mathematical equations. A typical example of such knowledge for an operational farm system could be the rate equation which governs the work capacity of a machinery set based upon its operational parameters.

Qualitative knowledge, generally difficult to express in numeric form, consists of a set of heuristics and rules of thumb governing the system. The preference of the farmer for one field over another in allocating available machinery for an operation is a typical example of this type of knowledge for a farm system.

Procedural knowledge, also referred to as prescriptive knowledge by Williamson (1985), deals with prescribing a course of action in the presence of certain conditions. A

set of conditions to be checked prior to starting an operation on a field is a typical example of this knowledge type.

Knowledge-Based Decision Systems

Knowledge-based decision systems are integrated systems which combine simulation and knowledge systems and necessary databases (Shannon, 1984). The main idea behind these systems is to impart a component of "intelligent" functions in the simulation process, generally performed by human experts, to the computers. Such programs are reported to have high acceptability by agricultural computer users (Smith et al., 1988).

In a knowledge-based decision system, simulation provides a model of the world which can be used to test ideas before they are tried in the real world. The knowledge system, on the other hand, provides a model of an expert which can be used to discuss ideas before they are tried in real world. Thus, the integrated system can be used for discussing possible results of a problem without trying them in the real world (Gaines and Shaw, 1986). The integration could also provide a media which would be a two-way interaction between a computer and its user (IntelliCorp, 1988).

Approaches to development

The approaches to developing knowledge-based decision systems can be broadly classified as hybrid approach and

knowledge-based simulation approach (Moser, 1986; Beck and Jones, 1988; O' Keefe, 1986; Umphress and Pooch, 1987; Shaw and Gaines, 1987).

Hybrid approach is the most popular form of integrating simulation and knowledge systems. In this approach, the knowledge systems and simulation models, developed separately, are combined to work together. The simulation models available for the system and/or specially developed for the purpose are written using one of the procedural languages such as FORTRAN, BASIC or PASCAL. The knowledge systems are written in one of the artificial intelligence languages or specialized shells. In this combination the simulation model provides the knowledge about the system for the defined set of circumstances and knowledge system interprets it for the user.

Most agricultural decision systems, discussed later in the chapter, have been developed utilizing the hybrid approach. One problem of this approach is incompatibility of languages used for the two components, the simulation model and the expert systems. This increases the resource (both software and hardware) and time requirements for their development.

In the knowledge-based simulation approach the simulation and knowledge systems are written using one single environment thus providing a seamless system. In this approach, components of many models may be organized into a model-base

(Reddy et al., 1986). The model-base acts as a library which can be consulted for developing a specific model to address a particular problem. The artificial intelligence techniques are used to organize the model-base and also to identify those models needed for a specific goal and design. In addition, models can be integrated with databases and other application programs.

The concept of modelling in knowledge-based simulation approach is fundamentally altered (Oren and Zeigler, 1979; Oren, 1986). The artificial intelligence techniques capture knowledge about models themselves (meta-knowledge) in addition to knowledge of biological and physical processes captured within the models.

Languages for development

The computer programs written in conventional programming languages speak in an imperative mode. They say, "do this, then do this." The programmer has to define every step the computer should go through in order to attain the objective. The knowledge-based systems needs to represent the system in more descriptive rather than in an imperative mode. The programmer, known as knowledge engineer, in this case describes the system in the form of knowledge-bases. He needs to describe more clearly what to do rather than how to do. This makes most conventional computer languages less suitable

for developing such systems. Specialized languages and shells used to facilitate their implementation are discussed below.

Object-oriented programming (OOP). In these languages and environments (various versions of SmallTalk (Stefik and Bobrow, 1986)), the "objects" become the principal focus of attention. The whole world of interest is expressed as a collection of objects grouped into different classes (referred to as object-class in this dissertation) and with a mechanism by which they can communicate with each other and generate new objects (Stefik and Bobrow, 1986; Lal et al., 1987; Pascoe, 1986). A specific instance of an object-class (referred to as object-item in this dissertation) is generated when unique values are allocated to different attributes of an object-class. The representation of knowledge in different classes and subclasses in an hierarchical manner facilitates the development, utilization and maintenance of knowledge-based systems. However, PC-based systems are not yet sufficiently powerful to handle operational scale applications using this approach.

PROgramming in LOGic (PROLOG). PROLOG is one of the most widely used language for writing knowledge systems in Japan and Europe. It is considered as a development tool. It contains a basic scheme for representing knowledge (a common characteristics of a programming language) and also

has a built-in inference engine which conducts its searches to achieve its goal.

A program in this language is a collection of facts and rules about different object-classes and their items of the world of interest. From this collection, PROLOG derives solutions to the questions by searching for clauses that are true. The symbol and list processing capabilities, and recursion in PROLOG facilitate handling qualitative knowledge in addition to quantitative knowledge.

In recent years, there have been a few attempts to develop simulation programs and simulation languages using PROLOG as a base language (Adelsberger, 1984; Futo and Gergely, 1986; Kornecki, 1986; Yokoi, 1986). Efforts have also been made to develop object-oriented programming languages using PROLOG (Futo and Gergely, 1987; Fan and Sackett, 1988). The simulations in PROLOG have also been referred to as goal-oriented simulation or logic-based simulation.

PROLOG provides a unique environment for developing seamless knowledge-based systems (Shintani et al., 1986; Walker et al., 1987) and was also selected for this dissertation work. Both simulation and knowledge systems can be developed under one single environment. In addition, it provides the following advantages for writing simulation: 1) It allows specification of the system in a much more descriptive manner than most procedural languages.

2) It provides a convenient design facility. It permits natural-language-like sentences which facilitate better understanding of program logic.

3) The inferencing capability of PROLOG can be utilized to diagnose causes for poor performance of the real system. The system can then help the user in the process of designing an improved plan for operations.

LISP Processing (LISP). LISP is the second oldest computer language. It has two data structures: atoms and lists. An atom is an element which cannot be divided further. A list may be made up of atoms and of other lists. Lists are separated from each other by parentheses. People who find it difficult to keep track of the proper combination of parentheses call LISP "Lot of Irritating Silly Parentheses" (Williamson, 1985, Page 63).

LISP works by manipulating lists. It can also handle recursive procedures. The data and procedure in LISP take exactly the same form. Both are written as lists, what is a procedure in one program can be data in another.

The LISP programs are also very demanding of computer memory. The microcomputers generally run out of memory before LISP can perform any significant task. Therefore, specialized LISP machines have been developed to handle LISP programs.

Specialized development shells. The specialized development shells accommodate the "synthesis" concept of problem solving. It is an intermediary concept between "thesis" and "antithesis" (Walker et al., 1987).

The concept of thesis states that the general methods of expert problem solving can be found, made computational, and applied to many different problems (Newell and Simon, 1963).

On the other hand, the theory of antithesis put forward by Feigenbaum (Feigenbaum and McCorduck, 1983) states that rather than generality, we should set out empirically to capture human knowledge and procedures for each specific task. Intelligence and reasoning power alone are not enough, knowledge is needed for each specific problem. It means that we should be willing to write a new program for each task.

The theory of synthesis essentially takes the middle ground. The idea is that many tasks have requirements in common, and these requirements can be met by knowledge systems development shells.

These shells are knowledge-less problem solver tools to which the knowledge about particular tasks can be added. They consist of a knowledge representation scheme, an inference or search mechanism, a means of describing the problem and a way to determine the status of the problem while it is being solved (Citrenbaum et al., 1987).

There are a large number of commercial shells available in the market. They range from interpreters of relatively

simple languages to very elaborate development environments (Pepper, 1986).

The selection of an appropriate tool for developing knowledge-based systems depends upon factors such as cost, availability of hardware, and type and specification of problem domains (Elmaghraby and Jagannathan, 1986). Engel et al. (1988) provide comparative evaluation of different knowledge systems shells available on the market and a procedure for selecting the most appropriate for the specific applications.

Steps in development

The steps in development of knowledge-based systems can be broadly divided into 1) problem identification and conceptualization, 2) knowledge acquisition, representation and manipulation, and 3) verification and qualification of the system (Waterman, 1986; Hayes-Roth et al., 1983; Halterman et al., 1986). All of these stages are interrelated and interdependent. They need continuous recycling until the system consistently produces the correct solutions.

The problem identification for knowledge-based system is a critical step in its development process. Some of the criteria for identifying candidate problems are 1) recognized expert(s) exist and can solve the problem significantly faster and/or more accurately than non-experts, 2) experts are available to work on the project, 3) the problem solving task

routinely needs to be taught to others, 4) the problem is well-bounded, 5) solutions can be validated and 6) the impact of the system can be measured (IntelliCorp, 1988).

The successful implementation of a knowledge-based system for an appropriate problem depends upon acquiring the knowledge from domain expert(s) and representing it for manipulation within the framework of the selected tool. A great amount of work is being done in developing efficient means for acquiring knowledge from domain experts (Jones et al., 1986; Prerau, 1987). However, a new trend is for the experts to write their own knowledge systems utilizing their own knowledge (Ogilvie, 1988, Personal Communication).

Finally, the development of a knowledge-based system is not complete until it has been verified for its functionality. In general, the model testing can be divided into verification and validation. However, in case of the knowledge systems, the validation has been termed as "qualification" (Wildberger, 1988).

The qualification process of knowledge-based system involves letting a team of peers evaluate and critique the system in terms of its knowledge, logic, functioning, user-interface, etc. It is analogous to a procedure used to certify a human expert for his expertise. A knowledge system designed to identify crop insect(s) and to make recommendation on their control should be subjected to the same kind of the qualification tests as an entomologist.

The verification and qualification of knowledge systems may never end as they are never complete. Like human experts, they should grow and adapt to additional knowledge as it is gained. However, researchers are working on algorithms to verify the consistency and completeness of knowledge-bases and the recommendations developed by such systems (Nguyen et al., 1987).

Applications

Applications of knowledge-based systems can be found virtually in every walk of human life. They range from making financial investment and marketing decisions (Thieme et al., 1985; Bulkeley, 1986; Phillips and Harsh, 1987), to better management of resources (Coulson, 1987; Davis and Nanninga, 1985), to factory design and industrial research (Fisher, 1986; Moralee, 1986).

Some of the earlier, more popular and commonly cited knowledge systems (Morrison, 1985) are DENDRAL (for identification of chemical structure of compounds), MACSYMA (for complex symbolic mathematics), MYCIN (for diagnosing bacterial infection and suggesting treatments), XCON (for designing microcomputer configuration) and PROSPECTOR (for identifying potential locations of mineral deposits). Most of these applications are pure rule-based with no integration of simulation or any other type of models.

The agricultural applications of knowledge-based systems can be broadly divided into two categories namely 1) pure rule-based systems and 2) knowledge-based decision systems.

Most of the pure rule-based applications are built using development shells on microcomputers. They do not operate any external programs or access any data produced by them. Their applications range for selecting a particular type of machinery (Gibson et al., 1986; Negahban et al., 1988), for identifying pests and diseases and recommending appropriate control remedies (Jones et al., 1986b, Donahue et al., 1988; Gibson et al., 1986), for irrigation design, planning and control of soil erosion (Thomson and Peart, 1986; Bennett et al., 1988; Engel et al., 1988a), for controlling and managing green houses (Jones and Haldman, 1986), to forest road planning (Thieme et al., 1986), for managing animal waste, weed growth in soybean and other crop production related problems (Smith et al., 1985; Ruckman, 1986; Kalkar and Goodrich, 1986; Nagarajan et al., 1987; Rettinger et al., 1987), and for determining bottlenecks in on-farm grain processing systems (Loewer et al., 1988). Role of such systems in technology transfer, education and research has also been emphasized by many researchers (Barrett et al., 1985; Jones and Hoelscher, 1987; Lal et al., 1987; Slocombe, 1986).

The knowledge-based decision systems for agricultural applications have been developed using both hybrid and knowledge-base simulation approach (Beck and Jones, 1988).

The typical examples of the hybrid approach are the systems reported by Jones et al. (1987), Batchelor et al. (1987), Lemmon (1986), Kline et al. (1987), Khuri et al. (1988) and Yost et al. (1986). Jones et al., 1987 describe a system that recommends insecticide applications for the soybean crop. A built-in economic model determines treatment thresholds based upon market value of the crop, cost of insecticide, stage of crop growth and expected yield. It then utilizes the built-in heuristics to select the appropriate insecticide.

The soybean pest management system (SMARTSOY) of Batchelor et al. (1987) uses scouting data and expert knowledge to project insect population for a week in the crop. The estimates of damage potential by the insects in the field are computed and are then used as input to the SOYGRO a soybean crop growth model (Jones et al., 1988) to estimate yield losses if control actions are not taken. The system then recommends the type of insecticide to apply depending upon the yield loss and grower sensitivity to risk. The knowledge system is written in Level5 and the simulation model is in FORTRAN.

The COMAX/GOSSYM (McKinion and Lemmon, 1985a; McKinion and Lemmon, 1985b; Lemmon, 1986) works on a concept similar to SMARTSOY. It recommends optimum scheduling of fertilization, irrigation and harvesting for cotton crop.

GOSSYM is a cotton growth simulation model written in FORTRAN and COMAX is the knowledge system part written in common LISP.

A system for analyzing the results of a linear programming model aimed at selecting optimum combinations of machinery systems for a crop production have reported by Kline et al. (1986, 1987).

Citrus Harvest Expert Simulation System (CHESS), developed by Khuri et al. (1988), interprets the results of Citrus Harvesting Simulation Model (MACH II) written in the SLAM II simulation language (Pritsker, 1984) and recommends to the user how the harvesting operation can be improved.

The system developed by Yost et al. (1986) recommends the appropriate dosage of lime for crop production for tropical soils. It uses a set of equations to compute the lime requirement and the relative yield loss without lime application based upon crop, location and its soil type. The system checks the preconditions and qualifications which must be evaluated in order to use these equations properly.

Agricultural applications using the knowledge-based simulation approach are rather few and most of them are in their initial stages of development.

COTFLEX is a farm level decision support system for individual crops (Stone et al., 1986; Helms et al., 1987). The frame based representation of farm knowledge allows to store the attributes of the object-classes in their slots. The values for these slots can come from one of four possible

sources namely 1) a simulation, 2) an expert system, 3) a data base or 4) the user himself.

CALEX is a farm level decision system which consists of an executive program which provides and controls the access to data files, the knowledge systems and the simulations. Facilities are provided by which the expert system can call a simulation and vice-versa. It stresses the record keeping aspect of farm management decisions (Plant and Wilson, 1986; Plant et al., 1987).

POMME is a decision support system for apple orchards (Roach and Virkar, 1986) which can advice on various production practices including pest control recommendations.

An ADaptive Assembler for Models (ADAM) organizes equations for calculating parameter values when several alternative equations may be available (Whittaker et al., 1987). It has been applied to the domain of soil erosion modelling. It can help identify the erosion equation appropriate for a particular situation.

The WHeAt Modelling expert system (WHAM) also uses heuristic rules to assemble model components from a model-base into a simulation to answer user's question (Rimington et al., 1987). In operation, the user specifies a set of output variables to be predicted by the model. The model-base is then activated to locate and assemble models which can satisfy this goal.

Some of the most recent developments of integrating knowledge engineering concepts with data processing techniques as applied to agriculture were presented and are compiled in the proceeding of a conference on "Integration of expert systems with conventional problem solving techniques in agriculture" (AAAI and ASAE, 1988).

CHAPTER III SYSTEM DESIGN AND IMPLEMENTATION

An integrated decision support system should interact with the user to collect information about the system, analyze it utilizing the knowledge (both procedural and heuristics) captured within itself, and provide the user with expert help to improve the productivity of the system in the context of the input knowledge as depicted in Figure 3.1.

A decision support system consisting of four components: Operations Simulator, Expert Analysis System, Yield Estimation System and Information Management System has been designed and implemented using knowledge engineering concepts of Artificial Intelligence. The integrated system has been referred to as FARMSYS and discussed in this chapter.

The Operations Simulator simulates the field operations for the complete cropping season with a one-day time step. The Expert Analysis component examines the simulation reports and makes recommendations for planning decisions of an operational farm. The Yield Estimation component estimates the crop yields and profits from different fields, crops and the whole farm, and the Information Management System is an intelligent user-interface to gather information from the user

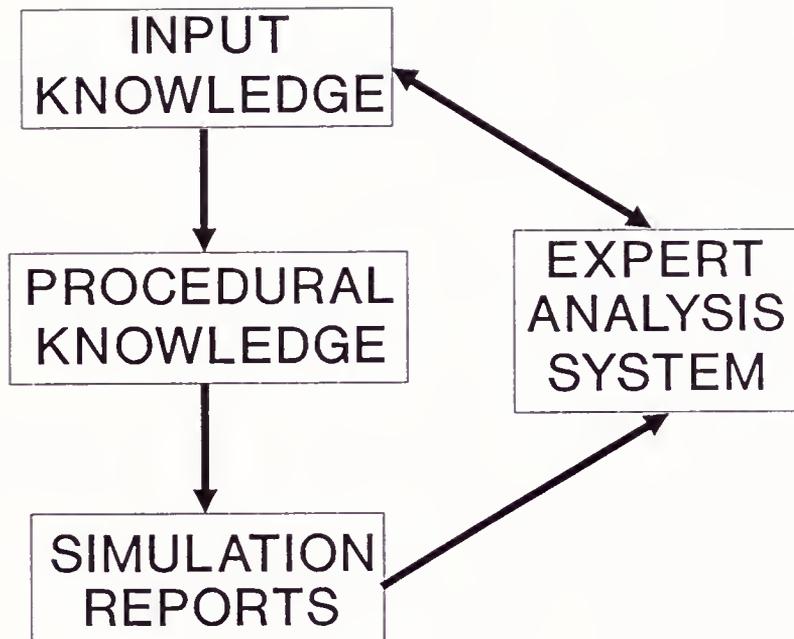


Figure 3.1 Functioning of an integrated decision support system

and to let him make changes in the existing farm knowledge. The hierarchical representation of these components of FARMSYS is presented in Figure 3.2.

The overall goal of such a system is to provide a planning and/or management tool for researchers, educators, perhaps farmers, to "test" combinations of resources such as equipment, crop mixes, labor, procedures (no-till, minimum till, etc.) and weather for timeliness, yields and efficiency of utilization of resources. It can be utilized for evaluating the performance of different farms for a particular weather year or to evaluate the performance of the same farm over different weather years with different combinations of labor, machinery and crop combinations.

The languages and development shells explored for the dissertation work are SmallTalk V (Digitalk, 1986), VP Planner and VP Expert, a combination of spread sheet and expert system shell (Paperback Software International, 1985) and Turbo PROLOG (Borland, 1986). And the Turbo PROLOG was selected for the final implementation of the system, because it provided an environment for simulating field operations in an object-oriented manner, an efficient depth-first search mechanism for the expert systems inferencing and an appropriate frame work for representing farm, regional and expert knowledge in the format of semantic nets, a popular AI knowledge representation technique.

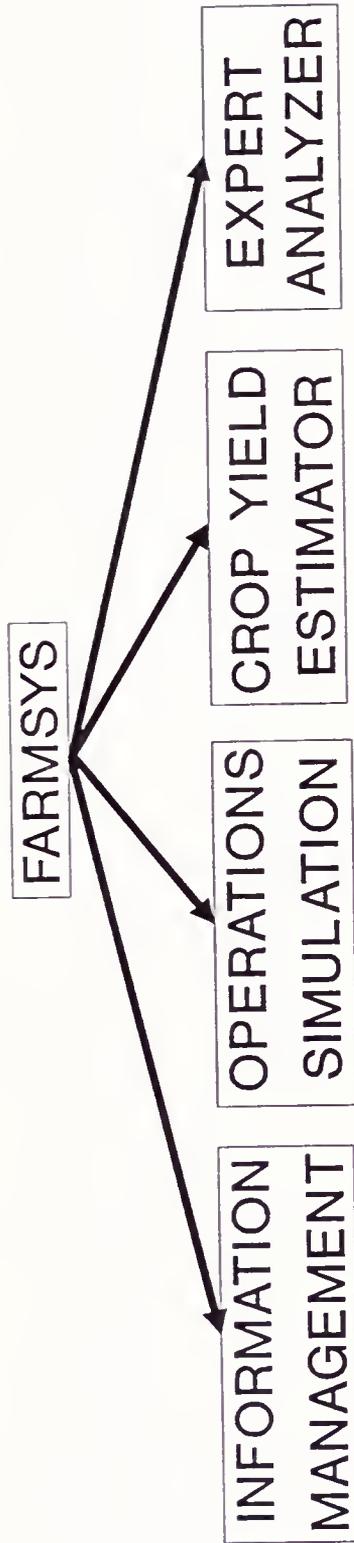


Figure 3.2 Components of FARMSYS and their hierarchical representation

Simulation in an object-oriented manner implies representation and manipulation of farm knowledge as a collection of objects along with a mechanism by which they interact with each other and with the environment over time. This approach enabled modelling the decision behavior of the farm manager in a more realistic fashion than is generally captured in the conventional approach to simulation in procedural languages.

In addition, PROLOG also permits compiling the final program in the format of a executable file which can be distributed to users with no obligations of any run-time versions.

Object-Oriented Simulation

The model simulates field operations for an operational farm system with given machinery, labor, crop combinations, work schedule and other farm specific parameters. The process of simulating agricultural field operations involves both discrete events and continuous processes comprising of many activities and numerous state variables.

The decision about the earliest start and latest finish times for different operations in a cropping sequence is a typical example of discrete events. The scheduling of these operations within their agronomic work windows and allocating the machinery as well as labor to different fields to carry out these operations are the examples of event happenings.

On the other hand, once an operation starts its execution is then controlled by a continuous rate process depending upon the machinery capacity, climatic and other management factors.

The general sequencing of the operation types for crop production is land preparation, fertilizer application and planting, plant protection and harvesting. Each of these could contain one or more operations and are the processes of the system which are controlled by their state variables which change dynamically with time.

All these factors make the task of developing a model which would emulate the behavior of an operational farm system very complex and challenging.

The inputs for the model for a farm can come either from its present mode of activities or from a representative farm of the region. The initial cropping systems for a farm can be selected from the ones most appropriate for the region which depend upon climate, government policies and other market information about the region as depicted in Figure 3.3. Farmers goals and his preferences, irrigation facilities at his disposal, and his financial situation play key role in making selection of cropping systems for his farm situation. The selected cropping systems along with machinery and labor, agronomical inputs and management decisions when applied to farm fields result in crop production for the market and/or home consumption.

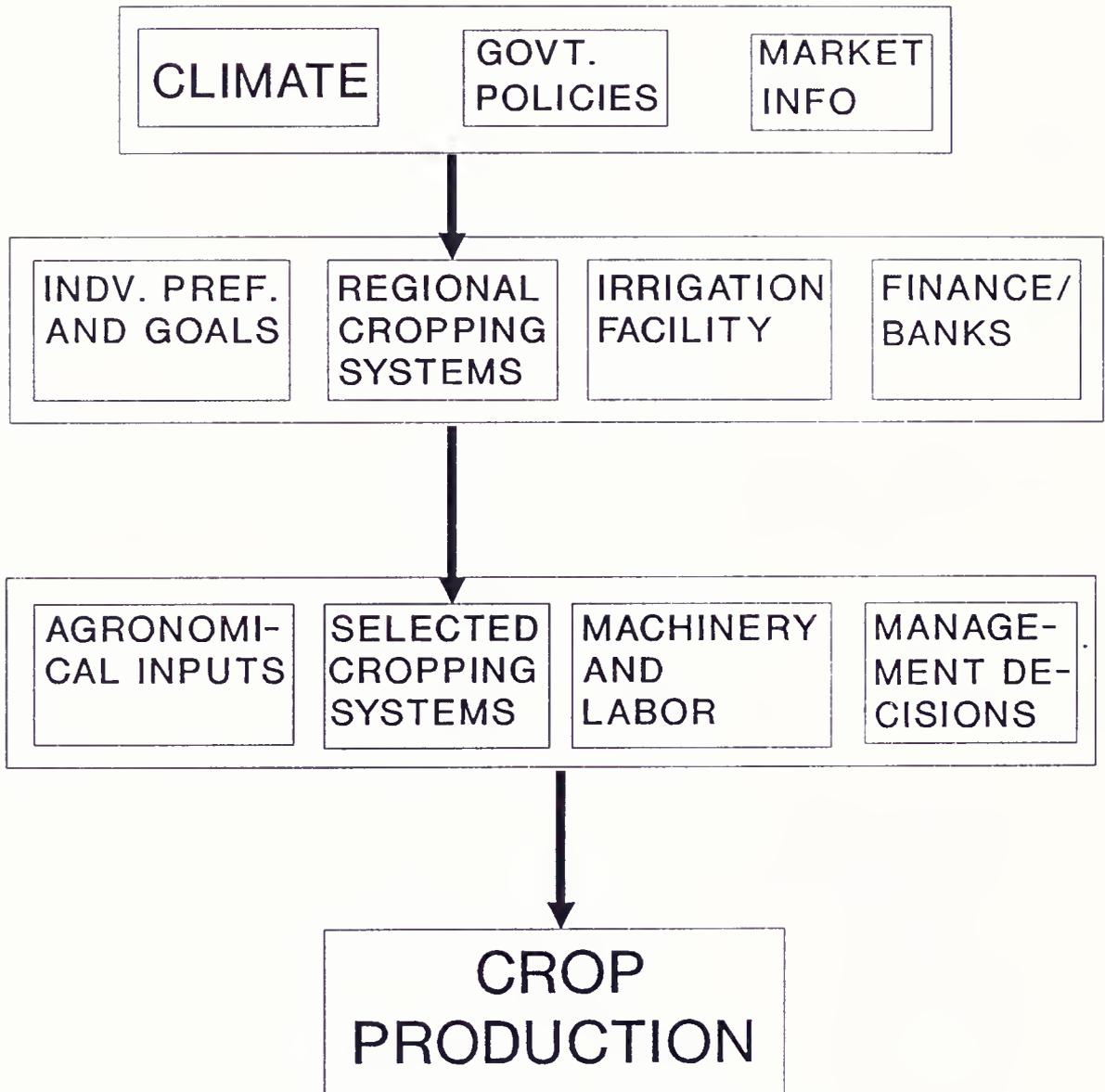


Figure 3.3 Objects and environment of an operational farm system with crop production activity.

The inferencing capability of logic programming (PROLOG) employed for writing the simulation facilitated to incorporate several expert systems within the simulation to make heuristic decisions and other types of searches at various stages of the simulation.

Farm as a System

The farm is a complex system and is considered as an intermediate level in hierarchical representation of an agricultural system (Hart, 1984). It contains many subsystems and sub-subsystems. Each of these systems contain many object-classes and object-items along with their attributes and values. Table 3.1 presents the object-classes with their attributes needed for simulating field operations for an operational farm system, and the specific items of each type of these object-classes with their attributes are depicted in Figure 3.4 for the Davis farm used as a test farm for the operational verification of the model discussed in chapter IV.

The interrelationship between different farm object-classes in the form of the semantic nets is shown in Figure 3.5 and discussed below.

The farm has a name, location, total area, cultivated area, a principal activity and an extension code name for storing its (farm) information. It possesses fields, crops, implements, tractors, laborers and a work schedule.

```

farm("DavisFarm", "SantaRosa", 400, 400,
     "CropProduction", "dvs")

labor("Laborer_1", "Jerry", "m", ["General"],
     "Avail")

tractor("Tractor_1", "JD4450", 148, ["Jerry"],
     "Avail")

implement("Implement_2", "LandPrep", "DiskHarrow",
     ["Tractor_1", "Tractor_2"], 4.74, 7, "Avail")

crops("Crop_3", "Wheat", ["Fertilize", "Harrowing",
     "Plowing", "Planting", "FertiSpreading",
     "Harvesting"])

fieldi("Field_1", "field1", 20, "SandyLoam",
     ["Cotton", "Wheat"], ["DPL41", "FL302"],
     [150, 165], ["Fertilized", "Fertilized"],
     "NotIrrigated", "NotNeeded", [750, 125],
     [1200, 110], "Avail")

operations("PlantFertiOps", "Planting", "Soybeans",
     136, 182, [[0, 3, 1], [3, 8, 0.5]], ["Planter_77"],
     0.76)

```

Figure 3.4 Examples of items of different object-classes of Davis farm in the format of dynamic databases of PROLOG.

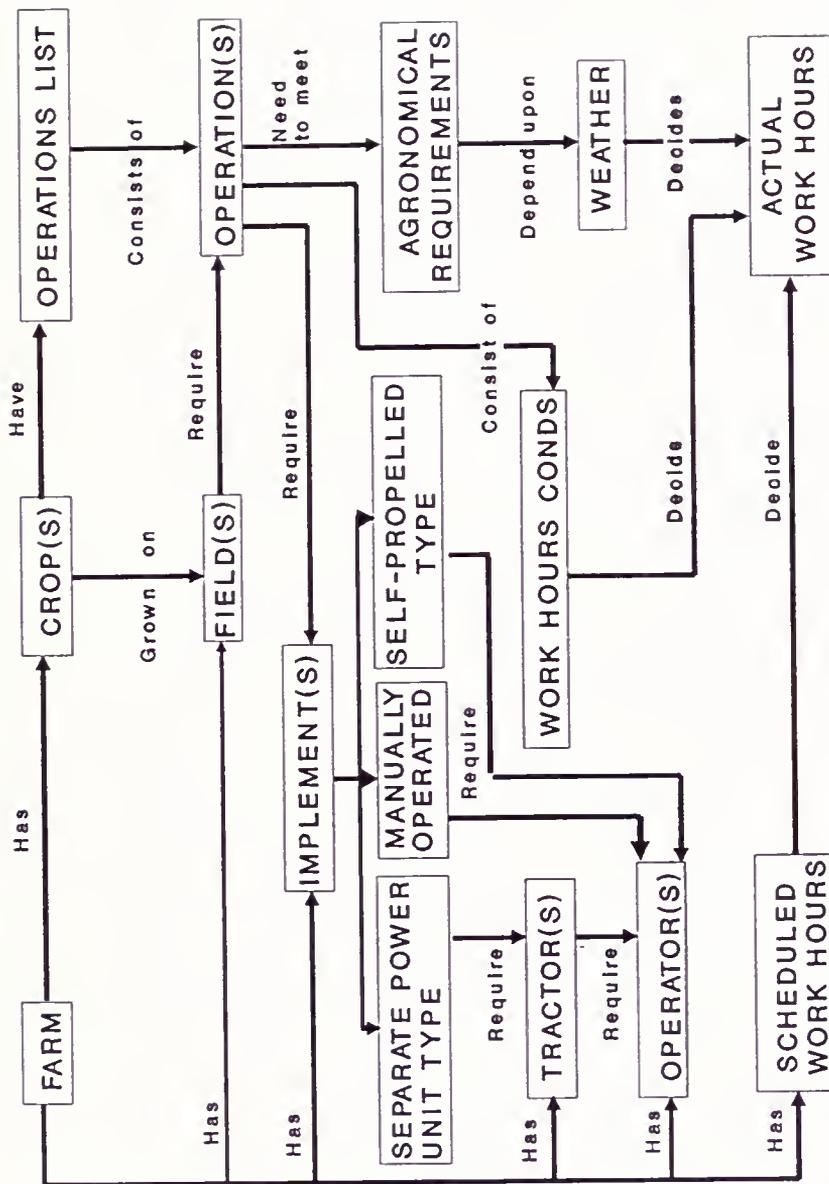


Figure 3.5 Semantic representation of farm knowledge

Table 3.1 Farm Objects and their Attributes

Objects	Attributes
Farm	Name, Location, Total Area, Cultivated Area, Principal Activity, Farm Code
Labor	Number, Name, Sex, Functions, Availability
Tractor	Model, Hp, Operators List, Availability
Implement	Number, Type, Name, Tractors/Operator List, Width, Speed, Availability
Irrigation Equipment	Number, Specification, Capacity, Operators List, Availability
Crop	Number, Name, Operations List
Field	Number, Identification, Area, Soil Type, Crop List, Variety List, Maturity Days List, Fertilizer Type List, Irrigation Type, Production Cost List, Sale Price List, Availability
Operations	Operation Type, Operation Name, Crop, Earliest Start Time, Latest Finish Time, Work Hours Conditions List, Implement List, Efficiency
Scheduled Work Hours	Month, Schedules Daily Working Hours

A laborer has his/her name, sex, functions and availability status. The number of laborers available on a farm could depend upon its size, mechanization level and cropping intensity.

