AN ABSTRACT OF THE THESIS OF

Yimin Zeng for the degree of Doctor of Philosophy in Forest Products presented on August 17, 1995. Title: Integration of an Expert System and Dynamic Programming Approach to Optimize Log Breakdown Using 3-Dimensional Log and Internal Defect Shape Information.

Abstract approved: _ Signature redacted for privacy. _____

James W. Funck

A computer program, SAW3DG, was developed to optimize log breakdown using 3-dimensional log and internal defect shapes. The program was based on SAW3D, a log breakdown optimization program considering only 3-dimensional log shape. SAW3D was modified to include internal log defects in 3-dimensional representation and integrated with SLGRADER, an expert system for softwood lumber grading, resulting in a system that is able to optimize log breakdown based on lumber grade.

SAW3DG allows live, cant, around, and multi-thickness sawing methods. The system uses a polygonal cross-section model to represent the log and its internal defects. It consists of four basic components: headrig optimization, edging optimization, trimming optimization, and lumber grading. The headrig optimization component begins the log breakdown process by mathematically rotating and skewing the log into a position and then uses a programming (DP) algorithm to find the optimum sawing pattern. The profile of each piece cut from the log along with its defect information is then passed to the edging optimization component where the piece is optimally positioned and then edged using another DP algorithm. Information about the untrimmed lumber and its defects is sent to the trimming optimization component where the sizes of each finished piece of lumber and defects exposed on

its four faces are finally determined. This information is then sent to the lumber grading component to determine the lumber grade. Lumber value is determined by its grade and size and used by a third DP algorithm to decide the optimum trimming pattern. Solutions are provided both in text and graphic formats. Twelve computer generated logs of various sizes and in ellipsoid, horn-down, and S-twisted shapes with a number of internal defect types and distributions were used to evaluate SAW3DG. Results indicated that SAW3DG provided better solutions than those models that consider only the true external log shape or that treat the log as a defect free truncated cone. In addition, effects of log rotation and skewing operations, flitch/cant pitch, and different sawing methods on SAW3DG solutions were also studied.

Integration of an Expert System and Dynamic Programming Approach to
Optimize Log Breakdown Using 3-Dimensional Log and
Internal Defect Shape Information

by

Yimin Zeng

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Completed August 17, 1995
Commencement June 1996

ACKNOWLEDGEMENTS

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

INTEGRATION OF AN EXPERT SYSTEM AND DYNAMIC PROGRAMMING APPROACH TO OPTIMIZE LOG BREAKDOWN USING 3-DIMENSIONAL LOG AND INTERNAL DEFECT SHAPE INFORMATION

CHAPTER 1 INTRODUCTION

Lumber production at sawmills involves a complex sequence of interrelated decisions at several manufacturing stations. These stations include log bucking where long logs are cut into short segments, primary log breakdown at the headrig where the bucked logs are cut into flitches and/or cants, edging and resawing centers where flitches and cants are further edged into specified widths, and finally the trimming station where rough lumber from the previous steps is trimmed into final product lengths. One of the most important factors that affects a sawmill's profit is the efficiency of raw material usage. Value yield or volume recovery could be decreased at any of those machine centers if the cutting decisions made there are not optimum. The process of converting logs into lumber is complicated by log geometry, log quality, saw kerf, sawing variation, sawing method, edging and trimming methods, and the mix of final product dimensions desired. Given so many factors, it is very difficult if not impossible for a human operator to make an optimum sawing decision. Use of computer technology to make optimum decisions at those stations has been an ongoing effort for three decades. 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 to determine optimum breakdown patterns. In modern sawmills, computer-aided decision tools have been extensively used at the various machine centers to help mills increase lumber recovery and thus, net revenue.

In recent years, the wood processing industry has experienced major changes in timber supply. Logs are smaller and thus the timber supply contains a higher proportion of juvenile wood; there will be more species in the log mix, including

currently under-utilized hardwoods and conifers; large clear logs will be scarce and therefore, more expensive. All these suggest that the industry requires even better optimization techniques than the current level to further increase the efficiency of raw materials usage.

