Irrigation Scheduling Configuration System Based on Expert Systems and Operations Research

Mostafa Mahmoud, **Iman Hassan**
Central Lab for Agriculture Expert System
Agriculture Research Center
Egypt

Mohamed Rasmy, Ahmed Rafea
Faculty of Computer and Information
Cairo University
Egypt

Abstract -*This paper presents configuration of knowledge to solve irrigation scheduling problem in agriculture domain and a general approach for configuration management knowledge throughout the integration between operations research and expert systems approach. This work has established new configuration structure for an Irrigation Scheduling Configuration System suitable for CommonKADS Layers to produce the configuration irrigation scheduling system. This structure supports the knowledge acquisition process and the generation of suitable alternative scheduling systems. This helps the knowledge engineer to build a new simpler, quicker and more effective system, and reusing the already existing components in similar systems. By using the proposed tool, the developers can generate or update their systems to include the growing expertise and techniques. This work is also new in generating multiple suitable alternative scheduling systems in the agricultural domain according to the user requirements, and aids the decision maker to select the appropriate one.*

Keywords: Expert Systems, Knowledge Base, Operations Research, Scheduling, Configuration.

1 Introduction

Recent approaches integrate both the Operations research (OR) and Expert Systems (ES) together under a unified framework to benefit from each one of them in solving the scheduling problems [1]. Since solving these problems need to adopt heuristic techniques as there is no single best approach in OR for solving the scheduling problem, because the quality of the solution depends on many parameters that until now no complete comparison between them can be made. On the other hand most of the already existing irrigation scheduling systems has been built from scratch, often using the same or similar components within different systems. All of them include sophisticated scheduling strategies, but the systems are not as modular, and the components are not as reusable, as they should be.

In order to build an effective configuration irrigation scheduling system, we have analyzed most of the developed systems in the irrigation scheduling to identify the scheduling primitives needed and technical problems [2,3,4,5,6,7,8]. The analysis of these systems has revealed a set of problems to be addressed when designing a scheduling configuration system. During the analysis stage, the tasks and inferences that include problem-solving methods and primitive problem-solving methods, in the irrigation scheduling systems have been identified to form a library. Common domain knowledge, i.e. the part of the domain knowledge that can be reused in other similar agriculture scheduling systems without modification, which considered as generic domain knowledge, has been identified and presented in another library. This work has established new configuration structure suitable for CommonKADS Layers [9, 10]. This structure supports the knowledge acquisition process and the generation of suitable alternative scheduling systems. This helps the knowledge engineer to build a new simpler, quicker and more effective system, and reusing the already existing components in similar systems.

In this work an agriculture irrigation scheduling configuration tool (ISCT) has been developed to configure an expert system automatically based on user requirements.

This tool has been verified throughout building a real system that has been generated from the tool for strawberry irrigation. This process emphasized the capability of the tool to generate an identical system compared with the already existing one in addition to producing numerous alternative systems that use different alternative problem-solving methods. Those alternative systems help the expert to choose the more appropriate configuration components that match his inference and experience. Integer Linear Programming has integrated to determine the suitable configuration to maximize the economic return.

In the following sections, the framework of irrigation scheduling configuration described in section 2, then an agriculture Irrigation Schedule Configuration Tool (ISCT) component is described in section 3. In section 4 the processes of ISCT tool are presented. The evaluation of the tool is presented in section 5, and finally the conclusion is in section 6.