A tractor has a number, make and model, horse power, a list of operators and its availability status. The horse power of the tractor controls the type of implement it can operate. The operators list contains the names of the farm laborers who can operate the tractor.

Irrigation equipment, similar to a tractor, has a number, specification, capacity, list of operators and its availability status. If the equipment does not need any operator during its operation, then its operator list could contain "NotNeeded."

An implement is characterized by its type, name, working width, speed of operation and a list of tractors or operators needed to operate it. For a self-propelled and manually operated implements, the list contains the names of the laborers who can operate the implement. In case of the implement which needs a separate power unit for its operation, this list consists of tractors or animals needed to power the implement. For completing the machinery system for such an implement, the laborer needed to operate the implement is selected from the list of laborers attached to the selected power unit for use with the implement.

A crop has a number, its name and a list of operations needed to grow it. In addition, the variety, number of days for maturity after planting, the fertilization level, the production cost and the sale price associated with the crop are also assigned to the field on which it is grown.

A field has a number, its name, the area, the soil type, irrigation facility, etc. It can be utilized to grow two crops in a year. A farm could have any number of fields as long as the sum total of the fields' area does not exceed the cultivated area of the farm.

An operation has a type, name, earliest start time, latest finish time, work hour conditions, efficiency and an implement list. The earliest start time is the time before which the operation cannot start, and the latest finish time is the time after which the operation should no longer be continued. The work hour conditions associated with an operation are a set of conditions which define how many hours the operation should be carried out depending upon the rainfall and the scheduled work hours for the day. The implement list contains implement names which could be utilized for the operation.

The scheduled work hours are daily working hours for each month of the year. It is utilized to estimate the actual work hours for different operations based upon their work hours conditions and rainfall of the day.

In addition to the farm objects, the simulator also needs local weather and evapo-transpiration data and crop irrigation requirements for scheduling irrigations.

Knowledge Representation

The information about different farm objects for the simulation is stored in the form of relational tables (referred as dynamic databases for Turbo PROLOG). It is a special feature of the logic programming environment which facilitates representing facts about the objects and their relationships within the system. It can include qualitative knowledge, in addition to quantitative and procedural knowledge, which would have been difficult to do in conventional programming environments.

The preference of the farmer for selecting operators out of his labor crew for a tractor and assigning an order of priority for each of them is a typical example of qualitative knowledge. It has been captured in PROLOG by representing the selected operators in a list and attaching it to a tractor. The search for an operator to work with the tractor is carried from left to right. So, the laborers in this list should be listed in order of preference to operate the tractor. Similar logic applies to the list of the tractors for an implement and to the list of the implements for an operation. In addition, this representation also provides the facility of listing

backup implements for an operation which are generally ignored in conventional programming models of simulation.

Model Description

The simulation model is structured in two distinct components: a) farm and region specific knowledge, and b) procedural simulation knowledge. The farm and the regional knowledge needed as an input to the simulation is maintained separate from its procedural knowledge. This has made the model a versatile and flexible. It can be utilized for different kinds of farm settings with little or no modifications.

Farm specific knowledge consists of a number of ASCII files containing information about the different objects of the farm system in the form of dynamic databases of PROLOG (Figure 3.4 and Appendix I). This component needs to be developed by the user for his specific farm setting utilizing Intelligent Information System discussed later in this chapter.

Procedural simulation knowledge is a PROLOG program consisting of several predicates and clauses (discussed later in this chapter). It carries out the actual simulation. The compiled version of the program cannot be accessed by the user for any modification. The execution of the simulation goes through the following steps in sequence as depicted in Figure 3.6.

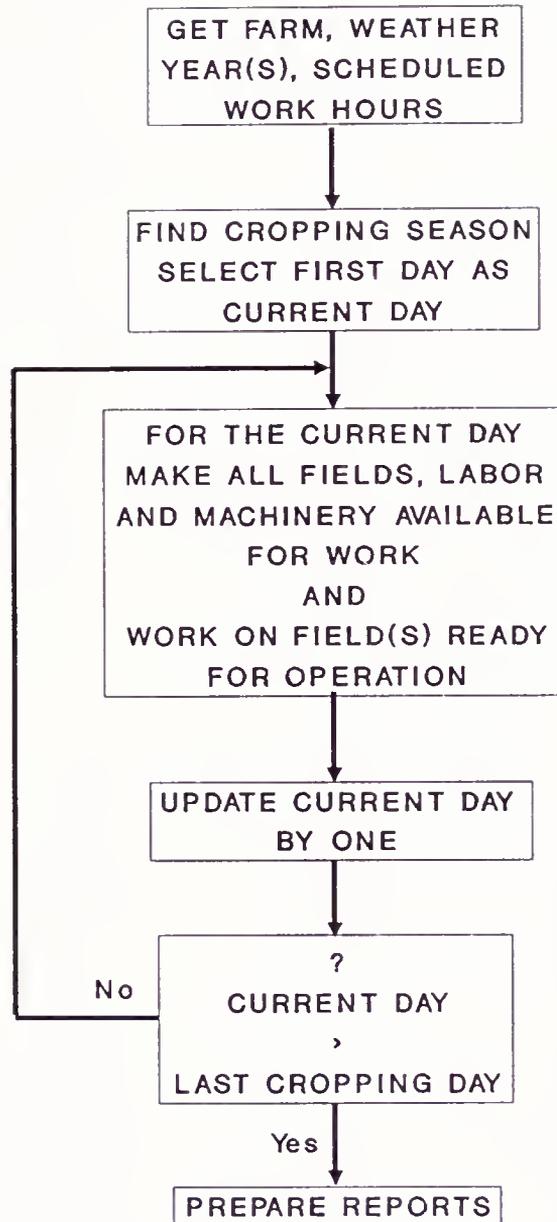


Figure 3.6 Flow chart of execution of the operations simulation

Step 1. Get farm name and consult all related files.

Step 2. Let the user select weather year and allow him to adjust work schedules, if he so desires.

Step 3. Simulate the operations for the cropping season. This step finds out the first and last day of the cropping season and carries out the following during this time interval.

I. Make the first day of the cropping season as the current day and carry out the simulation for the current day using following sub-steps.

1. Make all fields, labor and machinery available for work.

2. Pick up the first field from the field file and carry out the checks and the actions listed in Figure 3.7 for scheduling irrigation and other field operations.

3. Repeat the above procedure for all fields of the farm.

II. Update the current day by one day.

III. Check if new day is greater than the last day of the cropping season, if not repeat the above process, else terminate the simulation.

Step 4. Preparation of simulation reports.

The simulator prepares five reports. Specific contents of these reports are listed in Table 3.2. The user can access any of them by use of any editor to analyze the simulation results. However, the summary report is automatically displayed on the screen on completion of the simulation.

- 1) Calculate soil moisture of the day based upon yesterday's ET, rain and irrigation, if any.
- 2) Check if the field has a irrigation facility and if so find out the irrigation equipment used on the field.
- 3) Check the availabilities of irrigation equipment and any operators needed to operate the equipment.
- 4) Check number and amount of pending irrigations
- 5) Irrigate the field, if soil moisture is below threshold, number or amount of pending irrigations is not zero and irrigation equipment and operator are available.
- 6) Make the field, the equipment and the operator associated with the irrigation non-available for the day, else do not change their status. Update the soil moisture status of the field irrespective of decision about the irrigation.
- 7) If the field is not being irrigated, get the list of operations associated with the crop being grown on the field.
- 8) If the operations list is not empty, pick up first operation from the list.

Figure 3.7 Tasks performed by the Operations Simulator for scheduling irrigation and other field operations for a field on daily basis.

9) Check if the operation is the type of operation being tried now.

10) Get details for the operation such as agronomical work window, implement list, etc.

11) Check suitability of the "current day" for the operation with respect to its agronomical work window. The "current day" should be within the work window of the operation.

12) Check the availability of the required machinery set for the operation; the implement and the operator in case of self-propelled implements, and the implement, the tractor (power unit) and the operator in the case of other types of implements.

13) If all the above conditions are satisfied carry out the operation based upon the implement characteristics; working width, speed of operation and actual work hours and operation efficiency.

14) Record the work and no work as applicable and remove the operation from the list, if it has been completed or its latest finish time has passed.

15) Make the machinery set (Implement, Tractor and Operator) and the field not available for the day.

Figure 3.7--continued

Table 3.2. Simulation Reports and their Contents

Report Description	Contents
Work Report for Irrigation	Julian day, field name, irrigation applied (mm of water), irrigation equipment and operator used, day's soil moisture, pending number of irrigations and their amount.
Work Report for Field Operations	Julian day, month, field, crop, operation, total area, accumulated done area, day's work, implement, tractor, operator, and number of hours worked.
No-Work Report for Irrigation	Julian day, month, field, crop, irrigation equipment and its availability report, operator and its availability report.
No-Work Report for Field Operations	Julian day, month, field, crop and operation, total area, done area, reason of no-work, implement list and its availability report, operator list and its availability report, tractor list and its availability report.
Summary Report for Field Operations	Field, crop, operation, scheduled start and finish times, actual start and finish times, number of days and hours worked, total done area, number of non-working days.

In the process of the simulation the fields are served on a first-come first-serve basis. The user should enter them in order of the priority that he wants them to be checked for different operations.

The priority for different operations is built-in the program code. Irrigation, if the field has irrigation facility, is given the highest priority followed by harvesting, plant protection, planting and land preparation operations. However, the structure of the simulator permits one changing the priority to suit specific cases.

Machinery and labor present on the farm are assumed to be available for work for the complete duration of scheduled work hours. They are also assumed to have 100% reliability within the operational efficiency defined for the operation they are assigned to perform.

Simulation in PROLOG

PROLOG is an AI language which has been utilized for developing knowledge systems. The use of this language for simulation is rather recent and limited. Therefore, this section explains how PROLOG has been utilized for carrying out the present simulation.

A program in Turbo PROLOG consists of five sections namely domains, databases, predicates, clauses and a goal. The sections of predicates and clauses are the principal ones. A predicate is declared by stating its name and the domains

of its arguments. A clause is either a fact or a rule corresponding to one of the declared predicates. Borland (1986) describes the contents of each section in detail.

PROLOG has a built-in inference engine which searches through the knowledge-base in a backward chaining manner. In this mode, a goal to be achieved is pursued by searching through the true clauses. It includes a pattern matcher, which retrieves stored (known) information by matching answers to the questions. The information supplied to PROLOG (the facts about the objects of the system and rules governing their actions and behavior) become the finite set of knowledge to work on.

In addition to logically finding answers to questions, PROLOG can deal with alternatives and find all possible solutions rather than only one. Instead of proceeding from the beginning of the program to the end, PROLOG can actually back up and look for more than one way of solving each part of the problem. This process of backing up is known as "back tracking" and can be achieved by various ways as discussed by Flowers (1988). The process of "back tracking" has been utilized for simulation in PROLOG to develop an effect similar to looping in conventional procedural languages.

A simplified version of the actual program code of the simulator is presented in Figure 3.8. The given code consists of three principal predicates, "doSimulation",

```

doSimulation:-
    getFarmKnowledge(labor, field, equipment,
        crop,irrigate, operation, tractor,
        implement),
    getOtherKnowledge(expertfile, weatherfile,
        workhrsfile),
    findSimulationDuration(SDay, FDay),
    simulateSeason(SDay, FDay),
    writeResultFiles(result1, result2, result3, result4).

simulateSeason(SDay, FDay):-
    asserta(currentday(Sday)),
    repeat,
    currentday(SimDay),
    makeAvailAll,
    simulateADay(Simday, "Irrigation"),
    simulateADay(Simday, "HarvestingOps"),
    simulateADay(Simday, "PlantProtectionOps"),
    simulateADay(Simday, "PlantFertiOps"),
    simulateADay(Simday, "LandPrepOps"),
    Sday1=Simday+1,
    changeDay(Sday1),
    Sday1>Fday.

simulateADay(SDay, OpsType):-
    fieldc(FieldN, Crop),
    chkCondAndWork(FieldN, Crop, Sday, OpsType),
    fail.

simulateADay(,):-!

chkCondAndWork(FieldN, Crop, ,):-
    fieldo(FieldN, , , , Crop, OpsList, ,
    _/_/_/_/_),
    OpsList=[],!.

```

Figure 3.8 Simplified code of operations simulation in PROLOG

```

chkCondAndWork (FieldN, Crop, SDay, OpSType) :-
  fieldo (FieldN, Tarea, CUarea, Darea, Crop,
    OpList, __, __, __, __, __, __),
  OpList=[Op1|_],
  type (OpSType, OpSTypeList),
  not (member (Op1, OpSTypeList)),
  operations (_, Op1, Crop, __, FtOpera, __, __, __),
  chkTimeFinish (Sday, FtOpera, AnstT),
  adjustParameters (OpList, Tarea, CUarea, Darea,
    AnstT, "No", OpList1, Tareal, CUareal, Dareal),
  recordWork (FieldN, Crop, Tareal, CUareal, Dareal,
    OpList1, "Avail"), !.

chkCondAndWork (FieldN, Crop, SDay, OpSType) :-
  fieldo (FieldN, Tarea, CUarea, Darea, Crop,
    OpList, __, __, __, __, __, __),
  OpList=[Op1|_],
  fieldi (FieldN, __, __, __, [Crop1, Crop2],
    __, __, __, __, __, __),
  Crop=Crop2,
  not (Crop1="NoCrop"),
  fieldo (FieldN, __, CultArea, __, Crop1,
    CurOpList, __, __, __, __, __, __),
  decideWork3 (CurOpList, CultArea, Reason),
  recordNoWork (Sday, FieldN, CUarea, Crop, Op1,
    OpSType, Area, Reason, [], "NotChecked", [], "
    NotChecked", [], "NotChecked"),
  operations (_, Op1, Crop, __, FtOpera, __, __, __),
  chkTimeFinish (Sday, FtOpera, AnstT),
  adjustParameters (OpList, Tarea, CUarea, Darea,
    AnstT, "No", OpList1, Tareal, CUareal, Dareal),
  recordWork (FieldN, Crop, Tareal, CUareal, Dareal,
    OpList1, "NotAvail"), !.

chkCondAndWork (FieldN, Crop, Sday, OpSType) :-
  fieldo (FieldN, Tarea, CUarea, Darea, Crop,
    OpList, __, __, __, __, __, __, "NotAvail"),
  OpList=[Op1|_],
  recordNoWork (Sday, FieldN, CUarea, Crop, Op1,
    OpSType, Area, "FieldNotAvail", [],
    "NotChecked", [], "NotChecked",
    [], "NotChecked"),
  operations (_, Op1, Crop, __, FtOpera, __, __, __),
  chkTimeFinish (Sday, FtOpera, AnstT),
  adjustParameters (OpList, Tarea, CUarea, Darea,
    AnstT, "No", OpList1, Tareal, CUareal, Dareal),
  recordWork (FieldN, Crop, Tareal, CUareal, Dareal,
    OpList1, "NotAvail"), !.

```

Figure 3.8--continued

```

chkCondAndWork(FieldN,Crop,SDay,OpsType):-
  fieldo(FieldN,Tarea,CUarea,Darea,Crop,
    OpsList,_,_,_,_,_,_),
  OpsList=[Ops1|_],
  operations(_,Ops1,Crop,StOpera,FtOpera,
    OpsCondList,ImpList,Eff),
  chktime(SDay,FieldN,Crop,Ops1,StOpera,
    FtOpera,AnsT),
  decideWork1(SDay,FieldN,CUarea,Crop,Ops1,
    OpsType,Darea,Ans),
  MachinesLAvail(ImpList,Implement,Tractor,
    Operator,ImpWidth,Speed,AvailI,TraList,
    AvailT,OprList,AvailO)
  decideWork2(SDay,FieldN,CUarea,Crop,Ops1,
    OpsType,Darea,ImpList,AvailI,TraList,
    AvailT,OprList,AvailO),
  weather(Sday,Month,_,_,_,Rain),
  workHrs(Month,WrkHrs),
  calculateHrsIndOps(OpsCondList,Rain,
    WorkHrs,WrkHrs),
  dayWork=Speed*ImpWidth*Eff*WorkHrs/10.0,
  TDarea=Darea+DayWork,
  ChangeOpStday(Sday,FieldN,Crop,Ops1),
  chkTimeFinish(Sday,FtOpera,AnsT),
  chkAreaFinish(CUarea,TDarea,AnsA),
  adjustParameters(OpsList,Tarea,CUarea,TDarea,
    AnsT,AnsA,OpsList1,Tareal,CUareal,Dareal),
  recordWork(FieldN,Crop,Tareal,CUareal,Dareal,
    OpsList1,"NotAvail"),
  makeMachinesOccp(Tractor,Implement),
  makeEntityOccp(labor,Operator),
  minR(TDarea,CUarea,DareaR),
  maxR(CUarea,DareaR,CUareaR),
  writeworkreport(SDay,Month,FieldN,Crop,Ops1,
    CUareaR,DareaR,DayWork,Implement,
    Operator,Tractor,WorkHrs),!.

chkCondAndWork(, , , , ) :- !.

```

Figure 3.8--continued

"simulateSeason" and "simulateADay" which work in an hierarchical order. They can be thought of as subroutines in procedural languages.

The predicate "doSimulation" creates the necessary conditions for the simulation by bringing the farm as well as other knowledge to the memory using predicates "getFarmKnowledge" and "getOtherKnowledge." It then calls "simulateSeason" and writes reports on successful completion of the seasons's simulation.

The "simulateSeason" predicate has two arguments: start day (SDay) and finish day (FDay). The Turbo PROLOG predicate "asserta" puts "SDay" into the dynamic database "currentday." Then it collects the same day value by use of the database predicate "currentday(SimDay)." Then it calls "simulateADay" five times with the current day and different operation types. The order of calling this predicate with different operation types determine their priority over one another. Under the current format irrigation gets the highest priority followed by harvesting, plant protection, planting and fertilizer application, and land preparation operations.

On the successful completion of the one-day simulation, the current day is updated by one day ($Sday1=Sday+1$), and the content of the "currentday" database is changed by the use of predicate "changeDay." Finally, a check is made to see if the updated day is greater than the last day for the simulation ($Sday1>Fday$). This condition is satisfied when the simulation

for the whole period is completed, otherwise the condition fails and backtracking begins.

The backtracking proceeds up through different predicates until it encounters "repeat." The "repeat" predicate is a non-deterministic clause which always reverses the backtracking process. On the reversal of the process, PROLOG finds the content of the "currentday" database changed and picks it up for further actions of calling the "simulateADay" predicate with the new day. The process is repeated with one day increments until $(Sday1 > Fday)$ becomes true and the execution of the predicate "simulateSeason" is successfully completed.

The "simulateADay" predicate is designed to carry out one day simulation for the current day (Sday) and the given operation type (OpsType). The dynamic database "fieldc" consists of entries of all the fields with the crop being grown on them. These entries are made in order of priorities of the fields to be served by the simulator. During execution of this predicate, the first field (FieldN) and the crop (CropN) being grown on it are given to predicate "chkCondAndWork" to carry out its task.

The predicate "chkCondAndWork" checks various conditions and carries out different actions as listed in Figure 3.8 for the simulation. It also calculates the amount of work done based upon the available machinery set and prepares different reports.

On successful completion of the task of "chkCondAndWork" for the current field, crop and operation type, the predicate "fail" causes the clause to fail and forces it to backtrack.

In this case, the backtracking occurs up to database predicate "fieldc" which acts like a non-deterministic clause. At this stage PROLOG picks the contents (FieldN and CropN) of the next entry in the database and repeats the steps of this clause in hope of succeeding. However because of the "fail" predicate, it fails again. The process is repeated until all the entries of the "fieldc" database have been tried. Having worked on all the entries of the database "fieldc", the clause fails even prior to calling predicate "chkCondAndWork." Then, PROLOG searches for an alternative "simulateADay" clause or rule, and finds "simulateADay(_,_) :- !.", which always succeeds for any value of SDay and OpSType, thus successfully completing the task of simulation for the instantiated values of the operation type and current day for all fields on the farm. The same process is repeated for all the operation types.

The simulation checks the possibility of carrying out all types of operations on every field daily throughout the cropping season. The predicate "chkCondAndWork" does this task very efficiently. Its structure and sequencing avoids deep level searches for the operations on the fields not yet ready to receive them. It is so structured that the actual time for a day's simulation decreases as the work on different

fields is completed and the operations lists associated with them become empty.

The "chkCondAndWork" is a multi-clause predicate and it makes use of many other PROLOG user-defined predicates detailed in Table 3.3. The tasks performed by different clauses in order of their listing (Figure 3.8) are as follows.

The first clause checks if the current operation list is empty (OpsList=[]). If this condition fails, which means that there are still some pending operations for the field, PROLOG goes on to the next clause; else, it assumes that the work for a particular call of the predicate is done and returns that message to its user clause ("simulateADay"). The second clause determines if the operation being tried is of the type intended to be examined. If this conditions succeeds, the clause fails and control is passed on to the next clause. Else, appropriate adjustments in the field database are made using "adjustParameters" and "recordWork" predicates without carrying on any further action.

The third clause takes care of the fields with sequential cropping systems. It checks if all the operations associated with the first crop grown on the field have been completed. If this condition succeeds, the clause fails and passes the control to the next clause. Else, it prepares a no-work report with the reason determined by predicate "decideWork3" and adjusts the field's records accordingly.

Table 3.3 User-defined PROLOG predicates and their description, used by different clauses of "chkCondAndWork" of the Operations Simulator for scheduling field operations on daily basis.

Predicate	Description
fieldc	a database predicate to control sequencing of different fields in the simulation process.
fieldo	a database predicate containing additional information about fields not included in the "fieldc".
field	a database predicate to keep track of actual status of fields during various stages of simulation.
member	checks if an object (Ops1) is a member of an object list (OpsTypeList1).
operation	a database predicate to store details about operations associated with different crops. The attributes associated with the operation-class are listed in Table 3.1.
chkTimeFinish	a predicate with three arguments. It takes "Sday" and "FtOpera" as inputs and checks if current day (SDay) has passed the FtOpera and returns an answer "AnsT" (Yes or No) to that effect.
adjust-Parameter	adjusts the values of "OpsList", "Tarea", "CUarea" and "Darea" respectively to "OpsList1", "Tareal", "CUareal" and "Dareal" based upon the values of "AnsT" and "Ansa" (Yes and No) which are also supplied as input. It removes the completed operation from the current operations list and set the done area back to zero for the next operation based upon the values of "Ansa" and "AnsT".

Table 3.3--Continued

Predicate	Description
chkAreaFinish	checks if "Darea" for the current operation has equaled or exceeded the currently used area of the field.
recordWork	records the current status of the field. It removes the old status of the field from the database "field" and asserts a new entry with updated parameters which are received from the predicate "adjustParameters".
recordNoWork	collects and records information about the work which were attempted but could not be done including its reasons.
decideWork?	three predicates: "decideWork1", "decideWork2" and "decideWork3" are designed to decide further course of action based upon the instantiated values of different variables of their arguments.
machineLAvail	checks the availability the implement(s) from the implement list "ImpList" associated with the operation. The predicate returns the name of the machinery set (Implement, Tractor and Operator) and its operational parameters (ImpWidth and Speed).
weather	a database predicate stores the daily weather conditions such as Rain, etc.
workHrs	a database predicate stores the daily scheduled work hours on the monthly basis.
calculate- HrsIndOps	calculates the actual working hours (WrkHrs) for the operation based its operating conditions (OpsCondList), rainfall (Rain) and scheduled work hours (WorkHrs) for the day.

Table 3.3--Continued

Predicate	Description
changeOpStdday	changes the earliest start time for harvesting operation based upon the planting date and the number of days required to mature the crop.
makeMachines-Occp	makes the machinery set "Tractor" and "Implement" including their operator, engaged for an operation occupied; and not available for any other activity for the day.
minR	finds out the minimum value "DareaR" of two supplied input values (TDarea and CUareaR)
maxR	finds out the maximum value "CUareaR" of two supplied input values (CUarea and DareaR).
writeWork-Report	takes various parameters of the day's work and asserts them to the daily work report.

The fourth clause makes sure, before passing command to fifth clause, that the field being worked upon is available for receiving the services and is not engaged in any other operation, such as irrigation.

The fifth clause, the heart of this predicate, carries out the actual task once it receives the command. It makes a few additional checks prior to carrying out the actual task. It checks if the current day is within the operation's work window, searches for the machinery needed for the operation and determines its operational parameters, estimates the actual work hours for the operation for the day and calculates the day's work and accumulated area done by the operation. It adjusts field databases, records work or no-work reports, and makes the machinery and labor utilized in the operation occupied and not available for any other operation for the day.

The sixth clause is the terminating clause which always succeeds to continue the process if all the above five clauses fail due to some reason or other.

Expert Simulation Analysis

The foremost question in analyzing farming operations is, was a particular operation started and/or completed on time? If an answer to such a question is negative, then the next logical question is, what factors are responsible for the delays, and how they can be overcome?

Farm managers also need to know how different farm implements, power sources and labor performed on his farm during the cropping season. They want to identify under-utilized and/or over-burdened machinery and labor in order to better plan their operations for the next season. They may like to arrange supplementary units for the most needed machines or withdraw under utilized machines and labor, if possible.

The reports on the performance of any selected combination of items for different farm object-classes can be of great help to farm managers. These reports can be utilized for preparing budgets of different activities either for their own records or for tax purposes.

Answers to all above queries are available but hidden in the reports produced by the operations simulation discussed in earlier in this chapter. The output reports produced by the simulation are bulky and complex. Their complexity increases as the farm size (number of fields, crops and machinery resource) increases. Interpretation of these reports in the context of the available machinery and labor resources to respond to above queries is the most critical step in engineering farm knowledge for the decision support system.

An expert analysis system is designed and implemented in PROLOG to answer to the above queries based the simulation reports. It is designed to implement a logical reasoning

approach which characterizes delays in operations based upon the built-in or user-supplied heuristics, studies labor and machinery utilization on the farm during the cropping season, and makes recommendations for their efficient planning and management.

The expert analysis system consists of three components: 1) Operation Analysis Report, 2) Machinery Usage Report and 3) Accumulated Work Report. All these components are structured to work independently but in harmony with one another. The user can work with them in any sequence he desires. In addition, they work in an interactive and repetitive mode as depicted in Figure 3.9. The user can utilize them to prepare as many reports as he desires. On the completion of each report, the user has the option to quit the program or to get another report for a different set of inputs.

The Operations Analysis Report analyzes different operations for their timeliness in start and/or completion. It identifies factors responsible for the delays, if any, and presents them to the user along with the recommendations for their remedies in a cost effective manner.

The Machinery Usage Report provides the user with the seasonal and monthly utilization of items of different farm object-classes such as implements, laborers and tractors. It identifies the most and the least utilized items of the

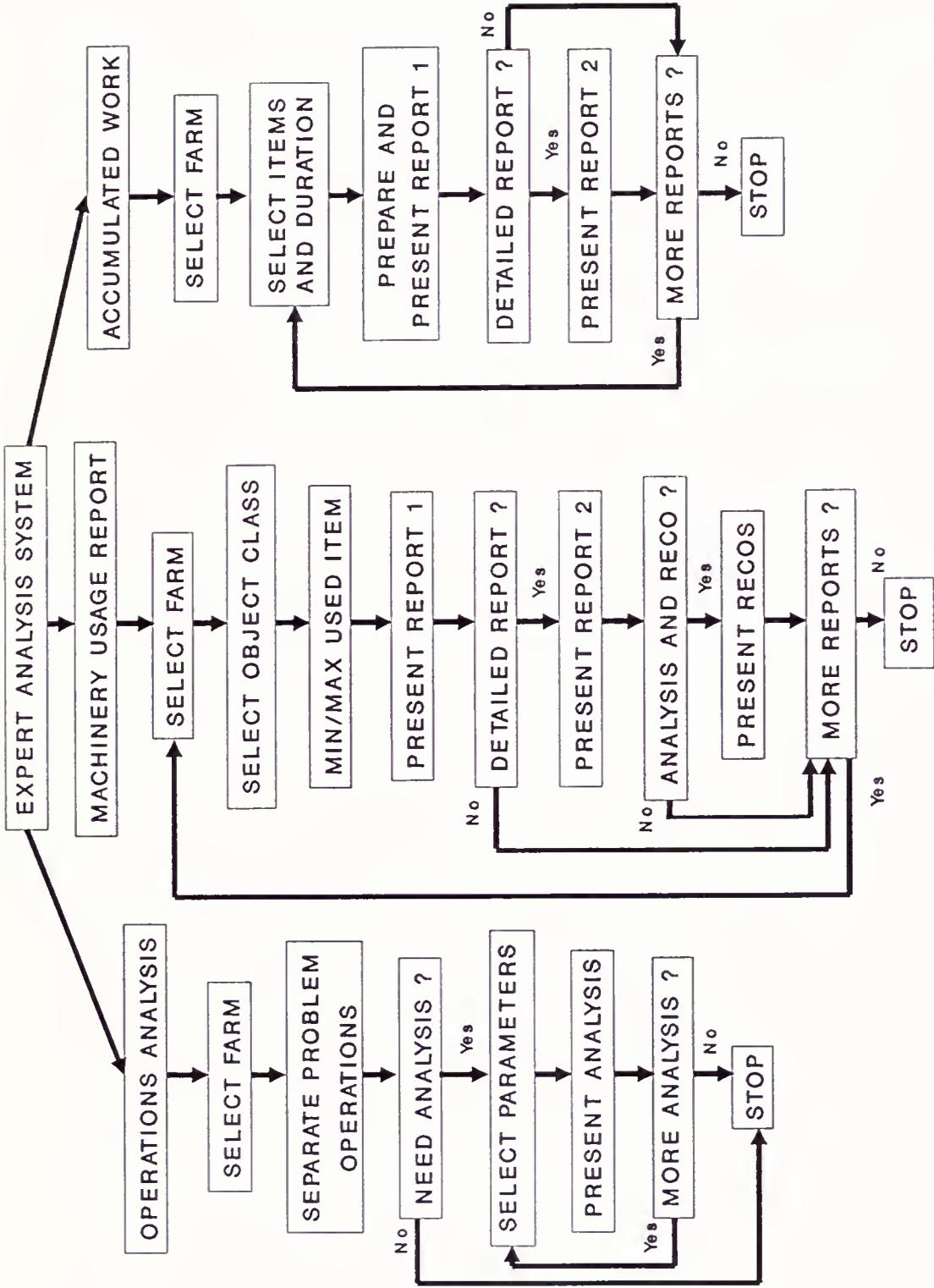


Figure 3.9 Flow chart of functioning of Expert Analysis System

selected class and recommends either to withdraw or to keep the least utilized item based upon its utilization level and its allocation strategy for different jobs supplied by the user for his farm.

The Accumulated Work Report provides the user with an aggregate work summary for any combination of the items of different farm object-classes for the desired time interval. Utilizing this report, the user can get answer to questions such as how many hours "Laborer_1" worked with "Tractor_2" for "Soybeans" fields during January 1 to June 30?