The early optimization systems at the log breakdown stations used only one geometric dimension of a simplified log model. The cutting patterns were also limited by the number of pre-determined patterns in a look-up table. Later, systems using log models closer to real log shapes were developed. Systems using real-time algorithms instead of look-up tables were also developed. Recently, commercial scanning systems that are able to obtain 3-dimensional log shape have been installed in some mills. Optimization software systems that use such real log shape to make log breakdown decisions are also being developed.

One of the most recent software systems is SAW3D (Zeng, 1991), a log breakdown optimization program using 3-dimensional log shape and dynamic programming. SAW3D has such advanced features as log modeling in real shape; real-time computation instead of a look-up table; log, flitch and cant positioning operations; multi-thickness sawing patterns; and various sawing methods.

However, currently available models, including SAW3D, make their "optimum" decisions based only on the log's or board's geometric dimensions. An important aspect ignored by optimization models is the effect of log internal defect type, size, and location on final lumber grade and thus lumber recovery and quality. As such, the resulting solutions are not usually optimum. 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 to predict lumber grade when making cutting decisions.

This study improves SAW3D by considering log internal defects in addition to other characteristics so that it can provide better sawing solutions than other systems that are based only on log external geometric information.

CHAPTER 2 OBJECTIVES AND SCOPE

This research was an extension of the SAW3D research project. It improves SAW3D by considering log internal defects in addition to all other features included in SAW3D. In addition, the around-sawing method was also considered by this research. Therefore, the system retains all the features found in SAW3D while adding improved features.

The objectives of this research were to:

1) develop a log breakdown optimization system considering 3-dimensional log shape and internal defects by modifying SAW3D and embedding SLGRADER, which is an expert system for softwood lumber grading;

2) develop a version of such a system for the around-sawing method; and

3) analyze the effect of using real log shape, log internal defects, log positioning operations, and various sawing methods on value and volume yields.

The system needed to be capable of:

1) accepting and using log geometric and internal defect scanning data to construct wire-frame models for the log and internal defects in 3-dimensional space;

2) positioning a log, flitch and cant along with all defects in 3-dimensional space;

3) live sawing, cant sawing and around-sawing breakdown of logs;

4) providing optimum sawing, edging and trimming patterns and lumber grades; and

5) providing optimum log, flitch and cant positions.

The system was to be developed by modifying SAW3D and integrating an expert system for softwood lumber grading, SLGRADER, with the modified version of SAW3D. Therefore, only softwood lumber grades were considered. Many grade

categories are included in SLGRADER, so the embedded expert system is able to work on any of those grade categories. Because this thesis focused on the system development methodology, however, the current versions of the integrated system considered only three grade categories, including Commons, Light Framing, and Joists and Planks. Similarly, only selected defect types were considered, including knots, wane, checks, splits, pockets, decay, stain, pitch, and pith.

The primary objective of this research was to develop the system as a tool for further research. The analysis plans just described were only for preliminary evaluation purposes and not as thorough investigations on log breakdown approaches. Therefore, the testing samples were also limited to selected log sizes and defect types.

CHAPTER 3 LITERATURE REVIEW

Efforts to optimize lumber recovery have been done since the late 1960's. Many of those efforts focused on systems considering only log exterior geometry while a few studied the effects of internal defects.

The development of log breakdown optimization systems was based on the advancement of scanning technologies. Over the past three decades, log shape scanning techniques have evolved from one-axis curtain-type scanning, to two-axis scanning, and to the most recent ones that provide scanning data of real log shape profiles. The one-axis curtain-type scanning system can obtain only two scanning points on a cross section, thus only a circular cross-section can be constructed. The two-axis scanning systems can obtain four scanning points per cross-section. Such a system can be used to construct an elliptical cross-section. The most recent scanning systems, such as The PerfectShape Scanner from Applied Scanning Technology and Opcon 400 DynaVision Scanners from Cutler-Hammer Products, are able to obtain hundreds of scanning points around the perimeter of a cross-section, providing the capability of constructing a real log shape profile. As log scanning systems have been evolving along that path, optimization systems have been improved from using simplified log exterior shapes to more realistic "real" exterior shapes.

