


AN ABSTRACT OF THE THESIS OF

Yimin Zeng for the degree of Master of Science in
Industrial and Manufacturing Engineering presented on
May 5, 1993

Title: An Expert System For Softwood Lumber Grading

Abstract approved: 

Dr. Sabah Randhawa

The focus of this research is to develop a prototype expert system for softwood lumber grading. The grading rules used in the knowledge base of the system are based on Western Lumber Grading Rules 88 published by the Western Wood Products Association. The system includes 27 grades in Dimension, Select/Finish, and Boards categories.

The system is designed to be interactive and menu-driven. The user input to the system consists of lumber size, grade category, and type, location and size of defects for each face. The system then infers the grade corresponding to each face, and an overall grade for the lumber. The system provides limited explanation capabilities.

Evaluation of the system was performed using 85 samples of pre-graded Siberian larch 2x4x12s in Structural Light Framing category. The initial evaluation was performed using the two wide faces of boards. Results indicated a 60 percent match between the grade assigned by

the human expert and the system. The largest cause of deviation was exclusion of defects on the two narrow faces. The knowledge base was expanded to include the two narrow faces; the match rate improved to 76.5 percent. Evaluations for other grading categories need to be conducted in the future to assess the adequacy of the knowledge base.

The prototype development concentrates on selected defect characteristics for each grade. These characteristics are clearly defined and described in the rule book, and are usually the most frequently encountered defects on softwood lumber. The knowledge base needs to be refined and expanded if additional factors such as knot positions relative to each other, warp, manufacturing imperfections and clustering of defects are to be considered.

(C) Copyright by Yimin Zeng
May 5, 1993

All Rights Reserved

An Expert System For Softwood Lumber Grading

by

Yimin Zeng

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

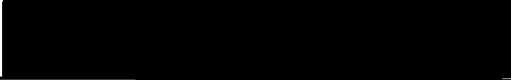
Completed May 5, 1993

Commencement June 1993

APPROVED:



Professor of Industrial and Manufacturing Engineering
in charge of major



Head of Department of Industrial and
Manufacturing Engineering



Dean of Graduate School

Date thesis is presented May 5, 1993

Typed by Yimin Zeng for Yimin Zeng

ACKNOWLEDGEMENTS

I wish to express my deepest gratitude and appreciation to my major professor, Dr. Sabah Randhawa, for his guidance, encouragement and support throughout the course of my graduate study. His tremendous patience and judicious suggestions made possible the successful completion of this thesis.

I wish to thank Dr. Kenneth Funk and Dr. Eldon Olsen for contributing their time and expertise while serving on my graduate committee.

My very special appreciations go to Dr. James Funk. He spent tremendous amount of time to teach me about lumber production and grading, and to help me to evaluate the system using real lumber.

I would also like to thank Dr. Robert L. Ethington and Dr. Rakesh Gupta for providing me with lumber samples.

I am grateful to my friend Mr. Johannes B. Forrer. The discussions with him always inspired me.

I am deeply indebted to my wife, Yuan Zhong, for her love, understanding, encouragement and help in all aspects of my daily life and graduate studies.

I am deeply indebted to my parents and sisters for their deep and eternal love, understanding, encouragement and support.

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION	1
CHAPTER 2 RESEARCH OBJECTIVES	7
CHAPTER 3 LITERATURE REVIEW	8
3.1 Programs Using Algorithmic Approach	9
3.2 Programs Using Expert System Approaches	12
3.2.1 Structure of Expert Systems	12
3.2.2 Expert Systems in Lumber Grading	15
3.3 Systems Developed in Industry	17
CHAPTER 4 SYSTEM DEVELOPMENT	19
4.1 Problem Domain	19
4.1.1 Softwood Lumber Grading Practices	19
4.1.2 Grading Categories	21
4.2 Knowledge Acquisition and Representation	31
4.2.1 Knowledge Acquisition	31
4.2.2 Knowledge Representation	32
4.3 Inference Mechanism	38
4.4 Tool Selection	43
4.5 Prototype Development	48
4.5.1 Prototype Components	48
4.5.2 Knowledge Base Implementation	51
4.5.3 Grading Process, Input, and Output	57
CHAPTER 5 EVALUATION AND ENHANCEMENTS	67
5.1 System Evaluation	67
5.2 System Improvement	69
CHAPTER 6 CONCLUSIONS	73
BIBLIOGRAPHY	76
APPENDICES	
Appendix A. Source Code for Structural Light Framing Category	78
Appendix B. Test Results Using Generated Data	93

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
1.1. Flow chart of a typical sawmill	2
1.2. Flow chart of a typical planer mill	3
3.1. Basic structure of an expert system	13
4.1. Grading categories, sub-categories, and grades included in this project	22
4.2. Basic structure of the prototype	49
4.3. Grading process	58
4.4. Opening window of the run-time version of the system	61
4.5. Dialog box for selecting input method	62
4.6. Dialog box for entering lumber thickness	62
4.7. Dialog box for confirming lumber dimension	62
4.8. Dialog box for selecting grade category	63
4.9. Dialog box for selecting dimension sub-category	63
4.10. Dialog box for selecting lumber face	64
4.11. Dialog box with partial defect list	64
4.12. Dialog box with another partial defect list	65
4.13. Dialog box for entering knot type and location	65
4.14. Dialog box for entering knot diameter	66
4.15. System output display	66
5.1. Evaluation results considering all factors	68
5.2. Evaluation results when warp, manufacturing imperfections, and measurement bias were not considered.	71

LIST OF TABLES

<u>Table</u>	<u>Page</u>
4.1. Grading categories	21
4.2. Grade characteristics considered by the project for grades in Select/Finish's Select sub-category	24
4.3. Grade characteristics considered by the project for grades in Select/Finish's Finish sub-category	25
4.4. Grade characteristics considered by the project for grades in Boards' Commons sub-category	26
4.5. Grade characteristics considered by the project for grades in Boards' Alternate sub-category	27
4.6. Grade characteristics considered by the project for grades in Dimensions' Light Framing sub-category	28
4.7. Grade characteristics considered by the project for grades in Dimensions' Structural Light Framing sub-category	29
4.8. Grade characteristics considered by the project for grades in Dimensions' Stud sub-category	30
4.9. Characteristics permitted and limiting provisions for CONSTRUCTION grade in Light Framing sub-category of dimension lumber	35
4.10. Grading rules for the CONSTRUCTION grade represented using production rules	36
4.11. Source files and their contents	50

LIST OF APPENDICES TABLES

<u>Table</u>	<u>Page</u>
B1. Specifications of grades for a piece of lumber in the Dimension category (lumber size = 2"x4"x10')	94
B2. Test results of grading the lumber in the Dimension category	95
B3. Specifications of grades for a piece of lumber in the Select/Finish category (lumber size = 1"x8"x12')	96
B4. Test results of grading the lumber in the Select/Finish category	97
B5. Specifications of grades for a piece of lumber in the Boards category (lumber size = 1"x8"x12')	98
B6. Test results of grading the lumber in the Boards category	99

AN EXPERT SYSTEM FOR SOFTWOOD LUMBER GRADING

CHAPTER 1

INTRODUCTION

Lumber sawmilling is one of the oldest industries in America. In mills, logs are converted into lumber by sawing, edging and trimming operations (Williston, 1988). At the sawing station, logs are broken down into rough, unedged and untrimmed lumber. The rough lumber is then cut into green lumber in specified width and length at edging and trimming stations. Rough green lumber is then dried and surfaced into finished products. Figures 1.1 and 1.2 show the primary steps in sawmill and planer mills for lumber manufacturing. Configuration of different mills may vary depending upon the log supply and finished products.

One of the most important issues concerning the lumber manufacturing industry is the efficiency of raw material usage. Lumber recovery, a term used to describe how much finished lumber is produced from a log or a given population of logs, is used as the primary measure for the performance of a sawmill. The trend is to use computer-based technology to optimize lumber recovery. Trimming and edging stations first used computer technology to obtain optimum cutting patterns of unedged and untrimmed lumber. Later, optimization techniques were used at log breakdown stations where the optimum log breakdown patterns are determined.

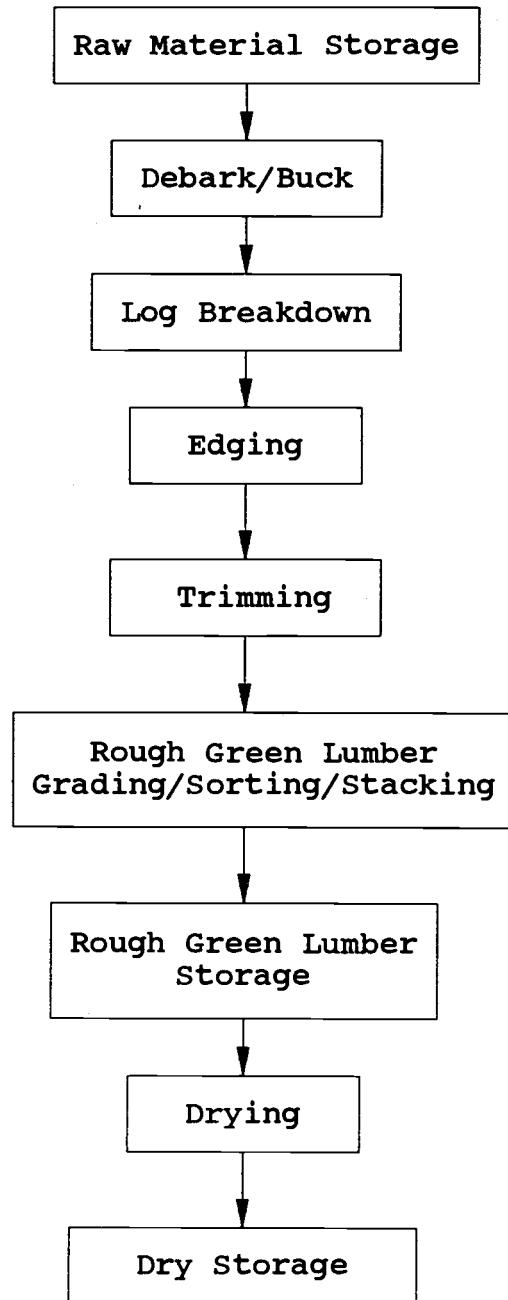


Figure 1.1. Flow chart of a typical sawmill

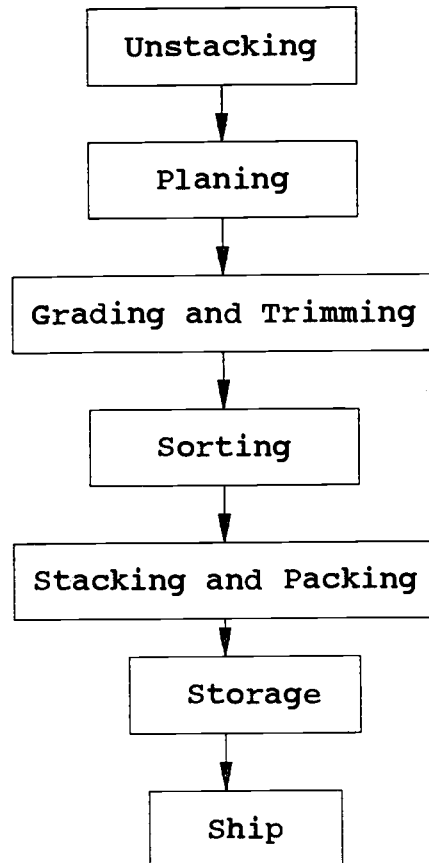


Figure 1.2. Flow chart of a typical planer mill

The use of optimization techniques at the log breakdown stations has evolved from initially considering only one geometric dimension of a log to where three dimensional features of the log are now being considered.

However, currently available models make their "optimum" decisions based only on the log's or the board's geometrical dimensions. An important aspect ignored by the optimization models is the effect of log internal defect type, size, and location on final lumber grade (i.e., recovery and quality). The presence of defects needs to be considered in all manufacturing steps, especially in the log breakdown, edging and trimming operations. More value recovery is expected if an optimization system can combine both geometrical and quality information, and predict lumber grade when making cutting decisions.

Natural log characteristics and manufacturing imperfections have a great impact on lumber quality. To provide specified quality levels for end-use, lumber must be graded according to its quality characteristics. Lumber grading is performed in a sawmill or a sawmill and planing mill, and is monitored by authorized agencies who publish grading rules which conform to the American Softwood Lumber Standards (ASLS, 1970). Traditionally, lumber grading is performed by trained human graders through visual inspection. Although computer aided manufacturing techniques have been widely used in modern sawmills, development of an automated lumber grading system was not initiated until recently because of the complexity of grading rules and the difficulties in the detection and recognition of defects using a machine vision system.

A lumber grading software package can be invaluable to the lumber manufacturing industry in the following aspects:

1. It can be linked to a log breakdown optimization program, such as SAW3D (Zeng, 1991), to improve lumber recovery. As mentioned before, lumber breakdown optimization programs have been used extensively in sawmills. After receiving log scanning data, the optimization system can determine optimum sawing, edging and trimming patterns to find the best lumber recovery from the given log. Nevertheless, such optimization systems do not consider external and internal defects of the log. As such, the resulting solution is not optimal. From a value stand point, an advanced log breakdown optimization system that considers log defects in addition to other characteristics will be able to provide better solutions. The lumber grading program can be embedded in a log breakdown program (such as SAW3D).

2. It can be coupled with a machine vision system to make an on-line automatic lumber grading system. An on-line automatic lumber grading system consists of two basic components, a machine vision system which performs as human eyes to capture the size and quality characteristics of lumber, and a grading software program which uses the information provided by the machine vision system to determine the grade of lumber. An integrated system should be able to provide more reliable and accurate grading

results than a human grader, and it should be more economical than using a human grader.

3. It can be used as an instruction tool to aid in training novices and upgrading knowledge of grading personnel.

CHAPTER 2

RESEARCH OBJECTIVES

The objective of this research was to develop an expert system for softwood lumber grading. The primary objective of developing this system was to link it to a log breakdown optimization system, SAW3D (Zeng, 1991), so that the integrated system can consider both log geometric and quality information when making sawing, edging, and trimming decisions to optimize lumber recovery.

Grading rules used in this system were based on "Western Lumber Grading Rules 88" published by the Western Wood Products Association. The research focused on feasibility analysis, and the conceptual and structure design of the system. Therefore, only selected grades and wood characteristics were included in the system. When SAW3D makes optimum log breakdown decisions, some defects such as manufacturing imperfections do not exist. Such defects were not included in the grading system.

CHAPTER 3

LITERATURE REVIEW

Although the significance of computer lumber grading programs has been recognized and some researchers developed programs for lumber grading in the early 1970's (Hallock and Galiger, 1971), such programs have not been extensively used in industry. One possible reason for lack of use is that a prerequisite for a defect detection system is a machine vision system for lumber grading. Developing a reliable machine vision system is a difficult task. As research on machine vision has evolved, activity in the development of lumber grading programs has also increased. Some prototype automatic grading systems have been recently developed in the sawmill industry.

Currently available computer grading programs have been developed for hardwood lumber grading, and use grading rules established by the National Hardwood Lumber Association (NHLA). There are two basic problems that exist while grading lumber: 1) the recognition of the type and seriousness of each defect, and 2) the calculation of usable areas. In grading hardwood lumber, all grades are based on the recognition of defects and the calculation of usable area.

The focus of this research is softwood lumber, for which little has been developed in way of automatic grading

systems. In contrast to hardwood lumber, softwood lumber grading is primarily based on the recognition of defects; only grades of Factory Lumber need to calculate usable area. Since grading problems for hardwood lumber and softwood lumber are very similar except for the differences in using different grading rules and grading procedures, it is worthwhile to review some of the research for hardwood lumber grading.

3.1 Programs Using Algorithmic Approach

Most of the early programs were developed using an algorithmic programming approach. Although the expert system approach has an obvious advantage over the algorithmic approach, lack of good development tools and the requirement of expensive computing facilities prevented use of the expert system approach until recently.

Hallock and Galiger (1971) developed a computer program for grading hardwood lumber. The program analyzes the board with its defects, finds the number of cuttings of the sizes permitted, and assigns a grade according to NHLA rules. It classifies lumber according to six major classifications. The board and all its defects are described by rectangles in a Cartesian coordinate system; the X-Y coordinates of those rectangles are entered on a series of cards. Once the program receives the input data,

it begins the grading process by assigning the board a potential grade based on the board size. This is the highest possible grade for the board without considering defects. Then the defects are checked and the board is assigned a potentially lower grade because of the impact of defects. Next, sizes and number of usable areas are calculated, and the board is assigned the final grade. The program was developed in FORTRAN. The authors believed the program's accuracy to be well within NHLA tolerances. However, some limitations exist. For example, grading rules and procedures were all coded into the program algorithms. This programming practice makes modification and expansion of grading rules difficult. Additionally, the algorithms considered only one face of the board, whereas both faces of a board are evaluated in the actual grading process.