The Operations Analysis and Machinery Utilization reports work in a competitive manner with each other. The Operations Analysis report identifies bottlenecks in the farm system and always attempts to recommends for increasing the machinery capacities. On the other hand, the Machinery Utilization report identifies under-utilized machines and laborers, and attempts to recommend their withdrawal from the farm. Therefore, these two reports in conjunction with the Accumulated Work report can be a very effective means of arriving at an appropriate level of machinery and labor for a given farm system.

Operations Analysis Report

The delays in critical operations such as planting, plant protection and harvesting in crop production can cause serious damage to crop yields. These delays could be due to lower

working hours put in during the operation and/or insufficient working capacity of the machinery utilized. In addition, the delay in one operation may cause holdups in subsequent operations because of their sequential nature. The present report is prepared by utilizing the work summary report of the field operations simulator discussed earlier in this chapter, and it makes use of built-in or user-supplied heuristics to develop various recommendations. Figure 3.10 presents the flow chart of functioning of the report and its development process. For preparing this report, the system goes through a decision process of characterization of the delays, identification of operation(s) to be analyzed and the problems causing these delays, and development of appropriate recommendations. The steps involved in this decision process are discussed below.

Characterization of delays

An operation is said to be delayed if it is not carried out within a critical time period. This period for an operation could depend upon its type, location, crop and soil characteristics. It is not possible to develop any global rule set that would be applicable uniformly for all types of agricultural operations. However, for the sake of the demonstrating the functions of the analyzer, a rule set depicted in Figure 3.11 and stated in Table 3.4 has been built into the system. According to this set, an operation is

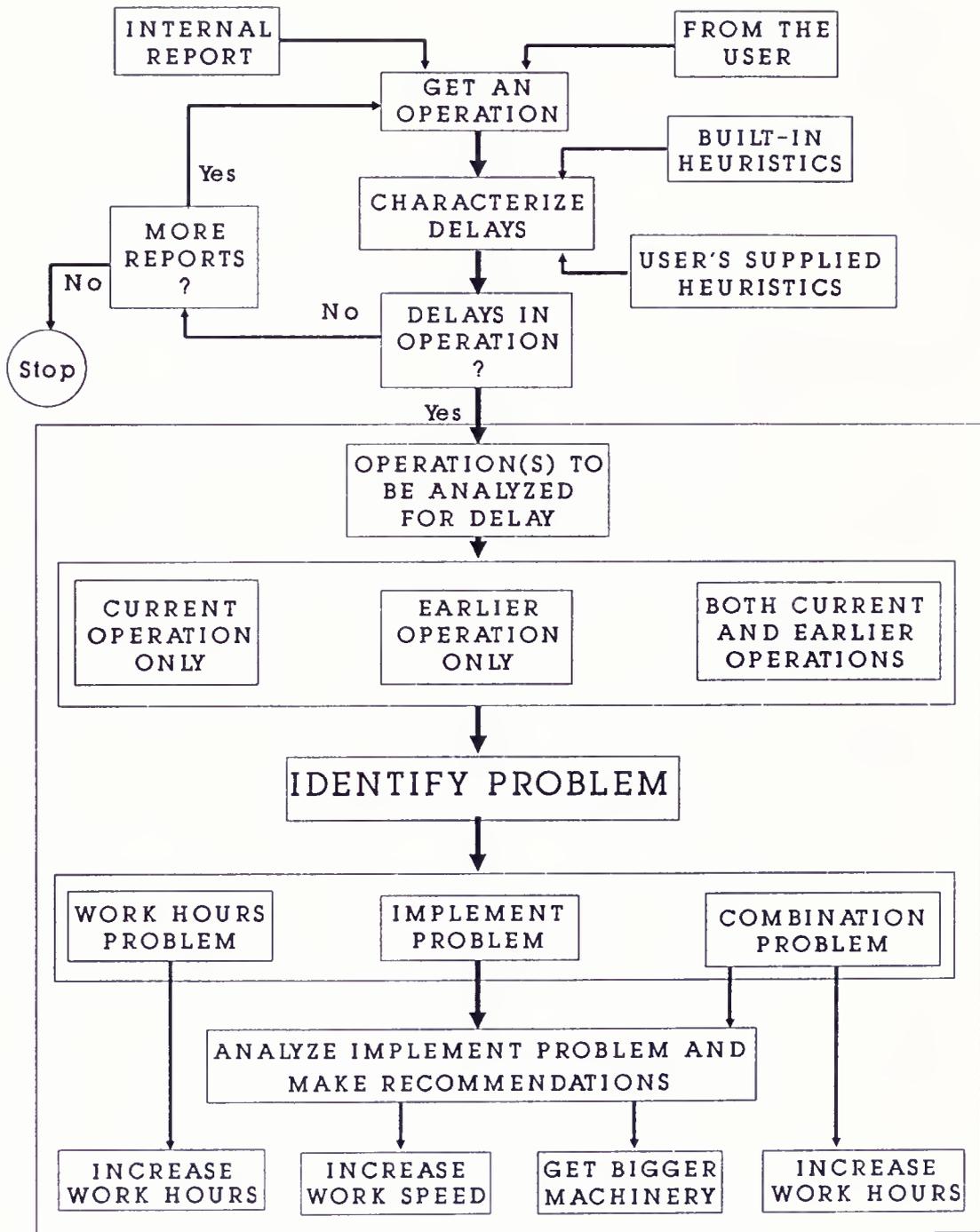


Figure 3.10

Flow chart of functioning and decision process in the Operations Analysis Report

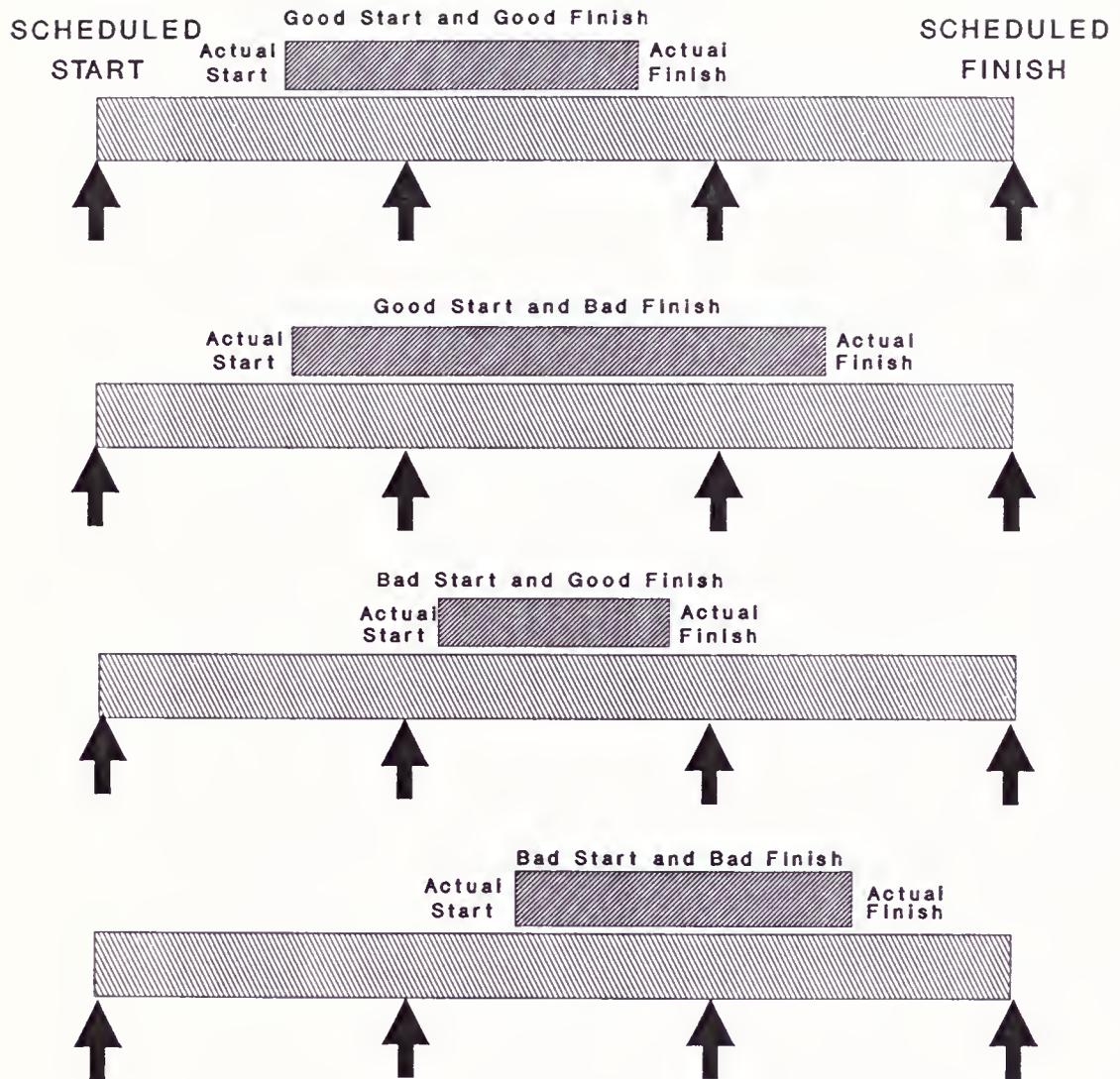


Figure 3.11 Schematic representation of built-in heuristics for the Operations Analysis report of the Expert Analysis System

Table 3.4 Rule set for qualifying delays in an operation based upon its actual start and finish times within its agronomical work window using the one-third heuristics.

Conditions	Decisions
The operation starts within first one-third and finishes before last one-third of the operational work window.	Good Start and Good Finish
The operation starts within first one-third and finishes during last one-third time of the operational window.	Good Start and Bad Finish
The operation starts after first one-third and finishes before last one-third of the operational window.	Bad Start and Good Finish
The operation starts after first one-third and finishes during last one-third of the operational window.	Bad Start and Bad Finish

considered as delayed start if it commences after one-third of its agronomic window, and is characterized as delayed finish if it continues in the last one-third of the window.

A second version of the analyzer permits the user to define the critical window within the agronomic window for each operation individually.

Operation(s) to be analyzed

The delay in an earlier operation on a field could holdup the start of the current operation. For such a situation, the system need to analyze the earlier operation either by itself or in combination with the current operation for identifying causes of delays in the current operation. This leads to three following combinations to be analyzed: 1) earlier operation only ("LastOpsOnly"), 2) current operation only ("CurrentOpsOnly") and 3) both operations ("BothOps") as shown in Figure 3.10. The system identifies such situations by looking at the actual completion of earlier operation and comparing it with the actual start of the current operation. Actual done area for the operation ("DoneArea") is also considered in making these decisions. Table 3.5 presents these criteria in the form of induction rules and Figure 3.12 shows some examples of converting the entries from this table into actual rules. The Example Rule 3 in Figure 3.12 states that the earlier operation (LastOpsOnly) need to be analyzed

```
Example Rule 1 (Entry No. 1)  
  IF Good Start and Good Finish  
  AND Done Area=Cult. Area  
  THEN Every thing as required  
  
Example Rule 2 (Entry No. 2)  
  IF Good Start and Good Finish  
  AND Done Area <Cult. Area  
  THEN Analyze the CurrentOpsOnly.  
  
Example Rule 3 (Entry No 4)  
  IF Bad Start  
  AND Good Finish  
  AND Cult.Area=Done Area  
  AND The current operation started  
    immediately after completion  
    of last operation  
  THEN Analyze the LastOpsOnly.  
  
Example Rule 4 (Entry No. 5)  
  IF Bad Start  
  AND Good Finish  
  AND Cult.Area>Done Area  
  AND The current operation started  
    immediately after completion  
    of last operation  
  THEN Analyze BothOps.
```

Figure 3.12 Examples of a rule set for identifying operations to be analyzed for delays in the current operation based upon the Table 5.2

Table 3.5 Induction table for developing rules for identifying the operation(s) to be analyzed for the delays in the current operation.

No.	Good Start	Spec. Cond.	Good Finish	DoneArea	Operations to be Analyzed
1	Yes		Yes	=Cult.Area	"Every Thing OK"
2	Yes		Yes	<Cult.Area	"CurrentOpsOnly"
3	No	(StCA=FnAL+1)	Yes	=Cult.Area	"LastOpsOnly"
4	No	(StCA>FnAL+1)	Yes	=Cult.Area	"CurrentOpsOnly"
5	No	(StCA=FnAL+1)	Yes	<Cult.Area	"BothOps"
6	No	(StCA>FnAL+1)	Yes	<Cult.Area	"BothOps"
7	Yes		No	=Cult.Area	"CurrentOpsOnly"
8	Yes		No	<Cult.Area	"CurrentOpsOnly"
9	No	(FnAL>=StCS)	No	0.0	"LastOpsOnly"
11	No	(FnAL<StCS)	No	0.0	"BothOps"
12	No		No	0.0	"MisMatchedImp"
13	No		No	=Cult.Area	"BothOps"
14	No		No	<Cult.Area	"BothOps"

where

StCA - Actual start of current operation

StCS - Scheduled start of current operation

FnAL - Actual finish of last operation

FnCS - Scheduled finish of current operation

Cult.Area- Cultivated Area

LastOpsOnly - Only last operation responsible for delay

CurrentOpsOnly-Only current operation responsible for delay

BothOps - Both operations (current and last) responsible for delay

if the operation had delayed start, timely finish and it started immediately after completion of the earlier operation.

Identification of problem

The delays in an operation can be caused because of shorter working hours and/or low working capacity of the machinery utilized. The low machinery capacity can be further attributed to either low working speed and/or to its size in general. The system is designed to identify all these factors using the following procedure.

1) The system estimates average work hours for the operation and compares them with the recommended daily work hours stored in an expert file. The average work hours for an operation is calculated by dividing total work hours by number of its working days. The recommendation is made to increase the working hours during the period of agronomic window of the operation, if the average work hours are less than the recommended work hour for the operation type.

2) In case there is no possibility of increasing work hours for the operation, the system analyzes the implements associated with the operation(s) in order to identify their (implements) operational parameters for improving timeliness of the operation.

The implement list(s) attached to the operation(s) being analyzed is(are) divided into three sub-lists of different implements types namely self-propelled, perfectly-matched and

under-sized implements. The self-propelled implements have their own built-in power source. On the other hand the perfectly-matched and under-sized implements are those which need a separate power unit for their operation.

An implement is considered as a perfect match to a power unit if the power available from the unit almost equals the power required by the implement. The implement is classified as perfect match to a list of power units if more than 50% of the power units have perfect match to the implement individually. The implements which fail to meet these tests are classified as under-sized implements.

Development of recommendations

The system makes one or more of the following recommendations in order to reduce the delays in an operation:

- 1) Increase working hours during the operation window,
- 2) Increase speed of operation or working width of under-sized implements,
- 3) Increase operational capacity of self-propelled implements and
- 4) Increase operational capacity of perfectly matched implements.

The system uses the following steps to develop the recommendations.

- 1) It identifies the operation(s) to be analyzed and their problems using the criteria discussed earlier.

2) It recommends increasing daily work hours in case it is considered as a problem. Else, it analysis the implements associated with the operation and makes recommendations to adjust their operational parameters.

For perfectly-matched implements, it recommends increasing the size of the implements and their associated power unit(s).

For the under-sized and self-propelled implements, the system compares the speed at which the implement is being operated to the recommended speed for the operation type. If it is lower than the recommended speed for the operation type, it recommends increasing it within allowable limits for the operation type. In case of under-sized implements, it calculates the possible increment in implement speed to match up its power units (not exceeding the recommended speed) and advises accordingly. Else, it recommends a bigger size implement and the associated power units, if applicable.

For a single operation case ("CurrentOpsOnly" and "LastOpsOnly", Table 3.5), it is either a "WrkHrsProb" or "ImpProb" (Table 3.6a). However, for both operations case ("BothOps", Table 3.5), it could be any combination of the two types of problems (Table 3.6b). The expert analyzer can handle all these situations.

Table 3.6a Induction table for identifying the actual problem(s) with the operation(s) to be analyzed for the delays in the current operation. a) case of one operation being analyzed, b) case of both operations (current and previous operations being analyzed)

Condition	FurtherAction
AvgWrkHrs<WrkHrsReco	"WrkHrsProb"
AvgWrkHrs>=WrkHrsReco	"ImpProb"

Table 3.6b Induction table for identifying the actual problem(s) with the operation(s) to be analyzed for the delays in the current operation. a) case of one operation being analyzed, b) case of both operations (current and previous operations being analyzed)

Current Operation	Last Operation	FurtherAction
AvgWrkHrs<WrkReco	AvgWrkHrs<WrkReco	WrkHrsBoth
AvgWrkHr>=WrkReco	AvgWrkHrs<WrkReco	CurOpsImpLastOpsWrkHrs
AvgWrkHr<WrkReco	AvgWrkHrs>=WrkReco	CurOpsWrkHrsLastOpsImp
AvgWrkHr>WrkReco	AvgWrkHrs>=WrkReco	ImpBoth

Where

AvgWrkHrs- Calculated average work hours for the operation
 WrkReco- Recommended work hours for the operation
 WrkHrsBoth- Recommend increased work hours for both operations
 CurOpsImpLastOpsWrkHrs-
 Check implement list for the current operation and recommend increased work hours for the last operation
 CurOpsWrkHrsLastOpsImp-
 Recommend increased work hours for the current operation and check implement list for the last operation
 WrkHrsProb- Work hours problem for the operation being analyzed and recommend increasing them
 ImpProb- Problem with implement used for the operation, check the implement list of the operation being analyzed

Execution procedure

The two versions of the analyzer (with a built-in heuristics and user-supplied heuristic) make the same types of recommendations and use similar development process with slightly different execution procedures.

The analyzer with the built-in heuristics automatically studies all the operations carried out by the simulation model and separates those which are considered having delays in their start and/or completion. It identifies the causes for the delays and remedies to overcome them, and stores this information in a separate file for presentation of the reports.

The reports are presented using the screen layout shown in Figure 3.13. The user selects the field, the crop and the operation from the pop-up menus for which he wants the system's analysis and recommendations. The report and the recommendations are then presented in the lower half of the screen (Figure 3.13) which contains 1) Problem type, 2) Operation(s) analyzed for the delays, 3) Possible remedies.

The second version of the analyzer carries out its execution in following steps.

- 1) Similar to version 1, it lets the user select the crop, the field and the operation, for which he wants the analysis.

- 2) It presents to the user the agronomic work window of the selected combination.

```
* FARMSYS-A Decision Support System for Multi-Crop *
*               Production Systems                   *
*   Expert Advisor-Operations Analysis Report       *
*****
Press <Enter> to Select Items of Farm Objects

Field  [ ] Crop  [ ] Operation [ ]
=====
                                OPERATIONS ANALYSIS REPORT
```

Figure 3.13 Screen layout for the Operations Analysis report with the built-in heuristics.

3) It gets the heuristic from the user to characterize the delays.

4) It then analyzes the operation based upon the supplied heuristics and presents the user with the reports using the steps discussed for earlier version of the analyzer with built-in heuristics.

The modified screen layout utilized for second version of the analyzer is presented in Figure 3.14.

Machinery Usage Report

In an operational farm system, the machinery and labor constitutes a major component of the capital outlay in the production cost of different crops. Farm managers attempt to make maximum utilization of the available machinery (tractors and implements) and labor on the farm to keep production cost low without sacrificing the product quality. They can increase the utilization level of this valuable resource-base by increasing the cultivated area, adjusting and changing the crop mixes or by withdrawing under-utilized machinery and labor from their farm.

This report analyzes the usage of machinery and labor for the given farm system based upon the work report of the operations simulation. It also identifies the least utilized item for an object-class and analyzes its usage and allocation strategies in order to withdraw it from the farm, if possible. The criteria utilized for recommending withdrawal of an

```

* FARMSYS-A Decision Support System for Multi-Crop *
*           Production Systems                       *
*   Expert Advisor-Operations Analysis Report       *
*****
Press <Enter> to Select Items of Farm Objects
Field  [ ] Crop  [ ] Operation [ ]

Work Windows      Earliest Start      Latest Finish
                Month   Date          Month   Date
Initially Supplied [ ] [ ] [ ] [ ]

Operation Consi-  Start After          Finish After
dered Delayed If [ ] [ ] [ ] [ ]
=====
                                OPERATION ANALYSIS REPORT

```

Figure 3.14 Screen layout for the Operations Analysis report with the user-supplied heuristics.

object-item, and development and execution process of the report follow.

Criteria for withdrawal

In addition to an overall utilization of an object-item, the decision to withdraw it from the farm could depend upon factors such as the importance of the item for the farm, and the effect of its withdrawal on other items of the same object-class. In addition, every farm could also have a minimum acceptable utilization level of a machinery unit not to warrant its withdrawal. Many commercial farms own machinery (both implements and tractors) more than what they generally require to complete their operations timely and efficiently during a normal year of cropping season. However, extra pieces of equipment are maintained to serve as a backups to most needed machinery to meet unwarranted situations for critical operations.

The present analysis system recommends withdrawal of an item if it is utilized less than 20% of the maximum utilized item of the same object-class as detailed in Figure 3.15a. If the 20% test condition succeeds, the second level search is made for the object-item in different knowledge-bases (Table 3.7) to evaluate its importance for the current farming system. The search aims to find out the total number of places the object-item, being considered for withdrawal, appears as a possible candidate for any work. This number is

Rule 1

IF Amt of Most Util Item = Amt. of Least Util Item
THEN Recommend not to withdraw the Least Utilized Item,
BECAUSE Farm has only one item of this type,
OR Items are being utilized evenly.

Rule 2

IF Amt of Least Util Item = 0.0
THEN Recommend to withdraw of the Item,
BECAUSE Farm is not using this Item at all.

Rule 3

IF Amt of Least Util Item ≥ 0.20 * Amt of Most Util Item,
THEN Recommend not to withdraw of the item,
BECAUSE Farm has its reasonably balanced usage.

Rule 4

IF Amt of Least Util Item < 0.20 * Amt of Most Util Items,
AND Amount of Least Utilized Item > 0.0 ,
THEN Go for Further Analysis.

Amt=Amount Util=Utilized

Figure 3.15a Rules for recommending withdrawal of the least utilized item of farm objects (machinery and labor). a) based upon the actual utilization of the items of farm object class, b) based upon the allocation strategy for the least utilized item.

Table 3.7 Knowledge-bases searched for different object-classes to find out the number of places the item appears as possible candidate for a job on the farm.

Entity Type	Knowledge-bases searched
Implements (All Types)	Operations
Tractors	Implement
Operators	Tractors, Implement and Irrigation Equipment

further divided into 1) number of places it appears as a unique candidate for the job and 2) number of places it has one or more complementary units which could assume the task of the least utilized object-item after its withdrawal.

Based upon the second level search, the rule set presented in Figure 3.15b decides and recommends for the withdrawal, if appropriate.

Development and execution process

The report is prepared utilizing the work report for the operations simulation and the other information related to the object-item supplied by the user. The system separates entries from the work report for the selected object-class. Actual work hours and days worked in by different items of the object-class are added to find out their seasonal utilization. The utilization levels, both in terms of working hours and work days, of the least and the most utilized object-items are then categorized up on a monthly basis.

The seasonal work report of the least and the most utilized object-items is presented to the user to give him a global idea about their performance. The user can also assess the monthly breakup of utilization of these items, if he desires, in order to study the variation in their usage over the cropping season. Finally, the system analyzes and presents its report on the least utilized item of the selected object-class.

Rule 5

IF NumCom=0.0,
 AND NumUnq = NumTot,
 THEN Recommend not to withdraw the Item,
 BECAUSE Item is allocated alone everywhere,
 AND It has no complimentary unit on farm.

Rule 6

IF NumCom=NumTot,
 AND NumUnq = 0.0,
 THEN Recommend strongly to withdraw the Item,
 BECAUSE Item has a complimentary unit everywhere
 it is allocated.

Rule 7

IF NumCom \geq 0.50*NumTot,
 THEN Recommend to withdraw the Item,
 BECAUSE More than 50% times the Item has
 a complimentary unit.

Rule 8

IF NumCom $<$ 0.50*NumTot,
 THEN Recommend not to withdraw the Item,
 BECAUSE Less than 50% times the Item has
 a complimentary unit.

Figure 3.15b Rules for recommending withdrawal of the least utilized item of farm objects (machinery and labor). a) based upon the actual utilization of the items of an object class, b) based upon the allocation strategy for the least utilized item.

NumTot= Total number of places the item appears as a possible candidate for a job.
 NumUnq= Number of places the item appears as a unique candidate for a job.
 NumCom= Number of places the item has at least one complimentary unit which can be utilized for the job if the item being analyzed is not available

In the execution of this report, the user is asked to select the object-class such as Tractor, Laborers, or Implements through a pop-up menu. In the case of Implements, he is asked to make second choice for the implement type class such as land preparation, planting, harvesting, etc. On the final selection of the object-class, the system goes through the preparation and presentation of the reports.

The system prepares and guards information for all the stages of the report. However, the user is not obliged to go through all of them. He can quit the advisor at any stage he wants to make changes in the farm resource-base. In making these changes, the user can use the recommendation of the expert system or his own experience.

Accumulated Work Report

This report works as a support to the earlier two reports: Operations Analysis and Machinery Usage reports. The user can utilize it to get additional details to further analyze the recommendations made by these reports and/or to get information about the items not covered by either of them. This report does not make any specific recommendations for the selected combination of object-items of the object-classes. However, its contents facilitate the user making management and planning decisions.

Development and execution process

The report is prepared utilizing the work report of the operations simulation. The user selects object-items for different farm object-classes such as crop, field, tractor, implement, etc. and the time interval for the report through pop-up menus on the screen layout presented in Figure 3.16. He can select one or all items from an object-class.

On the selection of different object-items, the accumulator separates all the entries from the simulation report for the selected combination of object-items over the desired period. It then adds up the work hours and work days of the separated entries and prepares and presents two reports.

The first report, in a concise format, consists of dates of actual start and finish of the usage of the selected object-items, total accumulated work hours, number of days actually worked, average daily work hours, average monthly work hours and list of the months during which the selected items actually worked. The second report, in a detailed format, consists of a complete listing of selected entries from the simulation work report on the daily basis over the desired time period.

The concise report is automatically presented on the lower half of the screen (Figure 3.16), however the presentation of the detailed report is optional and the user can by-pass it, if he so desires.

```

* FARMSYS-A Decision Support System for Multi-Crop *
*                               Production Systems   *
*       Expert Advisor-Accumulated Work Report      *
*****
Press <Enter> to Select          Write in Date and
Duration of the Report          Press <Enter>
Start Month  [XXXXXXXXXX]       Start Date  [XXXXXXXXXX]

Finish Month [XXXXXXXXXX]       Finish Date [XXXXXXXXXX]

Press <Enter> to Select Items of Farm Objects
Crop  [XXXX]  Field [XXXXXX]  Operation [XXXXXXXXXX]

Tractor [XXXX]  Implement [XXXXXX]  Operator [XXXXXXXXXX]
=====
                                SUMMARY AND DETAILED REPORT

```

Figure 3.16 Screen layout for the Accumulated Work Report.

Knowledge-Based Yield Estimation

The ultimate goal of a farm system with crop production activities is to produce crops, either for the market and/or for home consumption (Figure 3.3). The factors which most influence crop yields can be broadly classified into natural resource factors (soil types, climate, etc.) and management factors (crop varietal selection, irrigation and fertilization levels and timings of critical operations).

Farmers have little control on natural resource factors. However, farm productivity can be greatly improved by appropriate selection of management factors and their timely implementation. Farmers can benefit significantly with timeliness of critical operations in a crop production process. One of the important factors which controls the timeliness of field operations is the machinery and labor availability on the farm.

Farmers want to maintain an appropriate size and mix of machinery to complete their operations on time and economically. They neither want to be over-equipped with machinery which increases their fixed costs, nor do they want to be in short supply of equipment which could delay their critical operations and drastically reduce their crop yields and profits. They also want to know how an addition of a particular implement would alter the timeliness of different operations and would influence the crop yields and overall farm production and profits.

A knowledge-based yield estimation system which appraises the farm production of different crops and their gross and net returns has also been developed and integrated with the farm decision system. The yield assessments for different fields are based upon the user supplied management inputs such as crop variety, irrigation and fertilization levels and simulated timings of different operations on an individual field. These yield values for different fields are then converted into the overall farm production and profit based upon the cost of production and expected sale price for different crops.

The user can utilize this system, as demonstrated in chapter IV, to assess overall production, net and gross returns from his farm under his current farming system and to study the effect of variation in sale prices of different produce on his overall net returns. In combination with the operations simulation, the yield estimation system can also be utilized to study the effects of various machinery management decisions (daily work hours, machinery size and their working speed) on the timings of different operations and their influence on farm production and profit.

The yield estimation system is structured in two components: crop yield knowledge-base(s) and an expert search system. The crop yield knowledge-bases can be developed using the crop growth models, acquiring knowledge from regional crop experts or from any other source available in the region.

The expert search system is a PROLOG program. It collects crop related management input factors from the user-supplied field file and dates of critical operations from the simulation reports for each field and searches through the crop yield knowledge-bases to find the yield levels for the crops grown on different fields. The overall production for a field is then calculated by multiplying the yield times its harvested area.

Crop Yield Knowledge-Bases

The crop yield knowledge-bases are the relational tables of the input factors (both natural resource and management) and the expected yields for different crops. They are stored in the form of dynamic databases of PROLOG. The general format along with some hypothetical examples of these tables is presented in Appendix I.

To demonstrate the functioning of the estimation system, a set of these tables for soybeans, peanut and wheat were developed using IBSNAT crop growth simulation models (IBSNAT, 1987). A set of input values required as input for the models, and presented in Table 3.8 were selected from their respective databases. These values were chosen to approximately represent the test farm situation. The agronomic work window for planting for each crop was divided into weekly intervals. Simulation runs of the models were then made using the selected values of input factors and

Table 3.8 Values of input parameters selected for running crop models for developing hypothetical crop yield knowledge-bases

Parameter	Value selected
Weather year(s)	Tallahassee (FL), 1954 (Considered as dry and also used for operations simulation)
Soil Type	Milhopper Fine Sand
Crop Variety	
Soybean	Bragg
Wheat	Condo
Peanut	Florunner

middle date for each week within agronomic work window for planting for each crop. These runs were made using batch files specially made for each model. A single line output for each run consisting of input parameters and the crop yield and its components were saved in a separate file for every crop. These output files then served as induction tables for developing the required knowledge-bases in the form of dynamic databases.

The cotton yield values, to complete the knowledge-bases for all crops of the test farm, were then estimated on a hypothetical crop yield distribution curve (Figure 3.17). In this distribution a linear decrease in yield was assumed with delayed planting after first two weeks of the agronomic work window.

Expert Search System

This system searches through the crop yield knowledge-bases to estimate the crop yields for different fields on the farm. It can be used to get report for a single crop grown on different fields, for a single field (both or any one crop) or for the whole farm with all fields and crops.