In addition to log exterior shape, log internal defects such as knots, pockets, pitch streak, stain, decay, etc. and other characteristics such as growth ring patterns also have great impacts on lumber recovery. For instance, one research project showed that log breakdown considering internal defects resulted in a 13.56 percent higher recovery than that without log internal defect information (Wagner et al., 1990). Obviously, if a log breakdown optimization system is capable of making decisions based on both true log shape and internal defects, it will provide even better solutions. Log internal defect scanning systems are very difficult and costly to develop, though much fundamental research in the area has been done and progress has been made. Given the situation that scanning systems capable of obtaining log

internal defect information in an industrial environment have not been developed yet, most research considered only the effect of internal defects rather than the actual development of log breakdown optimization systems using internal defect information.

This chapter first briefly reviews the research considering only log exterior shape and then discusses the research considering log internal defects. Some updated information is added while others were discussed in Zeng (1991) in greater detail.

3.1 Systems Considering Only Log Shape

The effectiveness of a log breakdown system is based on how the system models a log and how it arrives at the solution. Based on log modeling techniques, previous log breakdown research can be classified into categories of systems using a simplified log model and systems using true-log shape. Based on solution-searching approaches, they can be classified into categories of systems using computer simulation techniques and systems using mathematical programming methods.

3.1.1 Log Modeling Techniques

When developing a log breakdown optimization system, one of the critical aspects is how the log shape is described to obtain an optimum breakdown solution. No matter what kind of optimum searching algorithm is engaged, the final optimum solution is obtained by operating on that log model. Obviously, the closer to the real log shape the log model is, the better the solution will be.

There are several log modeling methods and each has its own advantages and disadvantages. One can define log models as whole log, cross section, and computer array models (Alleckson, 1980). Whole log models use a mathematical equation to describe a log. Examples of this approach are cylinder and truncated-cone models. The early systems widely used such a simplified log model instead of real log shape

(Geerts, 1984; Hallock and Lewis, 1971, 1973, 1978; Lewis, 1985a, 1985b; Tejavibulya, 1981). The cylinder model is the simplest log shape model. Given a diameter and length, the log is represented by the equation for a cylinder. Because taper is not considered, no log is really correctly described by this model. The truncated-cone model is better than the cylinder model since taper is included. Given the small-end diameter, taper, and length, the log can be represented by a truncated cone. These two models are easily formulated by mathematical equations and are very easy to solve with any optimum searching algorithm. As a result, most existing programs use some form of either cylinder or truncated-cone log models.