Klinkhachorn et al. (1988) created a hardwood lumber grading program as part of an automated lumber process system. The program was implemented using FORTRAN 77 for an IBM PC or compatible. After reading in the board dimensions and defect types and their location on each face, the program assigns a potential grade to each face. Next the program calculates usable area to determine the final grade assigned to the board. This program improved upon the previous efforts by considering both faces of the board and applying a more flexible grading procedure which provides the option of admitting or excluding certain types

of defects when searching for cuttings. In addition, the program uses the BLOCK DATA feature of FORTRAN 77 to logically separate grading rules from the program logical code so that reconfiguration of the grading rules is easier. When grading rules need to be modified, only the data specified in the BLOCK DATA segment need to be changed.

Klinkhachorn et al. (1989) presented a computer-aided instruction tool, named HaLT (Hardwood Lumber Training) for training hardwood lumber graders. The HaLT system incorporates a graphic user interface and explanation facility in the hardwood lumber grading program discussed above. The interface provides graphic views of the board including each side of the entire board with defects or the expanded view giving the detailed section of the board, asks questions, compares answers, and provides reasoning. The program runs on an IBM PC or compatible. The objective is to provide the lumber industry with an interactive learning tool to educate and train their graders on-site.

Klinkhachorn et al. (1991) improved their previous program (Klinkhachorn et al, 1988) by converting the code from FORTRAN 77 to C, and restructuring it. The grading rules for various species, the mechanical dimensions of a board, the violations of the various rules, and the grade of the board are all mathematically represented in terms of program variables in data structure so that the program

data are logically separated from the program code. This enhances the program's capabilities, its adaptability and portability, and makes the restructuring of grading rules easier.

3.2 Programs Using Expert System Approaches

3.2.1 Structure of Expert Systems

An expert system is a computer program that uses knowledge, facts, and reasoning techniques to solve problems that normally require the abilities of human experts. Expert system architectures include different components. However, certain components are common to most expert systems. These include a user interface, a knowledge base, and an inference engine. Figure 3.1 shows the basic structure of an expert system.

The interface provides the communication between the user and the system. It allows the user to query the system, enter facts about a specific situation that is relevant to the system's subject domain, and receive advice, consultation and explanation. Many expert systems also accept new knowledge through the user interface.

The knowledge base contains expert-level knowledge on the system's subject domain. The knowledge can be obtained from books and publications, and/or from one or more human experts, and is stored using a knowledge representation

scheme.

The inference engine provides the reasoning capabilities of the system. It uses the domain specific knowledge in the knowledge base and facts provided by the user to infer new facts that may be conclusions to goals and subgoals as required by the system. The inference method is built into the inference engine, while the domain specific knowledge is contained in the knowledge base. The general problem solving knowledge and the domain specific

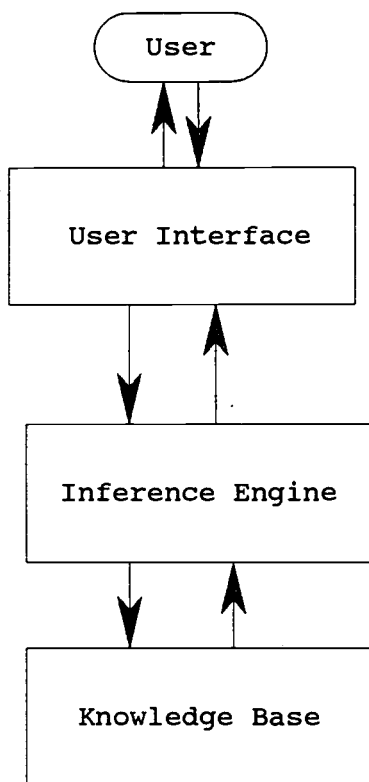


Figure 3.1. Basic structure of an expert system

knowledge are separated, thus the same inference engine can be used to reason with different knowledge bases.

In addition to those basic components, many expert systems have various supplemental components, including knowledge acquisition facilities, knowledge-base maintenance and update facilities, explanation facilities, self-training facilities, and validation tools.

Expert system techniques are very appropriate to problems for which human expertise is required, which need symbolic reasoning but cannot be easily handled numerically or algorithmically, or for which information required for a solution is incomplete or uncertain.

The lumber grading problem is declarative rather than algorithmic in nature. The grading process involves symbolic rather than numeric reasoning. Maintaining and updating grading rules is required as criteria change over time, and from one mill to another. In some cases, data about a defect may be uncertain. If used as an instruction tool, explanation on the actions and conclusion would be necessary. All these facts indicate that while an algorithmic approach is possible, the expert system approach is more appropriate. Using the expert system approach for lumber grading has the following advantages over the algorithmic approach:

1. In knowledge-based systems, lumber data and grading rules are stored in a knowledge base which is decoupled

from the program logic. Hence, modification and expansion of grading rules can be made easily without affecting the program logic.

2. When used as a computer-aided instruction (CAI) tool, knowledge-based systems are more appropriate because they provide consultation and explanation capabilities required in training situations.

3. Rules-of-thumb used by grading experts can also be incorporated in the knowledge base to supplement the rules from publications, and to speed up the grading process as these may result in reducing the search space. Furthermore, due to the separation of the knowledge base from the program logic, customization of such rules can be easier.

3.2.2 Expert Systems in Lumber Grading

Huang and Sparrow (1989) developed a CAI tool for hardwood lumber grading using the expert system approach. The expert system consists of five main components: a knowledge base, an inference engine, a procedure for calculating the usable area, a user interface facility, and an explanation subsystem. The knowledge base contains facts and rules. Facts include lumber data, current data, and information derived from the rules. Rules are the conditions and restrictions for judging the grades of the boards. Knowledge is represented by rules and working memory elements. A forward-chaining inference mechanism is

used. The system was implemented using the LISP language and runs on the VAX 11/780 computer. The program facilitates the training of professional hardwood lumber graders by providing consultation and explanation of lumber grading procedure in an interactive question-answer process. Lumber data, including lumber dimensions, types and sizes of defects, need to be entered by the user. The program instructs the user in three steps: 1) to judge grades of the two faces based on a combination of defect types and sizes; 2) to judge grades of the two faces based on calculation of usable areas on both faces; and 3) to judge the overall grade of the board. At each step, it asks the user to determine a tentative grade, and then explains the user's mistakes, if any. Otherwise, it provides an explanation of the process of reaching the correct grade. The program reduces the training costs by decreasing the need to rely solely on an expensive human instructor, and increases the speed and retention rates of students because it offers a flexible time table that makes it more convenient for the student to work at his/her own pace. The knowledge base is limited in the system. Among over 100 types of hardwood lumber, only two types (walnut and poplar) were chosen for this initial work.

Softwood lumber grading programs using the expert system approach have not been developed.

3.3 Systems Developed in Industry

As mentioned before, lumber recovery optimization systems based on dimension scanning have been widely used in sawmills. While more complete geometric information is included in recent three-dimensional scanning and optimization systems, higher recovery is to be expected if the defect information along with the dimensional data is used in making cutting decisions. Though some corporations have been developing grade scanning and optimization systems which combine both dimension and quality information for logs and lumber in making optimum decisions, there is little information available in published literature. In addition to the difficulty of accessing information from industry, available papers focus on defect detection systems, as compared to implementing grading rules in their programs. This is not surprising since the development of real time defect detection systems has been a long time barrier in this area.

VisionSmart Inc. of Canada developed a Grade-based Optimizer system which uses a camera to scan all features related to dimension and wane, and an X-ray sensor array to detect knots and other density related features such as compression wood and pitch pockets (Flatman and Bodell, 1989; Flatman and Kenway, 1991; Kenway, 1990; Aune, 1991). Other features such as shakes, checks, holes, and

stains are detected by cameras operating under different lighting than the lighting for the camera used for dimension and wane. The system combines both profile and defect data and then determines optimum cutting solutions for maximum value return. The system can be used either for green lumber or for dry lumber.

A Grade-Based Edger Optimizer was installed in MacMillan Bloedel's Alberni Pacific Division sawmill in March, 1989. The Edger Optimizer scans wet green lumber for grading features and based on the scanning data, uses a dynamic programming algorithm to determine an optimum edging solution. The Edger Optimizer showed a \$160,000 return per month, as measured by the mill. A Grade-Based Trimmer Optimizer was also installed in CANFOR'S Fort St. John Mill in May 1990, where dry lumber is scanned after planing and the optimum trimming solutions calculated.

The Coe Manufacturing Company developed the D*TEC GradeScan system that is able to detect knots, splits, pockets, stain and decay, as well as wane and dimensions¹. The edger, trimmer and planer mill applications manufactured by the company have been equipped with the GradeScan system so that both defects and shape can be considered when optimizing for value recovery.

¹ The Coe Manufacturing Company. 1991. Company brochure. Portland, Oregon.

CHAPTER 4

SYSTEM DEVELOPMENT

4.1 Problem Domain

4.1.1 Softwood Lumber Grading Practices

Lumber grading is performed in a sawmill or a planer mill by human graders. The graders evaluate each piece of lumber and make their decision according to grading rules as specified by regional association, proprietary, or national grading rules. The following descriptions are given in Western Lumber Grading Rules 88 (1988):

The grading of lumber cannot be considered an exact science because it is based on either a visual inspection of each piece and the judgment of the grader or on such visual inspection and judgment combined with the results of method of mechanically determining the stiffness characteristics of structural lumber. The results are, however, sufficiently explicit to establish a maximum of 5 percent below grade as a reasonable variation between graders.

In general, American Standard softwood lumber is classified according to use, manufacturing, and size

classifications. Grading of lumber is based upon a visual inspection and is primarily a judgment of end-use strength or appearance. Natural log characteristics which have an effect upon strength are taken into account along with manufacturing imperfections. In most cases these include limiting provisions on characteristics such as stain, checks, grain, skip, cup, crook, wane, twist, knots, holes, pitch, pocket, split, steaks and patches, shakes, decay, and cutout. Definitions and detailed descriptions of these characteristics can be found in the Western Lumber Grading Rules 88 (1988).

Grading rules applied to softwood lumber for domestic market are governed by National Bureau of Standards Voluntary Product Standard PS 20-70 (American Softwood Lumber Standard, 1970). All of the grading rule books which are published by grading organizations are based on this standard. The two largest grading organizations on the West Coast are the Western Wood Products Association (WWPA) and the West Coast Lumber Inspection Bureau (WCLIB). In 1986, nearly 35 percent of the nation's softwood lumber production was graded under WWPA supervision. This research project will use the grading rules published by the WWPA as described in the Western Lumber Grading Rules 88 (1988).

4.1.2 Grading Categories

According to the WWP, lumber is classified into one of the following five categories (Table 4.1) based on the size and end-use: Select and Finish lumber, Boards, Dimension, Timbers, and Factory lumber and related products.

Table 4.1. Grading categories

Name of Category	Nominal Size		Judgment on Use Emphasis
	Thickness	Width	
Selects and Finish	3/8" - 16/4"	>= 2"	appearance
Boards	3/4" - 16/4"	>= 2"	appearance
Dimension	2" - 4"	>= 2"	strength
Timber	>= 5"	>= 5"	strength
Factory	any	any	remanufacturing

Each major category is further grouped into many sub-categories and each sub-category includes many grades. Since the focus of this project is prototype development, it considers only grading rules for Select/Finish, Boards, and Dimension categories. Figure 4.1 show these categories and their sub-categories, and the grades selected for this project implementation.

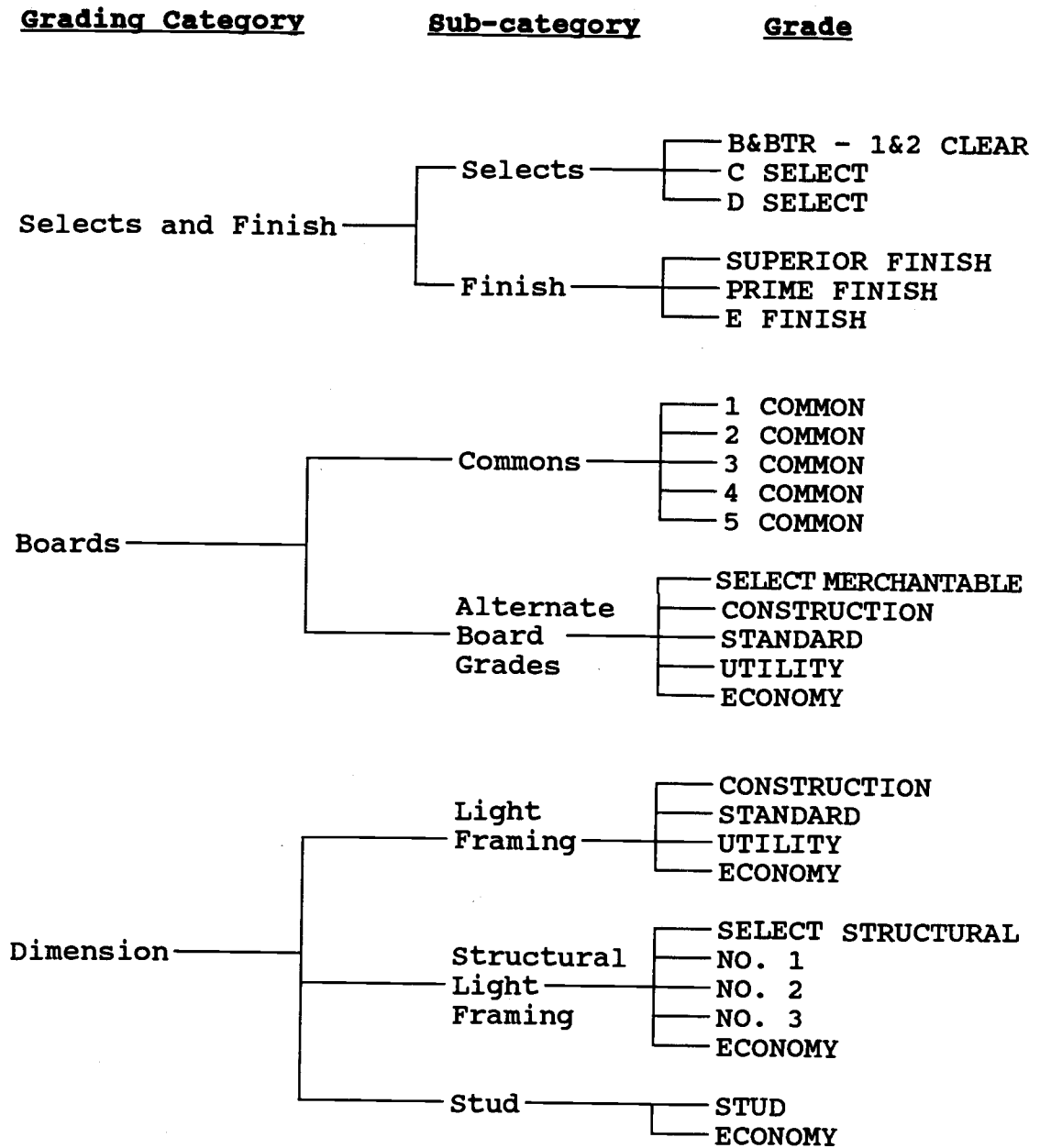


Figure 4.1. Grading categories, sub-categories, and grades included in this project

Characteristics and limiting provisions for each grade include wood defects and manufacturing imperfections, and their type, size, location and distribution. Tables 4.2 to 4.8 give the characteristics for each grade as described by the rule book, and show the characteristics included in the scope of this project. There are two primary reasons for focusing on a subset of the grading rules. First, in many applications, not all characteristics listed in the rule book appear on lumber. At this initial development stage, only the most frequently occurring characteristics are included in the system. The focus was on conceptual design and testing the feasibility of the expert system approach. The system design is such that the knowledge base can be easily refined and extended. Second, the system is planned to be used as a component embedded into SAW3D (Zeng, 1991), a log breakdown optimization system. When SAW3D makes optimum breakdown decisions, manufacturing imperfections are not included. For example, warp, such as bow, crook, cup and twist, are not included.