The screen layout which allows the user select choices about the field(s) and crop(s) through pop-up menus is presented in Figure 3.18. The flow pattern of the functioning of the system is depicted in Figure 3.19. Similar to expert analysis system, it also works in a repetitive mode and the user can utilize it to get reports for different combinations of fields and crops for the selected farm. Table 3.9 presents

Table 3.9. Type of reports produced with different combination of selected entities

Selection		Type of Report
Field	Crop	
One Field	One Crop	One crop grown on the selected field
One Field	All Crops	All crops grown on the selected field
All Fields	One Crop	One crop grown on all farm fields
All Fields	All Crops	All crops grown on all farm fields

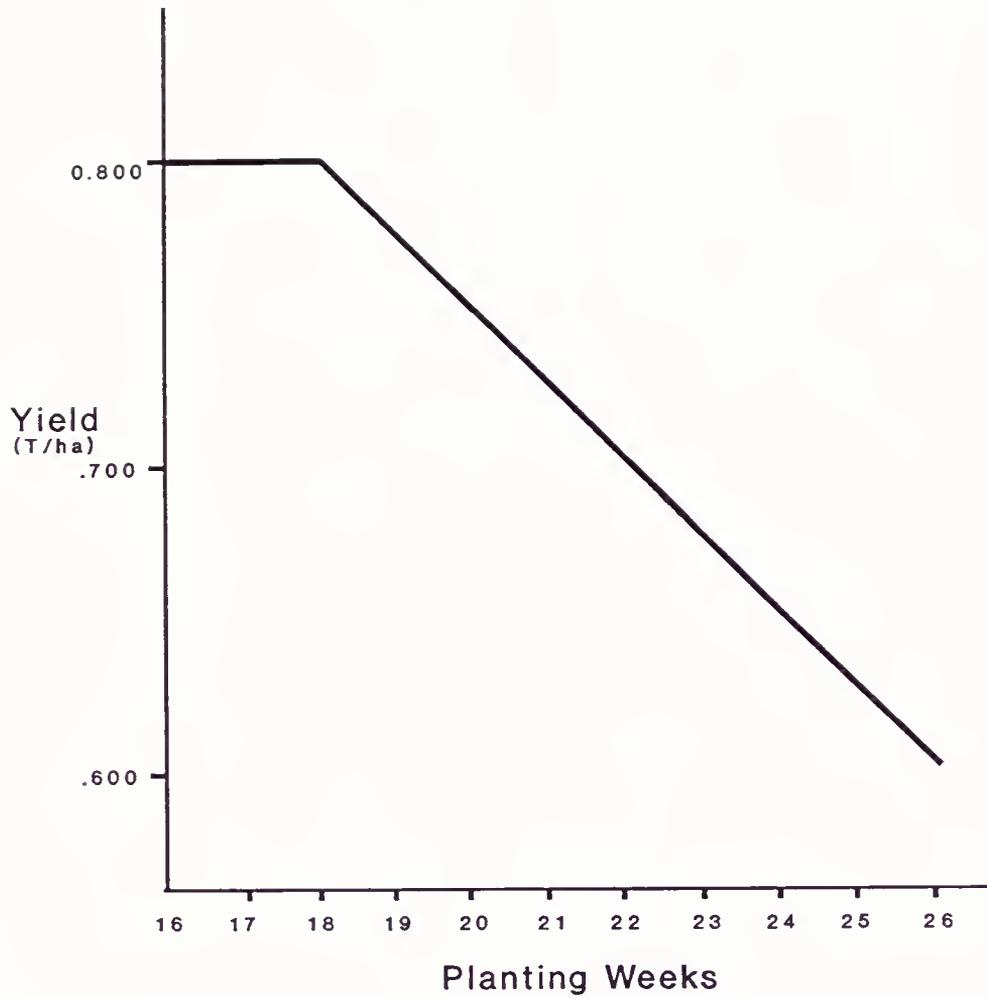


Figure 3.17 Hypothetical yield distribution of cotton crop as affected by its planting weeks.

```
* FARMSYS-A Decision Support System for Multi-Crop *
*               Production Systems                 *
*               Crop Yield and Profit Estimator     *
*****
Press <Enter> to Select Items of Farm Objects

Field(s) ████████████████████████████████████████ Crop(s) ████████████████████████████████████████

=====

                                YIELD AND PROFIT REPORT
```

Figure 3.18 Screen layout for presenting Yield Estimation Report

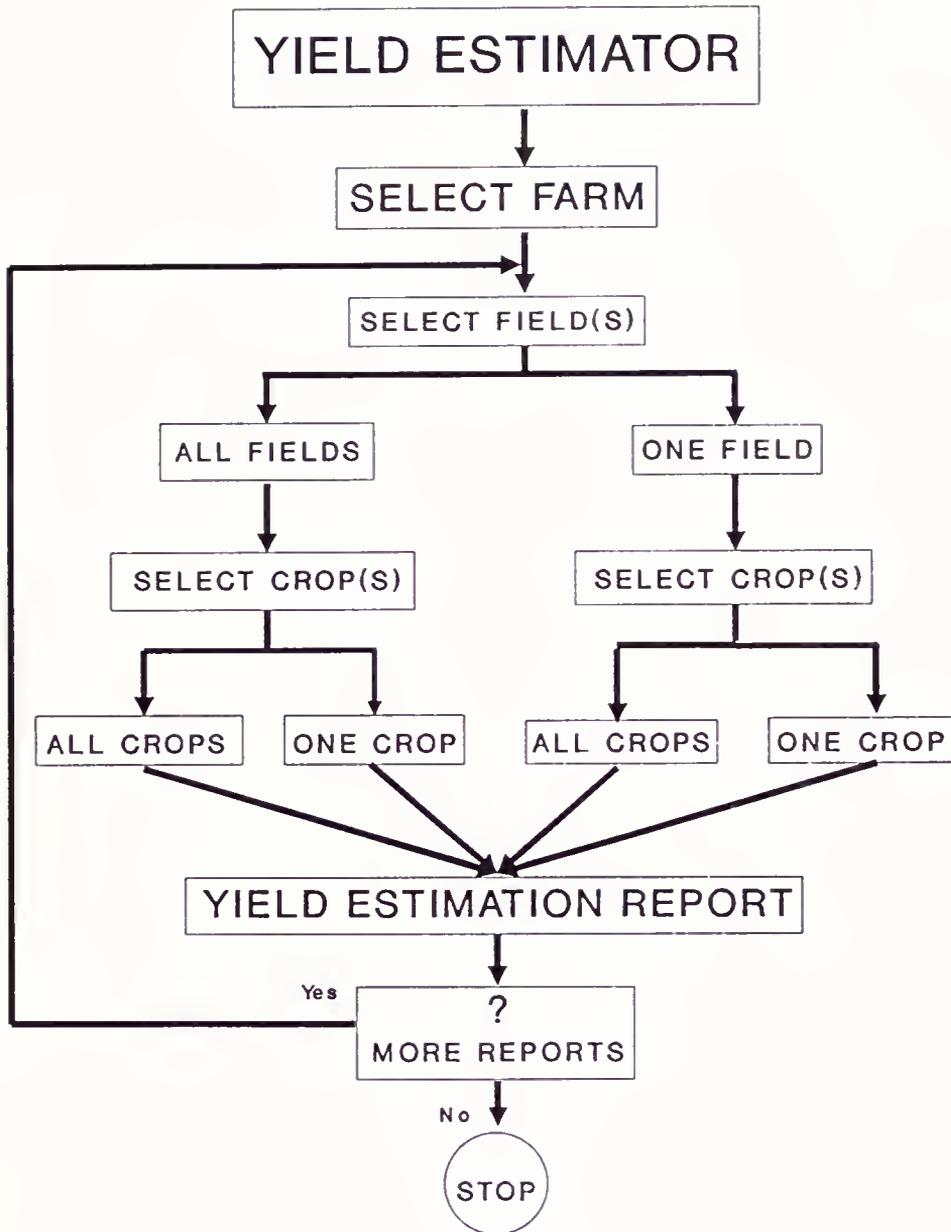


Figure 3.19 Flow chart of functioning of the Yield Estimation System.

the types of reports produced with different combinations of the selected object-items from the menus. The total production, cost, gross and net returns for a crop produced on a field are estimated using following equations.

$$P = Y * HA \text{ ----- (1)}$$

$$GR = P * SP \text{ ----- (2)}$$

$$TC = AA * PC \text{ ----- (3)}$$

$$NR = GR - TC \text{ ----- (4)}$$

$$ANR = NR/CA \text{ ----- (5)}$$

where P = Total Production (Ton)
 Y = Yield (Tons/ha)
 HA = Harvested Area (ha)
 GR = Gross Returns (\$)
 SP = Expected Sale Price (\$/Ton)
 TC = Total Cost (\$)
 AA = Average Area = (HA+CA)/2.0
 NR = Net Returns (\$)
 ANR = Average Net Return (\$/ha)
 CA = Cultivated Area
 PC = Production Cost (\$/ha)

The average production cost and expected sale price for the crop is supplied by the user. The Average Area (AA) for estimating total production cost accounts for the situations where machinery constraints result in harvested area less than the cultivated area of a field.

The report generated by the expert search system contains field name, its cultivated and harvested areas, crop name(s), weather set, yield, production, sale price, cost

price, total sale, total cost, net profit and net profit per unit area for each field and for complete farm.

An Intelligent Information Management System

Information management deals with control of inflow and outflow of information, while assuring its adequacy and accuracy (Walton, 1967). A good and intelligent information management acquires great rewards for an enterprise whereas mismanagement of information carries the threat of very high penalties. In large industrial enterprises specialists, referred to as Information Managers, are hired to carry out this task.

A software-based information management system for large and complex simulation models can increase their ease of operation and also their credibility as decision-aid tools. It can help to collect knowledge, needed as an input for the model, from the user and let him update it in a friendly and interactive manner.

The simulation model, discussed earlier in this chapter, requires a huge amount of farm specific knowledge in a predefined format for its functioning. A conventional procedural program with input data files would require a good amount of planning, organizational skill and typing ability on the part of the user to provide the model with the needed knowledge even for a small farming enterprise. The complexity of the task increases manyfold if one is required to manage

a large enterprise comprising many farms with different types of cropping systems.

A software-based information management system was designed and implemented. It is currently structured to collect farm information and works as an intelligent interface between the user and a field operations simulation model.

Object-items of an operational farm system can be classified into two categories namely 1) physical object-class and 2) abstract object-class (Table 3.10). The details about each farm object-class are stored in separate files and are discussed later in this chapter.

Items of a physical object-class can be numbered from 1 to "n", where "n" is the total number of items of the object-class available on the farm. The typical examples are tractors, implements and fields. The items of these object-classes are shared by other farm object-items. For example, Tractor_2 could be shared by five different implements to carry out seven types of operations on ten different fields.

Items of an abstract object-class are not numbered. They do not exist physically on the farm. Unlike physical object-items, they cannot be shared by other farm object-items. A typical example of this object-class is the operations list associated with a crop. This list is specific to a crop and cannot be shared by other crops. The planting operation, as an example, for different crops could have totally different requirements in terms of their timings and implement list.

Table 3.10 Categorization of different farm object-classes

Category	Object-class
Physical objects	laborers, tractors, implements, equipments, crops and fields
Abstract objects	operations

In addition, the objects of an operational farm system are inter-dependent on each other (Figure 3.5). For example, an operator is needed to make a tractor operational, and a tractor is needed to make an implement work in the field as explained earlier in this chapter. Therefore, development of a new set or updating information of an existing object-class file would require information from other related files. It could also require adjustments to one or more farm files to make the changes effective.

Table 3.11 presents the files required to carry out different actions on various farm object-classes. For instance, the action to "add New Item" or to "modify An Item" in the tractor file would require information from farm and labor files, in addition to tractor file itself. Therefore, all these files should be present and should be accessible to the system in order to carry out these actions.

Table 3.12 presents the listing of files affected by different updating options on various farm files. For example, an addition of a new laborer on the farm would not

Table 3.11 List of files needed to carry out different actions on various farm objects files.

File Name	Action Performed	Files Required
farm	list All Items	farm
farm	delete An Item	farm
farm	modify An Item	farm
farm	add New Item	Action Not Allowed
farm	develop New Set	Action Not Allowed
labor	list All Items	labor
labor	delete An Item	labor
labor	modify An Item	Action Not Allowed
labor	add New Item	farm and labor
labor	develop New Set	farm
tractor	list All Items	tractor
tractor	delete An Item	tractor
tractor	add New Item	farm, labor and tractor
tractor	modify An Item	farm, labor and tractor
tractor	develop New Set	farm and labor
implement	list All Items	implement
implement	delete An Item	implement
implement	add New Item	farm, labor, tractor, implement
implement	modify An Item	farm, labor, tractor, implement
implement	develop New Set	farm, labor and tractor
equipment	list All Items	equipment
equipment	delete An Item	equipment
equipment	add New Item	farm, labor and equipment
equipment	modify An Item	farm, labor and equipment
equipment	develop New Set	farm and labor
crop	list All Items	crop
crop	delete An Item	crop
crop	modify An Item	farm and crop
crop	add New Item	farm and crop
crop	develop New Set	farm

Table 3.11--continued

File Name	Action Performed	Files Required
field	list All Items	field
field	delete An Item	field
field	modify An Item	farm, equipment, crop and field
field	add New Item	farm, equipment, crop and field
field	develop New Set	farm, equipment and crop
operation	list All Items	operation
operation	delete An Item	Action Not Allowed
operation	modify An Item	farm, crop, implement and
operation	add New Item	Action Not Allowed
operation	develop New Set	farm, crop, implement

Table 3.12 List of affected files due to various updating options of different farm objects files.

File Name	Action Performed	Affected Files Names
farm	delete An Item	all files deleted
farm	add New Item	action not allowed
farm	modify An Item	field
farm	develop New Set	Action Not Allowed
labor	delete An Item	tractor, implement and equipment
labor	add New Item	tractor, implement and equipment
labor	modify An Item	Action Not Allowed
labor	develop New Set	tractor, implement and equipment
tractor	delete An Item	implement
tractor	add New Item	implement
tractor	modify An Item	none
tractor	develop New Set	implement
equipment	delete An Item	field
equipment	add New Item	field
equipment	modify An Item	none
equipment	develop New Set	field
crop	delete An Item	field and operation
crop	add New Item	field and operation
crop	modify An Item	operation
crop	develop New Set	field and operation
field	delete An Item	none
field	add New Item	none
field	modify An Item	none
field	develop New Set	none
operation	delete An Item	Action Not Allowed
operation	add New Item	Action Not Allowed
operation	modify An Item	none
operation	develop New Set	none

be effective until the changes in tractor, implement and equipment files are made to allocate the new laborer to different tractors and implements.

Characteristics

The principal characteristics of an Intelligent Information Management System for a farming enterprise with the crop production as principal activity can be listed as follows.

1) It should be able to adopt itself to the size and the complexity of a variety of farm systems from different locations.

2) It should be very friendly and interact with the user through menus requiring him to do a minimum amount of typing.

3) It should be able to store information about many farms to manage them individually whenever required. The information stored should be accessible to other components of the decision support system such as Operations Simulator, Yield Estimator, and Expert Advisor described earlier in the chapter.

4) It should work for developing a completely new farm set, as well as for updating the information about existing farms by modification, addition and deletion of an entry from different object-class files. It should prohibit the user from making duplicate entries of the same object-item.

5) It should carry out various integrity checks such as cultivated area vs. total area, matching implement power requirement to power available from the power source, etc. during development and updating of farm knowledge.

6) It should be able to decide and collect the information needed to complete different types of machinery set such as self-propelled and the implements needing a separate power unit.

7) The same information should not be asked repeatedly. The information once supplied should be presented to the user, whenever it is a possible selection as an input henceforth.

8) It should keep track of the status of different farm files as affected by any change in the related file(s) and should advise the user accordingly.

Components and their Functions

An information management system with the above characteristics was designed and implemented in Turbo PROLOG and is named as Information Manager. It consists of two components: 1) Information Generator and 2) Information Modifier. Information Generator lets the user develop information for a completely new farm in a predefined logical sequence. The user fills in information on the specially designed forms on the computer screen. Once any information such as tractor make or model is entered, it need not be typed in again, as it will appear on a pop-up menu the next time

when it is a possible selection as an input, thus saving many typing errors. The information collected is then stored in different files.

Information Modifier lets the user update information about the existing farms. It permits him to access any farm file in a developed set and make changes such as addition, deletion of an object-item, or modification of one or more of its attributes in its object-class. The general decision tree of the Information Manager is depicted in Figure 3.20 and is discussed briefly hereafter.

Information Generator

Information Generator lets the user develop information about different farm object-classes of a new farm. It creates separate files for each class with the extension code supplied by the user. The sequence of collecting information for different object-classes is logically structured. The most needed information is collected first.

On the successful completion of the general farm file, the name of the new farm with its extension code is registered with a built-in file manager. On the next usage of Information Manager, the name of the new farm shows up in the menu of farms' name to allow the user to update its information, if he so desires.

On initiation of Information Manager, the user is given options to select a farm from a pop-up menu which also

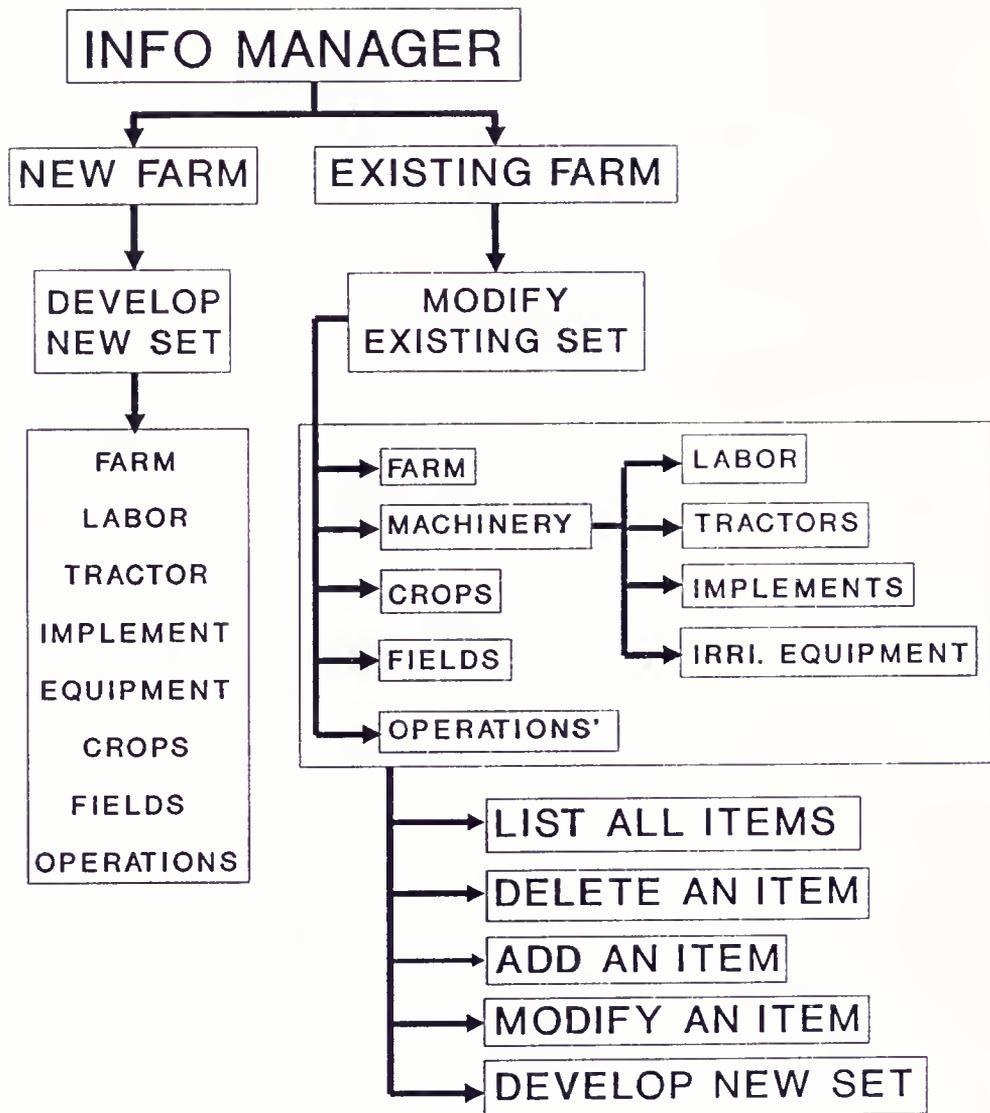


Figure 3.20 Flow chart of functioning of Information Management System

includes "NewFarm" in addition to the already existing farms. The selection of "NewFarm" leads the user to Information Generator where he can develop information about different objects of his farm. The general farm information (name, location, etc) is developed first followed by labor, tractor, implement, irrigation equipment, crop, field and operation.

The development process, in general, of a new farm information involves asking the user for the number of items of the object-class he possesses on his farm. The Information Generator then interacts with the user to collect information about each of them.

The "operation" is an abstract object-class (Table 3.10) and its development is controlled internally by the system. In developing the operation file, the user is directly presented with the screen to enter its details rather than asking for their total number as in case of physical object-classes. The development of the operations file is controlled by the crop file which has a list of operations for each crop. The system picks the first crop from the crop file and obtains details about all the operations present in its list. This process is then repeated for all the crops in the crop file.

Information Modifier

Information Modifier provides five options: list all items, delete an item, add new item, modify an item, and develop new set for updating knowledge about existing farms.

List all items lists all items for the selected object-class with their attributes on the screen. The user can examine them to decide his further action.

Delete an item allows the user to withdraw one object-item from its class at a time. In order to withdraw more items, the action should be repeated that many times. The user selects the object-item to delete from the menu of the object-class.

Add new item allows the user to add one more item to its existing object-class file at a time. The process involves counting number of object-items in the class, allocating an identification number for the new item and receiving and saving its details.

Modify an item allows the user to change one or more attributes of an item an object-class. The process involves letting the user make selection of the object-item, presenting him old attributes of the selected item, letting him make modifications and saving the item with the new attributes.

Develop a new set allows the user to develop an entirely new set for any farm object-class. The process involves letting the user select the object-class, receiving number of items of the selected class, then collecting and saving information about each item of the object-class.

On selecting one of an already existing farm, the user is led into Information Modifier where he is presented with a menu of different object-classes (farm, machinery, crop,

field and operations) to let him select one to update information. The machinery is further subdivided into labor, tractor, implement and irrigation equipment. On selecting the object-class from this menu, the user is presented with updating options menu (Table 3.11, Figure 3.19).

In updating existing information, the Information Modifier first verifies the status of the files needed to carry out the desired action (Table 3.11). In case of non-existence of any of these files, it advises the user to develop them prior to carrying out the desired action. At this stage, the user is also informed of the effects of his action on other farm files (Table 3.12) and is warned that his updates would not be effective if he does not make changes in other affected files.

It then brings the necessary files to memory, initializes screen layout, lets the user select a object-class and its item he would like to update, presents him the old information, if needed, then allows him to make changes in the selected object-item. Finally, it writes the file with updated information to the disk.

Knowledge-Bases Utilized and Generated

The knowledge manipulated by Information Manager can be broadly classified into a) knowledge utilized and b) knowledge generated by the user.

The knowledge utilized consists of 1) type file, 2) screen layout files, 3) month details file, 4) manager file, and 5) expert file.

The farm knowledge generated is stored in different files: 1) farm file, 2) labor file, 3) tractor file, 4) implement file, 5) irrigation equipment file, 6) crop file, 7) field file and 8) operation file. Appendix I presents a simplified version of these files for the Davis Farm located in Santa Rosa County of northwest Florida. This farm has been used as a test farm for operational evaluation and verification of the integrated decision support system.

Knowledge utilized

The general format of different databases utilized by Information Manager is presented in Table 3.13 and discussed briefly below.

Type file. This file consists of entries of a single database called "type." The database consists of two elements: 1) object-class name and 2) a list of object-items associated with the class. The list associated with certain object-classes can be modified by the user interactively. The typical example is crop and its varietal list. Appendix I gives the complete listing of this file for the test farm.

This knowledge-base is utilized to prepare menus to let the user select object-items specific to his farm. The user

Table 3.13 Format of PROLOG knowledge-bases of different files utilized by Information Management System to let the user generate farm knowledge

File Name	Knowledge-bases and their format
type file	type(LandPrepImp, ["MBPlow", "AddNewItem", "DeleteAnItem"])
month file	month("January", 1, 31)
screen files	field("name", str, 8, 28, 29) txtfield(8, 12, 12, "Name of Farm") windowsize(20, 77)
manager	filesmanager("FarmName", "FarmCode", "Exist", "Exist", "Exist", "Exist", "Exist", "Exist", "Exist", "NotExist")
expert	et("October", 2.55) file cropirr("Peanuts", "ClayLoam", 10, 200) soilchar("FineSand", 200.0, 120.0, 140.0) opcond("Soybean", "LandPrep", [[0.0, 3.0, 1.0], [3.0, 8.0, 0.5]]) opstypedetail("LandPrepOps", 8.0, 7.5, 7.0, 0.76)

can select a single item or a list of items depending upon the type of object-class. The selection for a variety of a crop to be grown on any field is a typical example of single object-item selection from a given list. On the other hand, selecting a list of tractors for an implement is a typical example of selecting a list from an existing list.

An example of this database, presented in Table 3.13, signifies that Information Manager currently knows only "MBPlow" as a land preparation type implement. However it can be changed by selecting "AddNewItem" or "DeleteAnItem" from the pop-up menu.

Screen layout files. These files are utilized for creating screen layouts for collecting information about different farm object-classes. There is a separate file for each class and is saved as "*****.scr," where the first word represents the name of the class such as farm, tractor, implement, etc.

These files consist of entries of three databases and have been adapted from screen handler program of Turbo PROLOG ToolBox (Borland, 1987). Figures 3.21-3.28 present the screen layouts generated by these files. Appendix I gives the complete listing for collecting general farm information.

Month file. This file contains entries of a single database with 3 elements namely month name, start Julian day

and last julian day. This database is utilized to check the validity of date entered by the user for any month as well as to convert julian day to calendar month and date and vice versa.

A typical format of this database in Table 3.13 signifies that the first and the last julian day for the month of January are 1 and 31 respectively.

Manager file. This file consists of entries of a single database named "fileManager." It has only one entry to start with. The other entries are automatically added on after the successful completion of the development of the general farm file of the new farms. It keeps track of the status ("Exist"/"NotExist") of different files of the farms available to the system.

This database consists of 10 elements: farm name, farm code and status of eight object-class files (farm, labor, etc.). This file is used to prepare the menu of farm names for letting the user select a farm with which he wants to work.

Expert Knowledge File. This file contains expert knowledge which can be acquired either from the regional experts and/or from the published literature. It stores entries for five databases namely "cropirr", "soilchar", "opcond", "et" and "opstypedetail."

The "cropirr" contains information about the irrigation requirement for various crops when grown on different soil

types. It consists of four elements: crop name, soil type, the amount and number of irrigations that could be applied during the crop cycle.

The database "soilchar" contains soil moisture characteristics of different soil types with four elements: soil type, maximum water holding capacity, wilting point, and the threshold limit to start irrigation.

The database "et" of two elements stores daily average evapo-transpiration values for different months of the year.

The "opscond" database stores knowledge for developing rules to estimate daily working hours for different operations. This database consists of three elements: crop name, operation type and a set of lists of real numbers which is used to estimate actual working hours for the day. Figure 3.29 presents an example of using this database for developing the rule set for actual work hours based upon the rainfall and scheduled work hours for the day.

The "opstypedetail" database consists of five elements to provide operational details for different operation types. These elements are operation type and its upper allowable limits for the recommended daily work hours, draft requirement and the speed of operation.

This knowledge is used to conduct various types of consistency checks on the information supplied by the user.

General format

```
opcond(Crop,SoilType,OpsType,
      ListOfListOfRealNumbers)
```

Specific example

```
opcond("Soybean","Sand","LandPrep",
      [[R1,R2,1.0],[R2,R3,0.5]])
```

Rule_1

```
IF rainfall > R1 mm for the day
AND rainfall <= R2 mm for the day
THEN do 100% of scheduled work hrs
```

Rule_2

```
IF rainfall > R2 mm
AND rainfall <= R3 mm
AND schedule Work Hrs for the day
are >4.0 Hrs
THEN do 50% of scheduled work hrs
```

Rule_3

```
IF rainfall > R2 mm
AND rainfall <= R3 mm
AND Scheduled work Hrs <=4.0 Hrs
THEN do 100% of scheduled work hrs
```

Rule_4

```
IF rainfall > R3 mm
THEN don't do any work
```

Figure 3.29 Rules for deciding actual work hours for the day based upon the operating conditions (ListOfListOfRealNumbers) of an operation type (OpsType), rainfall and scheduled work hours for the day.

For example, Information Manager identifies the inappropriate parameters for an implement in relation to its power sources. In case, the power requirement for an implement exceeds the power available from the smallest tractor selected for the implement, the user is prompted and advised to reduce either the speed or the working width of the implement.

Knowledge developed by the user

The knowledge developed by the user through Information Manager is stored in eight files of dynamic databases. The attributes of different farm object-classes and their typical format, as stored in the form of dynamic databases were discussed earlier in this chapter (Table 3.1 and Figure. 3.4). The number of entries for the physical object-class files equals their numbers present on the farm. A brief description of the mechanism of developing information about different farm object-classes follows.

Farm file. Most of the user supplied information on this screen is entered except for the principal activity which is selected from a pop-up menu. A built-in checker prevents the user to enter a cultivated area more than the total farm area. The updating options "add Mew Item" and "develop New Set" are not permitted on this file. However, the user can change cultivated area and/or total area using "modify An Item" option.

Labor file. Most of the information is user entered. This file is utilized to generate the list of operators on the farm for allocating them to different farm implements, equipment and tractors.

Tractor file. The make, model and horse power of the tractor are entered. The operators list is selected from a pop-up menu consisting of the laborers available on the farm.

All updating options are permitted on this file. The user can modify the make, model, horse power and/or operators list associated with the tractor.

This file is utilized to generate the list of tractors on the farm for allocating them to different implements.

Implement file. The implement type, its name, and tractor/operator list are selected from the pop up menus, and working speed and width are entered by the user.

The pop-up menu of the implement names can be changed interactively by the user. The menu of tractors available on the farm also includes "NotNeeded" option which can be selected for the self-propelled and the manually operated implements. When "NotNeeded" is selected, the user is presented with another menu of laborers available on the farm, and then asked to select the operators list for the implement.

All updating options are permitted on this file. The user can change operational parameters (working speed and working width) of the implements and the tractor/operator list associated with it.

This file is used to prepare the menu of implements to let the user select them (implements) for different operations while developing the operations database.

Irrigation equipment file. The equipment specifications such as model and its capacity are entered by the user, and operators list is selected from a pop-up menu consisting of laborers available on the farm. The "NotNeeded" option is also available for the equipment requiring very nominal amount or no labor at all for its operation.

All updating options are permitted on this file. The user can change the capacity and the operators list for the equipment.

Crop file. Most of the information is collected by selecting object-items from the pop-up menus. The content of the crop menu for selecting crop(s) for the farm can be modified by the user to suit his specific requirements. The information for different operation types is collected separately and then joined together in a proper sequence to form an aggregate list for the crop. The user can select the sequence of operations of each type from the pop-up menus. He can even modify the content of the menus of the land preparation and the plant protection types of operations to include operations suiting to his farming practices for different crops and soil types.

All updating options are permitted on this file. The user can change the sequence and content of the list of

operations. This file is utilized to prepare a menu of crops for letting the user to select them for allocating to different fields.

Field file. Most of the information is collected through the pop-up menus. The menu for selecting crops for the field also provides the option of "NoCrop" in case the user decides to grow only one or no crop at all on any field. Information Manager keeps track of the unaccounted cultivated area of the farm and prohibits the user from creating fields with total area more than the cultivated area of the farm.

All updating options are permitted on this file. The user can change the crop and its characteristics, field area and its irrigation status.

Operations file. Number of entries in this file equals the sum total of the items in lists of operations associated with different crops in the crop file. The agronomic work windows (earliest start time and latest finish time) and implement list are collected through pop-up menus. The user can and should select one or more implements for each operation in order to successfully simulate the operation as discussed earlier in this chapter.

The updating options of "add New Item" and "delete an Item" are not allowed on this file. However, the user can modify agronomic work windows and the list of implements associated with an operation.