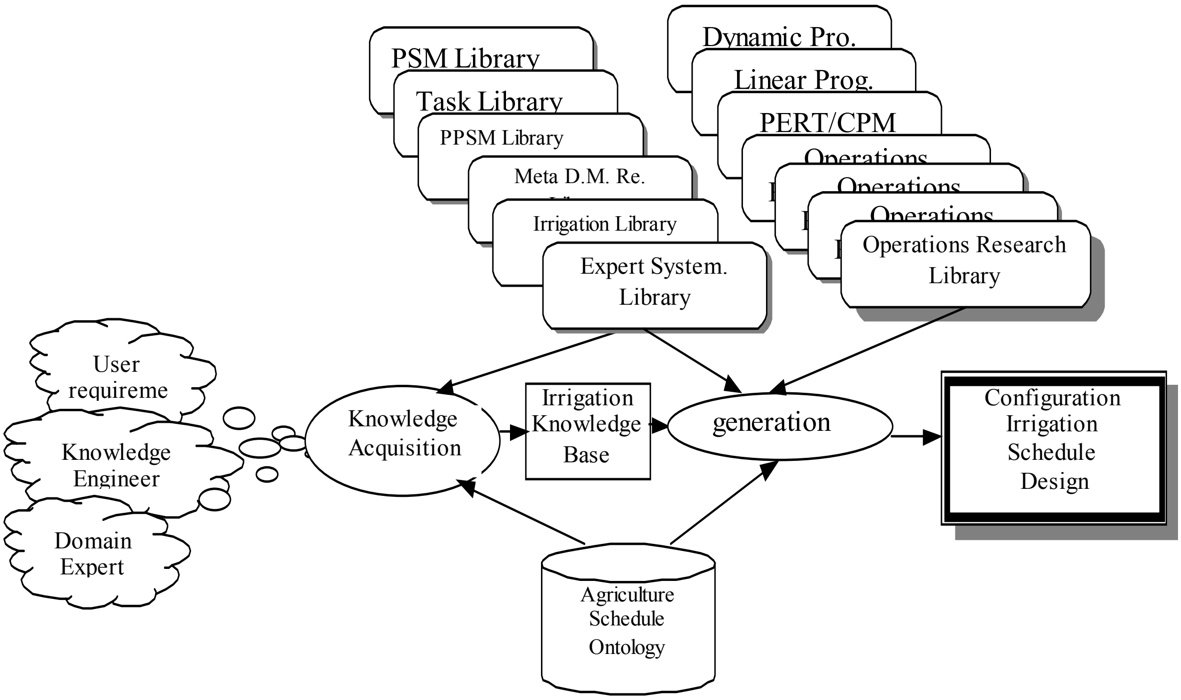

Figure 1: Generating Configuration Irrigation Schedule

2 Framework of Irrigation Scheduling Configuration

The framework of the proposed methodology for an Irrigation Scheduling Configuration system has two main stages: development of infrastructure and generation of agricultural schedule configuration. The infrastructure development consists of control knowledge and domain knowledge. Most of those components were decomposed into other small pieces. The components produced from the first stage will be used in the second stage to construct the framework of the agricultural schedule configuration system.

The scheduling configuration system components consist of five main components, namely, Control Schedule Library, Library of Mechanisms, Agriculture Schedule ontology, Common Knowledge Base, and Case-specific Knowledge Base. Each part of those five main components is a result of the first stage. Consequently, the domain knowledge in the proposed tool has two main components: agricultural schedule ontology and domain model. The domain model itself is classified into common and case-specific knowledge base. The final product is the configuration schedule system as a result of the generation process.

3 Irrigation Scheduling Configuration Tool (ISCT) Components

The Irrigation Schedule Configuration Tool (ISCT) consists of three main components, namely Agriculture Schedule ontology, OR library, and expert systems library. This tool produce one intermediate result called 'irrigation knowledge base' through the Knowledge Acquisition process, and it generate a configuration irrigation schedule design through the generation process as depicted in figure 1. The following subsections describe each component of the three main components in addition to the intermediate result and the final product of the tool.

3.1 Agriculture Schedule Ontology

An ontology for a knowledge based system is an explicit specification for the objects, concepts and other entities that are presumed to exist in some area of interest as well as the relationships that hold among them [11].

In ISCT, agriculture schedule ontology classified into five parts: resources, irrigation domain ontology (activity), irrigation schedule (product), user requirement (demand), and constraint. Resources are an entity that supports or enables the execution of activities [12]. Resources are generally in finite supply and their availability constraints when and how activities execute. In ISCT, we conceptualize the resource as consisting of objects (i.e. soil, water, climate, farm, previous crop, and plantation) that are elementary entities such as integers and string (i.e. soil salinity, soil texture).

The activity is a set of processing steps required to produce or provide the product (irrigation schedule). It is the set of activities that fulfill the demand. The Sequences of activity are expansion, Et0, EtCrop, SHWC, Water Requirement, and Frequency/Interval to achieve the user requirement. Irrigation domain ontology (domain model) defines declaratively the set of terms and relations of a domain independent of any problem solving method. The domain model contains static information about the application environment.