Table 4.2. Grade characteristics considered by the project for grades in Select/Finish's Select sub-category

Characteristics	Grade (Selects/Finish - Selects)		
	B&BTR - 1&2 CLEAR	C SELECT	D SELECT
Stain	Y	Y	Y
Checks	Y	Y	Y
Splits	N/A - Y	N/A - Y	Y
Torn Grain	N	N	N
Skips	N	N	N
Cup	N	N	N
Crook	N	N	N
Wane	Y	Y	Y
Twist	N	N	N
Knots	Y	Y	Y
Holes	N/A - Y	Y	Y
Pitch	Y	Y	Y
Pockets	Y	Y	Y
Pitch Streak	N/A	N	N
Cutout	N/A	N/A	N

- Y : Listed in the rule book and considered by the project
- N : Listed in the rule book, but not considered by the project
- N/A : Not listed in the rule book and not considered by the project
- N/A - Y : Not listed in the rule book, but considered by the project

Table 4.3. Grade characteristics considered by the project for grades in Select/Finish's Finish sub-category

Characteristics	Grade (Selects/Finish - Finish)		
	SUPERIOR FINISH	PRIME FINISH	E FINISH
Stain	Y	Y	Y
Checks	Y	Y	Y
Splits	Y	N/A - Y	Y
Torn Grain	N	N	N
Skips	N	N	N
Cup	N	N	N
Crook	N	N	N
Wane	Y	Y	Y
Twist	N	N	N
Knots	Y	Y	Y
Holes	N/A - Y	Y	Y
Pitch	Y	Y	Y
Pockets	Y	Y	Y
Pitch Streak	N	N	N
White Speck	N/A	N	N
Cutout	N	N	N
Chip Marks	N	N	N

- Y : Listed in the rule book and considered by the project
- N : Listed in the rule book, but not considered by the project
- N/A : Not listed in the rule book and not considered by the project
- N/A - Y : Not listed in the rule book, but considered by the project

Table 4.4. Grade characteristics considered by the project for grades in Boards' Commons sub-category

Characteristics	Grade (Boards - Commons)				
	1 COMMON	2 COMMON	3 COMMON	4 COMMON	5 COMMON
Stain	Y	Y	Y	Y	Y
Checks	Y	Y	Y	Y	Y
Torn/Raised Grain	N	N	N	N	N
Skips	N	N	N	N	N
Cup	N	N	N	N	N
Crook	N	N	N	N	N
Wane	Y	Y	Y	Y	Y
Twist	N	N	N	N	N
Splits	Y	Y	Y	Y	Y
Streaks and Patches	N/A	N	N	N	N
Pitch	Y	Y	Y	Y	Y
Pockets	Y	Y	Y	Y	Y
White Speck Honeycomb	N/A	N/A	N	N	Y
Unsound Wood	N/A	N/A	N	N	Y
Shake	N/A	N	N	N	N
Roller Checks	N/A	N	N/A	N	N
Pith	Y	Y	N/A	N/A	Y
Knots	Y	Y	Y	Y	Y
Holes	N/A -Y	Y	Y	Y	Y

- Y : Listed in the rule book and considered by the project
- N : Listed in the rule book, but not considered by the project
- N/A : Not listed in the rule book and not considered by the project
- N/A - Y : Not listed in the rule book, but considered by the project

Table 4.5. Grade characteristics considered by the project for grades in Boards' Alternate sub-category

Characteristics	Grade (Boards - Alternate)				
	SEL-MERCH	CONST	STAND	UTIL	ECONOMY
Stain	Y	Y	Y	Y	Y
Checks	Y	Y	Y	Y	Y
Torn/Raised Grain	N	N	N	N	N
Skips	N	N	N	N	N
Cup	N	N	N	N	N
Crook	N	N	N	N	N
Wane	Y	Y	Y	Y	Y
Twist	N	N	N	N	N
Splits	Y	Y	Y	Y	Y
Streaks and Patches	N	N	N	N	N
Pitch	Y	N/A	N/A	N/A	Y
Pockets	Y	Y	Y	Y	Y
White Speck Honeycomb	N	N/A	N	N	Y
Unsound Wood	N	N/A	N/A	N	Y
Shake	N/A	N/A	N	N	N
Roller Checks	N	N/A	N/A	N/A	N
Pith	N/A	N/A	N/A	N/A	Y
Knots	Y	Y	Y	Y	Y
Holes	Y	Y	Y	Y	Y
Burls	N/A	N/A	N/A	N	N

- Y : Listed in the rule book and considered by the project
N : Listed in the rule book, but not considered by the project
N/A : Not listed in the rule book and not considered by the project

Table 4.6. Grade characteristics considered by the project for grades in Dimensions' Light Framing sub-category

Characteristics	Grade (Dimension - Light Framing)			
	CONST	STAND	UTIL	ECONOMY
Checks	Y	Y	Y	Y
Knots	Y	Y	Y	Y
Holes	Y	Y	Y	Y
Manufacture	N	N	N	N
Pitch	Y	Y	Y	Y
Pitch Streaks	N	N	N	N
Pockets	Y	Y	Y	Y
Shake	N	N	N	N
Skips	N	N	N	Y
Slope of Grain	Y	Y	Y	Y
Splits	Y	Y	Y	Y
Stain	N	N	N	N
Wane	Y	Y	Y	Y
Warp	N	N	N	N
Unsound Wood	N/A -Y	Y	Y	Y
White Speck	N/A	N	N	N

- Y : Listed in the rule book and considered by the project
- N : Listed in the rule book, but not considered by the project
- N/A : Not listed in the rule book and not considered by the project
- N/A - Y : Not listed in the rule book, but considered by the project

Table 4.7. Grade characteristics considered by the project for grades in Dimensions' Structural Light Framing sub-category

Characteristics	Grade (Dimension - Structural Light Framing)				
	SEL-STR	NO.1	NO.2	NO.3	ECONOMY
Checks	Y	Y	Y	Y	Y
Grain	N	N	N	N/A	N
Knots	Y	Y	Y	Y	Y
Holes	Y	Y	Y	Y	Y
Manufacture	N	N	N	N	N
Pitch	Y	Y	Y	Y	Y
Pitch Streaks	N	N	N	N	N
Pockets	Y	Y	Y	Y	Y
Shake	N	N	N	N	N
Skips	N	N	N	N	N
Slope of Grain	Y	Y	Y	Y	Y
Splits	Y	Y	Y	Y	Y
Stain	N	N	N	Y	Y
Wane	Y	Y	Y	Y	Y
Warp	N	N	N	N	N
Unsound Wood	N/A-Y	N/A-Y	Y	Y	Y
White Speck	N/A	N/A	N	N	N

- Y : Listed in the rule book and considered by the project
- N : Listed in the rule book, but not considered by the project
- N/A : Not listed in the rule book and not considered by the project
- N/A - Y : Not listed in the rule book, but considered by the project

Table 4.8. Grade characteristics considered by the project for grades in Dimensions' Stud sub-category

Characteristics	Grade (Dimension - Stud)	
	STUD	ECONOMY
Checks	Y	Y
Grain	N/A	N/A
Knots	Y	Y
Holes	Y	Y
Manufacture	N	N
Pitch	Y	Y
Pitch Streaks	N	N
Pockets	Y	Y
Shake	N	N
Skips	N	N
Slope of Grain	Y	Y
Splits	Y	Y
Stain	N	Y
Wane	Y	Y
Warp	N	N
Unsound Wood	Y	Y
White Speck	N	N

- Y : Listed in the rule book and considered by the project
- N : Listed in the rule book, but not considered by the project
- N/A : Not listed in the rule book and not considered by the project
- N/A - Y : Not listed in the rule book, but considered by the project

4.2 Knowledge Acquisition and Representation

4.2.1 Knowledge Acquisition

Knowledge acquisition deals with the issue of acquiring information about the problem domain. Knowledge used in problem solving may be based on experiences of one or more human experts, case histories, reference sources, etc. The key source of knowledge for this project is the grading rules. These are based upon the "Western Lumber Grading Rules 88" published by the Western Wood Products Association (1988).

Compared with many other expert systems, the knowledge acquisition task for this project is relatively simple because most grading rules are well defined in the rule book. However, the rule book cannot give all the information required to perform the grading task accurately and efficiently. First, some of the rules are not explicitly defined. For example, certain grades allow "several" as the number of certain defects. However, "several" is not precisely defined. Second, the rule book just stipulates characteristics and limiting provisions, but not the grading procedures. In practice, grading procedures affect the speed with which lumber is graded. For instance, knots are the most frequently encountered defects. Looking for frequently encountered defects accelerates the grading process. Third, as mentioned

before, grading is more of an art than a science, and the results depend on the inspection and judgement of the grader. Five percent variation from standards is accepted as a reasonable difference between graders. Some mills and markets may use their local rules, and take advantage of the allowed variations. This may involve specific knowledge not documented in the rule book. Thus, the knowledge base using grading rules from the rule book can be refined based on actual grading practices.

4.2.2 Knowledge Representation

The domain knowledge must be well organized and structured in the knowledge base, thus allowing the inference mechanism of the expert system to operate on it effectively and efficiently. There are many ways to represent domain knowledge, including production rules, frames, semantic networks, and first-order logic. Perhaps the most frequently used representation scheme is production rules.

Production rules represent knowledge in the following form:

```
IF (list of antecedents)
    AND ...
    OR ...
    NOT ...
THEN (list of consequents or actions).
```

This format is almost identical to the description of grading rules stipulated in the rule book. Thus, the scheme of production rules is intuitively the best representation scheme for the domain knowledge for this project.

The grading rules are of the following form:

IF (list of characteristics permitted and limiting provisions)

THEN Grade x.

For example, consider the CONSTRUCTION grade in the Light Framing sub-category of dimension lumber. The characteristics permitted and limiting provisions for this grade are summarized in Table 4.9. These can be listed as the condition part of a production rule; the conclusion of the rule will be grade CONSTRUCTION if all the conditions are satisfied. However, such a single rule containing a description of all the characteristics will be inefficient because the inference process will be long. A better representation mechanism is to break down a single rule into several smaller rules where each component rule focuses on a distinct characteristic. Since the example in Table 4.9 has eleven components, the 11 rules shown in Table 4.10 may be used to represent the CONSTRUCTION grade. Finally, Rule 12 integrates the conclusions derived from

the individual sub-components.

In this illustrative example, not all characteristics are included in the rules. For instance, spike knots and stains are not considered. Also, streaks and pockets are not listed because they have no impact on the grading result. Some rules, such as Rules 5, 7, and 11, need more details or other higher level rules to provide the antecedents.

Table 4.9. Characteristics permitted and limiting provisions for CONSTRUCTION grade in Light Framing sub-category of dimension lumber

Characteristics	Provisions												
Checks	Surface seasoning checks not limited. Through checks at ends are limited as splits.												
Knots	<p>Sound, firm, encased and pith, must be tight and are permitted in the following sizes or their equivalent displacement:</p> <table border="1"> <thead> <tr> <th>Nominal Width</th> <th>Anywhere On Wide Face</th> <th>Unsound or Loose Knots and Holes</th> </tr> </thead> <tbody> <tr> <td>2"</td> <td>3/4"</td> <td>5/8"</td> </tr> <tr> <td>3"</td> <td>1&1/4"</td> <td>3/4"</td> </tr> <tr> <td>4"</td> <td>1&1/2"</td> <td>1"</td> </tr> </tbody> </table> <p>Knots spiked entirely across the wide face are limited to a displacement of approximately 1/4 the cross section.</p>	Nominal Width	Anywhere On Wide Face	Unsound or Loose Knots and Holes	2"	3/4"	5/8"	3"	1&1/4"	3/4"	4"	1&1/2"	1"
Nominal Width	Anywhere On Wide Face	Unsound or Loose Knots and Holes											
2"	3/4"	5/8"											
3"	1&1/4"	3/4"											
4"	1&1/2"	1"											
Manufacture	Standard "E".												
Pitch and pitch steaks	Not limited.												
Pockets, pitch or bark	Not limited.												
Shake	Several heart shakes up to 2' long, similar to seasoning checks, none through.												
Skips	Hit and miss on 10% of the pieces.												
Slope of grain	1 in 6.												
Splits	Equal in length to the width of the piece.												
Stain	Stained sapwood. Firm heart stain or firm red heart.												
Wane	1/4 the thickness, 1/4 the width. 5% of the pieces may have wane up to 1/2 the thickness and 1/3 the width for 1/4 the length.												
Warp	1/2 of medium.												

Table 4.10. Grading rules for the CONSTRUCTION grade represented using production rules

<p>Rule 1: IF NO through checks at ends OR longest through check at end \leq lumber width THEN Checks satisfy grade CONSTRUCTION</p>
<p>Rule 2: IF NO firm knots on face OR (lumber width = 2" AND diameter of largest firm knot \leq 0.75") OR (lumber width = 3" AND diameter of largest firm knot \leq 1.25") OR (lumber width = 4" AND diameter of largest firm knot \leq 1.5") THEN firm knots satisfy grade CONSTRUCTION</p>
<p>Rule 3: IF NO loose knots on face OR (lumber width = 2" AND diameter of largest firm knot \leq 5/8") OR (lumber width = 3" AND diameter of largest firm knot \leq 3/4") OR (lumber width = 4" AND diameter of largest firm knot \leq 1") THEN loose knots satisfy grade CONSTRUCTION</p>
<p>Rule 4: IF NO holes on face OR (lumber width = 2" AND diameter of largest hole \leq 5/8") OR (lumber width = 3" AND diameter of largest hole \leq 3/4") OR (lumber width = 4" AND diameter of largest hole \leq 1") THEN holes satisfy grade CONSTRUCTION</p>
<p>Rule 5: IF manufacture standard is "E" THEN manufacture satisfies grade CONSTRUCTION</p>
<p>Rule 6: IF NO shakes OR (NO through shakes AND length of largest shake \leq 2') THEN shakes satisfy grade CONSTRUCTION</p>

Table 4.10. (Continued)

<p>Rule 7: IF NO skips OR hit and miss on $\leq 10\%$ pieces THEN skips satisfy grade CONSTRUCTION</p>
<p>Rule 8: IF slope of grain $\leq 1/6''$ THEN slope of grain satisfies grade CONSTRUCTION</p>
<p>RULE 9: IF NO splits OR length of longest split \leq lumber width THEN splits satisfy grade CONSTRUCTION</p>
<p>RULE 10: IF NO wane OR (thickness of wane $\leq 1/4$ lumber thickness AND width of wane $\leq 1/4$ lumber width) THEN wane satisfies grade CONSTRUCTION</p>
<p>RULE 11: IF NO warp OR warp $\leq 1/2$ medium THEN warp satisfies grade CONSTRUCTION</p>
<p>RULE 12: IF checks satisfy grade CONSTRUCTION AND firm knots satisfy grade CONSTRUCTION AND loose knots satisfy grade CONSTRUCTION AND holes satisfy grade CONSTRUCTION AND manufacture satisfies grade CONSTRUCTION AND shakes satisfy grade CONSTRUCTION AND skips satisfy grade CONSTRUCTION AND slope of grain satisfies grade CONSTRUCTION AND splits satisfy grade CONSTRUCTION AND wane satisfies grade CONSTRUCTION AND warp satisfies grade CONSTRUCTION THEN grade is CONSTRUCTION</p>

4.3 Inference Mechanism

Given a knowledge base, an expert system performs inference or deduction to arrive at conclusions. For rule-base systems, the logic deduction is based on the following modus ponens:

```
Given the rule:      IF      antecedent 1
                    AND      antecedent 2
                    .....
                    AND      antecedent N
                    THEN     conclusion 1

and facts:          antecedent 1
                    antecedent 2
                    .....
                    antecedent N,
```

it is possible to infer that conclusion 1 is true.

Facts and rules are basic components of the knowledge base. The reasoning process involves looking for facts and using rules to arrive at conclusions.

Inference strategies used in expert systems include backward chaining, forward chaining, breadth-first search, depth-first search, heuristic search, problem reduction, pattern matching, hierarchical control, unification, and event-driven control. Selection of a proper strategy for

an expert system depends on the knowledge representation scheme used in the system. Matching the task to the correct combination of knowledge representation and inference strategy is a complex task. For rule-based systems using production rules as the representation scheme, the major inference strategies used are backward chaining, forward chaining, or a combination of these two strategies. These strategies are primarily used to specify how rules contained in a knowledge-base are to be processed.

Backward chaining is also called goal-driven inference. It begins with the selection of a goal using some selection criterion, then tries to prove that the goal is true. A goal is the conclusion of a rule. The process usually starts with matching existing facts with antecedents of that rule. If the required facts cannot be found, the inference engine then looks for rules that may have the required facts as their conclusions. This may direct the search process to sub-goals. If the required facts still cannot be found, the system may begin a query process if the facts can be obtained from the user. Otherwise the system will select another goal and repeat the process until a solution is found or there are no more goals to be checked.

To illustrate how this strategy works, assume that there is a piece of lumber 2 inches in nominal thickness,

4 inches in width, and 10 feet long. There are no defects on its face except for a few well spaced firm knots. Assume that the largest knot has a diameter of 0.5 inches. The rules in Table 4.10 show that the grade of this piece of lumber is CONSTRUCTION in the light framing category of dimension lumber. The backward chaining process starts by selecting a grade from the 27 grades listed in Figure 4.1, and then tries to prove that this grade is the appropriate one. Assume that the grade selected is CONSTRUCTION. The system first looks for facts that are antecedents of Rule 12. Since these facts do not exist at this stage, it looks for rules 1 through 11 which have those facts as conclusions and tries to prove that rules 1 through 11 are true. To prove these rules, additional information is required. This can be obtained from the user, or by a search for more rules. Once rules 1 through 11 are proven to be true, all antecedents in Rule 12 are generated as new facts. This in turn proves that Rule 12 is true and the grading process is completed. If the selected initial grade is not CONSTRUCTION, the process may fail to find a solution; another grade may then be selected, and the process is repeated until a grade is found.

