ARO Report 94-1

TRANSACTIONS OF THE ELEVENTH ARMY

CONFERENCE ON APPLIED MATHEMATICS

AND COMPUTING

Sponsored by

The Mathematical and Computer Sciences Division of the

ARMY RESEARCH OFFICE

IMPLEMENTING MIXED CHAINING IN A CLASSIFICATION TYPE EXPERT SYSTEM.

Andrew W. Harrell

U.S. Army Engineer Waterways Experiment Station,

Vicksburg, MS 39180

ABSTRACT. Because of the general to specific nature of the backward ordered reasoning (from goals to input data) in some expert systems, it is hard to organize sets of rules that lead to multiple goals. In classification type expert systems, in particular, it is often difficult to organize the rules. Normally it is desired that under all circumstances they ask for all the information required. They should then conclude with a report which contains all the conclusions the system should reach in this situation. In this project, an auxiliary computer program was written to topologically sort the 120 rules in the knowledge base of an expert system. The conclusions of the rules were used as the means by which to define a partial order of the logic flow in the knowledge base.

Key words - Expert System, mixed chaining, knowledge base, topological sort.

INTRODUCTION. Generic categories of expert systems applications include decision management, diagnosis /troubleshooting (determining malfunctions from symptoms and other observable facts), classification and interpretation of situations (concluding situation descriptions from the data and facts encountered), planning and scheduling analysis, manufacturing design, configuring objects under constraints, instruction and intelligent documentation, configuration design, and process control (programs to govern the overall behavior of systems).

In 1989 the US Army Engineer Waterways Experiment Station (WES) established a research and development work unit within the Civil Works Research and Development Program's Flood Control Channels Budget Package entitled "Gravel and Boulder Rivers" (#32553). This effort has two major goals: the first being to develop an understanding of the physical sedimentary processes in rivers and streams, the second being to develop a conceptual model of these processes. An initial stream reach inventory form was developed and validated during 1989-1991. Based on the data gained by a nationwide inventory conducted by MCI Consulting Engineers, INC. for WES, a lack of understanding of and data for boulder/gravel systems became apparent. Work was done to:

a. Establish a systemic procedure for collecting and analyzing geomorphological, geometric, hydraulic, and sedimentary data using a stream reach inventory process.

b. Identify sediment sources and deposition zones.

c. Identify channel bed and bank forms which are hydraulic influencers.

d. Relate channel processes to channel features and link the sedimentation patterns to river engineering factors.

Efforts were conducted to develop technical guidance documents for use by District personnel. The end product was envisioned to be the basis for uniform data collection methods for boulder/gravel river systems. As a result of work conducted on this project in 1992 an existing set of separate stream bed channel flow rules was organized into a classification type computer expert system by the author using the methodology explained in this report.

This paper describes a knowledge based expert system entitled CHANNEL-FIX. The program is intended to serve the hydraulic engineer as a Boulder/Gravel River sedimentation analysis tool. CHANNEL-FIX provides guidance in the fluvial geomorphic processes occurring in a Boulder/Gravel river reach linked to 5 of the 6 stream channel design variables.

In terms of the description of how the rules are implemented, the scope of this study is limited to a particular version of the expert system software shell used (Level5 ver 1.1 for the MacIntosh). The general procedures to be explained in this report are applicable to this type of software expert system shell in general but the specific syntax and grammar of the rules in the knowledge base and system specific functions will be different for other shells.

Since the stream bed flow expert system program falls under the type of expert system used for classification and interpretation of situations some of the specific characteristics for expert systems in these areas will be briefly described below.

Classification expert systems help the user to choose products, procedures, or processes from a large or complex set of alternative possibilities. These programs identify a hypothesis based on the pattern of data that the user enters in response to a series of questions. Since the questions are asked in response to a set of presupplied hypotheses (that is they are framed and scheduled from the general to the specific) these systems are basically backward-chaining. However, as will be explained below, in some situations the information that accumulates as the data is entered may influence the order in which the questions should be asked. To take account of this the knowledge base and inferencing strategy also need to continue accumulating information even after each partial conclusion is reached. This

may require the expert system to start back through the rules again or iterate repeatedly by forward chaining through the rule sets several times.