CHAPTER IV TESTING AND EVALUATION

The model testing and evaluation, in general, can be divided into verification and validation. Verification is a process by which programming logic is compared with the programmer's intentions. On the other hand validation deals with comparing the simulation results with the actual system's behavior. For knowledge-based systems, validation has been termed as "qualification."

The integrated decision support system, named FARMSYS, combine the four components: Operations Simulator, Expert Analysis, Yield Estimation and Information Management Systems discussed in Chapter III. Testing and evaluation of FARMSYS has been carried out in two stages referred as professional qualification and operational verification. The professional qualification involved letting a team of peers evaluate and critique the competence of the system in terms of its knowledge, logic, functioning, user-interface, etc. in order to identify its strengths and weaknesses. The operational verification was aimed to demonstrate the functioning of FARMSYS on a real farm data and also to show how it can be utilized as a planning tool for machinery and labor for multi-crop production systems. FARMSYS is operated using a pull-down menu of Turbo PROLOG Toolbox (Borland, 1987). The

schematic representation of the screen layout and the set of menus along with their final actions is depicted in Figure 4.1. On the selection of an option from the main menu (for example "Information Manager"), an executable file and other related information to that option are loaded to the computer memory. The user can then carry out the intended actions. When terminating the job the command is passed back to FARMSYS and the user can select the next option or can select "Quit" to return to DOS. This structure of FARMSYS has made it possible to manage more than 1200 K of executable files under the DOS environment with less than 640 K available memory.

The time required to operate different components of FARMSYS depends upon the farm size and type of computer utilized. The system required approximately 45 minutes for one simulation run for the test farm (Davis farm) for two consecutive years on 12 Mhz IBM AT 80286 with 6 Mhz math co-processor. The actions by other components such as Information Manager and Expert Analysis System are quick and interactive with little time delays.

Professional Qualification

For the professional level qualification a one-day mini-conference on Knowledge-based Decision Systems using Logic Programming was organized at the Department of Agricultural Engineering of the University of Florida. It was attended by

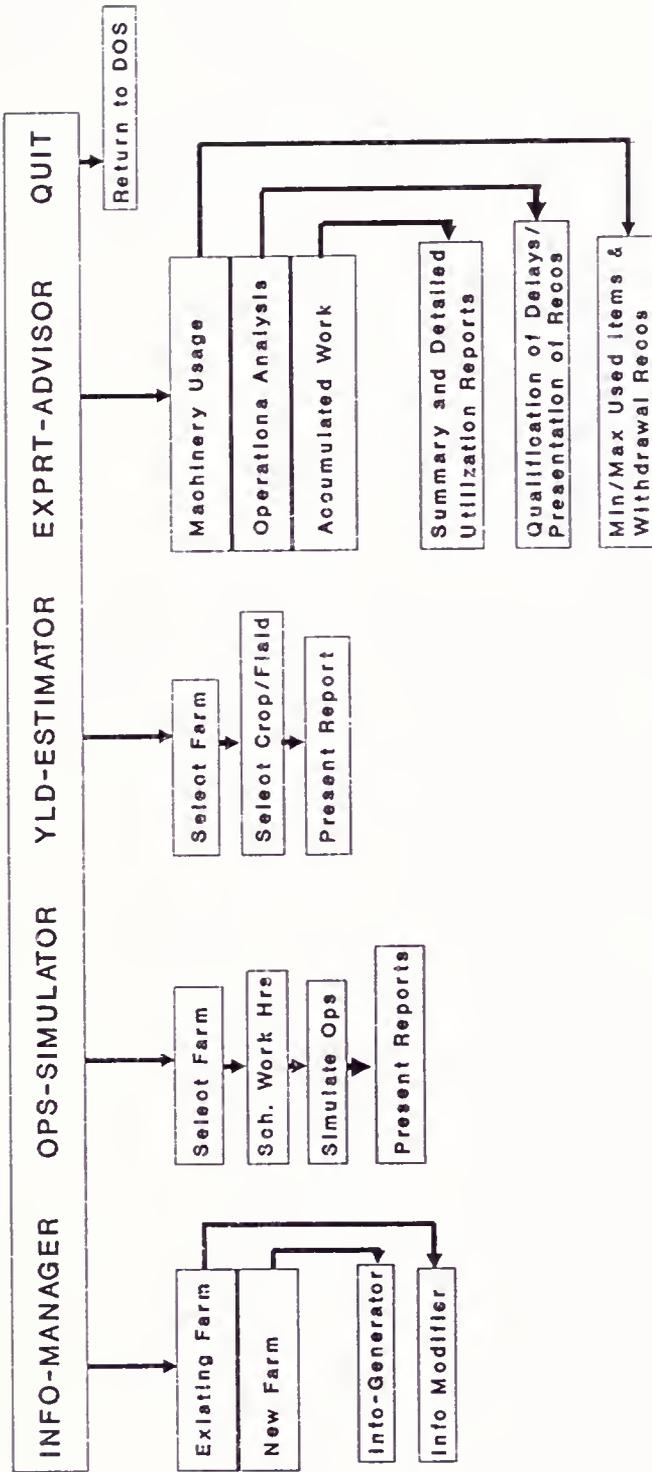


Figure 4.1 Schematic representation of FARM SYS screen layout and its principal menus

10 experts from U.S. and two from other countries (Appendix II), in addition to the local faculty members.

The structure, components and logic of FARMSYS were presented to the participants in detail. The participants witnessed a demonstration of FARMSYS with a real farm example. The eight professionals specially invited for the qualification session were then asked to evaluate FARMSYS and its different components using a specially designed questionnaire (Appendix II) which contained statements concerning various features of FARMSYS. They had 5 options to select for each option ranging from "strongly agree" to "strongly disagree." They were also encouraged to identify FARMSYS weaknesses and strengths in comparison to other similar tools.

The analysis of the questionnaire was carried out in two stages: structured response analysis and non-structured response analysis. In structured response analysis, the answers to different statements were analyzed by grouping and counting the similar responses. Tables 4.1 to 4.8 present these results. The actual comments made by the experts were grouped into four sub-headings: strengths, weaknesses, suggestions and concerns about FARMSYS as non-structured responses.

Table 4.1 Responses to the questionnaire about qualification of the Information Management System of FARMSYS

Responses	Statements/Questions ^a			
	1	2	3	4
Strongly agree	3	0	2	3
Agree	4	4	5	4
Undecided	0	1	1	0
Disagree	0	1	0	1
Strongly disagree	1	1	0	0
No response	0	1	0	0
Total	8	8	8	8

^a. The numbers correspond to the following statements

1. Screen layouts for collecting information about different farm entities are well designed.

2. Mechanism of letting the user develop and maintain knowledge-base specific to his farm is unique for FARMSYS.

3. Mechanism of retrieving, presenting old information, letting user make changes and saving new information are adequately designed.

4. Information Manager is qualified to let user develop new farm and modify information about existing farm.

Table 4.2 Responses to the questions about qualification of Operations Simulation System (Knowledge-bases) of FARMSYS

Responses	Statements/Questions ^a													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Strongly agree	0	2	1	1	1	1	1	2	0	0	1	4	3	6
Agree	3	5	4	5	5	3	2	2	2	0	3	3	3	1
Undecided	3	1	3	2	2	4	1	0	2	5	3	0	1	1
Disagree	1	0	0	0	0	0	2	4	4	3	1	1	1	0
Strongly disagree	0	0	0	0	0	0	1	0	0	0	0	0	0	0
No response	1	0	0	0	0	0	1	0	0	0	0	0	0	0
Total	8	8	8	8	8	8	8	8	8	8	8	8	8	8

^a. The numbers correspond to the following statements

1. The semantic representation of farm knowledge adequately expresses itself for multi-crop production systems.

The components of the following knowledge-bases are adequately structured

Farm Knowledge

- 2) Farm
- 3) Labor
- 4) Tractor
- 5) Implement
- 6) Equipment
- 7) Crops
- 8) Field
- 9) Operations

Regional Knowledge

- 10) Expert File
- 11) Weather File

Output Reports

- 12) Work Reports
- 13) No-Work Reports
- 14) Summary Report

Table 4.3 Responses to the questions about qualification of Operations Simulation System (Simulation Methodology) of FARMSYS

Responses	Statements/Questions ^a									
	1	2	3	4	5	6	7	8	9	10
Strongly agree	3	4	0	0	0	4	6	6	8	0
Agree	2	1	0	3	2	4	2	2	0	4
Undecided	1	0	4	2	4	0	0	0	0	2
Disagree	1	1	0	3	0	0	0	0	0	2
Strongly disagree	0	0	2	0	0	0	0	0	0	0
No response	1	2	2	0	0	0	0	0	0	0
Total	8	8	8	8	8	8	8	8	8	8

^a. The numbers correspond to the following statements

1. The semantic representation of the farm knowledge has facilitated keeping it separate from code of the simulation. It has resulted in a versatile and scale neutral simulator which can be utilized for different types of farm settings.

2. The assigning of more than one operator to a tractor and more than one tractor to an implement is more realistic for farm operations modelling than generally captured in conventional simulators.

3. The checking of conditions for the field operations is carried out more rigorously than most conventional simulator using procedural languages.

4. It has more realistic criteria for deciding actual work hours for the day's work.

5. It uses more robust and realistic mechanism for finding the available machinery set (Implement, Tractor and Operator) than conventional simulators.

6. The facility for adjusting the scheduled work hours for the farm prior to start of simulation is important.

7. The economics of operations is not considered in enough detail.

8. Scheduled work hours are decided for a whole farm rather than for individual operator(s) or tractor(s).

9. The expected future weather while deciding actual work hours is not considered.

10. The Operations Simulator is qualified to simulate field operations for multi-crop production systems.

Table 4.4 Responses to the questions about qualification of Yield Estimation System of FARMSYS

Responses	Statements/Questions ^a				
	1	2	3	4	5
Strongly agree	0	2	4	2	1
Agree	3	2	3	5	0
Undecided	1	0	0	1	3
Disagree	3	2	1	0	2
Strongly disagree	1	2	0	0	2
No response	0	0	0	0	0
Total	8	8	8	8	8

^a. The numbers correspond to the following statements

1. It is good to have yld-estimator as a component of the FARMSYS. However, its acceptance as a machinery management tool would not be affected significantly without it.

2. The use of crop simulation models for predicting crop yields for the farm level is not a good idea because they are not designed for this purpose.

The following features of the Yield Estimator are adequately structured:

3. Screen layout for letting the user select field(s)/crop(s).

4. Structure and contents of the final report.

5. The Yield Estimator is qualified to make yield and net return predictions if appropriate crop model(s) are used.

Table 4.5 Responses to the questions about qualification of Expert Analysis System (Machinery Usage Report) of FARMSYS

Responses	Statements/Questions ^a		
	1	2	3
Strongly agree	3	1	0
Agree	4	3	4
Undecided	1	1	1
Disagree	0	3	3
Strongly disagree	0	0	0
No response	0	0	0
Total	8	8	8

^a. The numbers correspond to the following statements

1. The objective of the report is sufficiently strong and impressive and users will make intensive use of this report.
2. The criteria utilized for identifying the least utilized item are adequate and realistic.
3. The criteria utilized for deciding recommendation about least utilized item are adequate and realistic.

Table 4.6 Responses to the questions about qualification of Expert Analysis System (Operations Analysis Report) of FARMSYS

Responses	Statements/Questions ^a					
	1	2	3	4	5	6
Strongly agree	1	1	1	4	2	3
Agree	4	3	1	3	3	2
Undecided	2	2	1	0	0	1
Disagree	1	2	3	1	3	2
Strongly disagree	0	0	2	0	0	0
No response	0	0	0	0	0	0
Total	8	8	8	8	8	8

^a. The numbers correspond to the following statements

1. The objective of the report is sufficiently strong and impressive and the user will make intensive use of this report.

2. The criteria utilized for identifying problematic operations is not adequate and needs improvement.

3. The break-up of the timing window is ok.

The criteria utilized to make the following recommendations are adequate and realistic.

4. Increase Scheduled Work hours

5. Increase Speed of operation

6. Increase Working width

Table 4.7 Responses to the questions about qualification of Expert Analysis System (Accumulated Work Report) of FARMSYS

Responses	Statements/Questions ^a			
	1	2	3	4
Strongly agree	1	2	7	3
Agree	6	4	1	4
Undecided	1	2	0	1
Disagree	0	0	0	0
Strongly disagree	0	0	0	0
No response	0	0	0	0
Total	8	8	8	8

^a. The numbers correspond to the following statements

1. The objective of the report is sufficiently strong and impressive and users will make intensive use of this report.

2. The interactive mode for letting users select different farm items and time interval is adequately designed.

The results of this report are presented in the form of two sub-reports; namely, Summary Reports and Detailed Report. Please evaluate these reports.

3. Both reports are essential.

4. Both reports are logically presented.

Table 4.8 Responses to the questions about overall qualification of FARMSYS

Responses	Statements/Questions ^a				
	1	2	3	4	5
Strongly agree	3	0	1	1	0
Agree	4	4	0	2	4
Undecided	0	2	3	4	2
Disagree	1	2	2	1	1
Strongly disagree	0	0	2	0	0
No response	0	0	0	0	1
Total	8	8	8	8	8

^a. The numbers correspond to the following statements

1. Information Manager is qualified.
2. Operations Simulator is qualified.
3. Yield Estimator is qualified.
4. Expert Advisor is qualified.
5. FARMSYS is qualified.

Structured Response Analysis

The analysis of the questionnaire suggested that all participants appreciated the general structure of FARMSYS and approach taken for its development. They had concern about the relevance of the some of the heuristics utilized in different components of FARMSYS for their global applicability.

In general, all participants found the Information Management System qualified for receiving farm information in an interactive and friendly manner (Table 4.1). This was considered one of the strength of FARMSYS.

There was some concern about the structuring and contents of different components of farm knowledge-bases (Table 4.2 and 4.3). However, mostly all participants found the contents of the reports produced by the simulation essential and adequately structured. They generally agreed that the facility of representing input knowledge using semantic nets has been able to capture the farm knowledge in a more realistic manner than conventional simulators and it has also made FARMSYS a versatile and flexible tool. However, they remained undecided or expressed their disagreement on the claim that FARMSYS checks conditions during simulation more rigorously than conventional simulators.

The screen layout for letting the user select field(s) and crop(s), and the structure and contents of the reports

produced by the Yield Estimation System received favorable rating (Table 4.4). However, they were undecided or had disagreement on its qualification to predict the farm yield.

The three mini-expert systems (Machinery Usage, Operations Analysis and Accumulated Work Reports) of the Expert Analysis System were considered as important components of FARMSYS (Tables 4.5-4.7). In general, they were considered qualified for their intended functions. However, some experts had concern about the applicability of built-in heuristics in these reports, specially the 20% criteria of Machinery Usage Report and the one-third rule for deciding "late" starts and finishes in the Operations Analysis Report. These comments led to further revisions in these reports which allow the user to specify the rules for late starts and finishes individually for each operation.

Table 4.8 presents the overall evaluation of FARMSYS and its different components. This table indicates that 50% of the participants considered FARMSYS qualified as a decision-aid tool and another 25% were undecided. The Operations Simulator and the Information Management System are its (FARMSYS) strong components. The table also indicates that 50% of the participants were undecided about the qualification of the Expert Analysis System. It was mainly because of the built-in heuristics for characterizing delays in the operations for the Operation Analysis report and the 20% criteria for the Machinery Usage report.

Insufficient time for a thorough presentation of FARMSYS and its different components at the one-day meeting probably accounts for some of the negative responses. However, some of these limitations have been already eliminated in the second version of the Operations Analysis report as discussed in Chapter III. The possible ways to implement some others are discussed in the plans for future work in the next chapter.

Non-Structured Responses

The non-structured responses mainly dealt with identifying strengths, weaknesses, suggestions and concerns about various components of FARMSYS.

Some of the strengths of FARMSYS, identified by the experts (Appendix II), are its comprehensive nature in machinery selection process for working in a field, its framework and structured approach, its ease of setting up and the flexibility of the produced reports. All these strengths can be mainly credited to the object-oriented and logic programming approach taken in its development.

The experts also made some suggestions for improvements which are listed in Appendix II. These suggestions included incorporation of location specific rules for various reports, detailed economics for various operations, the possibility of

including custom hiring of tractors and other machinery at peak periods and a facility to put more than one machine in the same field for the same operation.

Role of Mini-Conference in Evaluating Knowledge-based Systems

There is a general concern about the methodology for validating or qualifying knowledge-based systems. There are no standard procedures available for this purpose. Different authors (Khuri et al., 1988; Nguyen et al., 1987) have tried different approaches for validating or qualifying knowledge systems. The experiences with FARMSYS indicate that the methodology of a small group of experts meeting together to evaluate and critique the system offers a good and practical alternative for validation.

However, it is recommended that such a meeting should occur as two half-day sessions instead of one full day as in the case of FARMSYS. The first half-day session, preferably during an afternoon should be devoted to explaining and demonstrating the system. During the latter part of the afternoon or that evening, the participants should be encouraged to use the system. The second half-day session, preferably during the morning hours of the next day, should be devoted to the experts filling out questionnaires or other related material regarding the qualification of the system.

Operational Level Verification

For the operational level verification, contacts were established with three farmers from north Florida through the regional extension agents. Visits were made to their farms and the knowledge about one of them (Davis Farm) was collected using the Information Management System. The Davis farm knowledge (Appendix I) was then utilized as an input to FARMSYS to test its functionality as a decision-aid tool.

Additional input knowledge for operating FARMSYS was collected from either the published literature or was based upon estimates of experienced agricultural engineers. The actual weather data of Tallahassee, FL, for the years 1954 (Dry Year), 1960 (Normal Year) and 1964 (Wet Year) were selected to make the test runs.

The test runs are not intended to analyze any particular problem associated with the Davis farm. They are rather intended to show that FARMSYS actually performs as designed, how its different components can be utilized in an integrated form and types of problems it can address for a typical farm setting. Different components of FARMSYS work independently and can be utilized in any sequence the user desires. However, the test runs presented in this chapter were made utilizing different components of FARMSYS as depicted in Figure 4.2.

Description of the Test Farm

The test farm consists of about 400 hectares of cultivated area allocated to 23 fields. It has 4 laborers, 5 tractors and 25 implements. It grows 4 crops (cotton, peanuts, soybeans and wheat) and has 10 types of cropping systems as listed in Table 4.9 followed on different fields. Each of these crops and cropping systems require a different number of operations, ranging from 6 for wheat to 16 for cotton, during their production cycles. Each operation has its specific requirements in terms of its agronomic work window and implement list. Each implement has its specific characteristics in terms of tractor or labor requirements and its working width and speed of operation.

Setting up Farm Knowledge for Simulation Runs

All the crops, except wheat, are planted and harvested during the same calendar year. Wheat is planted in the second half of a year and harvested early in the following year. This requires the program to run for two consecutive years in order to simulate the complete cropping season of the farm. It also requires two years of weather input which were selected "Dry" after "Dry" year.

The crops of the second year were suffixed by character "2" and the agronomic work windows of different operations for the second year were adjusted by adding 365 to their values of the first year.

Table 4.9 Cultivated area and cropping systems of different fields on the Davis farm.

Field	Area (ha)	Cropping System
Field_1	102.2	Cotton-Cotton
Field_2	76.8	Cotton-Wheat
Field_3	6.0	Cotton-Soybeans
Field_4	41.6	Cotton-Peanut
Field_5	79.6	Soybeans-Cotton
Field_6	16.4	Peanut-Wheat
Field_7	10.0	Peanut-Cotton
Field_8	11.8	Soybeans-NoCrop
Field_8	8.0	NoCrop-Soybeans
Field_9	26.0	NoCrop-Cotton
Field_10	26.4	NoCrop-NoCrop
Total	404.8	

It is possible to simulate the test farm with its 23 fields. However, for the sake of simplicity of reporting the results, the test runs reported in this chapter were made by grouping all the fields with same cropping system into one single field. This led to a total of 10 fields (Table 4.9) with the total cultivated area of the farm unchanged.

The scheduled work hours is another factor which can be adjusted prior to the start of each simulation. It is done using the screen layout depicted in Figure 4.3.

The initial test runs were made with six hours of daily scheduled work for all months of the year. This resulted in zero "done area" for many operations on different fields. Then eight hours of scheduled work per day for the entire year also resulted in unfinished work for certain operations. Finally, it was possible to complete all operations on all fields within their operational windows utilizing the daily scheduled work hours as presented in Table 4.10.

These initial runs of simulation with FARMSYS showed that the Operations Simulation functioned properly, and they also showed the typical variation in requirements of work hours during different times of the cropping season. The results from the last run, referred as a "success run," are discussed in detail.

Table 4.10 Scheduled work hours for different months for simulating field operations on Davis farm

Month	Scheduled Work Hours
January	8
February	8
March	8
April	12
May	12
June	12
July	8
August	8
September	12
October	8
November	8
December	8

Operations Simulation Reports

The simulator produces three reports on field operations namely work report, no-work report and summary report.

The examples of work and no-work reports for the success run produced by the simulator are presented in Figure 4.4. These examples indicate that on Julian day 100, the simulator tried to carry out five tasks on different fields. It successfully completed three of them and failed to carry out the other two.

The tasks successfully completed (Work report) were 1) Plowing 14.2272 hectares on "Field_1" for "Cotton" with total cultivated area of 102.2 hectares using "BottomPlow," "Tractor_1" and "Jerry" by working 12 hours; 2) Harrowing 30.26 hectares on "Field_2" for "Cotton" using "Disk Harrow," "Tractor_1" and "Keith" by working 12 hours and 3) Fertilize 27.907 hectares on "Field_4" for "Cotton" using "FeriSpreaderLP," "Tractor_4" and "Joe" by working 12 hours.

The tasks to Fertilize on "Field_5" and "Field_8" of "Soybeans" were tried but could not be done because the machinery needed was not available ("MachinesNotAvail"). It required "FertiSpreaderLP" which was not available ("NotAvail"). It was being utilized for applying fertilization on "Field_4."

The complete summary report of the success run is presented in Table 4.11. The table presents the scheduled work window (ScSt to ScFn), actual work window (AcSt to AcFn),

Work Report**General format**

```
workreport(julianDay, Month, Field, Crop,
  Operation, CultArea, Day's DoneArea, TotDoneArea,
  Implement, Operator, Tractor, ActWorkHr)
```

For the Julian Day 100

```
workreport(100, "April", "Field_1", "Cotton",
  "Plowing", 102.2, 14.2272, 14.2272, "BottomPlow",
  "Jerry", "Tractor_1", 12)
```

```
workreport(100, "April", "Field_2", "Cotton",
  "Harrowing", 76.8, 30.26016, 30.26016,
  "DiskHarrow", "Keith", "Tractor_2", 12)
```

```
workreport(100, "April", "Field_4", "Cotton",
  "Fertilize", 41.6, 27.9072, 27.9072,
  "FertiSpreaderLP", "Joe", "Tractor_4", 12)
```

No Work Report**General format**

```
noworkreport(julianDay, Month, Field, DoneArea,
  Crop, Operation, DoneArea, ReasonOfNoWork,
  ImplementList, AvailStatus, WhereAboutImple-
  mentList, TractorList, AvailStatus, WhereAbout-
  TractorList, OperatorList, AvailStatus,
  whereAboutOperatorList)
```

For the Julian Day 100

```
noworkreport(100, "April", "Field_5",
  79.6, "Soybeans", "Fertilize", 0,
  "MachinesNotAvail",
  ["FertiSpreaderLP"], "NotAvail",
  [{"FertiSpreaderLP", "field4", "Cotton",
  "Fertilize"}],
  ["Tractor_4", "Tractor_5"], "Avail", [],
  ["Keith", "Ray", "Joe", "Jerry"], "Avail", [])
```

```
noworkreport(100, "April", "Field_8",
  11.8, "Soybeans", "Fertilize", 0,
  "MachinesNotAvail",
  ["FertiSpreaderLP"], "NotAvail",
  [{"FertiSpreaderLP", "field4", "Cotton",
  "Fertilize"}],
  ["Tractor_4", "Tractor_5"], "Avail", [],
  ["Keith", "Ray", "Joe", "Jerry"], "Avail", [])
```

Figure 4.4 Examples of work and no-work reports of the Operations Simulation of FARMSYS.

Table 4.11 Summary report of the "success run" of the Operations Simulation for Davis farm (Appendix I) for the years "Dry" followed by "Dry" (1954).

Operation	ScSt ----Julian	ScFn Day	AcSt -----	AcFn -----	Wrk Dys	Wrk Hrs	WArea (ha)	NoWrk R	Dys M

Field_1 (102.2 ha), First Year (Cotton)									
Fertilize	92	136	92	95	4	48.0	102.2	0	0
Harrowing	92	136	96	99	4	48.0	102.2	0	0
Plowing	92	151	100	108	8	90.0	102.2	1	0
Planting	106	161	109	112	4	48.0	102.2	0	0
Spraying1	106	167	113	114	2	24.0	102.2	0	0
Cultivation1	116	172	116	120	5	60.0	102.2	0	0
Spraying2	122	182	122	123	2	18.0	102.2	0	0
Cultivation2	126	189	126	130	5	60.0	102.2	0	0
Spraying3	131	212	131	131	1	12.0	102.2	0	0
Cultivation3	167	212	167	172	6	60.0	102.2	0	0
FertiSpreading	167	212	173	181	5	60.0	102.2	4	0
Spraying4	167	223	182	183	2	20.0	102.2	0	0
Spraying5	172	243	184	184	1	8.0	102.2	0	0
Spraying6	183	244	186	186	1	8.0	102.2	1	0
Harvesting	259	350	262	275	13	152.0	102.2	1	0
Mowing	289	381	289	294	6	48.0	102.2	0	0

Field_1 (102.2 ha), Second Year (Cotton)									
Fertilize	458	502	458	461	4	48.0	102.2	0	0
Harrowing	458	502	462	465	4	48.0	102.2	0	0
Plowing	458	507	466	474	8	90.0	102.2	1	0
Planting	472	533	475	478	4	48.0	102.2	0	0
Spraying1	473	533	479	480	2	18.0	102.2	0	0
Cultivation1	473	538	481	485	5	60.0	102.2	0	0
Spraying2	477	548	486	487	2	24.0	102.2	0	0
Cultivation2	492	553	492	496	5	60.0	102.2	0	0
Spraying3	497	578	498	499	2	18.0	102.2	1	0
Cultivation3	533	578	533	537	5	54.0	102.2	0	0
FertiSpreading	533	580	538	546	5	60.0	102.2	4	0
Spraying4	533	578	547	547	1	12.0	102.2	0	0
Spraying5	538	609	548	548	1	8.0	102.2	0	0
Spraying6	548	610	549	549	1	8.0	102.2	0	0
Harvesting	625	716	628	642	14	156.0	102.2	1	0
Mowing	655	730	655	660	6	48.0	102.2	0	0

ScSt-	Earliest Start Time			ScFn-	Latest Finish Time				
AcSt-	Actual Start Time			AcFn-	Actual Finish Time				
WrkDys-	No. of Days Worked			WrkHrs-	No. of Hours Worked				
WArea	-Worked Area			NoWrkDys-	No. of No-Work Days				
				R-Rain,	M-Machinery				

Table 4.11--Continued

Operation	ScSt ----Julian	ScFn Day	AcSt Day	AcFn Day	Wrk Dys	Wrk Hrs	WArea (ha)	NoWrk R	Dys M
Field_2 (76.8 ha), First Year (Cotton)									
Fertilize	92	136	96	98	3	36.0	76.8	0	0
Harrowing	92	136	100	102	3	36.0	76.8	0	0
Plowing	92	151	110	118	6	72.0	76.8	0	0
Planting	106	161	121	124	4	42.0	76.8	0	0
Spraying1	106	167	125	126	2	24.0	76.8	0	0
Cultivation1	116	172	127	130	4	48.0	76.8	0	0
Spraying2	122	182	131	133	2	18.0	76.8	1	0
Cultivation2	126	189	134	137	4	48.0	76.8	0	0
Spraying3	131	212	138	138	1	12.0	76.8	0	0
Cultivation3	167	212	173	180	4	48.0	76.8	4	0
FertiSpreading	167	212	182	187	5	44.0	76.8	1	0
Spraying4	167	223	188	189	2	16.0	76.8	0	0
Spraying5	172	243	190	191	2	12.0	76.8	0	0
Spraying6	183	244	192	192	1	8.0	76.8	0	0
Harvesting	259	350	276	291	15	120.0	76.8	1	0
Mowing	289	381	295	299	5	40.0	76.8	0	0
Field_2 (76.8 ha), Second Year (Wheat)									
Fertilize	289	350	299	305	5	40.0	76.8	2	0
Harrowing	289	350	306	310	5	36.0	76.8	0	0
Plowing	289	350	311	316	6	48.0	76.8	0	0
Planting	320	366	320	325	6	44.0	76.8	0	0
FertiSpreading	396	425	396	401	6	48.0	76.8	0	0
Harvesting	502	528	490	499	4	42.0	76.8	1	0
Field_3 (6 ha), First Year (Cotton)									
Fertilize	92	136	99	99	1	12.0	6.0	0	0
Harrowing	92	136	104	104	1	12.0	6.0	1	0
Plowing	92	151	119	119	1	12.0	6.0	0	0
Planting	106	161	125	125	1	12.0	6.0	0	0
Spraying1	106	167	134	134	1	12.0	6.0	0	0
Cultivation1	116	172	135	135	1	12.0	6.0	0	0
Spraying2	122	182	136	136	1	12.0	6.0	0	0
Cultivation2	126	189	137	137	1	12.0	6.0	0	0
Spraying3	131	212	138	138	1	12.0	6.0	0	0
Cultivation3	167	212	181	181	1	12.0	6.0	0	0
FertiSpreading	167	212	188	188	1	8.0	6.0	0	0
Spraying4	167	223	190	190	1	4.0	6.0	0	0
Spraying5	172	243	193	193	1	8.0	6.0	0	0
Spraying6	183	244	194	194	1	8.0	6.0	0	0
Harvesting	259	350	292	293	2	16.0	6.0	0	0
Mowing	289	381	300	300	1	8.0	6.0	0	0

Table 4.11--Continued

Operation	ScSt	ScFn	AcSt	AcFn	Wrk	Wrk	WArea	NoWrkDys	
	----Julian Day----				Dys	Hrs	(ha)	R	M
Field_3 (6 ha), Second Year (Soybeans2)									
Fertilize	458	502	462	462	1	12.0	6.0	0	0
Harrowing	458	502	466	466	1	12.0	6.0	0	0
Bedding	472	502	472	472	1	12.0	6.0	0	0
Planting	502	548	502	502	1	12.0	6.0	0	0
Cultivation1	502	563	503	503	1	12.0	6.0	0	0
Cultivation2	520	578	520	520	1	12.0	6.0	0	0
Cultivation3	549	589	549	549	1	8.0	6.0	0	0
Spraying1	563	594	563	563	1	8.0	6.0	0	0
Spraying2	580	640	580	580	1	8.0	6.0	0	0
Harvesting	650	681	637	637	1	12.0	6.0	0	0
Field_4 (41.6 ha), First Year (Cotton)									
Fertilize	92	136	100	101	2	24.0	41.6	0	0
Harrowing	92	136	105	106	2	18.0	41.6	0	0
Plowing	92	151	120	124	3	36.0	41.6	0	0
Planting	106	161	131	138	3	30.0	41.6	1	0
Spraying1	106	167	140	140	1	12.0	41.6	1	0
Cultivation1	116	172	141	142	2	24.0	41.6	0	0
Spraying2	122	182	143	143	1	12.0	41.6	0	0
Cultivation2	126	189	144	145	2	24.0	41.6	0	0
Spraying3	131	212	146	146	1	12.0	41.6	0	0
Cultivation3	167	212	182	184	3	28.0	41.6	0	0
FertiSpreading	167	212	189	192	4	28.0	41.6	0	0
Spraying4	167	223	193	193	1	8.0	41.6	0	0
Spraying5	172	243	195	195	1	8.0	41.6	0	0
Spraying6	183	244	196	196	1	8.0	41.6	0	0
Harvesting	259	350	294	303	8	64.0	41.6	2	0
Mowing	289	381	304	306	3	24.0	41.6	0	0
Field_4 (41.6 ha), Second Year (Peanut2)									
Fertilize	427	456	427	429	3	24.0	41.6	0	0
Harrowing	427	456	430	432	3	20.0	41.6	0	0
Plowing	427	456	433	437	5	40.0	41.6	0	0
Planting	472	487	472	473	2	24.0	41.6	0	0
Cultivation1	488	504	488	490	3	30.0	41.6	0	0
Spraying1	488	502	491	491	1	12.0	41.6	0	0
Cultivation2	512	563	512	515	2	24.0	41.6	2	0
Spraying2	533	563	533	534	2	18.0	41.6	0	0
Spraying3	540	578	547	547	1	12.0	41.6	0	0
Spraying4	549	594	549	549	1	8.0	41.6	0	0
Spraying5	562	609	562	562	1	8.0	41.6	0	0
Harvesting	611	640	613	621	9	96.0	41.6	0	0