This strategy is best suited for the problems with many initial facts but relatively few solutions. It has been successfully used for diagnostic problems. However, for the lumber grading problem, it is not the appropriate

method. One reason is that there are a large number of characteristics that need to be checked before a grade can be determined. It can be very inefficient if a wrong grade (a goal) is assumed. Another reason is that this inference mechanism is not analogous to the human grading process where conclusions are reached based on facts.

The forward chaining inference mechanism, also called data-driven chaining or reasoning, starts with existing facts and works on deducing new facts that will eventually lead to the goal. The inference process works in the following cycle:

1) Matching rule(s): find all rules whose antecedents are satisfied by working storage (that is, where existing facts are stored). The collection of such rules is called the conflict set.

2) Selecting rule(s): When more than one rule matches the facts, apply a conflict resolution strategy to determine which rule will be selected among those rules. Various conflict resolution strategies have been used for forward chaining systems, including:

Do One: Select the first rule that matches the facts.

Do All: Apply all the matching rules in one batch.

Do in Sequence: Apply the matching rules one at a time.

Do the Most Specific: Select the more specific rule.

Do the Most Recent: Select the one that matched a most recently derived fact.

- 3) Executing rule(s): the selected rule is proved, also called "fire the rule", and the working storage is updated with fact(s) of the newly fired rule.
- 4) The process is repeated until a solution is found or no more rules can be applied.

For the lumber example used to illustrate backward chaining, the forward chaining process will start by collecting all the defect information, and storing it in the working storage (memory). It then tries to find all rules whose antecedents are matched with the existing facts. For the given data, it will conclude that antecedents of rules 1 through 11 exist in the working storage. Thus rules 1 through 11 are placed in the conflict set. Selection and firing of these 11 rules result in the facts that match the antecedents of Rule 12; these are then placed in the working storage. Now Rule 12 is also put into the conflict set because all of its antecedents are satisfied. Finally, firing Rule 12 results

in the CONSTRUCTION grade for the specified input data.

The forward chaining procedure is best used to solve problems in which data is to be used as the starting point for problem solving. It is intuitively the most appropriate method for the lumber grading problem; it also matches the human grader's reasoning process. The inference used in this implementation is the forward chaining inference mechanism.

4.4 Tool Selection

After identifying an appropriate knowledge representation scheme and inference mechanism, the next phase in the development process is to select the implementation software. Four options were considered for this project: conventional programming languages, knowledge engineering languages, knowledge-based programming tool kits, and knowledge-based programming shells.

Conventional languages, such as C, C++ and Pascal, can be used to implement a knowledge-based system. They support a broad range of machine platforms. Compared with other options, they are very flexible and efficient in terms of execution speed and memory usage. The major disadvantage is that the development time is long requiring extremely large amount of coding. This option is obviously

not recommended for rapid prototyping and system design.

Knowledge engineering languages, such as LISP and PROLOG, support symbolic manipulation and/or have built-in inference engines. They are very powerful and efficient when used on the right platforms such as LISP machines and work stations. However, the opposite is true on microcomputers. Since this project selected microcomputers as the implementation platform, this option also did not represent the best choice.

Knowledge-based programming toolkits, such as KEE, OPS5, ART, Clips and Eclipse, are somewhat like high-level languages. While relieving the programmer from the burden of tedious coding by providing built-in knowledge representation schemes and inference engines, they do require some programming following their syntax so that maximum flexibility can be achieved. This option was considered appropriate for this project.

Knowledge-based shells usually provide a complete development environments. The programmer does not need to do any real programming. However, they are usually expensive and have limited flexibility. This option was prohibitive due to the cost involved and the flexibility required by the project, particularly interaction with external systems.

Several features were required of the development tool to be used. First, the tool should support rule-based

knowledge representation and a forward chaining inference mechanism. Second, the system should interact with C/C++ programs since the intended use of this product is to integrate it into SAW3D, a log breakdown optimization program written in C. Therefore, support of links between this project and any ANSI C/C++ compiler was considered to be an important feature of the development tool. Third, the tool should be run on an IBM-PC and support protected mode memory access. The knowledge base can potentially be so large that the 640k conventional memory limitation can become an obstacle. Fourth, the tool should be financially feasible. Other considerations included user interface and explanation facilities, ease of learning and use, documentation, and technical support.

Considering the above factors, Eclipse, a commercial rule-based programming tool from The Haley Enterprise (The Haley Enterprise, 1991), was selected as the implementation tool for this project. Eclipse supports the features described above, including rule-based knowledge representation, a forward chaining inference engine, running under protected mode and integration with other applications.

The syntax for Eclipse is derived from Inference Corporation's ART (for Automatic Reasoning Tool), and is compatible with NASA's Clips (for C Language Integrated Production System). The basic syntax uses LISP-like prefix

notation and parentheses. Eclipse uses the Rete Algorithm, which is one of the most efficient techniques for performing complex pattern matching of the type used in rule-based systems (Forgy, 1979, 1982). Although many other tools also use the Rete Algorithm, some benchmarks and applications have proven that Eclipse is superior to many other tools in terms of execution speed and program size² (Brooke, 1992). For example, it runs four times as fast as Clips on 386 machines and requires roughly one half the memory of Clips. Eclipse supports more advanced functionality and better integration, especially on personal computers. It supports more powerful logical reasoning capabilities and provides more expressive pattern matching.

One of Eclipse's best features is its integration capacity. It supports two-way integration, integrating other applications into Eclipse, and Eclipse into other applications. Other applications can be integrated into Eclipse by calling routines written in the C language, in some cases even Pascal or FORTRAN. Eclipse toolkits are provided with run-time libraries and source code necessary to construct applications which embed Eclipse. It allows declarative domain and procedural control knowledge to be explicitly separated and codified using rules or

2. The Haley Enterprise, Inc. 1992. Company brochure. Sewickley, PA.

procedures, retaining more of the advantages of a pure production system even when the requirement to procedurally control and embed a knowledge-base is implemented within any real application.

The Microsoft Windows version of Eclipse has a real-time WYSIWYG (What You See Is What You Get) development interface. It provides knowledge base monitoring windows which can be used to simultaneously view the relations between facts and goals, and the match states of any number of rule sets, agenda, rules, joins, patterns, templates, relations, facts, matches, etc. Each of these views is supported in real time as commands or rules are executed. In effect, these modeless dialog windows allow full manipulation and monitoring of all knowledge base components, scheduling and pattern matching activities. The windows version also supports dynamic data exchange (DDE). This support provides more communication methods between Eclipse and other applications. Using this feature, Eclipse commands can be passed from other applications and Eclipse can issue commands to other applications. Eclipse can request and respond to requests for data from other applications. Dynamic link libraries (DLLs) are also provided in the Windows version. This research uses the Windows version of Eclipse.

4.5 Prototype Development

Expert systems are best developed using an incremental prototyping methodology. A running prototype should be developed at an early stage. The prototype deals with a restricted portion of the problem domain; nonetheless, it will validate the feasibility and conceptual design of the system. After the prototype has been constructed, it can be extended into a fully functional system while its basic structure is preserved.

4.5.1 Prototype Components

The prototype for this project has four basic components as shown in Figure 4.2. The user interface collects data from the user and provides the grading results. The inference engine is a built-in component of Eclipse. The knowledge base contains selected grading rules. The explanation facility consists of a set of rules for tracing back the facts upon which the grading results are based.

At the beginning of system execution, the user loads the knowledge base, and then enters data describing lumber size and defect characteristics via the user interface. The inference engine tries to map the input data to one or more proper grades by performing the reasoning process on the input data and the knowledge base. Finally, the user

interface shows the final grades resulting from the matching process, along with explanations.

The knowledge base is implemented in several source files. Each file contains grading rules for a sub-category of a category (see Table 4.11 for explanation). The user can load one or more of them, depending on the grading categories to be worked on. The user interface is implemented using rules and functions, and is contained in

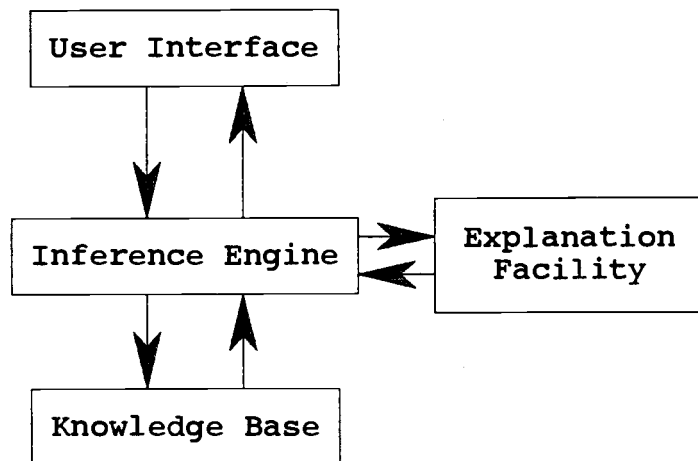


Figure 4.2. Basic structure of the prototype

a separate file. Also, separate files contain output printing and some supplemental control strategies. Table 4.11 summarizes the file list used in the system.

When using the Eclipse development toolkits, the development interface allows the system developer to watch how conclusions are reached and to monitor all facts in the knowledge base. Thus, an explanation facility is not necessary. However, the run-time version of the prototype does not include those knowledge-base debugging facilities.

Table 4.11. Source files and their contents

File Name	Content
Alt.clp	Grading rules for ALTERNATE sub-category of BOARD
Commons.clp	Grading rules for COMMONS sub-category of BOARD
Define.clp	Definition of relations used to describe facts
Finish.clp	Grading rules for FINISH sub-category of SELECT/FINISH
Lframe.clp	Grading rules for LIGHT FRAMING sub-category of DIMENSION
Query.clp	Rules and functions for query interface.
Select.clp	Grading rules for SELECT sub-category of SELECT/FINISH
Sframe.clp	Grading rules for STRUCTURAL sub-category of DIMENSION
Stud.clp	Grading rules for STUD sub-category of DIMENSION
System.clp	Rules for system control, explanation, and output printing

Instead, a simple explanation facility is coded into a set of rules contained in the system.clp file to explain how the conclusions are reached. If used as an instruction tool, the prototype should be expanded by developing more detailed explanation features.

4.5.2 Knowledge Base Implementation

The domain knowledge is described by objects, concepts, and relationships between them. The major objects in the problem domain are those defects considered by the system, including various types of stains, checks, splits, wane, knots, holes, pitch, and pockets. The major concepts are lumber grades.

As discussed earlier, this system is a rule based system, in which the knowledge base consists of a collection of rules completely describing the relationships between the objects and concepts. In Eclipse syntax, the general format of a rule is as follows:

```
(defrule rule_name "optional_comment"
  (pattern 1)
  (pattern 2)
  .
  .
  .
  (pattern N)
=>
  (action 1)
  (action 2)
  .
  .
  .
  (action M))
```

The first line is the rule header; "defrule" is a keyword indicating that this segment of code is a rule. Zero or more patterns are specified after the rule header. Each pattern consists of one or more fields. The patterns before the arrow symbol, "=>", are antecedents of an IF-THEN rule, and called the left-hand of the rule. The actions after "=>" are the THEN part of the IF-THEN rule, called the right-hand of the rule. Eclipse attempts to match the patterns of rules against existing facts. If all the patterns of a rule match facts, the rule is activated and put into an agenda (also called conflict set). The agenda is a collection of activated rules. Zero or more rules may be in the agenda. When more than one rule is in the agenda, Eclipse applies a conflict resolution procedure to select a rule to execute, called firing a rule. The actions following the arrow symbol will be executed when the rule in the agenda fires. Patterns, or antecedents of a rule are facts that are required to fire the rule. Facts are usually thought of as propositions, as in proposition calculus. A propositional fact is a fact that consists of a relation and one or more arguments following the relation. The relation is always in the first place in the fact with arguments following the relation. The relation describes how the argument relate to each other. For instance, a fact may be represented as (dia hole 0.5), where dia (for diameter) is the relation, hole and the

value of 0.5 are arguments. This pattern means that the hole has a diameter of 0.5 (inches).

To illustrate how a rule is implemented in Eclipse, consider Rule 2 in Table 4.10. The rule is modified to accommodate knot description. As stated in Table 4.10, the rule only considers knot type (firm or loose), but not knot location (at edge or center line). However, some grades in the light framing sub-category need to consider both knot type and location. To keep the query interface consistent, Rule 2 is modified as follows:

Modified Rule 2:

```

IF    NO firm knots at edge
      OR (lumber width = 2" AND diameter of largest
          firm knot at edge <= 0.75")
      OR (lumber width = 3" AND diameter of largest
          firm knot at edge <= 1.25")
      OR (lumber width = 4" AND diameter of largest
          firm knot at edge <= 1.5")
THEN firm knots at edge satisfy grade CONSTRUCTION of
Light Framing

```

The rule is implemented in Eclipse as follows:

```

(defrule CONST-LF-firm-edge-knot
  (declare (salience 1000))
  (face ?)
  (category light-framing)
  (face_width ?width)
  (or (not (symptom firm_knot_at_edge))
      (dia firm_knot_at_edge ?firmdia&:( or
        (and (= ?width 2) (<= ?firmdia 0.75))
        (and (= ?width 3) (<= ?firmdia 1.25))
        (and (= ?width 4) (<= ?firmdia 1.5))))))
=>
  (assert (satisfy firm_edge_knot CONST-LF)))

```

The name of this rule is CONST-LF-firm-edge-knot. The patterns before "=>" look more complicated than the general

format described above because many Eclipse-defined functions are used. A function looks like a fact pattern. The entire function is surrounded by parentheses. The name of function precedes the arguments. The arguments for the function are patterns of facts or other functions. Here "declare", "or", "not", "and", and "assert" are names of functions which perform the functions implied by their names. Any field with the symbol ? means that value of this field is a variable.

Comparing the modified Rule 2 and its Eclipse implementation, it becomes apparent that the first four patterns, or antecedents, within the implementation are not present in the modified Rule 2. These are supplemental patterns used for process control. The patterns after these process control patterns are just the Rule 2 expressed using Eclipse's syntax. An explanation of each component of the rule is given below.

1. (declare (salience 1000)): assigns a priority number to this rule. "Salience" is a keyword to indicate the priority of the rule, followed by a priority value. The system is driven by facts in the working memory. The system starts with matching the working memory facts with antecedents of rules. Once all antecedents of a rule are matched, that rule is put in the agenda. In many case more than one rule will be in the agenda waiting for firing. The system then applies a conflict resolution strategy to

fire those rules. However, sometimes priority ordering other than the Eclipse's predefined control strategy is necessary; "salience" orders the rule execution sequence.

2. (face ?): This has a special purpose. If both faces have the same defect characteristic described by a rule, without (face ?) once the first face has used in a rule, that rule will not be fired again for the second face, even if all the facts were present. This is due to the design of the Eclipse system. When grading different faces, ? in the pattern of (face ?) will have a different value. Therefore, even if no other facts in the rule change, the rule will still be activated because (face ?) pattern has changed.

3. (category light-framing): is used to prevent other grading categories from using this rule. This fact will be asserted into working memory during the query process when the user selects this category. Without using this pattern, Eclipse will try to match all antecedents of all grading rules, no matter what grading category the system is working on. That would be extremely inefficient.

4. (face_width ?width): is used to introduce variable ?width required in other patterns.

5. The patterns after (face_width ?width) and before "=>" are just statements of the modified Rule 2 in Eclipse syntax. They are written in compound logical "or", "not", and "and" format. The symbol "&:" has a special meaning.

The variable "?firmdia" before "&:" is tested in the calculations following "&:". For example, in the fact (dia firm_edge_knot 0.5), the ?firmdia is 0.5. If the lumber width is 2 inches, the ?firmdia will be tested to see whether the diameter of the knot is less than 0.75. The result will be true, and the system considers that the antecedent has been matched by the fact. If all the facts on the left-hand side of the rule are present in the working memory, the rule will be put on the agenda, and the new fact (satisfy firm_edge_knot CONST-LF) will be asserted into the working memory.

6. In the pattern of (satisfy firm_edge_knot CONST-LF), "satisfy" describes the relationship between the knot and the grade. Argument firm_edge_knot is an abstracted object, which is a combination of a knot object and its two attributes, type and location. Alternatively, the knot object can be represented using knot-type-value and knot-location-value triples, or using a knot template structure with type and location as its attributes. As a matter of fact, Eclipse has a "deftemplate" representation to support this kind of structure. Using deftemplate to represent a knot object was tried during prototyping, but was less preferable because of some inconvenience encountered in fact manipulation.