The following short glossary defines some of basic terms that will be used in explaining the problems that arose in designing the knowledge base for the expert system shell:

Terminology

A short list of some basic expert system terminology is listed below [1]:

Attribute -- Defines the qualities or values contained in a class and the type of information that make up a class. For example, the class car can have the attributes "type of engine" and "top speed".

Attribute value -- An actual number or confidence factor representing the degree of certainty with which a factor is known.

Backward-Chaining -- An inferencing strategy that is structured from the general to the specific. That is, it starts with a desired goal or objective and proceeds backwards along a series of deductive reasonings while it attempts to collect the hypotheses required to be able to conclude the goal. This process continues until the goal is reached and it then displays its conclusion. (See following sections for a more complete explanation and an example.

Class -- Defines the structure (in terms of its attributes) and behavior(in terms of its associated methods and procedures) of an <u>object</u>. When it becomes an <u>instance</u>, it then holds the actual data values of a particular realization of this type of object in the knowledge base. For example: a class called human beings might have attributes related to the parts that differentiate our physical beings and categories such as those related to its our mental and spiritual capacities. Some of the associated methods and procedures of this class could be thinking, talking, walking. It can be considered as a subclass of another class such as the class of living beings. The author and the reader are both specific <u>instances</u> of a human being <u>object</u>.

Antecedent -- The IF part of a conditional statement (synonymous with the term hypothesis in what follows).

Consequent -- The THEN part of a conditional statement (synonymous with the term conclusion in what follows).

See also, the Level5 object for Windows Users guide, Clips users manual, and a guide to expert systems by Waterman all of which are listed in the bibliography at the end of the report.

Control Rule -- A rule in the knowledge base that controls the order in which data is assimilated into the knowledge base.

Goal --- A top-level consequent of the rules in the knowledge base toward which Backward-Chaining may be directed. (It is a hypothesis that the program will try to determine if some group of rules can be instantiated together to satisfy)

Inference Mechanism -- The component of the expert system shell responsible for using the rules in the knowledge base to derive new facts from known information.

Instance or Instantiation -- Specific occurrence of an object. An object consists of its class structure, which defines its attributes and behavior and its instances, which hold the actual values of the object. An instance of the class human beings mentioned above would refer to an individual person, such as the reader of this report.

Knowledge Tree --- A graph showing the logic and data flow connections between rules and facts in the knowledge base. A knowledge tree presents a graphical representation of the complete structure of the knowledge base.

Method --- A procedure stored in an object's class structure that can determine an attribute's value when it is needed in the program , referenced in its class, or required to execute a series of procedures because another value in the program changes. "When needed methods" are executed during backward chaining to determine an attribute's value. "When changed methods" implement a procedure when a given attribute changes.

Node --- A vertex or point in the knowledge tree connecting the antecedents and consequents of rules in the knowledge base. In most conventions the nodes are the rules and the antecedents and consequents are the edges between the nodes or vertices.

Object -- General term for a programming entity that has a record type data structure along with attribute values and procedures that enable it to represent something concrete or abstract. It can be contrasted with other programming entities such as facts, rules, procedures, or methods. An object's structure is defined by its class and attribute definitions. A class declaration is a data template involved in representing knowledge which defines the structure of an object. For example, in the class "human being" mentioned above some of the attribute slots might be size, weight, hair color, and so forth.

Expert System -- A computer program that represents and uses expert human knowledge to attain high levels of performance in a problem area. An expert system has two basic components: a knowledge base which contains the information (facts, rules, and methods) found in the subject area of the problem area being represented, and an inference engine or mechanisms that make use of the knowledge base (by scheduling and interpreting the facts, rules, and methods) to make conclusions and decisions and solve problems that would normally take a human expert more effort.

Expert System Shell -- The interactive programming environment on the computer into which the user enters information, rules, and goals and which compiles the knowledge base, then runs the resulting expert system program.