3.2 Operations Research Library

Integer linear programming that included in operations research library is used to select the suitable configuration that maximizing the return. Integer linear programming formulation that used described as follows:

Maximize yield $R=\sum_{i=1}^{n} C_i X_i$ i= 1,2,3,…..,n

Subject to

$$\sum_{i=1}^{n} w_{ij} X_i \le L_i \quad \forall \quad j= 1,2,3,\ldots,9,\ i=1,2,3,\ldots,n$$

$$\sum_{i=1}^{n} W_i X_i \le LR \quad \forall \quad i=1,2,3,\ldots,n$$

Where

X_i Є {0,1} irrigation schedule system generated by the agriculture schedule configuration system of n alternatives irrigation schedule system.

$C_i \geq 0$ expected yield for each of n alternatives irrigation schedule system

$w_{ij} \geq 0$ water irrigation quantity in plant growth stage period j of nine irrigation periods, in i irrigation water schedule system from n alternatives irrigation schedule systems

$Lj \geq 0$, necessary water irrigation quantity for the plant in each growth stage period j of nine irrigation periods

$W_i \geq 0$, total irrigation water schedule of n alternatives irrigation schedule systems

$LR \geq 0$, necessary water irrigation quantity needed for the plant in all season

3.3 Expert Systems Library

Expert system library is a collection of problem solving method (PSM), Tasks, primitive problem solving method (PPSM), Meta domain knowledge representation, and irrigation libraries. The roles that need to be filled by a PSM prescribe what domain knowledge is expected from the expert. The domain knowledge is, therefore, no longer something that is acquired and represented independently of how it will be used in concrete problem solving. However, it is still explicitly and separately represented, giving the advantage of maintainability and systematic ness in knowledge acquisition. This library contains 21 PSMs irrigation schedule

The task reflects the general problem solving method for scheduling. A problem solving method can be defined as a knowledge-use-level characterization of how a task might be solved [10]. It can also be seen as an abstract model of how to solve certain problem. In irrigation schedule design, a generic problem solving method called "propose and revise" is based on mathematics model and heuristic model. Irrigation Task library contains 5 tasks and 57 sub-task for irrigation schedule. Irrigation is the main task of the irrigation generates design. This means that there are many alternative systems (hundreds of alternative systems) that can be constructed according to either the user needs or the knowledge availability. The goal of the irrigation task is to get irrigation schedule of the case description. Its inputs are the case description. Its output is the irrigation schedule of the case description.

4 Irrigation Schedule Configuration Processes

There are two main processes in the ISCT tool, the first one is a knowledge acquisition, and the second is generation as shown in figure 1. The first process produces the irrigation knowledge base, and the second one produce the irrigation schedule design. The next two subsections describe those processes.

```
task: Irrigation schedule ,
  task-definition:
    goal: the main goal of the irrigation is to determine the water requirement during
          cultivation.
    input: case-description
    output: irrigation schedule
  task_body
    type: Composite
    sub_tasks: Propose mathematic  irrigation  schedule
    control_structure:
              Propose mathematic  irrigation  schedule.
task: Propose mathematic  irrigation  schedule
  task-definition:
    goal: Generating an propose irrigation schedule
    input: case-description
    output: Propose irrigation schedule
  task_body
    type: Composite
    sub_tasks: Get case description , determine plant parameters,
               Initialize irrigation parameters,   Compute propose
               irrigation schedule,    irrigation units,
               Display irrigation  schedule
    control_structure:
               Get case description ,
                case description, expansion model -> expanded case  description,
                Initialize irrigation parameters,
                Compute propose irrigation schedule,
                 water requirement, unit model -> irrigation unit,
                  Display irrigation schedule.
```

Figure 2: task knowledge of irrigation

4.1 Knowledge Acquisition

The irrigation knowledge base is the result of the knowledge acquisition process. In the knowledge acquisition process, the knowledge engineer collecting the user requirement and eliciting the knowledge that used in agriculture schedule ontology and the expert system library by using a special knowledge acquisition tool that consists of five parts crop parameters, Evapotranspiration (Et0) method, EtCrop methods, Available water in soil methods, and water requirement methods. After the knowledge engineer collecting the knowledge relate to crops, the domain expert uses this tool to verify the crop parameters.