All grading rules in the knowledge base have a similar structure. The query interface, system output, and system

controls are all implemented using such rules. To illustrate, Appendix A gives the Eclipse implementation of all grading rules for the Structural Light Framing category.

4.5.3 Grading Process, Input, and Output

The system is menu-driven and interacts with the user through menus and a series of dialog boxes. The input process begins with loading the knowledge base and selecting the input method (interactively or loading a file containing facts). If the interactive input method is selected, a series of dialog boxes will guide the user to enter lumber size, select grade category and sub-categories, and specify the face where defects are located along with the defects' types, locations and sizes. The system then infers the grade corresponding to this face. Once the grade has been determined, the system allows the user to ask for explanation. The explanation provided by the system includes types of defects on the face, size of each defect, the grade each defect has satisfied, and the defects that determine the grade.

The system determines the grade for each face separately and the overall grade. The flow chart in Figure 4.3 shows the grading process. Figures 4.4 to 4.15 explain the various menus and dialog boxes in details. Initially, the prototype only considered the two wide faces. After

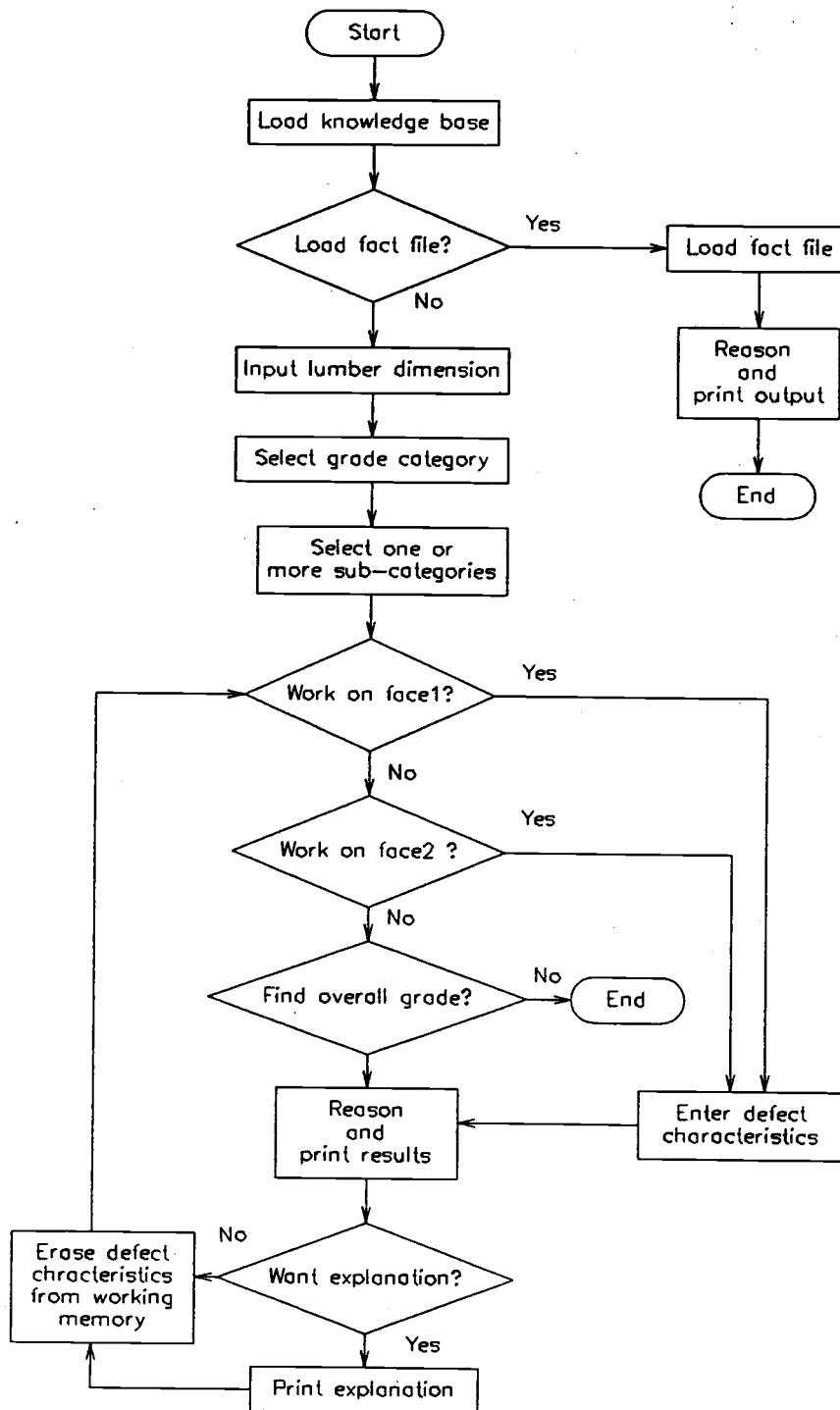


Figure 4.3. Grading process (This prototype has been improved by including all four faces as described in Chapter 5)

prototype evaluation, the system was enhanced to consider all four faces (see Chapter 5).

The system's initial window is shown in Figure 4.4. This window contains several pull-down menus for loading the knowledge base, cleaning memory space, and controlling system execution. The user can load a selected knowledge base into the system. Grading categories can be placed into one knowledge base, or several different knowledge bases, depending upon the user's preference. The development and maintenance of knowledge bases are done using Eclipse's development tool kits. Once the knowledge bases are error free, they can be saved as binary files. The run-time version of the system can only load knowledge bases in binary files. Binary files are efficient for fast loading and secure delivery.

The system allows the user to input lumber characteristics in two ways: either by entering the facts during the query process, or by loading a fact file at the beginning of the session (Figure 4.5). Once the user selects to input facts interactively, a series of dialog boxes will pop up to guide the user to enter the facts. First, the user enters lumber thickness (Figure 4.6); this is followed by lumber width and length. Each input dialog box is followed by a confirmation dialog box (Figure 4.7) so that the user has a chance to modify the input data.

Next the system asks the user to select a grade category (Figure 4.8) and sub-categories in the category (Figure 4.9). Although the same piece of lumber may have more than one grade in different categories, the current version of the system does not consider grade overlap between different categories. However, the system allows the user to select one or all sub-categories in the category, that is, grade overlap is allowed between sub-categories.

After selecting grade sub-categories, the user specifies the face of lumber to be graded (Figure 4.10) and then enters defects on this face. After a face has been graded, the dialog box in Figure 4.10 may again be used to specify another face. The overall grade is determined after both faces have been graded.

Figures 4.11 and 4.12 show the dialog boxes for entering defects. The defects shown in the figures include all defect types considered in the system. If the same defect type occurs more than once, only the worst defect is input. For example, if there are two holes, 0.1 inches and 0.5 inches in diameter, the larger hole (0.5 inches) is entered into the system.

Once a defect type has been selected, a dialog box will pop up to ask for more detailed information about the defect type and location, as appropriate. Figure 4.13 shows a dialog box for entering information on knots. This

dialog box asks for knot type and location. The information in the dialog box may differ across grade categories as various categories consider knot types differently. The last piece of information required for a defect is its size, as shown in Figure 4.14.

After all the defect data is entered, the user selects the `no_more_defects` item in Figure 4.12 to exit the input mode. The system maps the input data to appropriate grade(s). Figure 4.15 shows an example output. The output includes each face grade based on the defect distribution on that face, the overall lumber grade, and the corresponding sub-category. Overlap between sub-categories is allowed; Figure 4.15 shows the grades for a piece of dimension lumber.