Table 4.11--Continued

Operation	ScSt ----Julian	ScFn Day	AcSt -----	AcFn -----	Wrk Dys	Wrk Hrs	WArea (ha)	NoWrk R	Dys M
Field_5 (79.6 ha), First Year (Soybeans)									
Fertilize	92	136	102	105	3	36.0	79.6	1	1
Harrowing	92	136	107	110	3	36.0	79.6	0	0
Bedding	106	136	111	116	3	36.0	79.6	0	0
Planting	136	182	140	147	4	48.0	79.6	1	0
Cultivation1	145	197	150	153	4	48.0	79.6	2	0
Cultivation2	166	212	166	170	5	48.0	79.6	0	0
Cultivation3	183	228	186	191	6	44.0	79.6	1	2
Spraying1	197	228	197	197	1	8.0	79.6	0	0
Spraying2	214	274	214	214	1	8.0	79.6	0	0
Harvesting	284	315	282	287	6	48.0	79.6	0	0
Field_5 (79.6 ha), Second Year (Cotton2)									
Fertilize	458	502	463	465	3	36.0	79.6	0	0
Harrowing	458	502	467	470	3	36.0	79.6	1	0
Plowing	458	507	475	482	6	72.0	79.6	0	0
Planting	472	533	486	498	4	42.0	79.6	1	0
Spraying1	473	533	500	501	2	24.0	79.6	0	0
Cultivation1	473	538	502	506	4	48.0	79.6	1	0
Spraying2	477	548	507	508	2	24.0	79.6	0	0
Cultivation2	492	553	509	512	4	48.0	79.6	0	0
Spraying3	497	578	515	516	2	24.0	79.6	2	0
Cultivation3	533	578	538	545	4	48.0	79.6	4	0
FertiSpreading	533	580	547	552	5	44.0	79.6	1	0
Spraying4	533	578	553	553	1	8.0	79.6	0	0
Spraying5	538	609	554	554	1	8.0	79.6	0	0
Spraying6	548	610	555	556	2	12.0	79.6	0	0
Harvesting	625	716	648	662	15	120.0	79.6	0	0
Mowing	655	730	663	669	5	40.0	79.6	2	0
Field_6 (16.4 ha), First Year (Peanut)									
Fertilize	61	90	61	61	1	8.0	16.4	0	0
Harrowing	61	90	62	62	1	8.0	16.4	0	0
Plowing	61	90	63	64	2	16.0	16.4	0	0
Planting	106	121	106	107	2	18.0	16.4	0	0
Cultivation1	106	135	108	108	1	12.0	16.4	0	0
Spraying1	106	131	109	109	1	12.0	16.4	0	0
Cultivation2	151	197	151	151	1	12.0	16.4	0	0
Spraying2	167	197	171	171	1	12.0	16.4	0	0
Spraying3	172	212	172	172	1	12.0	16.4	0	0
Spraying4	183	223	184	184	1	8.0	16.4	0	0
Spraying5	192	243	192	192	1	8.0	16.4	0	0
Harvesting	245	274	247	250	4	42.0	16.4	0	0

Table 4.11--Continued

Operation	ScSt ----Julian	ScFn Day	AcSt -----	AcFn -----	Wrk Dys	Wrk Hrs	WArea (ha)	NoWrk R	Dys M
Field_6 (16.4 ha), Second Year (wheat)									
Fertilize	289	350	289	289	1	8.0	16.4	0	0
Harrowing	289	350	311	311	1	8.0	16.4	0	0
Plowing	289	350	317	318	2	16.0	16.4	0	0
Planting	320	366	326	327	2	12.0	16.4	0	0
FertiSpreading	396	425	402	403	2	16.0	16.4	0	0
Harvesting	502	528	500	500	1	12.0	16.4	0	0
Field_7 (10 ha), First Year (Peanut)									
Fertilize	61	90	62	62	1	8.0	10.0	0	0
Harrowing	61	90	63	63	1	8.0	10.0	0	0
Plowing	61	90	65	66	2	12.0	10.0	0	0
Planting	106	121	113	113	1	12.0	10.0	0	4
Cultivation1	106	135	114	114	1	12.0	10.0	0	0
Spraying1	106	131	115	115	1	6.0	10.0	0	0
Cultivation2	151	197	151	151	1	12.0	10.0	0	0
Spraying2	167	197	173	173	1	12.0	10.0	0	0
Spraying3	172	212	177	177	1	12.0	10.0	3	0
Spraying4	183	223	186	186	1	8.0	10.0	1	0
Spraying5	192	243	194	194	1	8.0	10.0	0	0
Harvesting	245	274	253	255	3	30.0	10.0	0	0
Field_7 (10 ha), Second Year (Cotton2)									
Fertilize	458	502	466	466	1	12.0	10.0	0	0
Harrowing	458	502	471	471	1	6.0	10.0	0	0
Plowing	458	507	483	483	1	12.0	10.0	0	0
Planting	472	533	499	499	1	12.0	10.0	0	0
Spraying1	473	533	505	505	1	12.0	10.0	1	0
Cultivation1	473	538	506	506	1	12.0	10.0	0	0
Spraying2	477	548	509	509	1	12.0	10.0	0	0
Cultivation2	492	553	510	510	1	12.0	10.0	0	0
Spraying3	497	578	511	511	1	12.0	10.0	0	0
Cultivation3	533	578	546	546	1	12.0	10.0	0	0
FertiSpreading	533	580	553	553	1	8.0	10.0	0	0
Spraying4	533	578	557	557	1	8.0	10.0	0	0
Spraying5	538	609	558	558	1	8.0	10.0	0	0
Spraying6	548	610	559	559	1	8.0	10.0	0	0
Harvesting	625	716	663	664	2	16.0	10.0	0	0
Mowing	655	730	670	670	1	8.0	10.0	0	0

Table 4.11--Continued

Operation	ScSt ----Julian	ScFn Day	AcSt Day	AcFn Day	Wrk Dys	Wrk Hrs	WArea (ha)	NoWrk R	Dys M
Field 8 (11.8 ha), First Year (Soybeans)									
Fertilize	92	136	106	106	1	6.0	11.8	0	10
Harrowing	92	136	117	117	1	12.0	11.8	0	0
Bedding	106	136	118	118	1	12.0	11.8	0	0
Planting	136	182	154	154	1	12.0	11.8	0	0
Cultivation1	145	197	155	155	1	12.0	11.8	0	0
Cultivation2	166	212	166	166	1	12.0	11.8	0	0
Cultivation3	183	228	192	192	1	8.0	11.8	0	2
Spraying1	197	228	198	198	1	8.0	11.8	0	0
Spraying2	214	274	215	215	1	8.0	11.8	0	0
Harvesting	284	315	289	289	1	8.0	11.8	0	0
Field 9 (8 ha), Second Year (Soybeans)									
Fertilize	458	502	467	467	1	12.0	8.0	0	0
Harrowing	458	502	473	473	1	12.0	8.0	0	0
Bedding	472	502	474	474	1	12.0	8.0	0	0
Planting	502	548	507	507	1	12.0	8.0	0	0
Cultivation1	502	563	508	508	1	12.0	8.0	0	0
Cultivation2	520	578	520	520	1	12.0	8.0	0	0
Cultivation3	549	589	551	551	1	8.0	8.0	1	0
Spraying1	563	594	565	565	1	8.0	8.0	1	0
Spraying2	580	640	581	581	1	8.0	8.0	0	0
Harvesting	650	681	642	642	1	8.0	8.0	0	0
Field 10 (26.4 ha), Second Year (Cotton)									
Fertilize	458	502	469	469	1	12.0	26.4	1	0
Harrowing	458	502	475	475	1	12.0	26.4	0	0
Plowing	458	507	484	485	2	24.0	26.4	0	0
Planting	472	533	500	501	2	24.0	26.4	0	0
Spraying1	473	533	517	517	1	12.0	26.4	0	0
Cultivation1	473	538	518	519	2	24.0	26.4	0	0
Spraying2	477	548	521	521	1	12.0	26.4	0	0
Cultivation2	492	553	522	523	2	24.0	26.4	0	0
Spraying3	497	578	524	524	1	12.0	26.4	0	0
Cultivation3	533	578	547	548	2	20.0	26.4	0	0
FertiSpreading	533	580	554	556	3	20.0	26.4	0	0
Spraying4	533	578	560	560	1	8.0	26.4	0	0
Spraying5	538	609	561	561	1	8.0	26.4	0	0
Spraying6	548	610	567	567	1	8.0	26.4	1	0
Harvesting	625	716	665	671	5	40.0	26.4	2	0
Mowing	655	730	672	674	3	20.0	26.4	0	0

total hours, days and worked area (Wrk Dys, Wrk Hrs and WArea) for each operation. In addition, the number of no-work days (NoWrkDys) due to rain (R) and due to non-availability of the machinery (M) are also shown in the table. The table indicates that the simulator was able to successfully complete its task for the entire cropping season which covered the period of day 61 (March 1 of the first year) to day 730 (December 31 of the Second Year).

All the operations for different crops were successfully completed on all the fields within their agronomic work windows. They were carried out in proper sequence, as planned. However, some of the later spraying operations (Spraying4, Spraying5, Spraying6) in cotton fields were carried out one after another with no time gaps between them. This occurred because they were planned to be carried out one after another, and the machinery and labor needed to carry them out were freely available with no other operations specified between the two sprayings. However, it can be avoided, if needed, by attaching some additional attributes to the operation object-class, by breaking up one big field into small fields and/or by defining the agronomic windows for subsequent sprayings with some time gaps between them.

Expert Analysis System

The reports produced by the simulation and discussed above do not provide any indication about the percentage

utilization of different machinery and laborers working on the farm, and/or the relative timeliness of different operations. This task is performed by the Expert Analysis System. It studies these reports in the context of the resource-base available on the farm. The following sections demonstrate the functioning of the three components of the Expert Analyzer and how the information generated by them can be utilized by the user to improve his farming operation. The three components are Machinery Usage, Operations Analysis and Accumulated Work reports.

Machinery usage report

This report identifies the least and the most utilized items of different farm object-classes, prepares their seasonal and monthly utilization and makes recommendations to keep them or to withdraw them from the farm, based upon their utilization level and their importance to the current farming operations, as discussed in chapter III.

During the initial test runs the Machinery Usage Report identified "GeneralPlanter," "PDSprayer" and "PDSprayer2" as not being used at all for any operation and recommended their withdrawal. They were withdrawn with no effect on the scheduling of different operations. On the Davis farm, they were probably older machines kept as backup.

Table 4.12 presents a summary of the Machinery Usage Reports of the test run after withdrawal of the implements not

Table 4.12 Summary of reports and recommendations by Machinery Usage Report of Expert Result Analysis System for the "success run" for Davis farm knowledge.

Object		Utilization		Recommendation and Reasoning
Class	Item	Hrs	Days	
Operator	Jerry	2820	286	Withdraw Ray. He has a complementary unit(s).
	Ray	142	16	
Tractor	Tractor_3	1432	131	Withdraw Tractor_5. It has complementary unit(s).
	Tractor_5	120	13	
Implement LandPrep	Bottom Plow	476	44	Keep Sub Soiler. It does not have any complementary unit(s).
	Sub Soiler	64	6	
Plant/Ferti	Planter_77	396	38	Keep Grain Drill. It does not have any complementary unit(s).
	Grain Drill	56	8	
Plant-Prot.	Sprayer	552	53	Withdraw PCultivator. It has complementary unit(s).
	PCultivator	24	2	
Harvesting	Cotton-Picker	684	72	Do not withdraw General Grain Combine. It does not have any complementary unit(s).
	General-Grain-Combine			
	Combine	130	14	

used at all and identified during initial runs. Table 4.13 presents their monthly usage for the first year of success run of the simulation.

The laborer "Ray" is the least utilized laborer on the farm. He works less than 20% of the work hours of "Jerry" (the most utilized laborer on the farm). In addition, he (Ray) has one or more complementary operator(s) that could carry out the tasks that were assigned to him. So the recommendation was made to withdraw him from the farm work force. In actual practice on the Davis farm, he mostly performs other non-crop operations.

The usage reports on usage of tractors and plant protection implements also produced similar recommendations for "Tractor_5" and "PCultivator" based upon their utilization level in comparison to the most utilized items ("Tractor_3" for tractors and "Sprayer" for plant protection implements) of their respective object-class.

The recommendations about Sub-Soiler, Grain Drill and General Grain Combine are different. Although their utilization was less than 20% of the maximum utilized item of their object-class, still the recommendations were not to withdraw them from the farm as they do not have complementary unit(s) to carry out tasks currently assigned to them.

The recommendations made by Machinery Usage Report to withdraw any object-item should be implemented with a little caution because of the following reasons. However, the user

Table 4.13 Monthly variation of most and least utilized items of different object-classes on Davis farm during first year of simulation of the "success run."

Item	Monthly Utilization (Hrs)											
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Operators												
Jerry	0	0	44	336	228	144	116	16	216	224	100	0
Ray	0	0	0	6	12	36	60	0	0	8	0	0
Tractors												
Tractor_3	0	0	0	174	312	180	68	0	0	0	56	0
Tractor_5	0	0	0	0	12	36	52	0	0	0	0	0
Land Preparation Implements												
BottomPlow	0	0	28	198	12	0	0	0	0	0	0	0
SubSoiler	0	0	0	0	0	0	0	0	0	64	0	0
Plant and Ferti Implements												
Planter_77	0	0	0	90	120	12	0	0	0	0	0	0
Grain Drill	0	0	0	0	0	0	0	0	0	56	0	0
Plant Protection Implements												
Sprayer	0	0	0	42	120	60	68	0	0	0	0	0
PCulti -vator ¹	0	0	0	0	0	0	0	0	0	0	0	0
Harvesting Implements												
Cotton Picker	0	0	0	0	0	0	0	0	0	0	168	184
General Grain Combine	0	0	0	0	0	0	0	0	0	0	0	56

¹. Used during the second year only.

can take help from other components of the Expert Analysis System (Accumulated Work and Operations Analysis reports) and advantage of his own farming experiences in making final decisions for the withdrawal as demonstrated further in this chapter.

1) The recommendation for withdrawal of an object-item is based upon one-season simulation and do not account for the changes in cropping systems and other factors over time.

2) The recommendations only state that there are one or more complementary units to substitute the object-item to be withdrawn. It does not guarantee that these complementary unit(s) would definitely be available for work when called upon to carry out the tasks currently performed by the object-item recommended for withdrawal.

3) The item being recommended for withdrawal, though used rarely, may be an important backup resource on the farm and cannot be afforded for withdrawal.

Accumulated work report

This report provides the user with an aggregate and daily work reports for any combination of object-items of farm entities over a desired interval of time. To further analyze the recommendations of the Machinery Usage Report about "Ray," "Tractor_5" and "PCultivator" this report is utilized to produce their aggregate and daily work reports for the entire cropping season (January 1 of the first year to December 31

Table 4.14 Daily utilization report of Mr. Ray, the least worked laborer on Davis Farm

Day	Field	Crop	Operation	Implement	Tractor
106	Field_8	Soybean	Fertilize	FertiSpreaderLP	Tractor_4
151	Field_7	Peanut	Cultivation2	MCultivator2	Tractor_5
173	Field_1	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
177	Field_1	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
182	Field_2	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
183	Field_2	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
184	Field_6	Peanut	Spraying4	Sprayer	Tractor_4
186	Field_7	Peanut	Spraying4	Sprayer	Tractor_4
188	Field_3	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
189	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
190	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
191	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_4
192	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Tractor_5
289	Field_6	Wheat	Fertilize	FertiSpreaderLP	Tractor_4
547	Field_5	Cotton2	FertiSpreading	FertiSpreaderPF	Tractor_5
549	Field_4	Peanut2	Spraying4	Sprayer	Tractor_4
Accumulated Use (Hrs) = 142			Average Working Hrs		
(Days) = 16			per working day = 8.9		
			per working month = 20.3		

Working Months

April, May, June, July, October, June2 and July2

of the second year). Tables 4.14, 4.15 and 4.16 present these reports in a concise form respectively for "Ray," "Tractor_5" and "PCultivator."

The aggregate work hours for different farm object-items being analyzed produced by this report (Tables 4.14-4.16) exactly matched the hours produced by the Machinery Usage report. It shows that the two reports are working correctly. In addition, the tables present some supplementary information which can be utilized to further analyze the recommendations of the Machinery Usage Report. It seems that "Ray" and "Tractor_5" have some things in common which should be considered in deciding their withdrawal from the farm. Both of them are used mainly for lighter operations such as fertilizer spreading and are used together for most of the time. For example, "Ray" operated "Tractor_5" for 10 out of 13 total working days during the cropping season.

It seems "Tractor_5" makes a good match with "Ray" for lighter operations such as fertilizer spreading and spraying. The withdrawal of either of them could lead to engaging heavier tractors and/or operators, which might not be economically viable and could lead to delays in other operations. However, the Davis farm has another tractor ("Tractor_4") of similar specifications which might be able to fill in its gaps at no extra cost.

To further analyze the recommendations, the user can and should also consult the Operations Analysis Report to receive

its advice about various operations carried out by the object-items being considered for withdrawal.

Operation analysis report

This report analyzes different operations for their timely start and timely completion. It identifies factors for their delays and makes recommendations to overcome them in a cost effective manner. The two versions of this report, as stated earlier, have been developed. They respectively work with a built-in (one-third rule) and user-supplied heuristics for qualifying delays in operations and subsequently finding their remedies.

Table 4.17 summarizes the recommendations of the Operations Analyzer with the built-in heuristics for the operations employing "Tractor_5," "Ray" and/or "PCultivator." This table shows that the operations performed employing these object-items either did not have a timeliness problem or their problems were not directly related with the object-items recommended for withdrawal. In most cases, the advice was to increase work hours or speed for the operations prior to ones engaging these object-items.

To verify the functioning of the second version of the Operations Analysis Report, the operation of "Cotton" planting on "Field_3" was selected for the analysis by it. This operation has scheduled work window between Julian days 106 to 161 (corresponding to April 14 to June 8) and operation

Table 4.17 Recommendation of Operations Analyzer for the operations carried out by "Tractor_5", "Ray" and/or "PCultivator"

Field	Crop	Operation	Prob ^a	Recommendation
Field_2	Cotton	FertiSpreading	1	Increase WrkHrs Jun-Jul.
Field_5	Cotton2	FertiSpreading	1	Increase WrkHrs Jun2-Aug2.
Field_8	Soybean	Fertilize	0	
Field_7	Peanut	Cultivation2	0	
Field_1	Cotton	FertiSpreading	2	
Field_6	Peanut	Spraying4	0	
Field_7	Peanut	Spraying4	0	
Field_3	Cotton	FertiSpreading	1	Increase WrkHrs Jun-Jul.
Field_4	Cotton	FertiSpreading	1	Increase WrkHrs Jun-Jul.
Field_6	Wheat	Fertilize	0	
Field_5	Cotton2	FertiSpreading	1	Increase WrkHrs Jun2-Aug2.
Field_4	Peanut2	Spraying4	0	
Field_5	Cotton2	Cultivation1	2	Increase speed of Sprayer
Field_7	Cotton2	Cultivation1	2	Increase speed of Sprayer

^a. Different numbers means as follow

0= No Problem

1= Problem with current operation only

2= Problem with last operation only

In all cases the operations had a delayed start but timely finish

completed in one day on the 125th Julian day (corresponding to May 3rd).

With the critical window between May 5 and May 25, the analyzer was able to identify the timely start and timely completion of the operation. However, when the critical window was changed to April 20 to May 25, the analyzer identified the delayed start and timely completion of the operation and recommended to increase the working width of the planter ("Planter_77") used for the planting operation. The scheduled work hours and speed of operation for "Planter_77" were at their upper limits and there was no possibility of increasing them.

Considering the recommendations of the different components of the Expert Analysis System, additional test runs were made to study the validity of these recommendations and demonstrate the functioning of FARMSYS as a decision-aid tool. The results of each run are described briefly below.

Test Runs with FARMSYS

Run 0. This run was made by withdrawing "Chisel Plow," "General Planter" and "PDSprayer" from the farm using the Information Management System and with no changes in work schedule. It produced exactly the same simulation reports (Table 4.11) as the success run. However, the Machinery usage report indicated that there were two more implements namely

"NoTillPlanter" and "PDSprayer2," which were not being utilized at all.

Run 1. This run was made by withdrawing "NoTillPlanter" and "PDSprayer2" in addition to "Chisel Plow," "General Planter" and "PDSprayer," and it also resulted in the same simulation report. However at this stage, the Machinery Usage Report indicated that there were no more object-items (machinery and labor) which were not being utilized at all. The earlier discussion about the recommendations of various reports of Results Analyzer was based upon this run.

Run 2. This run was made by withdrawing "Ray" from the farm with no other change in the work schedule or implement parameters. The simulation run was successfully completed with all operations being completed within their agronomic work window. However, there were slight shifts in the actual start and completion of certain operations which initially involved "Ray." Most of these were spraying operations on different fields (Table 4.18) which were delayed by a day or two. This happened because "Keith," the substituting operator, was busy elsewhere and was not available to start the operations on days carried out during earlier simulation.

To study the affect of withdrawal of "Ray" on the performance of "Tractor_5," the Accumulated Work Report was used to produce its daily utilization report (Table 4.19). Although, the types of operations carried out by "Tractor_5" during the two simulations were similar, there were some

Table 4.18 Effect of withdrawal of "Ray" and "Tractor_5" on start and finish of different operations

Operation	S. Window		Case 1		Case 2		Case 3	
	SSt	SFn	ASt	AFn	ASt	AFn	ASt	AFn
Field_1 (102.2 ha), First Year (Cotton)								
FertiSpr-eading	167	223	173	181	173	181	179	186
Spraying4	167	223	182	183	182	183	187	189
Spraying5	172	243	184	184	184	184	190	191
Spraying6	183	244	186	186	186	186	192	192
Field_2 (76.8 ha), First Year (Cotton)								
FertiSpr-eading	167	212	182	187	182	189	190	197
Spraying4	167	223	188	189	190	192	198	200
Spraying5	172	243	190	191	193	193	202	202
Spraying6	183	244	192	192	194	194	203	203
Field_3 (6.0 ha), First Year (Cotton)								
FertiSpr-eading	167	212	188	188	190	190	202	202
Spraying4	167	223	190	190	193	193	203	203
Spraying5	172	243	193	193	195	195	204	204
Spraying6	183	244	194	194	196	196	206	206
Field_4 (41.6 ha), First Year (Cotton)								
FertiSpr-eading	167	212	189	192	191	193	204	209
Spraying4	167	223	193	193	194	194	210	210
Spraying5	172	243	195	195	197	197	211	211
Spraying6	183	244	196	196	198	198	212	212
Field_5 (79.6 ha), First Year (Soybeans)								
Spraying1	197	228	197	197	200	200	197	197
Field_5 (79.6 ha), Second Year (Cotton)								
FertiSpr-eading	533	580	547	552	547	553	548	555
Spraying4	533	578	553	553	554	554	556	556
Spraying5	538	609	554	554	555	556	557	557
Spraying6	548	610	555	556	557	557	558	558
Field_6 (16.4 ha), First Year (Peanut)								
Spraying4	183	223	184	184	184	184	183	183
Spraying5	192	243	192	192	195	195	192	192

Table 4.18--Continued

Operation	S. Window		Case 1		Case 2		Case 3		
	SSt	SFn	ASt	AFn	ASt	AFn	ASt	AFn	
Field_6 (16.4 ha), Second Year (Wheat)									
Fertilize	289	350	289	289	290	290	290	290	
Field_7 (10.0 ha), First Year (Peanut)									
Cultivat- ion2	151	197	151	151	151	151	152	152	
Spraying4	183	223	186	186	186	186	184	184	
Spraying5	192	243	194	194	196	196	193	193	
Field_7 (10.0 ha), Second Year (Cotton)									
FertiSpr- eading	533	580	553	553	554	554	556	556	
Spraying4	533	578	557	557	558	558	559	559	
Spraying5	538	609	558	558	559	559	560	560	
Spraying6	548	610	559	559	560	560	561	561	
Field_8 (11.8 ha), First Year (Soybeans)									
Fertilize	092	136	106	106	109	109	117	117	
Harrowing	092	136	117	117	117	117	118	118	
Bedding	106	136	118	118	118	118	119	119	
Spraying1	197	228	198	198	202	202	208	208	
Field_10 (26.4 ha), Second Year (Cotton)									
Cultiva- tion3	533	578	547	548	548	552	547	548	
FertiSpr- eading	533	580	554	556	555	557	557	558	
Spraying4	533	578	560	560	561	561	567	567	
Spraying5	538	609	561	561	567	567	568	568	
Spraying6	548	610	567	567	568	568	569	569	

SSt-	Earliest start time				SFn-	Latest finish time			
ASt-	Actual start time				AFn-	Actual finish time			
Case 1.	When both "Ray" and "Tractor_5" are available for work.								
Case 2.	When "Ray" has been withdrawn from the farm.								
Case 3.	When both "Ray" and "Tractor_5" have been withdrawn from the farm.								

Table 4.19 Daily utilization report of "Tractor_5", the least utilized tractor on Davis farm after withdrawal of "Ray"

Day	Field	Crop	Operation	Implement	Operator
109	Field_8	Soybeans	Fertilize	FertiSpreaderLP	Keith
151	Field_7	Peanut	Cultivation2	MCultivator2	Keith
173	Field_1	Cotton	FertiSpreading	FertiSpreaderPF	Keith
177	Field_1	Cotton	FertiSpreading	FertiSpreaderPF	Keith
182	Field_2	Cotton	FertiSpreading	FertiSpreaderPF	Keith
183	Field_2	Cotton	FertiSpreading	FertiSpreaderPF	Keith
190	Field_3	Cotton	FertiSpreading	FertiSpreaderPF	Keith
191	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Keith
192	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Keith
193	Field_4	Cotton	FertiSpreading	FertiSpreaderPF	Keith
547	Field_5	Cotton2	FertiSpreading	FertiSpreaderPF	Keith
Accumulated Use (Hrs)			= 108	Average Working Hrs	
(Days)			= 11	per working day	= 9.8
				per working month	= 21.6

Working Months

May, June, July, June2 and July2

changes in actual operations performed by it ("Tractor_5") in the two cases. It was also interesting to note that the overall utilization of "Tractor_5" decreased from 120 hours in 13 days to 108 hours in 11 days with the withdrawal of "Ray" from the farm. And "Keith," another laborer on the farm, became its operator for all eleven days. The reduction in work hours of "Tractor_5" may mainly be attributed to the non-availability of an operator to operate "Tractor_5" all the days it became a possible choice for work.

Run 3. This run was made by withdrawing "Tractor_5" in addition to "Ray" with no other change. The simulation was successful in completing all the operations within their agronomic work windows. There were some additional shifts in the start and/or completion of spraying operations (Table 4.18). However, it was interesting to note that some operations such as "Spraying4" and "Spraying5" on "Field_6" got expedited with the combined withdrawal of "Ray" and "Tractor_5." This may primarily be attributed to the priorities of different items of different object-classes (implement, tractor and operator) used to carry out various tasks. At this stage, "Tractor_2" became the least utilized tractor on the farm (Table 4.20). However, it was not recommended to be withdrawn from the farm because it was utilized more than 20% of the most utilized tractor ("Tractor_4"). However, the recommendation for withdrawal of "PCultivator" remained unchanged.

Table 4.20 Effect of withdrawal of the least utilized items of different farm object-classes the recommendations of Machinery Use Report

Run and Specification	Object-Item	Usage (Hrs)	Recommendation
1. With "Ray", "Tractor_5", "PCultivator" & "MCultivator"	Jerry	2820	
	Ray	142	Withdraw "Ray"
	Tractor_3	1432	
	Tractor_5	120	Withdraw "Tractor_5"
	Sprayer	552	
	PCultivator	24	Withdraw "PCultivator"
2. Without "Ray"	Jerry	2844	
	Keith	734	Do not withdraw "Keith"
	Tractor_3	1428	
	Tractor_5	108	Withdraw "Tractor_5"
	Sprayer	560	
	PCultivator	24	Withdraw "PCultivator"
3. Without "Ray" and "Tractor_5"	Jerry	2924	
	Keith	654	Do not withdraw "Keith"
	Tractor_4	1492	
	Tractor_2	516	Do not withdraw "Tractor_2"
	Sprayer	560	
	PCultivator	24	Withdraw "PCultivator"
4. Without "Ray", "Tractor", and "PCultivator"	Jerry	2912	
	Keith	654	Do not withdraw "Keith"
	Tractor_4	1480	
	Tractor_2	516	Do not withdraw "Tractor_2"
	Sprayer	560	
	MCultivator	72	Withdraw "MCultivator"
5. Without "Ray", "Tractor_5", "PCultivator" & "MCultivator"	Jerry	2896	
	Keith	630	Do not withdraw "Keith"
	Tractor_3	1468	
	Tractor_2	516	Do not withdraw "Tractor_2"
	Sprayer	560	
	SprayCoupe	280	Do not withdraw "SprayCoupe"

Run 4. This run was made by withdrawing "PCultivator." All the operations were successfully completed within their agronomic work windows with some additional shifts in start and/or completions times of the different operations. However, at this time, the Machinery Usage Report identified "MCultivator" as the least utilized plant protection implement (Table 4.20) and recommended its withdrawal from because it was also utilized less than 20% of the most utilized item (Sprayer) of the same object-class.

Run 5. This run was made by additionally withdrawing "MCultivator." The simulation completed all the operations within their agronomic work windows. After this simulation the recommendations of the Machinery Usage Report were not to withdraw any of the least utilized machinery or labor (Table 4.20) because they were either unique object-items on the farm and did not have any complementary unit to substitute for them or they had a reasonably good utilization level.

Some additional runs were made by removing some of the least utilized object-items (not recommended for withdrawal) to test the accuracy of recommendation. In these runs there were one or more operations not completed within the agronomic work windows which resulted in work area less than the cultivated area.

Yield Estimation System

This component of the decision system estimates the production of one or more crops grown on the farm. The knowledge-bases developed utilizing IBSNAT crop models and "Dry" weather year (1954, Tallahassee, FL) showed little variation in the peanut yields and an inverted U-shaped distribution for soybeans yields for different weeks during their respective planting windows. These yield results for this particular location and year make it difficult to demonstrate the effect of delayed operations utilizing these knowledge-bases.