Forward-Chaining -- Forward-chaining reasoning is an inferencing strategy in which the questions are structured from the specific to the general. That is, it starts with user supplied or known facts or data and concludes new facts about the situation based on the information found in the knowledge base. This process will continue until no further conclusions can be reached from the user supplied or initial data (using the rules and methods contained in the knowledge base). (See following sections for a more complete explanation and an example).

Vertex --- Same as node. (See above)

EXAMPLE OF HOW THE RULE-BASED SYSTEM CAN CLASSIFY THE PLANIFORM STABILITY OF A REACH IN AN ACTUAL STREAMBED

The WES CHANNEL-FIX rule based system contains about 80 rules, 101 facts in 950 lines of computer code for a MacIntosh personal computer. Once the program is started on the computer, a screen appears which explains the system which is driven by graphical menus and buttons. The user enters information into the program by either clicking buttons on the screen with a cursor directed by a keyboard mouse or by typing text from the keyboard in order to answer the questions that appear on the screen. Certain menu choices or questions in the program are preceded by explanatory pictures on the screen. These pictures give the user a graphical explanation of some of the menu choices that are displayed. Also, when the explain button appears above the question area on the screen window, the program will display explanatory text when this button is clicked. When the system has asked all the questions that it needs to determine which rules and facts may be applicable to the situation the session will occur a summary of all conclusions and determinations will be printed out on the screen and saved in a file in the program's directory work area. At any time during the series of questions that the program makes a partial conclusion the user may click on the explain button to see displayed which rules and facts were used to make that particular conclusion.

As an example of the steps involved in using the program we will display the questions and determinations for a session in which the user enters the information for a reach in the North Fork Licking river. Normally a reach of the river would be determined from the data from several cross-sections at the site. After the initial screen appears the next step in the program is normally initiated by clicking on the continue button that appears above the question area in the program's screen window.

a. SAMPLE PROGRAM RUN CROSS SECTION #! NORTH FORK LICKING RIVER

(1) The program asks for the name of the river which the user enters as **North Fork Licking** in this case.

(2) The program then asks for the type of bar that is present in this reach of the river survey. In this case the user responds: **Point Bar.**

(3) The program then asks for the active channel width[2] in feet. This is entered as **75.**

(4) The program then asks for the slope of the river bed and the water surface slope at this point in the stream. The answers entered in this case are: **.01 and .02.**

---

[2] See the reference Harrell[1993] for the definition of the hydrologic terms used in this example.

(5) The program then asks the user whether fines are present on the bar surface. The answer is **Yes** in this example.

(6) The program then asks whether large clasts are in direct contact at this point in the reach. The answer is **No** in this example.

(7) The program then asks whether imbrication is present. The answer give is **Yes.**

(8) The program then asks whether you can identify the evidence of fresh scour on the outside bank. The answer given is **Yes.**

(9) The program then asks whether there are fresh deposits on the bar. The answer given is **Yes.**

(10) The program then asks what is the average depth of the active channel. The answer given is **2 ft.**

(11) The program then asks whether there is fresh scour on the bar. The answer given is **No.**

(12) The program then asks whether there are diffuse gravel sheets. The answer given is **Yes.**

(13) The program then asks whether the Main Channel is increasing, stable, or decreasing. The answer given is **increasing.**

The program then concludes the session and prints out a screen displaying all the conclusions reached. This information is shown below:

Based on your description of this reach of North Fork Licking River, the
following conclusions can be drawn:

The sedimentary structure of the bar is Matrix Gravel
Large clast are not typically in direct contact in Matrix gravels.

The matrix consists of 30% or more sediment finer than fine gravel. Fluvial action will rapidly entrain the matrix sediment reducing the stability of the gravel clast.

This erosional process occurs at mean flow or higher. Field data indicates the even burial of clast to 75% does not increase stability.

Tractive or shear stress produced by mean flow will entrain the matrix or finer grain sediment. The lack of clast interlocking that is present in framework gravel reduces the stability although there is a high per cent of fine grained matrix material.