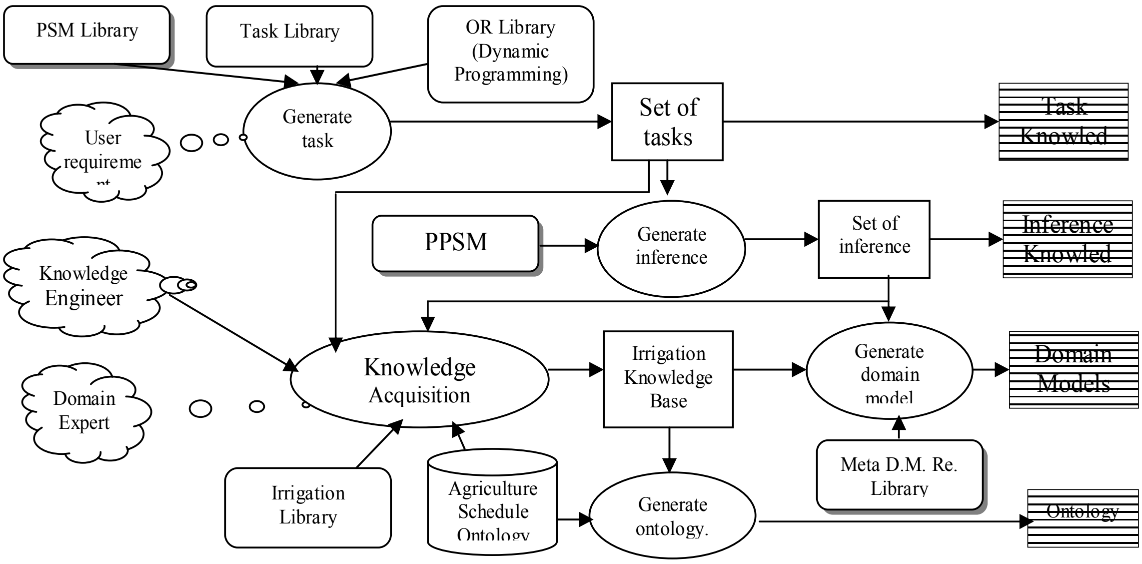

Figure 3: Detail Process of Generating Configuration Irrigation Schedule Design

4.2 Generation

The generation starts with the requirements definition where several questions are used to guide the generations of the irrigation schedule design for any crop. The criteria for generating design describe by the property (Attribute) and the configuration design for irrigation system is used to collect data on:

- o Generation model (e.g. mathematical, heuristic...)
- o Schedule type(daily, weekly, each-10-day, monthly)
- o The objective of irrigation management (i.e. maximize net return, maximize yield)
- o Scheduling problem solving containing propose and revise (i.e., propose, revise)
- o Yield response to water (i.e. forecasting expected yield)
- o Evaporation method (Et0 Penman, Eto modified Penman, Et0 Penman Mountance, Et0 Hargrve, ET0 Blaney, ET0 Radiation, ET0 Pan evaporation, Et0 Reference Sector, Eto Reference Governor)

The generation process is decomposed into four processes namely: generate tasks, generate inference, generate domain model, and generate ontology to produce the irrigation schedule system that contains four parts: tasks and sub-tasks, inferences, domain model, and ontology as shown in figure 3.

5 Evaluation

Integrating Operations research and Expert Systems under unified framework in agriculture domain is very important

This section aims to evaluate the introduced methodology for Irrigation Scheduling Configuration Tool and to show how it meets the requirements of the user through applying the proposed tool on different requirements of scheduling. In order to demonstrate the applicability of the presented architecture, it was applied on an already existed expert system for Strawberry Scheduling Irrigation system to show how the system is formed and executed. A sample run of the expert system produced by the tool compared with the already existed system is described in section three. In the following subsection a comprehensive discussion is provided to show the tool results under different user requirements and using evaluation criteria integrated in the tool to help the decision maker to select the more suitable configuration system from the alternative ones.

5.1 Experiment 1: Generate already existed system

The proposed architecture tool has been used to generate an already existed expert system for Strawberry Scheduling Irrigation system [5]. The objective of developing this system is to evaluate the capability of the tool to produce the same system with a little effort and comparing the two versions of the same system, and to demonstrate the final result for the alternative systems, also to assist the knowledge engineer and the domain expert to develop such system rapidly using this new facilities.