As cotton is the major crop of the test farm (Davis farm), a hypothetical distribution (Figure 3.16) of yield response to planting weeks within its agronomic window was assumed and a knowledge-base was developed varying the yield from 600 Kg/ha to 800 Kg/ha within the planting window utilizing this distribution. This knowledge-base has been utilized to demonstrate the functioning of the Yield Estimation System an integrated component of the decision system. Tables 4.21a and 4.21b present summary reports of this system for the runs with 12 and 10 daily work hours during April to June (the agronomic work window of cotton planting for the test farm). As expected, the shorter work hours caused delays in planting cotton, thus reducing yields and overall farm profit from this crop. The total area under

Table 4.21a Effect of scheduled work hours during planting period on yield, production and profits from cotton on Davis farm. a) 12 hours of daily scheduled work during planting window, b) 10 hours of scheduled daily work during planting window.

Field	Area (ha)	Planting St. -Jdays-	Fn.	Yield (T/ha)	Total Prod. Ton	Total Sale (\$)	Total Cost (\$)	Profit Total (\$)	/ha (\$)
field1	102.2	109	112	0.80	81.8	98112	76650	21462	210
field1	102.2	475	478	0.80	81.8	98112	76650	21462	210
field2	76.8	121	124	0.80	61.4	73728	57600	16128	210
field3	6.0	125	125	0.80	4.8	5760	4500	1260	210
field4	41.6	131	138	0.75	31.2	37440	31200	6240	150
field5	79.6	486	498	0.77	61.3	73550	59700	13850	174
field7	10.0	499	499	0.75	7.5	9000	7500	1500	150
field10	26.4	500	501	0.75	19.8	23760	19800	3960	150
Total	444.8				349.6	419462		85863	
Avg.				0.7775					183

Table 4.21b Effect of scheduled work hours during planting period on yield, production and profits from cotton on Davis farm. a) 12 hours of daily scheduled work during planting window, b) 10 hours of daily scheduled work during planting window.

Field	Area (ha)	Planting St. -Jdays-	Fn.	Yield (T/ha)	Total Prod. Ton	Total Sale (\$)	Total Cost (\$)	Profit Total (\$)	/ha (\$)
field1	102.2	115	120	0.80	81.8	98112	76650	21462	210
field1	102.2	479	484	0.80	81.8	98112	76650	21462	210
field2	76.8	129	140	0.75	57.6	69120	57600	11520	150
field3	6.0	141	141	0.72	4.3	5184	4500	684	114
field4	41.6	142	150	0.72	30.0	35942	31200	4742	114
field5	79.6	493	506	0.75	59.7	71640	59700	11940	150
field7	10.0	507	507	0.72	7.2	8640	7500	1140	114
field10	26.4	517	523	0.67	17.7	21225	19800	1425	54
Total	448.8				340.1	407975		74375	
Avg.				0.74215					139

Sale Price= \$ 1200.00/Ton
 Cost of Production= \$ 750.00/ha
 Total Production Cost = \$ 333600.00

cotton on the test farm of 444.8 hectares produced 349.6 Tons and had net profit of \$ 183/hectare with 12 hours of scheduled work. With the reduced work hours the production as well as net profit reduced because of lower yields from most of the fields. The overall production decreased to 340.1 Ton and net profit per hectare to \$ 139.5 representing approximately 23% reduction in net profits.

The test runs made to evaluate the recommendations of Expert Analysis System did not cause any change in the yield reports because in all these cases the shifts in timings were mainly for spraying operations to which the crop yield knowledge-bases are not currently designed to respond to. However, it is possible to modify them if appropriate information is available.

Integrated Decision System

The integration of the four components: Information Management, Operations Simulation, Expert Analysis and Yield Estimation Systems in a seamless manner has been successfully performed under the umbrella of FARMSYS. FARMSYS acts a controller for all four components which make use of common knowledge-bases stored in the form of dynamic databases of PROLOG. It can be utilized as decision-aid for a variety of farm systems under different climatic conditions.

A team of experts who evaluated the professional qualification of FARMSYS found its components and logic

captured in them adequate designed and widely applicable for a variety of farming situations.

Four components of FARMSYS functioned correctly, intelligently and in harmony with one another on the Davis farm knowledge used as a test farm for their operational verification.

The Operations Simulation successfully simulated farm operations for complete cropping season and responded, as expected, to the variations in scheduled work hours and to the withdrawal of machinery and laborers.

The Expert Analysis System has several ways to interact with the user to help him improve the productivity of his farm system. Its different components successfully imitated the behavior of machinery management expert(s) and acted according to the heuristics captured in them.

The Machinery Usage report was able to identify the machinery and labor, not very important to the current farming system of the test farm. The Accumulated Work report and Operations Analysis report respectively provided the detailed analysis about the usage and the operations carried out by object-items recommended for withdrawal by the Machinery Usage report. This analysis was helpful in making final decisions about their withdrawals which were accurate and effective in reducing machinery and labor on the test farm. The decisions made after these recommendations demonstrated the possibility of reducing machinery and labor force from the farm without

significantly affecting the timeliness of the majority of farm operations (Table 4.22).

The Yield Estimation system responded successfully and accurately to the variations in scheduled work hours during April to June. The lower scheduled work hours during this period resulted in delayed planting of cotton on most fields, thus reducing the cotton yields and profits from different fields.

The changes in farm knowledge-bases for all of the above tests were carried out using the Information Management System, thus proving its successful functioning as an integral component of FARMSYS, a knowledge-based decision support system for multi-crop production systems.

Table 4.22 Comparison of machinery and labor resources on Davis farm available and recommended by FARMSYS

	Available List	Recommended List
Implements	DiskHarrow	DiskHarrow
	ChiselPlow	-
	SCultivator	SCultivator
	MCultivator1	MCultivator1
	MCultivator2	-
	LBCultivator	LBCultivator
	PCultivator	-
	Sprayer	Sprayer
	PDSprayer	-
	PDSprayer2	-
	SprayCoupe	SprayCoupe
	General- Planter	-
	NoTilPlanter	-
	GrainDril	GrainDril
	Bedder Bedder	
	GeneralGrain- Combine	GeneralGrain- Combine
	PeanutCombine	PeanutCombine
	CottonPicker	CottonPicker
	BottomPlow	BottomPlow
	FertiSpreaderLP	FertiSpreaderLP
	SubSoiler	SubSoiler
	Planter_77	Planter_77
	BMower	BMower
	FertiSpreaderPF	FertiSpreaderPF
Operators	Jerry	Jerry
	Joe	Joe
	Ray	-
	Keith	Keith
Tractors (HP)	JD4450 (148)	JD4450 (148)
	JD4430 (125)	JD4430 (125)
	JD2950 (90)	JD2950 (90)
	JD3020 (70)	JD3020 (70)
	JD2640 (70)	-

CHAPTER V CONCLUSIONS AND RECOMMENDATIONS

The principal objective of this dissertation to represent and manipulate farm operational knowledge, including dynamic processes as well as heuristics for expert planning and management for machinery, labor and other resources was successfully accomplished utilizing knowledge engineering concepts of artificial intelligence. It has opened up a new horizon for developing decision-aid systems for complex agricultural problems. This new approach permitted seamless integration of simulation and knowledge systems, including their necessary databases.

The use of PROgramming in LOGic (PROLOG) for engineering farm knowledge facilitated simulating field operations in an object-oriented manner. The inferencing capability of PROLOG were utilized to incorporate several expert systems within the simulation to make heuristic decisions and other types of searches at various stages of simulation. The object-oriented approach to simulation helps in capturing system's knowledge and depicting its dynamic behavior in a much more realistic manner than the conventional approaches to simulation. This

approach also permits easy modifications and upgrades in the program code because of its modular structure.

An operational scale decision-support system, FARMSYS, consisting of an intelligent information management system, an object-oriented operations simulation, an expert analysis system and a crop yield estimation system was designed and constructed. It was also evaluated for its professional qualifications and operational capabilities using a real farm knowledge set.

The structure of the simulation in two distinct components: farm and regional knowledge, and simulation procedural knowledge has resulted in a versatile and flexible system which can be adjusted for a variety of farming situations ranging from highly mechanized farm (such as Davis farm) to a subsistence labor intensive farm with no mechanization at all.

The encouraging results of the evaluation session for the professional qualifications of FARMSYS by a team of experts and verification of its operational capabilities through the use of a real farm knowledge set clearly indicate the possibility of using it as a decision-aid tool for planning and managing farming operations under a variety of farm situations. The concept of seamless integration of simulation and expert systems and their ability to use common knowledge-bases can be used for a variety agricultural and industrial applications.

A few additional thoughts to further enhance the capabilities of FARMSYS as a decision-aid tool are presented below.

1) This research was mainly directed to handle the crop production component of a farm system. Most farms, especially small scale enterprises also have animal production as an integral component of their farming systems. The inclusion of an animal component in the representation of farm knowledge would result in a more practical farm model and decision-aid.

2) The scheduling of work hours for simulating field operations are currently adjusted on the global basis for the whole farm. All farm object-items are available for the entire cropping season and work the same number of hours for a given day. In real world situation, many farms have varying work hours for different members of their labor crew and even hire extra labor and machinery to meet their seasonal peaks.

This can be achieved by assigning a list of work hours to each item of machinery and laborer class of the farm. This would permit adjusting scheduled work hours for each of them individually and also inclusion of custom hiring in and out of extra machinery and labor during critical periods of the cropping season.

3) The actual work hours for an operation for a day are currently based upon its type, the scheduled work hours and rainfall of the day. However in the real world situation, the past and expected future weather play a very important role

in farmer's decision for his working hours for the day. The inclusion of this factor would make the simulation more accurate.

In addition, there is a need to employ some systematic studies to develop genuine heuristics which govern the actual work hours for different regions.

4) Most farmers are interested in estimating the cost of machinery and labor component in their crop production activities. The incorporation of this factor utilizing some of the existing packages or developing a specialized one for FARMSYS would improve its (FARMSYS) acceptability for commercial enterprises. This can be done by attaching another attribute (cost/hour) for each machinery and labor on the farm.

5) All agricultural operations for crop production are sequential. They can be carried out one after another. However, certain operations such as two consecutive applications of a pesticide may require a minimum time delay between them to be effective.

There is also a need to attach a factor of importance to different operations. This factor then would decide for the continuation of subsequent operations if any of the earlier operations have failed completely or partially. It would avoid the situation of not harvesting a field which has been planted, but could not cultivated.

All these can be achieved by assigning some additional attributes to the operations object-class.

6) The effect of delayed harvesting on yield loss is not included in the crop models utilized for generating crop yield knowledge-bases for the Yield Estimation System. Therefore, there is a need to develop functions and/or expert knowledge-bases on yield losses due to a late harvest of different crops under various environments.

7) FARMSYS is currently designed for strategic planning and not for tactical management in real time. In addition, it is structured to handle a farm with a certain level of mechanization either through tractor or animal traction. However, its structure permits modifying to work as a tactical management tool and/or to handle labor intensive farming with no mechanization at all.

For a tactical management decision-aid tool in real time the adaptation needed would be 1) to stop the simulation at pre-defined interval and save the current environment, 2) present the current situation to the user, 3) provide him with the opportunity to make decisions about further actions either from his own experience or based upon the recommendations from the Expert Analysis System, 4) letting the user make changes in the resource-base and/or management strategies and 5) picking up the saved environment and continuing for simulation with the new resource-base until the next pause stage.

The adaptation of FARMSYS for labor intensive farming for subsistence agriculture of developing countries would require 1) defining a field with a number of crops simultaneously and operations list attached directly to the field rather than the individual crop, 2) the possibility of defining a list of farm laborers (family or hired) to carry out different operations and to allocate them to the field whenever that operation is carried out and 3) a built-in mechanism to estimate the operational capacity of the crew of laborers working for an operation based upon age and sex of these individuals.

APPENDIX I
KNOWLEDGE-BASES

Crop Yield Knowledge-bases

General format

```
cropyld(CropName, Year of Simulation, Variety, Soil Type,  
Irrigation, Fertilization, Planting Week, DaysToMaturity,  
Irrigation, Yield)
```

It would contain numerous entries depending upon the combinations of various management inputs. Hypothetical examples for the some crops of the test farm follow.

```
cropyld("Soybeans", "Dry", "Kirby", "Sandy Loam",  
"NotIrrigated", "Fertilized", 19, 136, 0, 0.8)  
cropyld("Peanut", "Dry", "Florunner", "Sandy Loam",  
"NotIrrigated", "Fertilized", 14, 128, 0, 2.05)  
cropyld("Wheat", "Dry-Dry", "FL302", "Sandy Loam",  
"NotIrrigated", "Fertilized", 47, 145, 0, 5.8)
```

Knowledge Generated by Information Manager

The following are the contents of different farm files of Davis Farm used as a test farm for different components of FARMSYS.

Farm File

General format

```
farm(Name, Location, Total Area, CultArea, Activity, Code)
```

The farm file contains only one entry as follow.

```
farm("DavisFarm", "SantaRosa", 400, 400, "CropProduction",  
dvs")
```

Labor File

General format

```
labor(Num,Name,Sex,FunctionList,AvailStatus)
```

It contains 4 entries one for each labor.

```
labor("Laborer_1","Jerry","m",["General"],"Avail")
labor("Laborer_2","Joe","m",["General"],"Avail")
labor("Laborer_3","Ray","m",["General"],"Avail")
labor("Laborer_4","Keith","m",["Genral"],"Avail")
```

Tractor File

General format

```
tractor(Num,Model,HP,OperatorList,AvailStaus)
```

It contains 5 entries one for each tractor.

```
tractor("Tractor_1","JD4450",148,["Jerry"],"Avail")
tractor("Tractor_2","JD4430",125, ["Keith", "Jerry",
"Joe"], "Avail")
tractor("Tractor_3","JD2950",90,["Joe","Keith"],
"Avail")
tractor("Tractor_4","JD3020",70,["Jerry", "Joe", "Ray",
"Keith"],"Avail")
tractor("Tractor_5","JD2640",70,["Jerry", "Joe", "Ray",
"Keith"],"Avail")
```

Implement File

General format

```
implement(Num,Type,Name,TractorList,Width,Speed,
AvailStatus)
```

It contains 25 entries one for each implement, example for each type of implement follows.

```
implement("Implement_2","LandPrep","DiskHarrow",
["Tractor_1","Tractor_2"],4.74,7,"Avail")
implement("Implement_5", "PlantProtection",
"SCultivator", ["Tractor_3","Tractor_4",
"Tractor_5"], 3.6,7,"Avail")
implement("Implement_14","PlantFerti","GeneralPLanter",
["Tractor_3"],3.6,8,"Avail")
implement("Implement_22","Harvesting","PeanutCombine",
["Tractor_1"],1.8,3.5,"Avail")
```

Crop File

General format

crops(Num,Name,OperationsList)

The crop file contains 8 entries, 2 for each crop to account for two years, example for one year entry for each crop follow)

```
crops("Crop_3","Wheat",["Fertilize","Harrowing",
    "Plowing", "Planting", "FertiSpreading",
    "Harvesting"])
crops("Crop_7","Soybeans", ["Fertilize", "Harrowing",
    "Bedding"Planting", "Cultivation1", "Cultivation2",
    "Cultivation3", "Spraying1", "Spraying2",
    "Harvesting"])
crops("Crop_2", "Cotton",["Fertilize", "Harrowing",
    "Plowing", "Planting", "Spraying1", "Cultivation1",
    "Spraying2", "Cultivation2", "Spraying3",
    "Cultivation3", "FertiSpreading", "Spraying4",
    "Spraying5", "Spraying6", "Harvesting","Mowing"])
crops ("Crop_4", "Peanut", ["Fertilize", "Harrowing",
    "Plowing", "Planting", "Cultivation1", "Spraying1",
    "Cultivation2", "Spraying2", "Spraying3",
    "Spraying4", "Spraying5","Harvesting"])
```

Field File

General format

```
fieldi(Num, Name, CultArea, SoilType, CropList,
    VarietyList, M a t u r i t y D a y s L i s t ,
    FertilizationLevelList, IrrigationStatus,
    OperatorforIrrigation, CostofProductionPerHectList,
    SalePricePerTonList, AvailStatus)
```

It contains 23 entries, one for each field, example of two fields follows.

```
fieldi("Field_1", "field1", 20, "SandyLoam", ["Cotton",
    "Wheat"],["DPL41", "FL302"], [150, 165],
    ["Fertilized", "Fertilized"], "NotIrrigated",
    "NotNeeded", [750,125], [1200,110], "Avail")
fieldi("Field_12", "field12", 2.4, "SandyLoam",
    ["Cotton", "Cotton2"], ["DPL90","DES119"],
    [150,150], ["Fertilized", "Fertilized"], "NotIr-
    rigated", "NotNeeded", [750,750],
    [1200,1200],"Avail")
```

Operation File

General format

```
operations(Type, Name, Crop, EarliestStartTime,
           LatestFinishTime, WorkHrsCondList, ImpList, Eff)
```

It contains 84 entries, one for each operation example for planting operations for different crops follows.

```
operations("PlantFertiOps", "Planting", "Soybeans", 136,
182, [[0,3,1],[3,8,0.5]], ["Planter_77"], 0.76)
operations("PlantFertiOps", "Planting", "Wheat", 320,
366, [[0,3,1],[3,8,0.5]], ["GrainDril"], 0.76)
operations("PlantFertiOps", "Planting", "Cotton", 106,
161, [[0,3,1],[3,8,0.5]], ["Planter_77"], 0.76)
operations("PlantFertiOps", "Planting", "Peanut", 106,
121, [[0,3,1],[3,8,0.5]], ["Planter_77"], 0.76)
```

Type FileGeneral format

```
type(EntityName,ItemList)
```

It would contain variable number of entries depending upon the farm size and its activities. The contents of this file of the test farm are listed below.

```
type("opsimplementtype",["LandPrep", "PlantFerti",
"PlantProtection","Harvesting"])
type("LandPrepOps",["Fertilize","Bedding","Harrowing",
"Plowing","Mowing","End","AddNewItem",
"DeleteAnItem"])
type("PlantFertiOps",["End","FertiSpreading",
"Planting"])
type("PlantProtectionOps",["Spraying7","Cultivation4",
"Spraying1","Spraying2","Spraying3","Spraying4",
"Spraying5","Spraying6","Cultivation1",
"Cultivation2","Cultivation3","AddNewItem",
"DeleteAnItem"])
type("HarvestingOps",["Harvesting","End"])
type("LandPrepImp",["FertiSpreader","Ridger",
"DiskHarrow","DiscPlow","MBPlow","BottomPlow",
"MBPlow1","AddNewItem","DeleteAnItem"])
type("PlantFertiImp",["Planter_77","CottonSeeder",
"GeneralPlanter","FertilizerSpreader",
"AddNewItem","DeleteAnItem"])
```

```

type("PlantProtectionImp", ["SCultivator",
    "MCultivator1", "MCultivator2", "LBCultivator",
    "PCultivator", "Sprayer", "PDSprayer", "PDSprayer2",
    "SprayCoupe", "AddNewItem", "DeleteAnItem"])
type("HarvestingImp", ["BMower", "GeneralGrainCombine",
    "PeanutCombine", "CottonPicker", "AddNewItem",
    "DeleteAnItem"])
type("crop", ["Cotton", "Wheat", "Cotton2", "Soybeans",
    "Soybeans2", "Corn", "Peanut", "AddNewItem",
    "DeleteAnItem"])
type("Soybeans", ["Kirby", "Bragg", "Centennian",
    "AddNewItem", "DeleteAnItem"])
type("Peanut", ["Florunner", "AddNewItem", "DeleteAnItem"])

type("Cotton", ["SOCS453", "DLP90", "DES119", "SOS106",
    "DLP41", "AddNewItem", "DeleteAnItem"])
type("Wheat", ["FL302", "AddNewItem", "DeleteAnItem"])
type("irrigation", ["Irrigated", "NotIrrigated"])
type("soiltype", ["SandyLoam", "SiltLoam", "ClayLoam",
    "FineSand"])
type("fertilization", ["Fertilized", "NotFertilized"])
type("activity", ["CropProduction", "LiveStock",
    "MixedFarming"])

```

Knowledge Utilized by Information Manager

Screen Files

General format

```

field(Name, Type, StartRow, StartCol, Length)
    txtfield(StartRow, StartCol, Length, Text)
    windowsize(Hight, Width)

```

The contents of the screen file for constructing screen layout for collecting General Farm information are listed below.

```

field("action", str, 6, 35, 25)
field("name", str, 8, 28, 29)
field("location", str, 10, 28, 23)
field("tarea", real, 12, 35, 11)
field("cultarea", real, 14, 35, 11)
field("activity", str, 16, 47, 30)
field("filesexten", str, 18, 53, 4)
txtfield(8, 12, 12, "Name of Farm")
txtfield(10, 12, 8, "Location")
txtfield(12, 12, 14, "Total Area (ha)")
txtfield(14, 12, 19, "Cultivated Area (ha)")
txtfield(4, 0, 16, " ")
txtfield(2, 0, 10, " ")

```

```

txtfield(0,11,1,"*")
txtfield(1,11,1,"*")
txtfield(2,11,1,"*")
txtfield(3,0,67,"*
*****
txtfield(0,16,48,"FARMSYS-A Decision Support System for
Multi-Crop")
txtfield(0,66,3," *")
txtfield(1,66,3," *")
txtfield(3,68,1,"*")
txtfield(1,30,20," Production Systems")
txtfield(2,13,56," Farm Information Manager-General
Farm
Information *")
txtfield(6,12,23,"Action being Performed ")
txtfield(18,12,35,"Farm generally known as (3 letters)")
txtfield(16,12,31,"Press<Enter> to Select Activity")
windowsize(20,77)

```

APPENDIX II
MINI-CONFERENCE FOR FARMSYS QUALIFICATION
PROGRAM

Decision Support System with Logic Programming
Agricultural Engineering Department
University of Florida
November 14, 1988

- 8:30 a.m. -Purpose of the meeting, and schedule adjustments
Dr. Peart
- 8:45 a.m. -Introduction to FARMSYS and to the concepts of
"Qualifying" rather than "Validating" knowledge-based systems
Harbans Lal and Dr. Peart
- 9:30 a.m. - Orange juice break
- 9:45 a.m. - Demonstration of FARMSYS and introduction to
questionnaire
Harbans Lal and Dr. Peart
- 11:15 a.m. -Operations management research, MACH II and U.S.
Sugar
Dr. W. D. Shoup
- 11:45 a.m. - Lunch
- 1:15 p.m. - Qualification Session with FARMSYS using
questionnaire
Harbans Lal and Dr. Peart
- 2:45 p.m. - Break
- 3:00 p.m. - Review of the international DSSAT program
Dr. Shrikant Jagtap
- 3:45 p.m. - Open Session (Discussion on approaches and
procedures for evaluating knowledge-based systems)
- 5:00 p.m. - Adjourn

Table II-1 Participants Names and Addresses

Name	Address
Dr. John Ogilvie ¹	Director, School of Engineering University of Guelph, Ontario, Canada
Dr. Bill Chancellor ¹	Professor, Dept. of Agric. Engg. University of California, Davis, CA 95616
Dr. Maurice R. Gebhardt ¹	Professor, Agric. Engg. Bldg. USDA-ARS, University of Missouri Columbia, MO 65211
Dr. M. Zohadie Bardaie ¹	Professor, Agric. Engg. Dept. University Pertanian Malaysia 43400, Serdang, Selangor, Malaysia
Dr. Kenneth L. Von Bargaen ¹	Professor, Agric. Engg. L. W. Chase Hall, University of Nebraska Lincoln, NE 68583-0726
Dr. Thomas S. Colvin ¹	Agricultural Engineer, USDA/ARS 213 Davidson Hall, Iowa State University Ames, IA -50011
Dr. Garry D. Bubenzer ¹	Chairman, Agric. Engg. Dept. University of Wisconsin 460 Henry Hall, Madison- WI 53706
Dr. R. K. Sprinkel	Assoc. Professor, Int. Pest Management Entomology and Nematology, Univ. of Florida Gainesville-FL 32611

Table II-1--continued

Name	Address
Dr. Peter Hildebrand	Professor, FSR/E Food and Resource Economics Univ. of Florida Gainesville -Fl 32611
Dr. Larry Bagnell	Professor Dept. of Agric.Engg. Univ. of Florida Gainesville -FL 32611
Dr. R. M. Peart	Grad. Research Professor Dept. of Agric.Engg. Univ. of Florida Gainesville -FL 32611
Dr. J. W. Jones	Professor Dept. of Agric. Engg. Univ. of Florida, Gainesville -FL 32611

1- Participants who filled in the questionnaire on FARMSYS

FARMSYS Evaluation and Qualification
Harbans Lal and R. M. Peart

Please indicate the extent to which you agree or disagree with the different statements.

- 1--Strongly agree
- 2--Agree
- 3--Undecided
- 4--Disagree
- 5--Strongly disagree

Evaluation of different components

1. Information Manager

This component is designed to collect information about a new farm and let the user make changes in the already existing farm. The important requirements for this component are ease and user-friendliness in collecting and updating farm information and checking the integrity of the supplied information.

Let us see if we have been able to incorporate these features effectively.

Please evaluate screen layouts for collecting information about different farm entities.

STRONGLY STRONGLY
AGREE-----DISAGREE

Screen layouts for collecting information about different farm entities is well designed.	1	2	3	4	5
---	---	---	---	---	---

Mechanism of letting user develop and maintain knowledge-base specific to his farm is unique for FARMSYS.	1	2	3	4	5
---	---	---	---	---	---

Mechanism of retrieving, presenting old information, letting user make changes and saving new information are adequately designed.	1	2	3	4	5
--	---	---	---	---	---

Information Manager is qualified to let user develop new farm and modify information about existing farm.	1	2	3	4	5
---	---	---	---	---	---

2. OPS-SIMULATOR

The Ops-Simulator is designed to simulate the field operations to predict the scheduling of different field operations. The effectiveness of Ops-Simulator in depicting the real world situation depends on its formulation and incorporation of the factors controlling these operations in as realistic a manner as possible. The key features of this component have been identification of different components of an operational knowledge of a farm, its representation and manipulation to generate new knowledge which can be then utilized by other components of FARMSYS.

Please evaluate the approach to simulating field operations.

2.1 Knowledge-Bases

STRONGLY STRONGLY
AGREE----DISAGREE

The semantic representation of farm knowledge (to be explained) adequately expresses the farm knowledge for multi-crop production system.

1 2 3 4 5

The components of the following knowledge-bases are adequately structured (to be explained).

a) Farm Knowledge:

Farm	1	2	3	4	5
Labor	1	2	3	4	5
Tractor	1	2	3	4	5
Implement	1	2	3	4	5
Equipment	1	2	3	4	5
Crops	1	2	3	4	5
Fields	1	2	3	4	5
Operations	1	2	3	4	5

b) Regional Knowledge:

Expert File	1	2	3	4	5
Weather File	1	2	3	4	5

c) Output Reports:

Work Reports	1	2	3	4	5
No-Work Reports	1	2	3	4	5
Summary Report	1	2	3	4	5

Please evaluate the following features of the Ops-Simulator for farm operations simulation modelling. Do they allow us to develop a simulator which would predict the system's behavior in a more accurate and realistic manner than most conventional simulators.

Indicate to what extent you agree or disagree with the following statements.

STRONGLY STRONGLY
AGREE-----DISAGREE

The expected future weather while deciding actual work hours is not considered.

1 2 3 4 5

The Ops-Simulator is qualified to simulate field operations for multi-crop production systems.

1 2 3 4 5

Please help identify additional weaknesses and strengths of the Simulator in comparison to other similar programs familiar to you. (Use back side of page if needed.)

a) Weaknesses

1)

2)

b) Strengths

1)

2)

3. YLD-ESTIMATOR

The yld-estimator is designed to estimate the overall production and net profit from different crops grown on the farm. The power of this component depends upon the power of the mechanism utilized to predict yields of different crops grown on the farm under different management inputs. We utilized the IBSNAT crop models for making these estimations. However, we are interested in identifying other ways to predict crop yields. Please help us to do that.

4.1 Individual Reports Evaluation

a) Machinery Usage Report

Objective

To identify the least utilized object-item of the selected machinery type or laborers and develop a recommendation about whether or not to keep it. This report is designed to identify the excess machinery available on the farm. Please give your opinion.

STRONGLY STRONGLY
AGREE-----DISAGREE

The objective of the report is sufficiently strong and impressive and users will make intensive use of this report.

1 2 3 4 5

The criteria (to be explained) utilized for identifying the least utilized object-item are adequate and realistic.

1 2 3 4 5

The criteria (to be explained) utilized for deciding recommendation about least utilized object-item are adequate and realistic.

1 2 3 4 5

List other criteria which could be employed for developing recommendations for the least utilized object-item.

a)

b)

c)

b) Operations Analysis Report

Objective

1) Identification of Problems

To identify operations which had delays in their start and/or finish, along with the factors responsible for these delays. This report is designed to identify bottleneck in timely completion of the different operations such as insufficient scheduled work hours, low working speed, or smaller size machinery on the farm.

	<u>STRONGLY</u> <u>AGREE</u> -----		<u>STRONGLY</u> <u>DISAGREE</u>		
The objective of the report is sufficiently strong and impressive and users will make intensive use of this report.	1	2	3	4	5
The criteria (to be explained) utilized for identifying problematic operations is not adequate and needs improvement.	1	2	3	4	5
The break-up of the timing window is ok.	1	2	3	4	5
List other criteria which could be utilized for the purpose (including criteria for dividing timing window).					
a)					
b)					
c)					

2) Developing Recommendations

The report also develops cost effective recommendations to minimize these delays.

The criteria (to be explained) utilized to make the following recommendations are adequate and realistic.

	<u>STRONGLY</u> <u>AGREE</u> -----		<u>STRONGLY</u> <u>DISAGREE</u>		
Increase Scheduled Work hours	1	2	3	4	5
Increase Speed of operation	1	2	3	4	5
Increase Working width	1	2	3	4	5
Suggest methods for improving the above recommendations.					
1)					
2)					
3)					

c) Accumulated Work Report

Objective

To present an accumulated work report for any combination of object-items of farm entities over any desired interval of time period. This report is designed to let users know about the intensity of usage of different object-items of various farm entities. Using this report one can determine how many hours Tractor_1 worked on Soybean Fields during the period January 1 to Jun 15 etc.

STRONGLY STRONGLY
AGREE-----DISAGREE

The objective of the report is sufficiently strong and impressive and users will make intensive use of this report.

1 2 3 4 5

The interactive mode for letting users select different farm object-items and time interval is adequately designed.

1 2 3 4 5

The results of this report are presented in the form of two sub-reports namely Summary Reports and Detailed Report (to be explained). Please evaluate these reports.

Both reports are essential

1 2 3 4 5

Both reports are logically presented.

1 2 3 4 5

List other types of information which could be included to make these reports more useful.

a)

b)

c)

4.2 Overall Evaluation

The Expert Advisor presently prepares the three reports evaluated individually earlier. However, we would like to have your overall impression about these reports.

STRONGLY STRONGLY
AGREE-----DISAGREE

The Expert Advisor is an important component of FARMSYS.

1 2 3 4 5

The Expert Advisor is qualified to carry out its desired functions.

1 2 3 4 5

Please list additional reports and their objectives which you would like to see included in Expert Advisor.

a) Title:
Objective:

b) Title:
Objective:

5. FARMSYS

Finally, we would like to have your overall impression of FARMSYS as an integrated decision-aid tool.

FARMSYS is qualified as a decision-aid tool for machinery management for multi-crop production system.