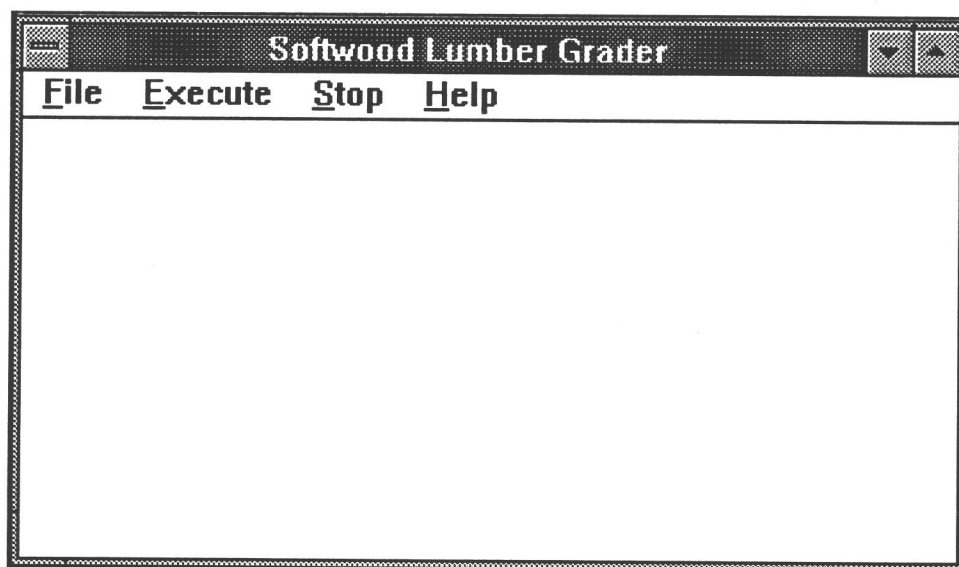


Figure 4.4. Opening window of the run-time version of the system

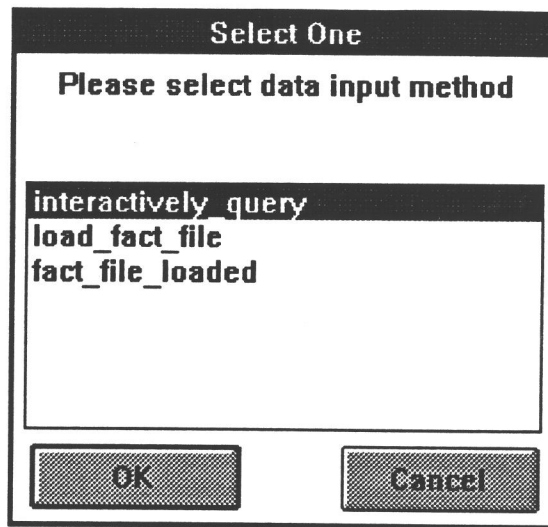


Figure 4.5. Dialog box for selecting input method

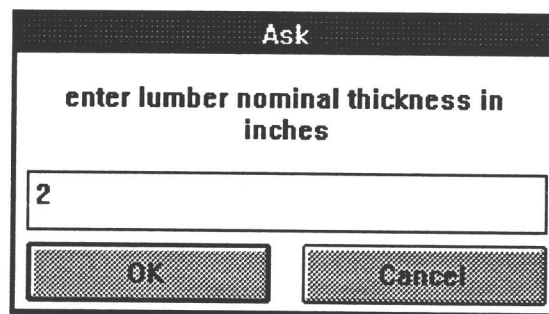


Figure 4.6. Dialog box for entering lumber thickness

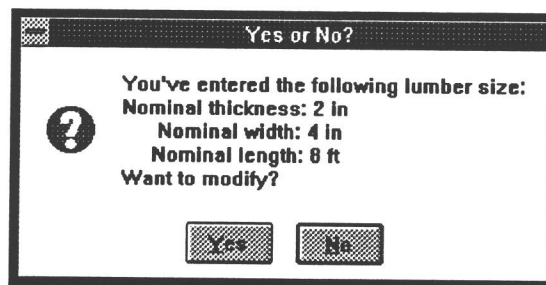


Figure 4.7. Dialog box for confirming lumber dimension

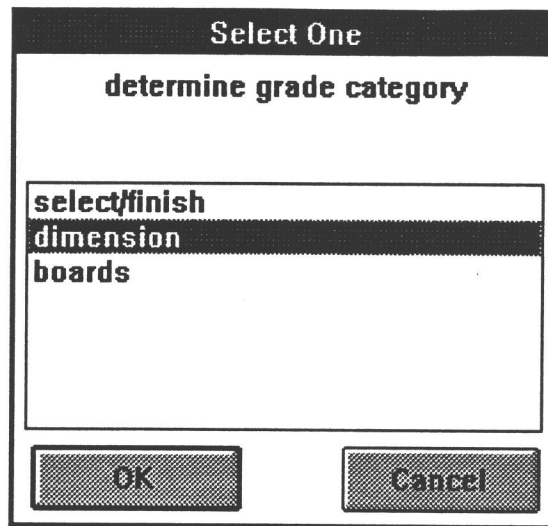


Figure 4.8. Dialog box for selecting grade category

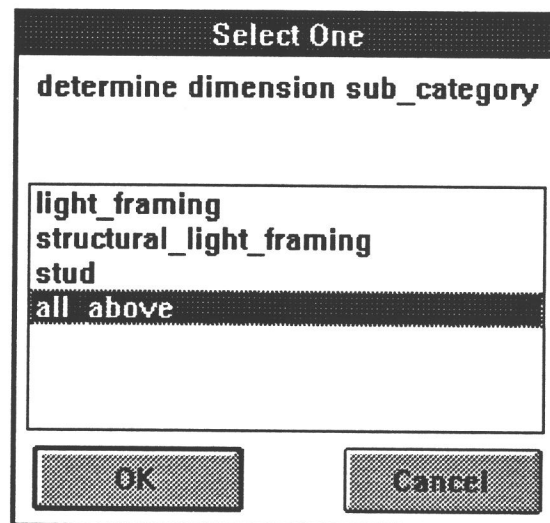


Figure 4.9. Dialog box for selecting Dimension sub-category

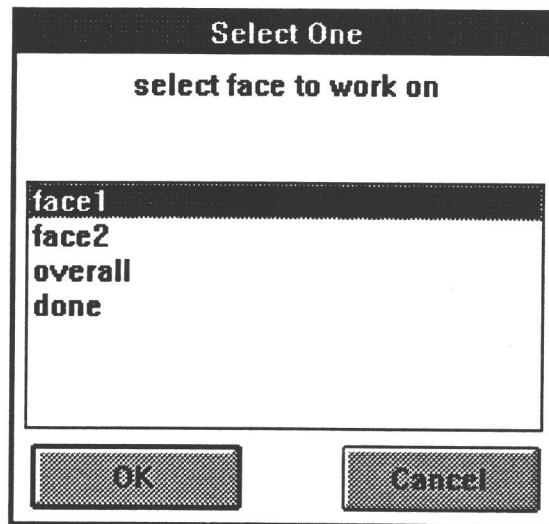


Figure 4.10. Dialog box for selecting lumber face

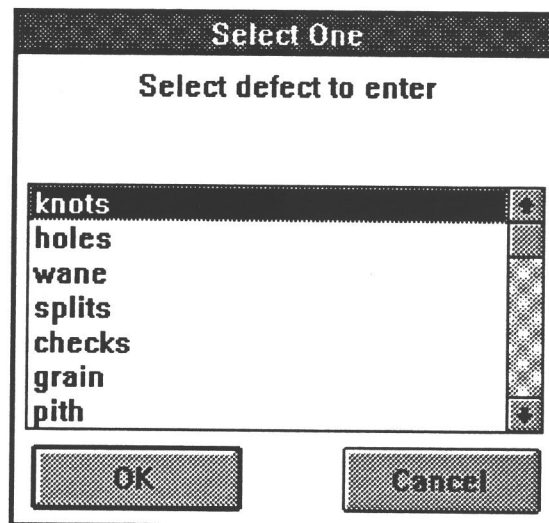


Figure 4.11. Dialog box with partial defect list

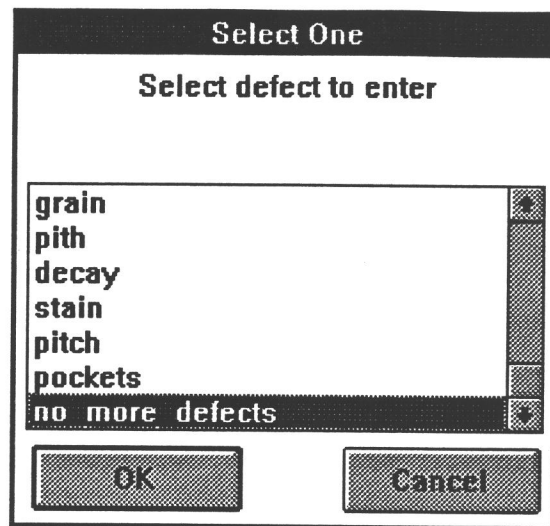


Figure 4.12. Dialog box with another partial defect list

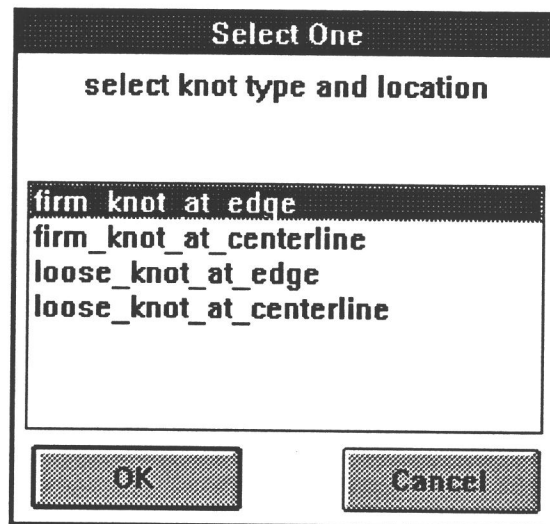


Figure 4.13. Dialog box for entering knot type and location

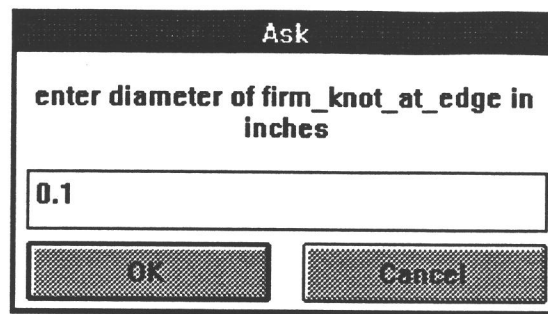


Figure 4.14. Dialog box for entering knot diameter

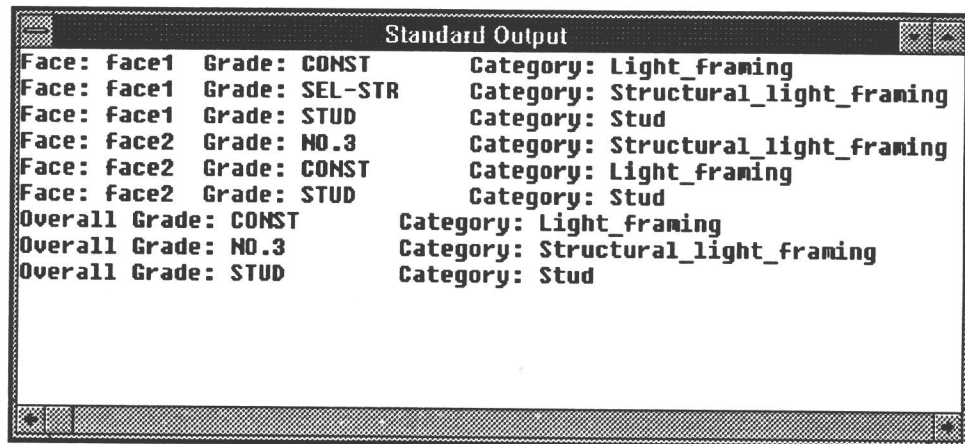


Figure 4.15. System output display

CHAPTER 5

EVALUATION AND ENHANCEMENTS

5.1 System Evaluation

The system was initially tested using generated input data. Simulated lumber and defect data were entered into the system. The resulting grades assigned by the system were compared with those assigned by using the rule book manually. All rules in the system were checked and 100 percent match was achieved for the simulated data. The test results are given in Appendix B.

The system was then evaluated using real lumber. Eighty five samples of Siberian larch 2x4x12 lumber were used to evaluate the system. The samples had been graded by a human grader from the West Coast Lumber Inspection Bureau (WCLIB). All samples belong to the Structural Light Framing category. Among the five grades in this category, 20 sample pieces were randomly selected from 61 pieces of Select Structural lumber, 20 from 31 pieces of No.1, 20 from 53 No.2, 20 from 73 pieces of No.3, and 5 from 10 Economy.

To test the performance of the system, defects on these pieces were measured by the author. These were then entered into the system. As shown in Figure 5.1, 60 percent of the samples (51 pieces) matched the original

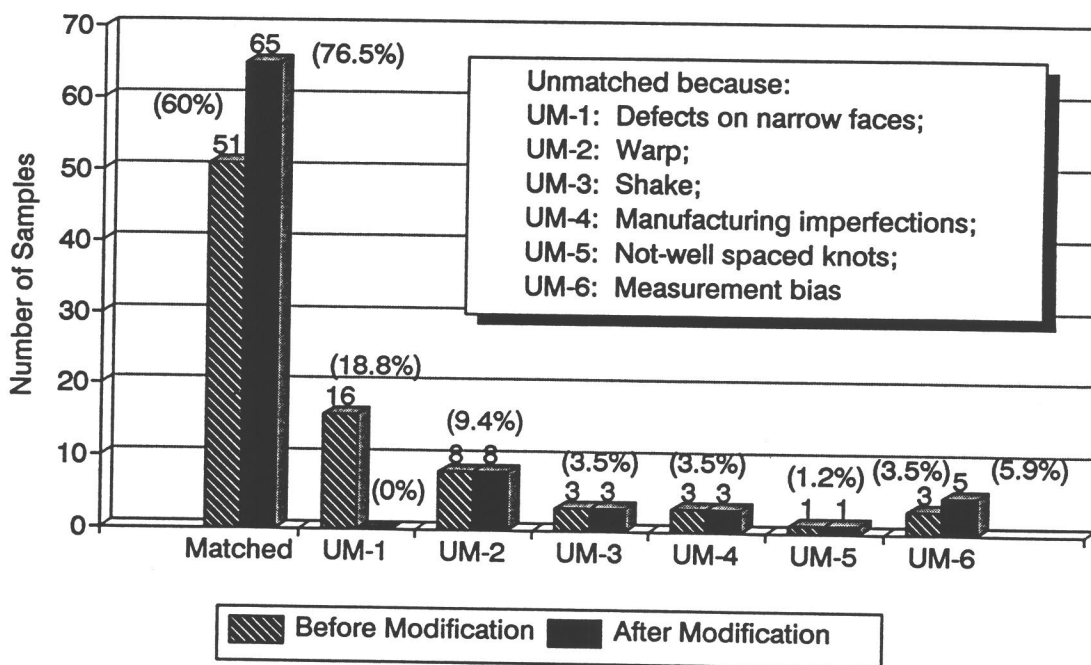


Figure 5.1. Evaluation results considering all

factors grades as judged by the human grader. A total of 31 pieces (36.5 percent) did not match the original grades because of limitations of the computer system, primarily not considering defects on the two narrow faces (edges), shake, warp, and manufacturing imperfections. An

additional 3.5 percent samples (3 pieces) did not match the grades assigned because of measurement bias or the mismatch could not be explained. For example, a firm knot at edge on a piece was measured as being 1.1 inches in diameter, which made this piece No.2. However, for some reason the human grader assigned Select Structural grade to the piece. Since the human grader was not present, it was difficult to correlate some of the mismatched observations.

If used as a sub-system of SAW3D, warp and manufacturing imperfections do not exist when SAW3D making decisions. If these defects are excluded, the matched rate between the grade assigned by the human grader and that assigned by the system is 72.9 percent.

Although some marginal errors are introduced due to measurement bias, a difference of 40 percent requires modifications of the system to improve its performance.

5.2 System Improvement

As shown in Figure 5.1, the largest cause of deviation between the grades assigned by the computer and those assigned by the human grader was exclusion of defects on the two narrow faces. Thus the focus of improvement was on including the narrow faces in the evaluation process.

The system was improved to allow defects on narrow faces to be input and evaluated. The enhanced model allows

the user to input defects on all four faces, assigns a grade to each of the four faces, and then determines the overall grade.

The eighty five samples tested earlier were again evaluated using the revised system. Figure 5.1 shows the grading results after the modification. The percentage of matched grades increased from 60 to 76.5 percent. The samples that did not match due to system limitations decreased from 36.5 to 17.6 percent. Samples which were assigned a grade different from that assigned by the human grader because of measurement error and unexplained differences increased from 3.5 to 5.9 percent.

All 15 unmatched samples caused by system limitations are upgraded in the revised system because defects such as shakes, warp and manufacturing imperfections are not considered by the system. All five unmatched samples caused by measurement difference are downgraded because defect measurements are larger than those permitted by the original grades.

Although warp and manufacturing imperfections also contributed to deviations in grade assignment, these are introduced as a result of the manufacturing process and do not exist before log breakdown. As such, they are not considered in a log breakdown model such as SAW3D. If the system is to be used for a log breakdown optimization system where warps and manufacturing imperfections are not

present, the percentage of matched samples increases to 72.9 percent before the modification, and to 89.4 percent after the modification (Figure 5.2). However, if the system is to be used as an instruction tool or a component of an automatic grading system, warp and manufacturing imperfection are important factors that affect grading accuracy, and therefore need to be included in the system.

The bias due to measurement error can be eliminated by having the individual who visually graded the samples also

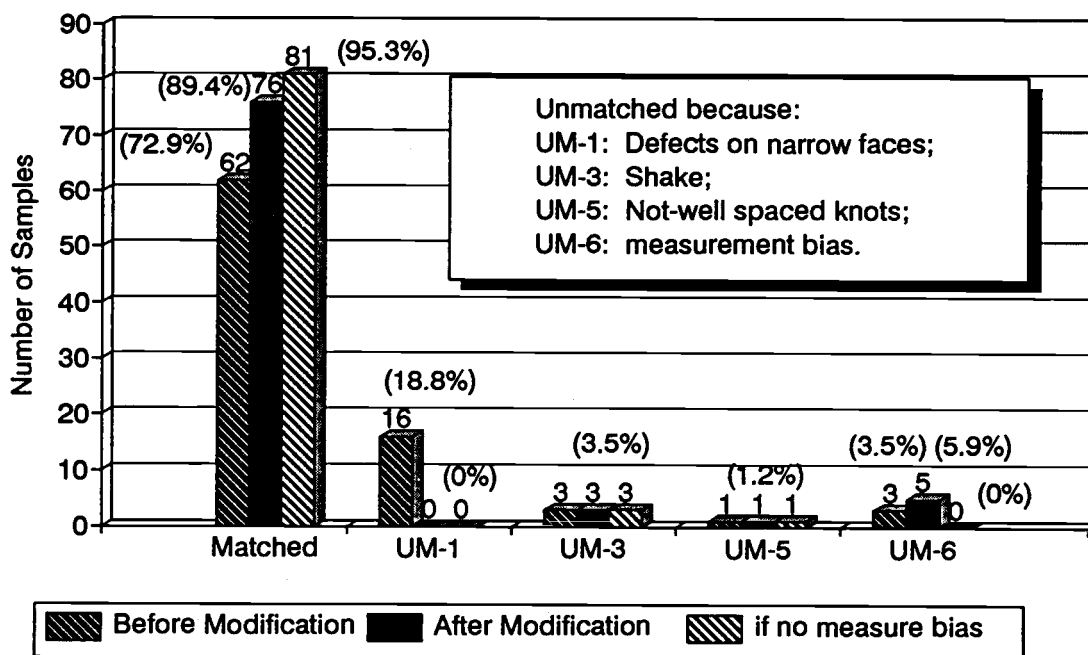


Figure 5.2. Evaluation results when warp, manufacturing imperfections, and measurement bias were not considered

measure the defects. In this case the matched samples represent 95.3 percent of the total boards (Figure 5.2). The number of unmatched samples due to shake and not-well-spaced knots is very small. These may also depend upon wood species; inclusion of these factors will further improve the system accuracy.

CHAPTER 6

CONCLUSIONS

A prototype expert system for softwood lumber grading was developed. The prototype can grade lumber for 27 grades in Dimension, Select/Finish, and Boards category. It can judge each of four faces of lumber individually and assign an overall grade based on the grades of the individual faces. The user interface helps to obtain facts about a piece of lumber, and provides grading results and explanations about how the conclusions are reached. The initial prototype considered only two faces of lumber. After an evaluation using real lumber samples, the prototype was modified to include all four faces resulting in significant performance improvement.

The intended use of the system is in conjunction with SAW3D, a log breakdown optimization system, to determine optimum log breakdown decisions. SAW3D will provide this grading system with lumber size and defect characteristics information. The expert system determines the lumber grade and will then pass this information back to SAW3D where SAW3D will determine the lumber value based on the grade, and use it to arrive at optimum log breakdown decisions. This application represents a new direction for improving lumber production efficiency: optimization of cutting strategies using both lumber geometric and defect

information.

The grading system can be used to build and maintain grading rules independently, and communicate with other components of the overall system in different ways. It can be embedded in the overall system, or it can exchange data with the system using Dynamic Data Exchange facility supported by MS-WINDOWS. The separation of the grading system from other parts of the overall system is very important. The knowledge base can be developed and maintained without interfering with any other program logic. This provides greater portability and flexibility of the grading system, and in turn simplifies the development and maintenance of the system.

Areas for further research include:

1. The evaluation using real lumber was only performed for the Structural Light Framing sub-category of the Dimension. Such a evaluation should be done for all grading categories.

2. For rapid prototyping, the development concentrates on selected defect characteristics for each grade. The selected characteristics are clearly defined and described in the rule book, and are usually the most frequently encountered defects on softwood lumber. Once the specific use of the system is determined, the knowledge base should be refined and more rules should be added to the knowledge base with the help of human grading experts. For example,

knot positions relative to each other, shake, warp and manufacturing imperfections need to be considered. Clustering of defects is another important feature that may effect a grade, and thus may need to be included in the knowledge base.

3. As mentioned before, this system is to be used as a sub-system of SAW3D. The system may also be used as an instruction or training tool. When used as an instruction tool, a better user interface should be developed. A help facility should be added to guide the user at each step, and more detailed explanations should also be provided at each step. A graphic display of defect distribution and grade stamp bitmaps may be very helpful. The system may need to include some controls allowing the user to grade the lumber, and then judge the results using the knowledge base.

BIBLIOGRAPHY

1. American Lumber Standards Committee. 1970. American Softwood Lumber Standard. National Bureau of Standards, U.S. Department of Commerce, Washington, D.C.
2. Aune, Jan. 1991. X-Ray edger optimizer makes money at Macmillan Bloederl's Alberni Pacific Division. Proceedings of the 4th International Conference on Scanning Technology in the Wood Industry, pp. Aune-1 - Aune-2, San Francisco, CA.
3. Brooke, Tom. 1992. The art of production systems. AI Expert, 7(1):31-35.
4. Flatman, Carl, and Egan Bodell. 1989. Computer-aided lumber grader-optimizer system. Proceedings of the Third International Conference on Scanning Technology in Sawmilling, pp. XIX-1 to XIX-15, San Francisco, CA.
5. Flatman, Carl, and Daniel J. Kenway. 1991. Mill experience with high speed grade based optimizers. Proceedings of the 4th International Conference on Scanning Technology in the Wood Industry, pp. kenway-1 to kenway-14, San Francisco, CA.
6. Forgy, L. Charles. 1979. On the efficient implementation of production systems. Ph.D Thesis. Carnegie-Mellon University.
7. Forgy, L. Charles. 1982. Rete: a fast algorithm for the many pattern/many object match problem. Artificial Intelligence, 19(1):17-37.
8. Haley Enterprise. 1992. Eclipse Reference Manual. The Haley Enterprise, Inc., Sewickley, PA.
9. Hallock, H., and L. Galiger. 1971. Grading hardwood lumber by computer. Res. Paper FPL-157. Forest Products Laboratory, Madison, WI.
10. Huang, S.S.L., and F.T. Sparrow. 1989. A computer-aided instruction tool for grading hardwood lumber. Forest Products Journal, 39(10):39-42.
11. Kenway, J. Daniel. 1990. A supercomputer-based machine vision grading system and trimmer optimizer for dimension lumber. Proceedings of Process Control/Production Management of Wood Products: Technology for the '90s, pp. 53-65, Athens, Georgia.

12. Klinkhachorn, P., J.P. Franklin, C.W. McMillin, R.W. Conners, and H.A. Huber. 1988. Automated computer grading of hardwood lumber. Forest Products Journal, 38(3):67-69.
13. Klinkhachorn, P., C.J. Schwehm, C.W. McMillin, and H.A. Huber. 1989. HaLT: a computerized training program for hardwood lumber grading. Forest Products Journal, 39(2):38-40.
14. Klinkhachorn, P., D. Yost, R. Kothari, C.W. McMillin, H.A. Huber. 1991. Computer lumber grading and HaRem for an automated lumber processing system. Proceedings of the First International Conference on Automated Lumber Processing Systems and Laser Machining of Wood, pp. 173-181, Michigan State University, East Lansing, MI.
15. Western Wood Products Association. Western lumber grading rule 88. 1988. Portland, OR.
16. Williston, Edward M. 1988. Lumber manufacturing - The design and operation of sawmills and planer mills. Miller Freeman Publications, Inc.
17. Zeng, Y. 1991. Log breakdown using dynamic programming and 3-D log shape. M.S. thesis. Oregon State University, Corvallis, OR.