Matrix gravel units appear to be a grouping clusters. This lack of stability and high erodibility factor leads the assignment of a stability rating of 4.

The relative stability of the bar (from 1 to 4) is: 4.00
The Active Channel Width is probably increasing

```
The slope is:increasing
The meander pattern is:decreasing
The stability of the channel is:decreasing
Conclusion:  The channel is migrating to the outside
The bank is providing transported sediment.
The sediment transport is:increasing
The Main Channel Depth is stable
The stability of the planiform is:     + 2.00
where +1 = change/increase
       0 = neutral
      -1 = change/decrease

The bank is eroding:           True
The bar is eroding:        False
The bar is migrating:      False
The bar is growing:        True


Report of conclusion for North Fork Licking River complete.
***End of session***
```

Note, that in this example the program had to ask the question whether the Main Channel was increasing, stable, or decreasing. For another set of reach information it is possible that the rule-based system would have been able to determine this from information already entered. In general, there are not enough rules to determine all the conclusions that may be required in order to proceed completely with any given set of facts. The program will then request the user to supply the answer to the missing information. The purpose of sorting the rules as explained earlier in the report in terms of the information required in the hypotheses of each rule is that the program will in all cases be able to proceed in a single program run in a manner which extracts all the information required to make all possible determinations that the rule-base will permit.

If we examine the stream bed flow program we see that it is a rule-based system which collects or makes a report of a series of conclusions, not just one. Therefore it does not fall in the area of backward goal searching diagnostic programs in which the questions are structured from the **general to the specific.** It is a forward chaining rule-based system in which the information is accumulated by asking a series of questions which are structured from the **specific to the general.**

The order of the goals in the program was restructured in order to make it consider all the rules in a single program run.

An abbreviated forward chaining flow chart for stream bed flow rule system as it now exists is illustrated below:

step 1

Determine the river name
Determine the type of bar
Determine the active channel width

step 2

Determine bar composition

step 3

Determine bar stability
Determine if the bank and/or bar is eroding or not
Determine channel depth
Determine channel slope
Determine if the bar is migrating or not

step 4

Determine if the active channel width is changing
Determine if the main channel depth is changing
Determine if the bar and/or the bank is providing transported sediment
Determine if the bar is eroding faster than the bank

step 5

Draw conclusions about the present state of the width, such as a point bar is forming
Draw conclusions about the effect of the bar on the active channel width
Draw conclusion if deposition is occurring on the inside of the bend
Draw conclusions about the affect of increasing channel width on slope
Draw conclusions about the affect of diffuse gravel sheets on slope
Draw conclusions about the affect of slope on channel stability

step 6

file a report of all the information entered and conclusions reached

The reason for this organization comes from both the way the software is written and the type of knowledge base that we want to create. The rest of the report will further elaborate on the organization and explain how it was arrived at.

The version of LEVEL5 that was used for this study is a backward chaining (goal driven) PROLOG[3] [4]type expert system shell based on predicate calculus. It provides a good graphical user interface, built-in database search predicates, and some object- oriented features. For classification problems which are data- driven and for which you need to record everything that can be determined about the situation, a forward- chaining LISP[5] [6] or CLIPS[7] type system with more object- oriented features is better.

---

[3] "A logic programming language based on predicate calculus", Barker, 1988.

[4] The book "PROLOG, Programming for Artificial Intelligence", by Ivan Bratko, Addison-Wesley Publishing Co,1986 contains a well-written and readable guide to understanding how Prolog type expert system programs work. See also, "Logic Programming and Knowledge Engineering" by Tore Amble, Addison Wesley Publishing Co.,1987.

[5] LISP - "A programming language well suited for list processing and symbolic manipulation. It is currently the most popular AI language in the United States", Barker, op.cit.

[6] LISP 3rd ed., by Patrick Henry Winston and Berthold Horn, Addison Wesley Publishing Co, 1989.

[7] CLIPS User's Guide, by Joseph C. Giarratono, NASA Lyndon B. Johnson Space Center, Information Systems Directorate, Software Technology Branch, 1991.