5.1.1 Strawberry Scheduling Irrigation System Description

CLAES[1] developed a version of Strawberry irrigation [5]. This version uses Modified Penman Method, EtCrop based

[1] Center Laboratory for agriculture expert systems, ARC, MOLAR. Egypt.

on Et0 and crop coefficients (KC), SHWC based on field capacity and permanent wilting point, and Leaching requirement method. Strawberry irrigation scheduling for open field determines the water requirement need for strawberry grow up during all growth stages under the available resources. Notice that the expert system package of irrigation strawberry implemented by KROL [13] has fixed task and inference step, it includes domain models expansion and computational model and the representations of stored knowledge are rules, tables and functions. The inputs are parameters related to: farm, soil, water, and climate. The output of the strawberry system is a schedule contains: week number, irrigation date, water quantity, interval between irrigation, motor work hours, and frequency of irrigation during day.

The problem solving method used in the system called “propose-revise” for solving strawberry irrigation schedule. The task includes two transfer tasks, one composite task "propose" and one primitive task "revise". The propose task consists of one primitive task "expand" and one composite task "irrigation weekly schedule". The inference structure includes three inference steps: expand, propose, and revise. The objective of the expand inference is to use known data to derive new ones using a set of relations forming the expansion model. The propose irrigation schedule compute the initial irrigation schedule using a set of function, table and relation from the computational model. Finally, the revise is adjusting the initial irrigation schedule.

5.1.2 Building Strawberry Irrigation using ISCT

In order to configure the task and inference knowledge for irrigation strawberry schedule, consult the configuration part of ISCT tool which is an expert system that produces the task and inference in CLIPS format [14]. Throughout four steps, we get the similar version of expert system for strawberry irrigation.

- **Step 1 : collect the user requirements**

The first step after running the configuration expert system is to collect the user requirements through the intelligent interface that appear to the user according to the current situation of the system. A set of questions including farm type [open field/ high tunnel/low tunnel], cultivation of crops [open field, high plastic tunnel, low tunnel], crop [crops/ vegetable / horticulture] , irrigation model [mathematical / heuristic], and so on.

- **Step 2 : configure task and inference**

The second step is the configuration process to generate the inference and task. It stores the results into 2 deferent files in CLIPS format. as per the CommonKADS methodology. After using our tool, providing a specific data input, we got the same configuration system that matched with this version. Consequently, by generating that version we had two similar versions of the strawberry irrigation scheduling system. One of them is the already existed system in CLAES, and the other is that generated by the tool.

- **Step 3 : integrate domain model with the task and inference**

The third step is to combine the domain model according to inference components with the result of step two. Therefore, the system is completed and ready to run under CLIPS environment.

- **Step 4 : running the system**

After generating all components of strawberry expert system as described in the three stages above, running the strawberry expert system under CLIPS environment was made successfully.

5.1.3 Comparing Results

In order to verify the results of this system and compare the same with other results of the expert system package of strawberry already implemented by other expert system tool called KROL, twenty experiments have been made; there is no significant difference between the two results.

5.2 Experiment 2: Configuring Several Strawberry Irrigation Scheduling Systems by ISCT

The tool can configure a lot of expert systems for strawberry irrigation using different components of schedule library, providing the tool with different inputs. The approach followed to generate strawberry irrigation system based on mathematical model and experiment.

5.2.1 Description of the different Configuration

There are many alternative approaches to calculate the following different methods:

- *evapotranspiration(Et0) (*there are six methods to calculate *it)*
  - Penman Method,
  - Modified Penman Method,
  - Hargerve Method,
  - Radiation Method,
  - Pan Evaporation Method, and
  - retrieval Eto according to sector, or governorate.
- *crop evapotranspiration (EtCrop), (two approaches each one has two methods)*
  - based on Et0 and crop coefficients (KC) together,
  - based on Et0, KC and green cover area together

Notice that the source of KC may be experiments or references

- *total available soil water (SHWC) (three methods, but their are two different sources of data)*
  - SHWC based on soil type,
  - SHWC based on field capacity and permint wilting point, and
  - SHWC based on soil bulk density and soil saturated percentage that used root depth (rd) and the rd data get from experiments or references.

Notice that the source of rd may be experiments or references in table or by equation

- *leaching requirement.*
  - Leaching requirement has three methods but there result not significant because there are limited on the quantity of leaching.