APPENDICES

Appendix A. Source Code for Structural Light Framing Category

```

;;;=====
; This file contains rules for Structural Light Framing
; category of Dimension lumber grades. This category
; contains the following four grades:
;
;     1) SEL-STR      (Select Structural, Structural Light
;                    Framing)
;     2) NO.1        (NO. 1, Structural Light Framing)
;     3) NO.2        (NO. 2, Structural Light Framing)
;     4) NO.3        (NO. 3, Structural Light Framing)
;     5) ECONOMY-SLF (Economy, Structural Light Framing)
;
; Validation of defects for each grade is checked by the
; rules separated from the grading rule.
;;;=====

;;;===== SEL-STR GRADE =====

;----Rules to check validation of defects for SEL-STR ----

(defrule SEL-STR_firm_edge_knot "Firm knots at edge <=
SEL-STR limit?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (face_width ?width)
  (or (not (symptom firm_knot_at_edge))
      (dia firm_knot_at_edge ?firmdia:( or
        (and (= ?width 2) (<= ?firmdia 0.375))
        (and (= ?width 3) (<= ?firmdia 0.5))
        (and (= ?width 4) (<= ?firmdia 0.75))))))
  =>
  (assert (satisfy firm_knot_at_edge SEL-STR)))

;-----

(defrule SEL-STR_firm_center_knot "Firm knots at center <=
SEL-STR limit?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (face_width ?width)
  (or (not (symptom firm_knot_at_centerline))
      (dia firm_knot_at_centerline ?firmdia:( or
        (and (= ?width 2) (<= ?firmdia 0.375))
        (and (= ?width 3) (<= ?firmdia 0.5))
        (and (= ?width 4) (<= ?firmdia 0.875))))))
  =>
  (assert (satisfy firm_knot_at_centerline SEL-STR)))

```

```

;-----
(defrule SEL-STR_loose_edge_knot "Loose edge knots <=
SEL-STR grade limit?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (face_width ?width)
  (or (not (symptom loose_knot_at_edge))
      (dia loose_knot_at_edge ?loosedia&:( or
        (and (= ?width 2) (<= ?loosedia 0.375))
        (and (= ?width 3) (<= ?loosedia 0.5))
        (and (= ?width 4) (<= ?loosedia 0.75))))))
=>
  (assert (satisfy loose_knot_at_edge SEL-STR))

```

```

;-----
(defrule SEL-STR_loose_center_knot "Loose cnt knots <=
SEL-STR grade limit?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (face_width ?width)
  (or (not (symptom loose_knot_at_centerline))
      (dia loose_knot_at_centerline ?loosedia&:( or
        (and (= ?width 2) (<= ?loosedia 0.375))
        (and (= ?width 3) (<= ?loosedia 0.5))
        (and (= ?width 4) (<= ?loosedia 0.75))))))
=>
  (assert (satisfy loose_knot_at_centerline SEL-STR))

```

```

;-----
(defrule SEL-STR_hole "Holes <= SEL-STR grade limit ?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (face_width ?width)
  (or (not (symptom holes))
      (dia hole ?holedia&:( or
        (and (= ?width 2) (<= ?holedia 0.375))
        (and (= ?width 3) (<= ?holedia 0.5))
        (and (= ?width 4) (<= ?holedia 0.75))))))
=>
  (assert (satisfy holes SEL-STR) )

```

```

;-----
(defrule SEL-STR_wane "Wanes <= SEL-STR grade limit ?"
  (declare (salience 1000))
  (face ?)

```

```

(category structural_light_framing)
(lumber_thickness ?thickness)
(lumber_width ?width)
(or (not (symptom wane))
    (and (thickness wane ?thx&:( <= ?thx (/ ?thickness
                                     4.0)))
         (width wane ?wid&:( <= ?wid (/ ?width 4.0))))))
=>
(assert (satisfy wane SEL-STR) ))

;-----

(defrule SEL-STR_split "Splits <= SEL-STR grade limit ?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (lumber_width ?width)
  (or (not (symptom splits))
      (length split ?splitlength&:(<= (- ?splitlength
                                             ?width) 0.000001)))
=>
(assert (satisfy splits SEL-STR) ))

;-----

(defrule SEL-STR_check "Checks <= SEL-STR grade limit ?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (lumber_width ?width)
  (or (not (symptom through_check_on_end))
      (length through_check_on_end
               ?checklength&:(<= (- ?checklength ?width) 0.000001)))
=>
(assert (satisfy through_check_on_end SEL-STR) ))

;-----

(defrule SEL-STR_slope "Slope within SEL-STR grade limit ?"
  (declare (salience 1000))
  (face ?)
  (category structural_light_framing)
  (or (not (symptom grain))
      (slope_of_grain ?slope&:( <= ?slope (/ 1 12.0))))
=>
(assert (satisfy grain SEL-STR) ))

;-----
;----- Grading rule for grade of SEL-STR -----
;-----

(defrule grade_SEL-STR "SEL-STR Grade"
  (declare (salience 900))

```

```

(not (symptom decay))
(satisfy firm_knot_at_edge SEL-STR)
(satisfy firm_knot_at_centerline SEL-STR)
(satisfy loose_knot_at_edge SEL-STR)
(satisfy loose_knot_at_centerline SEL-STR)
(satisfy holes SEL-STR)
(satisfy wane SEL-STR)
(satisfy splits SEL-STR)
(satisfy through_check_on_end SEL-STR)
(satisfy grain SEL-STR)
=>
(assert (grade Structural_light_framing SEL-STR 1)))

;;;===== NO.1 GRADE =====

;----- Rules to check validation of defects for NO.1 -----

(defrule NO.1_firm_edge_knot    "Firm knots at edge <= NO.1
grade limit ?"
  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (face_width ?width)
  (or (not (symptom firm_knot_at_edge))
      (dia firm_knot_at_edge ?knotdia:( or
        (and (= ?width 2) (<= ?knotdia 0.5))
        (and (= ?width 3) (<= ?knotdia 0.75))
        (and (= ?width 4) (<= ?knotdia 1.0))))))
  =>
  (assert (satisfy firm_knot_at_edge NO.1)))

;-----

(defrule NO.1_firm_center_knot    "Firm knots at center <=
NO.1 grade limit ?"
  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (face_width ?width)
  (or (not (symptom firm_knot_at_centerline))
      (dia firm_knot_at_centerline ?knotdia:( or
        (and (= ?width 2) (<= ?knotdia 0.5))
        (and (= ?width 3) (<= ?knotdia 0.75))
        (and (= ?width 4) (<= ?knotdia 1.5))))))
  =>
  (assert (satisfy firm_knot_at_centerline NO.1)))

;-----

(defrule NO.1_loose_edge_knot    "Loose edge knots <= NO.1

```

```

grade limit ?"
  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (face_width ?width)
  (or (not (symptom loose_knot_at_edge))
      (dia loose_knot_at_edge ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.5))
        (and (= ?width 3) (<= ?knotdia 0.75))
        (and (= ?width 4) (<= ?knotdia 1.0))))))
  =>
  (assert (satisfy loose_knot_at_edge NO.1))

;-----

(defrule NO.1_loose_center_knot    "Loose center knots <=
NO.1 grade limit ?"
  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (face_width ?width)
  (or (not (symptom loose_knot_at_centerline))
      (dia loose_knot_at_centerline ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.5))
        (and (= ?width 3) (<= ?knotdia 0.75))
        (and (= ?width 4) (<= ?knotdia 1.0))))))
  =>
  (assert (satisfy loose_knot_at_centerline NO.1))

;-----

(defrule NO.1_hole                "Holes <= NO.1 grade limit ?"

  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (face_width ?width)
  (or (not (symptom holes))
      (dia hole ?holedia&:( or
        (and (= ?width 2) (<= ?holedia 0.5))
        (and (= ?width 3) (<= ?holedia 0.75))
        (and (= ?width 4) (<= ?holedia 1.0))))))
  =>
  (assert (satisfy holes NO.1))

;-----

(defrule NO.1_wane                "wanes <= NO.1 grade limit ?"

```

```

(declare (salience 800))
(face ?)
(category structural_light_framing)
(not (grade Structural_light_framing SEL-STR 1))
  (lumber_thickness ?thickness)
(lumber_width ?width)
(or (not (symptom wane))
    (and (thickness wane ?thx&:( <= ?thx (/ ?thickness
                                         4.0)))
         (width wane ?wid&:( <= ?wid (/ ?width 4.0))))))
=>
(assert (satisfy wane NO.1))

;-----

(defrule NO.1_split      "splits <= NO.1 grade limit ?"

  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
    (lumber_width ?width)
  (or (not (symptom splits))
      (length split ?splitlength&:(<= (- ?splitlength
                                           ?width) 0.000001)))
=>
(assert (satisfy splits NO.1))

;-----

(defrule NO.1_check      "checks <= NO.1 grade limit ?"

  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
    (lumber_width ?width)
  (or (not (symptom through_check_on_end))
      (length through_check_on_end
               ?checklength&:(<= (- ?checklength ?width)
                                   0.000001)))
=>
(assert (satisfy through_check_on_end NO.1))

;-----

(defrule NO.1_slope      "slope is within NO.1 grade limit ?"
  (declare (salience 800))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
    (or (not (symptom grain))

```

```

    (slope_of_grain ?slope&:(<= ?slope (/ 1 10.0)))
=>
(assert (satisfy grain NO.1))

;-----
;----- Grading rule for grade of NO.1 -----
(defrule grade_NO.1      "NO.1 Grade"
  (declare (salience 700))
  (not (symptom decay))
  (satisfy firm_knot_at_edge NO.1)
  (satisfy firm_knot_at_centerline NO.1)
  (satisfy loose_knot_at_edge NO.1)
  (satisfy loose_knot_at_centerline NO.1)
  (satisfy holes NO.1)
  (satisfy wane NO.1)
  (satisfy splits NO.1)
  (satisfy through_check_on_end NO.1)
  (satisfy grain NO.1)
=>
  (assert (grade Structural_light_framing NO.1 2)))

;;;===== NO.2 GRADE =====
;----- Rules to check validation of defects for NO.2 -----

(defrule NO.2_firm_edge_knot      "Firm knots at edge <=
NO.2 grade limit ?"
  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (face_width ?width)
  (or (not (symptom firm_knot_at_edge))
      (dia firm_knot_at_edge ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.625))
        (and (= ?width 3) (<= ?knotdia 0.875))
        (and (= ?width 4) (<= ?knotdia 1.25)))))
=>
  (assert (satisfy firm_knot_at_edge NO.2)))

;-----

(defrule NO.2_firm_center_knot      "Firm knots at center <=
NO.2 grade limit ?"
  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (face_width ?width)
  (or (not (symptom firm_knot_at_centerline))

```

```

      (dia firm_knot_at_centerline ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.625))
        (and (= ?width 3) (<= ?knotdia 0.875))
        (and (= ?width 4) (<= ?knotdia 2.0))))))
    =>
    (assert (satisfy firm_knot_at_centerline NO.2)))
;-----

(defrule NO.2_loose_edge_knot    "Loose edge knots <= NO.2
grade limit ?"
  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (face_width ?width)
  (or (not (symptom loose_knot_at_edge))
      (dia loose_knot_at_edge ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.625))
        (and (= ?width 3) (<= ?knotdia 0.875))
        (and (= ?width 4) (<= ?knotdia 1.25))))))
    =>
    (assert (satisfy loose_knot_at_edge NO.2)))
;-----

(defrule NO.2_loose_center_knot  "Loose cnt knots <= NO.2
grade limit ?"
  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (face_width ?width)
  (or (not (symptom loose_knot_at_centerline))
      (dia loose_knot_at_centerline ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.625))
        (and (= ?width 3) (<= ?knotdia 0.875))
        (and (= ?width 4) (<= ?knotdia 2.0))))))
    =>
    (assert (satisfy loose_knot_at_centerline NO.2)))
;-----

(defrule NO.2_hole                "Holes <= NO.2 grade limit ?"

  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))

```



```

(face_width ?width)
(or (not (symptom holes))
    (dia hole ?holedia&:( or
        (and (= ?width 2) (<= ?holedia 0.625))
        (and (= ?width 3) (<= ?holedia 0.875))
        (and (= ?width 4) (<= ?holedia 1.25))))))
=>
(assert (satisfy holes NO.2)))

;-----

(defrule NO.2_wane          "wanes <= NO.2 grade limit ?"

  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (lumber_thickness ?thickness)
  (lumber_width ?width)
  (or (not (symptom wane))
      (and (thickness wane ?thx&:( <= ?thx (/ ?thickness
                                                3.0)))
           (width wane ?wid&:( <= ?wid (/ ?width 3.0))))))
=>
(assert (satisfy wane NO.2)))

;-----

(defrule NO.2_split        "splits <= NO.2 grade limit ?"

  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (lumber_width ?width)
  (or (not (symptom splits))
      (length split ?splitlength&:(<= ?splitlength (* 1.5
                                                       ?width))))))
=>
(assert (satisfy splits NO.2)))

;-----

(defrule NO.2_check        "checks <= NO.2 grade limit ?"

  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))

```

```

(lumber_width ?width)
(or (not (symptom through_check_on_end))
    (length through_check_on_end
      ?checklength&:( <= ?checklength (* 1.5 ?width))))
=>
(assert (satisfy through_check_on_end NO.2))

;-----

(defrule NO.2_slope "slope is within NO.2 grade limit ?"
  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (or (not (symptom grain))
      (slope_of_grain ?slope&:( <= ?slope (/ 1 8.0))))
=>
(assert (satisfy grain NO.2))

;-----

(defrule NO.2_decay "decay <= NO.2 grade limit ?"
  (declare (salience 600))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (lumber_thickness ?thickness)
  (lumber_width ?width)
  (or (not (symptom decay))
      (and (test (= ?thickness 2 ))
           (thickness decay ?thx&:( <= ?thx (/ ?thickness
                                                    3.0)))
           (width decay ?wid&:( <= ?wid (/ ?width 3.0))))))
=>
(assert (satisfy decay NO.2))

;-----
;----- Grading rule for grade of NO.2 -----
;-----

(defrule grade_NO.2 "NO.2 Grade"
  (declare (salience 500))
  (satisfy firm_knot_at_edge NO.2)
  (satisfy firm_knot_at_centerline NO.2)
  (satisfy loose_knot_at_edge NO.2)
  (satisfy loose_knot_at_centerline NO.2)
  (satisfy holes NO.2)
  (satisfy wane NO.2)
  (satisfy splits NO.2)
  (satisfy through_check_on_end NO.2)
  (satisfy grain NO.2)

```

```

(satisfy decay NO.2)
=>
(assert (grade Structural_light_framing NO.2 3))

;;;===== NO.3 GRADE =====

;----- Rules to check validation of defects for NO.3 -----

(defrule NO.3_firm_edge_knot      "Firm knots at edge <=
NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (face_width ?width)
  (or (not (symptom firm_knot_at_edge))
      (dia firm_knot_at_edge ?knotdia:( or
        (and (= ?width 2) (<= ?knotdia 0.75))
        (and (= ?width 3) (<= ?knotdia 1.25))
        (and (= ?width 4) (<= ?knotdia 1.75))))))
  =>
  (assert (satisfy firm_knot_at_edge NO.3)))

;-----

(defrule NO.3_firm_center_knot    "Firm knots at center <=
NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (face_width ?width)
  (or (not (symptom firm_knot_at_centerline))
      (dia firm_knot_at_centerline ?knotdia:( or
        (and (= ?width 2) (<= ?knotdia 0.75))
        (and (= ?width 3) (<= ?knotdia 1.25))
        (and (= ?width 4) (<= ?knotdia 2.5))))))
  =>
  (assert (satisfy firm_knot_at_centerline NO.3)))

;-----

(defrule NO.3_loose_edge_knot     "Firm knots at edge <=
NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))

```

```

(not (grade Structural_light_framing NO.1 2))
(not (grade Structural_light_framing NO.2 3))
(face_width ?width)
(or (not (symptom loose_knot_at_edge))
    (dia loose_knot_at_edge ?knotdia&:( or
      (and (= ?width 2) (<= ?knotdia 0.75))
      (and (= ?width 3) (<= ?knotdia 1.25))
      (and (= ?width 4) (<= ?knotdia 1.75))))))
=>
(assert (satisfy loose_knot_at_edge NO.3))

;-----

(defrule NO.3_loose_center_knot "Firm cntknots at edge <=
NO.3 grade limit?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (face_width ?width)
  (or (not (symptom loose_knot_at_centerline))
      (dia loose_knot_at_centerline ?knotdia&:( or
        (and (= ?width 2) (<= ?knotdia 0.75))
        (and (= ?width 3) (<= ?knotdia 1.25))
        (and (= ?width 4) (<= ?knotdia 2.5))))))
=>
(assert (satisfy loose_knot_at_centerline NO.3))

;-----

(defrule NO.3_hole "Holes <= NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (face_width ?width)
  (or (not (symptom holes))
      (dia hole ?holedia&:( or
        (and (= ?width 2) (<= ?holedia 0.75))
        (and (= ?width 3) (<= ?holedia 1.25))
        (and (= ?width 4) (<= ?holedia 1.75))))))
=>
(assert (satisfy holes NO.3))

;-----

(defrule NO.3_wane "wanes <= NO.3 grade limit ?"
  (declare (salience 400))