1 2 3 4 5

Things You Would Like To See Added To FARMSYS

ITEM:

Things You Would Like To See Deleted From FARMSYS

ITEM:

Name:
Address:

Non-Structured Responses about FARMSYSWeaknesses

- 1) Economic factors need to be incorporated more
- 2) Soil consideration on farming not adequately considered
- 3) Some obvious refinements not presented or communicated
- 4) Does not allow for information input from out side or even from internal previous day simulations in deciding what will be done on the following day.
- 5) Lack of use of established algorithms
- 6) Inability to adapt to tactical planning

Strengths

- 1) Comprehensive in selection process
- 2) Concept of no-work report is very good
- 3) structured approach
- 4) Excellent framework. Most suggestions made would be good and desirable but not essential.
- 5) global approach
- 6) Tries to include a broad range of factors and data.
- 7) Ease of setting up
- 8) Flexibility of reports
- 9) Adaptability to diverse areas of the world.
- 10) The mechanism of user information input

Suggestion

- 1) User definition of agronomic work window and critical timing of operations.
- 2) Include cash flow report, to keep track of cash requirement over time.
- 3) Looking at impact of soil beyond soil type.
- 4) Distinction between management and planning tool. They need different types of inputs.
- 5) Flexibility to account for individual criteria about machinery usage report and operations analysis report.
- 6) Need to accommodate more tools
- 7) Amount of resources needed by week or by month (fuel, labor etc.)
- 8) More rigorous role on soil/water/machinery management interactions.
- 9) multi-operations in field at same time or sub-divide the fields to allow this.
- 10) Utilize weather data to set initial time for operation to begin.
- 11) Include operation economics, cash flow projection and specific effect of operations upon profit.
- 12) Age of the implement for deciding recommendations about the machinery usage report, which gives idea of need for keeping back up. Other factors suggested for this report are a) potential decrease in market of machine value with

time, b) need for the machine as back up.

13) In critical periods day could be broken down to 2 hours interval or the next job could start immediately even though it is not the start of a new day.

14) Make facility to put more than one tool in same field for the same operation or subsequent operations.

15) Add economic data for each object-item e.g. US\$/hr for each worker and for each tractor and implement.

16) Graphics like Gnat chart to show actual happening vs. planned.

17) Realignability of windows based upon events which have developed on day to day basis.

18) The window should always be elongated (even as non-preformed choice) in order to get the land planted- no matter what.

19) A system that allows the daily choices to be made based upon the state of the fields, equipment, crop, operators and weather on the previous day.

20) Considering down time of first choice tractor, implement, etc.

21) Prepare maintenance report for tractors and implements.

22) Allow part time labor and machinery

Concerns

1) The user setting all the windows for work

2) I am not sure of validity of this evaluation. Time too much of limitation.

3) What about yesterday's work conditions influencing today's work.

4) I am troubled by lack of use of the established algorithm which have been field validated.

5) I really like its data input system and its concept of priority. I think its use of algorithm for appropriate tasks such as work days available need to be evaluated.

REFERENCES

- Addis, T. R. 1985. Designing knowledge-based systems: Knowledge for the new generation computers. Prentice-Hall, Englewood Cliffs, NJ.
- Adelsberger, H. H. 1984. PROLOG as a simulation language. In Proc. of the 1984 Winter Simulation Conf. ed. S. Sheppard, V. Pooch and D. Pegden. SCS, San Diego, CA. pp. 501-504.
- AAAI and ASAE. 1988. Integration of expert systems with conventional problem solving techniques in agriculture. Proc. of the Workshop, AAAI and ASAE. Aug. 10-12, San Antonio, TX.
- Barrett, J. R., J. B. Morrison and L. F. Huggins. 1985. Artificial intelligence and expert systems in agricultural research and education. ASAE paper 85-5516. ASAE, St. Joseph, MI.
- Batchelor, W. D., R. W. McClendon, J. W. Jones and D. B. Adams. 1987. An expert simulation system for soybean insect pest management. ASAE Paper 87-4501. ASAE, St. Joseph, MI.
- Beck, H. W. and J. W. Jones. 1988. Simulation and artificial intelligence concepts. Paper presented at the Expert Systems Workshop, Orlando, FL, Feb. 1988. ASAE, St. Joseph, MI.
- Bender, D. 1984. Hierarchical integration of simulation and linear programming to assess risk-efficient cropping systems. PhD Thesis. Purdue University, West Lafayette, IN.
- Bennett, T. B., R. S. Sowell and R. E. Sneed. 1988. An expert system for irrigation planning and design. ASAE Paper 88-5021. ASAE, St. Joseph, MI.
- Boggess, W. G., and C. B. Amerling. 1983. A bioeconomics simulation analysis for irrigation investments. S. Journal Agric. Econ. 16: 85-91.

- Boote, B. J. and Jones, J. W. 1986. Application of and limitations in crop growth simulation models to fit crops and cropping systems to semi-arid environment. Depts. of Agronomy and Agric. Engg. University of Florida, Gainesville, FL.
- Borland. 1986. Turbo PROLOG: The natural language for artificial intelligence. Borland International Inc., Scotts Valley, CA.
- Borland. 1987. Turbo PROLOG Tool Box: User's guide and reference manual. Borland International Inc., Scotts Valley, CA.
- Brown, T. L. 1974. Farm machinery selection by use of a zero-one mixed integer linear programming. MS Thesis. Purdue University, West Lafayette, IN.
- Bulkeley, W. M. 1986. Expert systems are entering into main stream of computers. Technology. The Wall Street Journal, Dec. 5. 1986.
- Burrows, W. C. and J. C. Siemens. 1974. Determination of optimum machinery for Corn-Soybean farms. Trans. of the ASAE 17(6):1130-1135.
- Candler, W. 1968. A linear program for selecting a corn production system. Conf. Proceedings: Computers and Farm Machinery Management, Dec. 1968. ASAE St. Joseph, MI. pp. 20-21.
- Chen, L. H. 1986. Microcomputer model for budgeting and scheduling crop production. Trans. of the ASAE 29(4):908-911.
- Chen, L. H. 1987. A microcomputer model for selection of machines and tractors. ASAE Paper 87-1050. ASAE, St. Joseph, MI.
- Chen, L. H. and R. W. McClendon. 1985. Soybean and wheat double cropping simulation model. Trans. of the ASAE 28(1):65-69.
- Citrenbaum, R., J. R. Geissman and R. Schultz. 1987. Selecting a shell. AI EXPERT. 2(9):30-39.
- Coulson, R. N., L. J. Folse and D. K. Loh. 1987. Artificial intelligence and natural resource management. Science 237:262-267.

- Davis, J. R. and P. M. Nanninga. 1985. GEOMYCIN: Towards a geographic expert system for resource management. *Journal of Environment Management* 85(21):377-390.
- Digitalk. 1986. *Smalltalk/V: Tutorial and program handbook*. Digitalk Inc., Los Angeles, CA.
- Donahue, D. W., R. S. Sowell and N. T. Powell. 1988. An expert system for diagnosing diseases of tobacco. ASAE Paper 88-5022. ASAE, St. Joseph, MI.
- Edwards W. and Michael B. 1980. Machinery selection considering timeliness losses. *Trans. of the ASAE-1980*. pp 810-815,820.
- Eisgruber, L. M. 1968. Where do we go from here ? - Panel discussion. *Conf. Proceedings: Computers and Farm Machinery Management, Dec., 1968*. ASAE St. Joseph, MI. pp 52.
- Elderen E. van. 1981. Scheduling of field operations: Research report. *Inst. of Agric. Engg. IMAG, Wageningen, Netherland* 81(3):865-871.
- Elderen, E. van. 1987. Scheduling farm operations: A simulation model, Pudoc Wageningen, Netherland.
- Elmaghraby, A. S. and V. Jagannathan. 1986. An expert system for simulationists. In: *AI Graphic and Simulation*. ed. Graham Birtwistle. Soc. for Comp. Simu. San Diego, CA. pp 106-109.
- Engel, B. A., D. D. Jones and J. R. Wright. 1988. Selection of an expert system development tool. ASAE Paper 88-5017. ASAE, St. Joseph, MI.
- Engel, B. A., D. B. Beasley and J. R. Barrett. 1988a. Knowledge engineering in soil erosion. ASAE Paper 88-2140. ASAE, St. Joseph, MI.
- Fan, I. S. and P. J. Sackett. 1988. A PROLOG simulator for interactive flexible manufacturing systems control. *Simulation* 50(6):239-247.
- Feigenbaum, E. A. and P. McCorduck. 1983. *The fifth generation: artificial intelligence and Japan's computer challenge to the world*. Addison-Wesley, Reading, MA.
- Fisher, E. L. 1986. An AI-based methodology for factory design. *AI Magazine*, Fall 1986. pp. 72-85.

- Flowers, E. B. 1988. Failing with grace. Turbo Technix. The Borland Language Journal 1(5):76-85.
- Futo, I. and T. Gergely. 1986. TS-PROLOG a logic simulation language, TRANS. of the SCS, 3(4): 112-119.
- Futo, I. and T. Gergely. 1987. Logic programming in simulation, Trans. of the Soc. for the Comp. Simu. 3(3):195-216.
- Gaines B. R and M. L. G. Shaw. 1986. Expert systems and simulation. In: AI Graphic and Simulation. ed. Grahm Birtwistle. Soc. of Comp. Simu. San Diego, CA. pp. 95-101.
- Gauthier, L. and R. Kok. 1988. A microcomputer-based farm management/ operating system. Can. Agric. Eng. 30(1):69-76.
- Geyer, F. P., R. M. Peart and W. H. M. Morris. 1963. Bottlenecks in handling corn at harvest. ASAE Paper 63-829. ASAE St. Joseph MI.
- Gibson, H. G., D. D. Jones, J. R. Barrett and C. H. Shih. 1986. Timber harvesting equipment selection: An expert system. ASAE Paper 86-1604. ASAE, St. Joseph, MI.
- Glunz, D. J. 1985. Comparative analysis of Florida citrus harvest systems. MS Thesis. University of Florida, Gainesville, FL.
- Gui X. and J. C. Siemens. 1986. An optimum machinery selection model for corn soybean farm. ASAE Paper 86-1523. ASAE, St. Joseph, MI.
- Halterman, S. T., J. R. Barrett and M. L. Swearingin. 1986. Expert systems development stages: Double cropping as an example. ASAE Paper 86-4516. ASAE St. Joseph, MI.
- Hart, R. D. 1984. Hierarchical agricultural systems. In: Prosepectives on farming systems research and extension. ed. Peter E. Hildebrand. Lynne Rienner Publishers, Boulder, CO. pp. 29-32.
- Hayes-Roth, F., D. A. Waterman and D. B. Lenatm. 1983. Building expert system. Addison-Wesley Publishing Co. Inc., Reading, MA.

- Helms, G. L., J. W. Richardson, M. E. Rister, N. D. Stone, D. K. Loh. 1987. COTFLEX: A farm-level expert system to aid farmers in making farm policy decisions. Paper presented at the 1987 Annual Meeting of Am. Agric. Econ. Assoc. East Lansing, MI.
- Hetz, E. J. and M. L. Esmay. undated. A field machinery selection model for wheat producers in the Andes Pre-Cordillera of south central Chile. Agric. Engg. Dept. Michigan State University, East Lansing, MI.
- Hoogenboom, G., Jones, J. W. and J.W. White. 1987. Use of models in studies of drought tolerance. Paper presented at the Workshop on Drought in Dry Beans. October 1987. Centro Internacional da Agricultura Tropical, Cali, Colombia.
- IBSNAT. 1986. Decision support system for agrotechnology transfer (DSSAT): Documentation for IBSNAT crop models input and output files, Version 1.0. Technical Report 5. International Benchmark Sites Network for Agrotechnology Transfer, University of Hawaii, Honolulu, HI.
- InelliCorp. 1988. How to get started in AI: IntelliCorp's guide to getting off on the right foot. IntelliCorp Inc., Mountain View, CA.
- Jones, D. D. and L. A. Hoelscher. 1987. Applications of expert systems in extension engineering. ASAE Paper 87-5021. ASAE, St. Joseph, MI.
- Jones, J. W., K. J. Boote, S. S. Jagtap, G. Hoogenboom, and G. G. Wilkerson. 1988. SOYGRO 5.4: Soybean crop growth simulation model-user's guide. Florida Agri. Expt. Sta. Journal No. 8304. Inst. of Food and Agric. Sci.. University of Florida, Gainesville, FL.
- Jones, J. W., Pierce Jones, P. A. Everett. 1987. Combining expert systems and agricultural models: A case study. Trans. of the ASAE 30(5):1308-1314.
- Jones, P., J. W. Jones, P. A. Everett and H. Beck. 1986a. Knowledge acquisition: A case history of an insect control expert system, ASAE Paper 86-5041. ASAE St. Joseph, MI.
- Jones, P., J. W. Jones, P. A. Everett, F. A. Johnson and R. K. Sprinkle. 1986b. Development of an expert system for making insects control decisions in soybeans. Int. Conf. on Computers in Agric. Ext. Prog. Feb 1986. Orlando, FL.

- Jones, P. and J. Haldeman. 1986. Management of a crop research facility with a microcomputer-based expert system. *Trans. of the ASAE* 29(1):235-242.
- Kalkar, S. and P. R. Goodrich. 1986. An expert system for animal waste management. ASAE Paper 86-4540. ASAE, St. Joseph, MI.
- Khuri, R. S., W. D. Shoup, R. M. Peart and R. L. Kilmer. 1988. Expert system for interpreting citrus harvest simulation. *Applied Engineering in Agriculture*. 4(3):275-280.
- Kline, D. E., D. A. Bender, C. E. Van Donge, B. A. McCarl and J.K. Schueller. 1986. Machinery selection using farm-level intelligent decision support systems. ASAE Paper 86-4519. ASAE, St. Joseph, MI.
- Kline, D. E., D. A. Bender, B. A. McCarl. 1987. Farm-level machinery management using intelligent decision support systems. ASAE Paper 87-1047. ASAE St. Joseph, MI.
- Konaka Toshio. 1987. Farm machinery utilization planning. *AMA* 18(3):75-81.
- Kornecki, A. 1988. Simulation and artificial intelligence as tools in aviation education. In: *AI Papers*. ed. R. J. Uttamsingh. Soc. for Comp. Simu., San Diego, CA. *Simulation Series* 20(1):121-126.
- Labiadh S. and J. C. Frisby. 1987. A simulation model to select alfalfa harvesting machines. ASAE Paper 87-1047. ASAE, St. Joseph, MI.
- Lal H. and L. C. Freire. 1984. Operational cost of animal-drawn agricultural implements in various sizes of farm holdings. *Research Bulletin* 21. CPATSA/EMBRAPA, Petrolina, Brazil.
- Lal, H., R. M. Peart and J. W. Jones. 1987. Expert systems for technology transfer. ASAE paper 87-5028. ASAE, St. Joseph, MI.
- Lemmon, H. 1986. Comax: An expert system for cotton crop management. *Science* 233:29-33.

- Link, D. A. 1967. Activity network techniques applied to farm machinery selection problem. Trans. of the ASAE 10(3):310-316.
- Link, D. A. and C. W. Bockhop. 1964. Mathematical approach to farm machine scheduling. Trans. of the ASAE 7(1):8-16.
- Loewer, O. J., M. F. Kocher and T. C. Bridges. 1988. An expert system for determining bottlenecks in on-farm grain processing systems. ASAE Paper 88-6073. ASAE, St. Joseph, MI.
- McKinion, J. M. and H. E. Lemmon. 1985a. Expert systems for agriculture. Computers and Electronics in Agriculture 1(1985):31-40.
- McKinion, J. M. and H. E. Lemmon. 1985b. Symbolic computers and AI tools for a cotton expert system. ASAE Paper 85-5520. ASAE, St. Joseph, MI.
- Meyer, G. E. and R. B. Curry. 1986. Soybean production decision-making with the REALSOY Model. ASAE Paper 86-4507. ASAE, St. Joseph, MI.
- Miles, G. E. and Y. J. Tsai. 1987. Combine systems engineering by simulation. Trans. of the ASAE 30(5):1277-1281.
- Moralee, D. S. 1986. The use of knowledge engineering in an industrial research environment - A retrospect. Expert Systems and Knowledge Engineering, Elsevier Science Publishers, Holland. pp. 101-109.
- Morrison, J. B. 1985. Forthcoming role of artificial intelligence in agricultural science and education system. Executive Summary, Agricultural Communication Service. Purdue University, West Lafayette, IN.
- Moser, J. G. 1986. Integration of artificial intelligence and simulation in comprehensive decision-support system. Simulation 47(6):223-229.
- Nagarajan K., D. Singh and J. K. Wang. 1985. Optimum allocation of farm power resources in multi-crop farming systems. ASAE Paper 85-5056. ASAE, St. Joseph, MI.
- Nagarajan, K., J. W. Mishoe and W. L. Currey. 1987. Development of an expert system for weed management in Soybean. ASAE Paper 87-9659. ASAE St. Joseph, MI.

- Neggahban, B., R. C. Fluck and W. D. Shoup. 1988. Selection of lawn and garden tractors using an expert system. *Applied Engineering in Agriculture* 4(3):211-216.
- Nelson, T. and W. Bowers. 1968. What programs are operational now? *Conf. Proceedings: Computers and Farm Machinery Management*, Dec. 1968. ASAE St. Joseph, MI. pp. 6-8.
- Newell, A. and H. A. Simon. 1963. GPS; A program that simulates human thoughts. In: *Computers and thoughts*. ed. E.A. Feigenbaum and J. Feldman. McGraw-Hill, New York, pp. 279-293.
- Nguyen, T A., W. A. Perkins, T. J. Laffey and Deanne Pecora. 1987. Knowledge-base verification. *AI Magazine*, Summer 1987. pp 69-75.
- O'Keefe, R. 1986. Simulation and expert systems; A taxonomy and some examples. *Simulation* 46(1):10-16.
- O'Keefe, R. M. and J. W. Roach. 1987. Artificial intelligence approaches to simulation. *J. Opl Res. Soc.* 38(8):713-722.
- Oren, T. I. 1986. Knowledge-bases for advanced simulation environment. *Proc. Conf. on Intelligent Simulation Environments*. Soc. for Comp. Simu., San Diego, CA. pp. 16-22.
- Oren, T. I. and B. P. Zeigler. 1979. Concepts of advanced simulation methodologies, *Simulation* 32(3):69-82.
- Ozkan, H. E. and W. M. Edwards. 1986a. A farmer oriented machinery comparison model. *Trans. of the ASAE* 29(3):672-677.
- Ozkan E. and W. Edwards. 1986b. Microcomputer software for machinery management. *Proc. of the Int. Conf. on Computers in Agric. Ext. Prog.*, Orlando, FL. pp. 701-706.
- Ozkan, H. E., W. M. Edwards & A. Saulman. 1984. A machinery selection model for farmer decision-making. *ASAE Paper* 84-1521. ASAE, St. Joseph, MI.
- Paperback Software. 1985. *VP-Planner: Spreadsheet flexibility with database power*. Paperback Software International, Berkeley, CA.
- Pascoe, G. A. 1986. Elements of object-oriented programming, *Byte* 11(9):139-144.

- Peart R. M. and J. R. Barrett. 1979. The role simulation. In: Modification of aerial environment of plants. ed. B. J. Barfield and J.F. Gerber. ASAE Monograph (2). ASAE St. Joseph, MI.
- Peart, R. M., J. R. Barrett and R. D. Smith. 1985. A cropping decision support system using artificial intelligence concepts. Agril. Engg. Dept. Purdue University, West Lafayette, IN.
- Peart, R. M., H. Lal and J. W. Jones. 1988. Developing integrated decision support systems using PROLOG. Presented at the AAAI workshop on AI in Agriculture, Aug., 1988. San Antonio, TX.
- Pepper, J. 1986. Knowledge craft: An environment for rapid prototyping for expert systems. Paper presented at the AI for Automotive Industry Conference. March 1986. Detroit, MI.
- Phillips, J. J. and S. B. Harsh. 1987. An expert system application to the financial analysis of lender case farm records. Paper presented at the Annual Meeting of Am. Agric. Econ. Assoc. Lansing, MI.
- Plant, R. E. and T. L. Wilson. 1986. A computer based pest management aid for San Joaquin Valley cotton. Proc. of the Beltwide Cotton Conf., Nation Cotton Council of Amer., Memphis, TN. pp.22-24.
- Plant, R. E., L. T. Wilson, L. Zelinski, P. B. Goodell and T. A. Kerby. 1987. CALEX/COTTON: An expert system based management aid for california cotton growers. University of California, Davis, CA.
- Prerau, D. S. 1987. Knowledge acquisition in the development of a large expert system. AI Magazine, Summer 1987. pp 43-51.
- Pritsker, A. A. B. 1984. Introduction to simulation and SLAM II. Second Edition. John Wiley and Sons, New York, NY.
- Pritskar, A. A. B. and C. D. Pegden. 1979. Introduction to simulation and SLAM. Systems Publishing Corp. West Lafayette, IN.
- Reddy Y. V., M. S. Fox, N. Hussain and M. McRoberts. 1986. The knowledge-based simulation system. IEEE SOFTWARE, March 1986. pp 26-37.

- Rettinger, J. C., G. E. Miles and G. E. Wilcox. 1987. Muskmelon production expert system. ASAE Paper 87-5029. ASAE, St. Joseph, MI.
- Rich, E. 1983. Artificial intelligence, McGraw-Hill Book Company, New York, NY.
- Rimmington, G. M., R. Berger, D. J. Connor and T. A. McMohon. 1987. Expert systems and crop simulation modelling. Proc. of the Simulation Society Conf., Melbourne, Australia. pp. 343-358.
- Roach, J. W. and R. S. Virkar. 1986. POMME: A computer based consultation system for apple orchard management using PROLOG. A technical report. Department of Computer Science and Applications, Virginia Polytechnic Institute and State University, Blacksburg, VA.
- Ruckman J. 1986. A model for computer integrated farming system. ASAE Paper 86-1519. ASAE, St. Joseph, MI.
- Rumsey, J. W., L. Gautz and W. J. Chancellor. 1986. Increasing farm machinery cost effectiveness in California. ASAE Paper 86-1520. ASAE, St Joseph, MI.
- Shannon, R. E. 1984. Artificial intelligence and simulation, Keynote Address. In. Proc. of Winter Simu. Conf. ed. S. Sheppard, U. Pooch, D. Pegden. Soc. for Comp. Simu., San Diego, CA. pp. 4-9.
- Shaw, L. G. and B. R. Gaines. 1987. A framework for knowledge-based systems unifying expert systems and simulation. In Intelligent simulation environment. ed. Paul Luker and H. Adelsberger. Soc. for Comp. Simu., San Diego, Simulation series, 17(1): 38-43.
- Shintani T., Y. Katayama, K. Hiraishi and M. Toda. 1986. KORE-A hybrid knowledge programming environment for decision support based on a logic programming language. In: Lecture Notes in Computer Science. ed. G. Goos and J. Hartmanis, Logic Programming' 86, Proc. of the 5th Conf., Tokyo, Japan. pp. 22-32.
- Siemens, J. C., K. Hamburg and A. T. Tyrrell. 1988. Farm machinery selection program. ASAE Paper 88-1056. ASAE, St Joseph, MI.
- Slocombe, J. W. 1986. Agricultural mechanization curricula in 2000. ASAE Paper 86-5515. ASAE, St. Joseph, MI.

- Smith, R. D. 1985. Biomass storage, decision support systems and expert systems in crop production and processing. PhD thesis. Perdue University, West Lafayette, IN.
- Smith, R. D., J. R. Barrett and R. M. Peart. 1985. Crop production management with expert systems. ASAE Paper 85-5521. ASAE St. Joseph, MI.
- Smith, O. R., L. D. Gaultney, J. R. Barrett, D. D. Jones and C. H. Castore. 1988. Acceptability of expert systems by agricultural computer users. ASAE Paper 88-5023. ASAE St. Joseph, MI.
- Smith, R. D., R. M. Peart and J. R. Barrett. 1985. Agricultural production management with decision support systems. ASAE Paper 85-3076. ASAE, St. Joseph, MI.
- Sowell, R. S., T. B. Bannett, D. W. Donahue and D. M. Crohn. 1988. Total farm machinery cost analysis with dBASE III and CLIPPER. ASAE Paper 88-1055. ASAE, St Joseph, MI.
- Stefik M. and D. G. Bobrow. 1986. Object-oriented programming: Themes and variations, AI Magazine, Winter 1986. pp. 40-59.
- Stone, N. D., R. E. Frisbie, J. W. Richardson and R. N. Coulson. 1986. Integrated expert system applications for agriculture. Proc. of Int. Conf. on Computers in Agric. Ext. Prog., Feb. 1986. Orlando, FL. pp. 836-841
- Thai, C. N. and C. P. Wilson. 1988. Simulation of peach post harvest operations. ASAE Paper 88-6058. ASAE, St. Joseph, MI.
- Thieme, R. H., D. D. Jones and H. G. Gibson. 1986. A knowledge-based approach to forest road planning. ASAE Paper 86-1606. ASAE, St. Joseph, MI.
- Thieme, R. H., J. W. Uhring, A. Dale Whittaker, R. M. Peart and J. R. Barrett. 1985. Grain marketing analysis with an expert system. ASAE Paper 85-5519. ASAE, St. Joseph, MI.
- Thomson, J. S. and R. M. Peart. 1986. An expert system for soil moisture-based scheduling of center pivot irrigation. ASAE Paper 86-4518. ASAE, St. Joseph, MI.
- Tsai, Y. J., J. W. Jones and J. W. Mishoe. 1987. Optimizing multiple cropping systems: A systems approach. Trans. of the ASAE 30(6):1554-1561.

- Umphress, D. A. and U. W. Pooch. 1987. A goal oriented approach to simulation. In: Methodology and validation, ed. Osman Balsci. Soc. for Comp. Simu., San Diego, CA. Simulation Series 19(1):44-49.
- Von Bargen K. and J. Hines. 1973. Economic Analysis of a complement of farm machines. ASAE paper 73-1529. ASAE, St. Joseph, MI.
- Von Bargen K. and R. M. Peart. 1969. Simulation of field machine and transport activities for a row crop planting system. ASAE Paper 69-657. ASAE, St. Joseph, MI.
- Walker A., M. McCord, J. F. Sowa and W. G. Wilson. 1987. Knowledge systems and prolog; Chapter 1; Knowledge systems: Principles and practices, Addison-Wesley Publishing Co. Inc., Reading, Massachusetts. pp. 1-22.
- Walton, T. F. 1967. Communications and data management. A Wiley-Interscience Publication. John Willey & Sons, New York, NY.
- Waterman, D. A. 1986. A guide to expert systems. Addison-Wesley Publishing Co. Inc., Reading, MA.
- Waund, D. and L. Mundell. 1968. Linear programming model for a vegetable processor. Conf. Proceedings: Computers and Farm Machinery Management, Dec., 1968. ASAE, St. Joseph, MI.
- Webster, N. 1979. Webster's new twentieth century dictionary of the English language. Second edition. Simon and Schuster, New York, NY.
- Whittaker, D. A., E. J. Monke and F. R. Foster. 1987. ADAM: An ADaptive Assembler of Models. ASAE paper 87-5536. ASAE St. Joseph, MI.
- Wildberger A. M. 1988a. Selection of tools for including expert system components in simulations. Paper presented at the 1988 Summer Comp. Simu. Conf., July 1988. Seattle, WA.
- Wildberger A. M. 1988b. Integrating an expert systems component into a simulation. In. AI Papers. ed. Ranjeet J. Uttamsingh. Soc. for Comp. Simu., San Diego, CA. Simulation Series. 20(1):132-135.
- Williamson, M. 1985. Artificial intelligence for microcomputers: The guide for business decision makers. Brady Communication Co. Inc., A Simon and Schuster Publishing Co., New York, NY.

- Yokoi S. 1986. A PROLOG based object-oriented language and its compiler. In: Lecture Notes in Computer Science. ed. G. Goos and J. Hartmanis, Logic Programming' 86. Proc. of the 5th Conf. Tokyo, Japan.
- Yost, R. 1985. Expert system to develop and transfer soil management research. Annual Report and Work Plan. TROPSOIL. Management Entity, Raligh, NC.

BIOGRAPHICAL SKETCH

HARBANS LAL was born on Jan. 6, 1949, in Kashipur (UP) a small town in the Tarai region of northern India. He had his secondary education in New Delhi and attended Indian Institute of Technology (IIT), Kharagpur, WB for his Bachelor and Master degrees in agricultural engineering from 1966 to 1973. He specialized in farm machinery and power as a major in agricultural engineering.

After graduation from IIT, he worked as a full-time volunteer with the Front for Rapid Economics Advancement (FREA), India to organize and manage an agro-service center to its Patoda base self-supporting.

In 1974, he joined the International Crops Research Institute for the Semi-Arid Tropics (ICRISAT), India, where he had the opportunity to work with professionals from different part of the world. As an Agricultural Engineer of the Farming Systems Research Team of the ICRISAT, he designed and developed agricultural implements for the improved farming systems developed at the ICRISAT. The adaptation of a traditional bullock cart to a animal-drawn tool carrier made such a technology within the economic reach of Indian farming community.

He then moved to Brazil in 1979 to work for the Inter-american Institute of Agricultural Science (IICA) as a Mechanization Specialist. He was stationed at the Center of Agricultural Research for the Semi-Arid Tropics (CPATSA) of the Brazilian Enterprise for Agricultural Research (EMBRAPA) and had the responsibility to organize and conduct mechanization program. The W-shaped soil management system, developed as a part of his research efforts, provided a tillage technique to stabilize and increase the production of arid northeast Brazil and other similar regions of the world. His research has resulted in over 20 publications in refereed journals from different parts of the world.

The desire to upgrade his qualifications and to learn modern techniques in agricultural engineering brought him back to school in 1986 for his Ph.D. The University of Florida (UF) was the choice because of its location in a tropical climate and a flexible curriculum in the Agricultural Engineering Department and a strong program in systems research. At UF, he concentrated in system engineering, learned about AI and simulation techniques and applied them to farm systems as demonstrated in this dissertation.

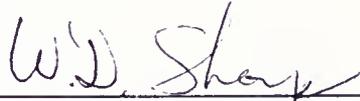
He is married with Chitra and has three daughters Bhawna, Monika and Prerna. He plans to find employment where he can enhance his recently acquired understanding about knowledge and systems engineering and make use of his past experience.

I certify that I have read this study and that in my opinion it confirms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.



Dr. R. M. Peart, Chairman
Graduate Research Professor of
Agricultural Engineering

I certify that I have read this study and that in my opinion it confirms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.



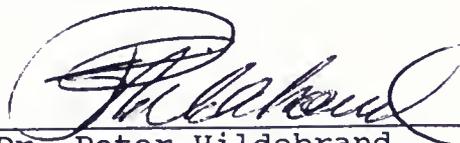
Dr. W. D. Shoup, Cochairman
Professor of Agricultural
Engineering

I certify that I have read this study and that in my opinion it confirms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.



Dr. J. W. Jones
Professor of Agricultural
Engineering

I certify that I have read this study and that in my opinion it confirms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.



Dr. Peter Hildebrand
Professor of Food and Resource
Economics

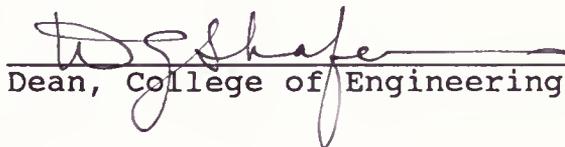
I certify that I have read this study and that in my opinion it confirms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as dissertation for the degree of Doctor of Philosophy.



Dr. Douglas D. Dankel
Assistant Professor of Computer
and Information Sciences

This dissertation was submitted to the Graduate Faculty of the College of Engineering and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December 1989



Dean, Graduate School

#48
075