By using a combinations of the above methods, we can permuting one of 6 methods for calculating Et0, one of 4 methods for calculating EtCrop, one of 3 methods for calculating SHWC, and one of 2 methods for calculating rd. The combination total number will be 144 permutations, each one of them constructs a new expert system for strawberry irrigation. Consequently ISCT tool used to configure several hundred versions of expert systems if all the necessary data are available. In actual case, by using this configuration tool, the user can provide the system during the configuration process with data and knowledge that can configure more than one solution. In this case, other criteria are to be applied to help the decision maker select the appropriate solution. An integer linear programming model is integrated within the tool to handle this exception.

The aim of this experiment is to evaluate the different solutions as a result from the different configurations systems.

Step 1: Running the ISCT tool to produce a configuration irrigation task and inference, through providing it with the necessary input data to generate a configuration system..

Step 2: Combine the domain model according to inference components with the result of step one.

Step 3: Running the expert system and save the irrigation schedule results.

Step 4: Repeat step 1 to 3 (20 times in this experiment) with different data input with another problem solved method

Result analysis of step 4: there is no two similar cases has the same water quantity. The range between the small quantity (980) and the large quantity (3984) based on:

- *The accuracy is increased upon increasing the number of parameters and calculation.*
- *The size of knowledge increases based on the degree of accuracy.*

5.2.2 Measuring the Result of the Different Configuration Systems

By applying the mathematical model in ISCT library for the above result in order to get the expected yield, the quantity of yield to water supply is quantified through the yield response factor (ky) which relates to the relative yield decrease (1-Ya/Ym) to the relative evapotranspiration deficit (1-ETa/ETm).

$$(1\text{-}Ya/Ym) = ky\ (1\text{-}ETa/ETm).$$
$$Ya/Ym = (1 - ky\ (1\text{-}ETa/ETm))$$
$$Ya = Ym * (1 - ky\ (1\text{-}ETa/ETm))$$

Where

Ya : actual yield

Ym: maximum yield level of a crop (Ym) is primarily determined

Ky : yield response of crop to water according to growth stage or all season

The graph represented in figure 4 showed the relations between the irrigation water quantity and the expected yield for each method.

5.2.3 Finding Optimal Configuration System

There is a relation between the irrigation water quantity and the expected yield, yet there are some cases in which we can expect the same amount of yield with different water quantity. In this case some other criteria can be used to

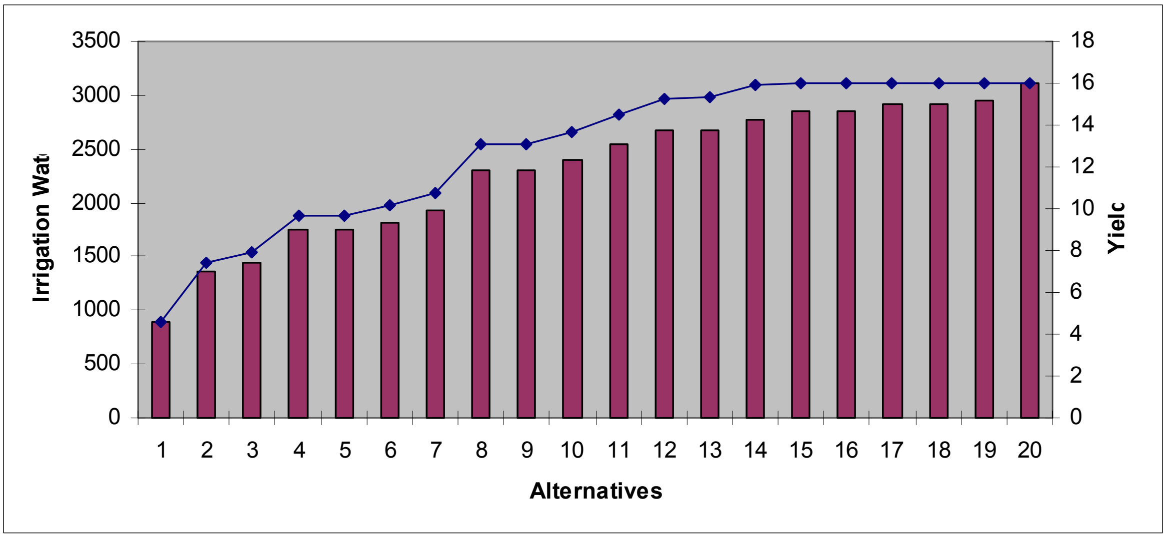

Figure 4: **Irrigation Water Quantity and Expected Yield in each Case**

assist the decision maker to take the right decision as described below. To identify the optimal alternative, Integer Linear programming model has been applied.