```

```

(face ?)
(category structural_light_framing)
(not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
(not (grade Structural_light_framing NO.2 3))
(lumber_thickness ?thickness)
(lumber_width ?width)
(or (not (symptom wane))
    (and (thickness wane ?thx&:( <= ?thx (/ ?thickness
                                         2.0)))
         (width wane ?wid&:( <= ?wid (/ ?width 2.0))))))
=>
(assert (satisfy wane NO.3)))

;-----

(defrule NO.3_split      "splits <= NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (lumber_length ?length)
  (not (grade Structural_light_framing SEL-STR 1))
    (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (lumber_width ?width)
  (or (not (symptom splits))
      (length split
                 ?splitlength&:(<= ?splitlength (/ (* 12
                                                         ?length) 6.0))))))
=>
(assert (satisfy splits NO.3)))

;-----

(defrule NO.3_check      "checks <= NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (lumber_length ?length)
  (not (grade Structural_light_framing SEL-STR 1))
    (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (lumber_width ?width)
  (or (not (symptom through_check_on_end))
      (length through_check_on_end
                 ?checklength&:(<= ?checklength (/ (* 12 ?length)
                                                         6.0))))))
=>
(assert (satisfy through_check_on_end NO.3)))

;-----

```

```

(defrule NO.3_slope "slope is within NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (or (not (symptom grain))
      (slope_of_grain ?slope&:(<= ?slope (/ 1 4.0))))
  =>
  (assert (satisfy grain NO.3)))

;-----

(defrule NO.3_decay "decay <= NO.3 grade limit ?"
  (declare (salience 400))
  (face ?)
  (category structural_light_framing)
  (not (grade Structural_light_framing SEL-STR 1))
  (not (grade Structural_light_framing NO.1 2))
  (not (grade Structural_light_framing NO.2 3))
  (lumber_cross_section ?lumberCS)
  (or (not (symptom decay))
      (cross_section decay ?decaySize&:(<= ?decaySize (/
        ?lumberCS 3.0))))
  =>
  (assert (satisfy decay NO.3)))

;-----
;----- Grading rule for grade of NO.3 -----
;-----

(defrule grade_NO.3 "NO.3 Grade"
  (declare (salience 300))
  (satisfy firm_knot_at_edge NO.3)
  (satisfy firm_knot_at_centerline NO.3)
  (satisfy loose_knot_at_edge NO.3)
  (satisfy loose_knot_at_centerline NO.3)
  (satisfy holes NO.3)
  (satisfy wane NO.3)
  (satisfy splits NO.3)
  (satisfy through_check_on_end NO.3)
  (satisfy grain NO.3)
  (satisfy decay NO.3)
  =>
  (assert (grade Structural_light_framing NO.3 4)))

;;;===== ECONOMY-SLF GRADE =====
;----- Grading rule for grade of ECONOMY-SLF -----
;-----

(defrule grade_ECONOMY-SLF "ECONOMY-SLF Grade"
  (declare (salience 200))
  (face ?)

```

```
(category structural_light_framing)
(not (grade Structural_light_framing SEL-STR 1))
(not (grade Structural_light_framing NO.1 2))
(not (grade Structural_light_framing NO.2 3))
(not (grade Structural_light_framing NO.3 4))
(not (symptom broken_ends ))
=>
(assert (grade Structural_light_framing ECONOMY 5))
;===== END OF STRUCTURAL LIGHT FRAMING GRADES =====
```

Appendix B. Test Results Using Generated Data

Tables B1, B3, and B5 show the grade specifications. Data in each column in the upper section of these tables represent the limiting provisions on defects; the grades are listed in the lower section. An X in the lower section means that data in the column are limiting provisions for the grade in the row. The grading rules for each grade in the knowledge base were tested using facts entered through the interactive query process. These facts represent realistic data values. Tables B2, B4, and B6 show the resulting grades. These tables use a format similar to the rule specification tables (Tables B1, B3, and B5). Each column contains data for a face of a lumber sample. Data in the top half of the table is the generated defect measurements (facts). Empty cells in a column reflect defects that do not exist in a sample. As in the specification tables, column and row intersections mean that the data (facts) in the column satisfies the grade in the row. Multiple Xs in a column means that the input data satisfy more than one grade. For example, a lumber sample with defects indicated in the first column of Table B2 satisfies CONST, NO.3, and STUD grades.

Table B1. Specifications of grades for a piece of lumber in the Dimension category (lumber size = 2"x4"x10')

Defect	Type/Location	Measure	Limiting Provisions (<= inches)											
			1.5	2	2.5	any	0.75	1	1.25	1.75	any	1.75	6	
Knot	firm-edge	dia	1.5	2	2.5	any	0.75	1	1.25	1.75	any	1.75	6	
	firm-center	dia	1.5	2	2.5	any	0.875	1.5	2	2.5	any	2.5	6	
	loose-edge	dia	1	2	2.5	any	0.75	1	1.25	1.75	any	1.75	6	
	loose-center	dia	1	2	2.5	any	0.75	1	2	2.5	any	2.5	6	
Hole		dia	1	1.25	1.5	any	0.75	1	1.25	1.75	any	1.5	any	
Wane		thx	0.5	2/3	1	any	0.5	0.5	2/3	1	any	2/3	2	
		wid	1	4/3	2	any	1	1	4/3	2	any	4/3	3	
Split		length	4	6	6	any	4	4	6	20	any	8	30	
Check	through-on-end	length	4	6	6	any	4	4	6	20	any	8	30	
Decay		thx	none	2/3	area	any	none	none	2/3	area	any	area	any	
		wid	none	4/3	8/3	any	none	none	4/3	8/3	any	8/3	any	
Grain		slope	1/6	1/4	1/4	any	1/12	1/10	1/8	1/4	any	1/4	any	
Sub-category	Grade													
Light Framing	CONST		X											
	STAND			X										
	UTIL				X									
	ECONOMY					X								
Structural Light Framing	SEL-STR						X							
	NO.1							X						
	NO.2								X					
	NO.3									X				
	ECONOMY										X			
Stud	STUD											X		
	ECONOMY												X	

- X - Data in the column is the specification for the grade in the row.
 none - Not allowed
 any - No limitations
 area - Width x length

Table B2. Test results of grading the lumber in the Dimension category

Defect	Type/Location	Measure	Facts (= inches)											
Knot	firm-edge	dia	1.5		2.5	3	0.75	1	1.25					
	firm-center	dia		2			0.875	1.5		2.5				
	loose-edge	dia	1		2.5		0.75			1.75	1.75			
	loose-center	dia		2			0.75		2		2.5			
Hole		dia	1	1.25	1.5		0.75	1	1.25	1.75	1.5			
Wane		thx	0.5	0.6	1		0.5	0.5	0.6	1	0.6			
		wid	1	1.3	2		1	1	1.3	2	2			
Split		length	4	6	6		4	4	6	20	8			
Check	through-on-end	length	4	6	6		4	4	6	20	8			
Decay		thx		0.5	1				0.66	1	1			
		wid		1	2.6				1.33	2.6	2.6			
Grain		slope	0.16	0.25	0.25		0.08	0.1	0.13	0.25	0.25			
Sub-category	Grade													
Light Framing	CONST		X				X	X						X
	STAND			X					X					
	UTIL				X									
	ECONOMY					X					X	X		
Structural Light Framing	SEL-STR						X							X
	NO.1							X						
	NO.2								X					
	NO.3		X	X							X	X		
Stud	ECONOMY				X	X								
	STUD		X	X			X	X	X			X	X	X

- 1) X indicates that input data in the column satisfies the grade in the row.
- 2) Blank cell means that no such characteristic exists.
- 3) For the same set of facts, grade overlaps are allowed. For example, facts in the first column match grades CONST, NO.3, and STUD.
- 4) The results satisfy the specification, indicating the system works well.

Table B3. Specifications of grades for a piece of lumber in the Select/Finish category (lumber size = 1"x8"x12')

Defect	Type/Location	Measure	Limiting Provisions (<= inches)					
			* 2	* 2	any	* 2	any	any
Knot	firm	number	* 2	* 2	any	* 2	any	any
		dia	0.5	0.75	any	0.75	any	any
	fixed	number	none	none	* 4	none	4	6
		dia	none	none	0.75	none	1	1.25
loose	number	none	none	none	none	none	none	
	dia	none	none	none	none	none	none	
Hole		dia	none	none	none	none	1/16	0.75
Wane	on face	thx	none	none	none	none	0.25	0.75
		wid	none	none	none	none	1	2
		length	none	none	none	none	24	48
	on back	thx	0.25	0.5	0.75	0.5	any	any
		length	24	36	36	36	any	any
Split		number	none	none	any	any	any	any
		length	none	none	24	24	24	48
Check	surface	number	3	any	any	any	any	any
		wid	1/32	1/32	1/32	1/32	1/32	any
		length	4	4	4	4	10	any
Pitch	light	wid	none	*area	any	*area	any	any
		length	none	576	any	576	any	any
	medium	wid	none	none	*area	none	*area	any
		length	none	none	768	none	768	any
	heavy	wid	none	none	none	none	none	any
		length	none	none	none	none	none	any
Pocket		number	* 1	* 2	* 4	* 4	4	6
		width	1/16	1/16	1/16	1/16	1/16	area
		length	3	3	6	6	8	4
Stain		type	light	medium	medium	medium	any	any
		wid	area	area	any	any	any	any
		length	115.2	384	any	any	any	any
Sub-category	Grade							
Select	B&BTR-1&2CLEAR	X						
	C SELECT		X					
	D SELECT			X				
Finish	SUPERIOR				X			
	PRIME					X		
	E FINISH						X	

- X - Data in the column is the specification for the grade in the row.
- none - Not allowed
- any - No limitations
- area - Width x length
- * - Only one of the characteristics with symbol * in the same column is allowed. For example, grade B&BTR-1&2CLEAR allows 2 firm knots or 1 pocket.

Table B4. Test results of grading the lumber in the Select/Finish category

Defect	Type/Location	Measure	Fact (<= Inches)					
Knot	firm	number	2			2		
		dia	0.5			0.75		
	fixed	number					4	6
		dia					1	1.25
	loose	number						
dia								
Hole		dia				1/16	0.75	
Wane	on face	thx					0.25	0.75
		wid					1	2
		length					24	48
	on back	thx	0.25	0.5	0.75	0.5		
		wid	0.5	1	2	1		
		length	24	36	36	36		
Split		number			2	1	1	1
		length			24	24	24	48
Check	surface	number	3	1	2	1	2	
		wid	1/32	1/32	1/32	1/32	1/32	
		length	4	4	4	4	10	
Pitch	light	wid						
		length						
	medium	wid			20		20	
		length			30		30	
	heavy	wid						
length								
Pocket		number		2			4	6
		width		1/16			1/16	1
		length		3			8	3
Stain		type	light	medium	medium	medium		
		wid	8	15	20	20		
		length	10	20	30	30		
Sub-category	Grade							
Select	B&BTR-1&2CLEAR	X						
	C SELECT		X					
	D SELECT			X	X			
Finish	SUPERIOR	X	X		X			
	PRIME			X		X		
	E FINISH							X

- 1) X indicates that input data in the column satisfies the grade in the row.
- 2) Blank cell means that no such characteristic exists.
- 3) For the same set of facts, grade overlaps are allowed. For example, facts in the first column match grades B&BTR-1&2CLEAR and SUPERIOR.
- 4) The results satisfy the specification, indicating the system works well.

Table B5. Specifications of grades for a piece of lumber in the Boards category (lumber size = 1"x8"x12')

Defect	Type/Location	Measure	Limiting Provisions (<= inches)									
			2.25	3	3.5	16/3	any	2.5	2.5	3.5	16/3	any
knot	firm-red	dia	2.25	3	3.5	16/3	any	2.5	2.5	3.5	16/3	any
	firm-black	dia	0.75	1.25	7/3	16/3	any	1.25	2.5	3.5	16/3	any
	loose	dia	none	none	1.5	2.5	any	none	1.25	1.75	2.5	any
Hole		dia	none	none	1.5	2.5	any	1/16	1.25	1.75	2.5	any
Wane	on face	thx	none	none	none	0.5	any	none	0.5	0.5	<1	any
		wid	none	none	none	1	any	none	1	4/3	2	any
		length	none	none	none	24	any	none	24	any	any	any
	on back	thx	0.5	0.5	2/3	any	any	any	any	any	any	any
		wid	1	4/3	2	any	any	any	any	any	any	any
	length	36	48	72	any	any	any	any	any	any	any	
Split		number	1	1	any	any	any	any	any	any	any	any
		length	4	8	24	48	any	8	8	24	36	any
Check	surface	number	4	2	any	any	any	any	any	any	any	any
		wid	1/32	1/32	1/32	any	any	1/32	any	any	any	any
		length	4	10	10	any	any	10	any	any	any	any
Pith	firm-heart	wid	0.25	0.5	any	any	any	?	?	?	?	?
		length	24	72	any	any	any	?	?	?	?	?
Pitch	light	wid	area	any	any	any	any	any	streak	streak	streak	any
		length	144	any	any	any	any	any	streak	streak	streak	any
	medium	wid	none	area	any	any	any	any	streak	streak	streak	any
		length	none	576	any	any	any	any	streak	streak	streak	any
	heavy	wid	none	none	area	area	any	none	streak	streak	streak	any
		length	none	none	576	576	any	none	streak	streak	streak	any
Pocket		number	2	3	any	any	any	any	any	any	any	any
		width	1/16	1/16	1/16	area	any	1/16	1/16	any	any	any
		length	3	6	12	4	any	12	12	any	any	any
Stain		type	light	medium	medium	heavy	any	medium	any	any	any	any
		wid	area	any	any	any	any	any	any	any	any	any
		length	384	any	any	any	any	any	any	any	any	any
Decay		wid	none	none	area	any	any	none	none	none	small	any
		length	none	none	144	any	any	none	none	none	small	any
Sub-category	Grade											
Commons	1 COMMON		X									
	2 COMMON			X								
	3 COMMON				X							
	4 COMMON					X						
	5 COMMON						X					
Alternate	SEL-MERCH							X				
	CONST								X			
	STAND									X		
	UTIL										X	
	ECONOMY											X

- X - Data in the column is the specification for the grade in the row.
- none - Not allowed
- any - No limitations
- area - Width x length

Table B6. Test results of grading the lumber in the Boards category

Defect	Type/Location	Measure	Facts (<= inches)										
			2.25	3	3.5	16/3	6	2.5	2.5	3.5	5		
knot	firm-red	dia	2.25	3	3.5	16/3	6	2.5	2.5	3.5	5		
	firm-black	dia	0.75	1.25	7/3	16/3		1.25	2.5	3.5	5		
	loose	dia			1.5	2.5			1.25	1.75	3		
Hole		dia			1.5	2.5		1/16	1.25	1.75	3		
Wane	on face	thx				0.5			0.5	0.5	***	1	
		wid				1			1	1.3	2		
		length				24			24	20	25		
	on back	thx	0.5	0.5	2/3								
		wid	1	4/3	2								
		length	36	48	72								
Split		number	1	1	2	1		1	1	1	1		
		length	4	8	24	48		8	8	24	36		
Check	surface	number	4	2	1			1					
		wid	1/32	1/32	1/32			1/32					
		length	4	10	10			10					
Pith	firm-heart	wid	0.25	0.5									
		length	24	72									
Pitch	light	wid	10										
		length	12										
	medium	wid		10									
		length		57.6									
	heavy	wid			10	10							
		length			57.6	57.6							
Pocket		number	2	3	2	1		2	1				
		width	1/16	1/16	1/16	1		1/16	1/16				
		length	3	6	12	4		12	12				
Stain		type	light	medium	medium			medium					
		wid	10	10	10			8					
		length	12	10	12			10					
Decay		wid											
		length											
Sub-category	Grade												
Commons	1 COMMON		X										
	2 COMMON			X									
	3 COMMON				X			X					
	4 COMMON					X			X				
	5 COMMON						X			X	X	X	X
Alternate	SEL-MERCH		X					X					
	CONST								X				
	STAND			X	X					X			
	UTIL											X	
	ECONOMY						X	X					X

- 1) X indicates that input data in the column satisfies the grade in the row.
- 2) Blank cell means that no such characteristic exists.
- 3) For the same set of facts, grade overlaps are allowed. For example, facts in the first column match grades 1 COMMON and SEL-MERCH.
- 4) The results satisfy the specification, indicating the system works well.