The irrigation of strawberry crop has nine periods, each period the plant should be irrigated with a specific amount of water; the suggested irrigation schedule should satisfy this condition, and otherwise the yield may be decreased or damaged. Consequently by applying the integer variables linear programming to formulate those condition we get the optimal solution. The objective function is to maximize the yield, and satisfy ten constraints. The first nine constraints related to the needed irrigation water for the nine growth stages of the plant, each constraint represent one of those nine plant growth stages. The last constraint represents the needed irrigation water for all the season.

Applying the above model that described in section 3.2 on our problem, the result after solving the problem by Integer Programming reveals that the alternative number seven is the optimal irrigation schedule.

6 Conclusion

Integrating Operations research and Expert Systems under a unified framework in agriculture domain is very important task specially in scheduling system like irrigation system to generate a configuration system to include these two approaches. This paper presents a methodology for building an effective configuration agricultural scheduling system. This system provides a hierarchical representation of the problem solving methods, tasks and primitive problem solving methods (inference step). It also contains generic domain knowledge, i.e. the part of the domain knowledge that can be reused in other similar agriculture scheduling systems without modification. This system also helps the knowledge engineer to build a new simpler, quicker and more effective system, and reusing the already existing components in similar systems. The developers can generate or update their systems to include the growing expertise and techniques. It also generates multiple suitable alternative scheduling systems in the agricultural domain; those alternative systems help the expert to choose the more appropriate configuration components that matches his inference and experience.

7 References

[1] Wayne L Winston (2004). Operations Research, Applications and Algorithms. Duxbury Press, 4th edition, 2004.

[2] CLAES (1999). Generic Irrigation Design Applied in Tomato Crop. Central Lab of Agricultural Expert Systems, TR/CLAES/72/99.5, May 1999.

[3] CLAES (2000). Amendment of Generic Irrigation Design Applied in Melon Crop. Central Lab of Agricultural Expert Systems, TR/CLAES/163/2000.10, October 2000

[4] CLAES (2001a). Amendment of Generic Irrigation Design Applied in Bean Crop (BEANEX). Central Lab of Agricultural Expert Systems, TR/CLAES/190/2001.1, January 2001.

[5] CLAES (2001b). Strawberry Irrigation Design (STRAWBEX). Central Lab of Agricultural Expert Systems, TR/CLAES/191/2001.1, January 2001.

[6] CLAES (2001c). Design of Irrigation System for Citrus Expert System (CITEX4). Central Lab of Agricultural Expert Systems, TR/CLAES/228/2001.10, Oct. 2001.

[7] CLAES (2003). Irrigation Design Expert System for Mango (MANGEX). Central Lab of Agricultural Expert Systems, TR/CLAES/268/2003.7, July 2003.

[8] Mahmoud M., El-Araby K., Rafea, A. (1995). LIMEX: An Integrated Expert System for Lime Crop Management. 2nd IFAC/IFIP/EnrAgEng workshop on Artificial Intelligence in Agriculture, 29 May - 1 June, 1995, The Netherlands.

[9] Breuker, J., and Van de Velde, W. (1994). CommonKADS Library for Expertise Modeling: Reusable Problem Solving Components. IOS-press, Amsterdam, August 1994.

[10] Wielinga, Bob J. (1994) Expertise Model Definition Document, ESPRIT Project P5248 KADS-II, Document Id.: KADS-II/M2/UvA/026/5.0, University of Amsterdam, 1994.

[11] Gruber T.R., (1993). A translation approach to portable ontology specifications, Knowledge Acquisition. 5 199–220.

[12] Yu, B. and MacCallum, K. (1995). Modelling of Product Configuration Design and Management by Using Product Structure Knowledge. Int. Workshop on Knowledge Intensive CAD, Finland, 1995.

[13] Rafea, A., Edree, S., El-Azhari, S., Mahmoud M., (1994). A Development Methodology for Agricultural Expert Systems Based on KADS, Proceedings of the Second World Congress on Expert Systems, January 1994.

[14] NASA (2002). CLIPS v. 6.2 Lyndon B. Johnson Space Center, Information System Directorate, Software Technology Branch, Houston, Tex.