Cross-section models use a series of cross sections at intervals along the length of the log to represent the log. The log surface is represented by straight lines connecting each pair of intervals and each pair of cross sections. Since data at each cross section can be different, some irregularities such as crook and sweep can be included in the model. Cross-section models include circular, elliptical, and polygonal cross-section models according to the shape of the cross sections. In a circular cross-section model all the cross sections are circular. By giving different diameters to each cross section, the irregularity along the length of the log can be represented to some extent. In an elliptical cross-section model all the cross sections are elliptical. The long axes and the short axes of all the cross sections are not necessarily parallel. Since this model contains more information on the cross sections, it is closer to the real shape of logs than the circular cross-section model. In a polygonal cross-section model, the log is represented by a series of polygonal cross sections at selected intervals along the length of the log. Scanning points representing vertices of polygonal cross sections can be provided by some of the latest log scanning systems, such as the systems supplied by Applied Scanning Technology and Cutler-Hammer Products. The surface of the log between two adjacent cross sections is described by straight lines connecting corresponding vertices on the two cross sections. Figure 3.1 shows one cross section of the log model, while Figure 3.2 shows the whole log modeled by this method. This model is much closer to a real log shape since the irregularities on the cross sections are also included in the model. Using an existing
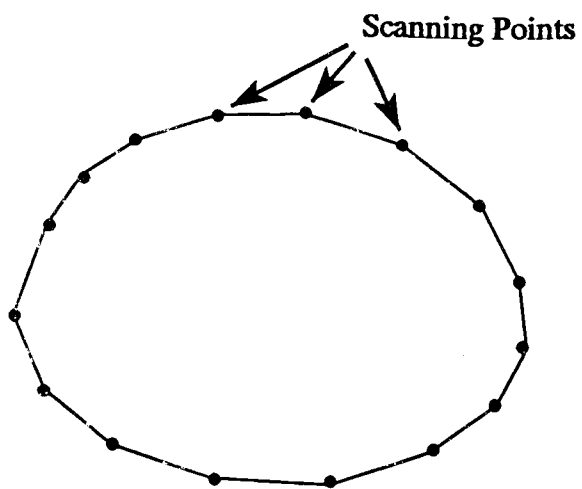
Figure 3.1. A cross section of the polygonal cross-section model.
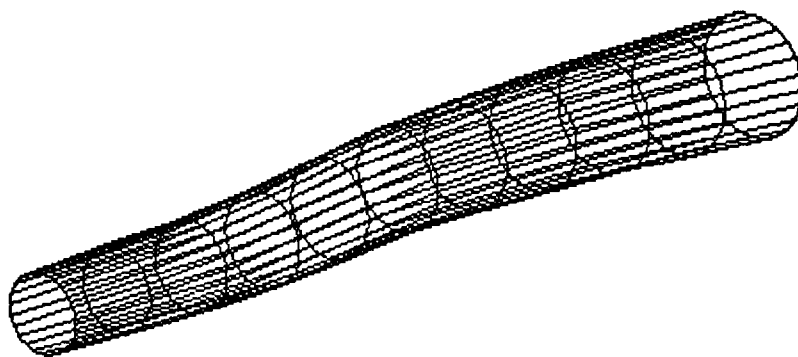
Figure 3.2. A log represented by the polygonal cross-section model.

curtain-type scanning system, a circular, elliptical, or an interpolated polygonal cross-section model can be constructed. Accuracy of a cross-section model is largely determined by the shape of the cross section. Mongeau et al. (1993) compared different cross-section models and studied their accuracy. They found that circular and elliptical models provided the poorest accuracy. Two interpolated polygonal cross-section models using Dyadic and Chaikin methods provided the best overall performance. The polygonal cross sections studied in that research were constructed using only traditional curtain-type scanning systems. Better accuracy is expected if the latest real shape laser scanning systems are used instead of the shadow scanners.

When choosing a log modeling method, two aspects deserve very careful attention, reality and complexity. If a model more closely describes the real shape of the log than other models, it will contain more information on the log. Therefore, the final solution could be better when given the same optimum searching algorithm, but the computing time is also increased. Among all the modeling techniques, the cylinder and truncated cone are very easy to handle, since the log can be described by simple mathematical equations, resulting in shorter computing time. The cross-section models are better models of a real log but need longer computing times.

3.1.2 Solution-searching Approaches

The solution-searching approach used is another aspect that can be used to distinguish between previous research. One can classify the optimization systems into simulation models and mathematical programming models. When the major concern of developing such a system was to analyze alternate sawing decisions, computer simulation techniques were used (Airth and Calvert, 1973a, 1973b; Allekson et al., 1980; Anderson and Reynolds, 1981; Hallock and Lewis, 1971, 1973, 1978; Leach et al., 1990; Lewis 1985a, 1985b; McAdoo, 1969; Occena et al., 1988; Pnevmaticos et al., 1974, 1976; Pnematicos and Mouland, 1978; Priasulmana, 1983; Richards, 1973,1977; Todoroki, 1990; Wagner and Taylor, 1975). Computer simulation models

formulate log models using either mathematical descriptions or from empirical data and simulate the sawing process given sawing parameters and sawing methods. When used to aid decision making, simulation models used the selected sawing procedures and/or obtained the best sawing solution by simply comparing all sawing solutions. Compared to other types of models, simulation models had two major advantages: 1) the capabilities of a real sawing process were considered so that the solutions were somewhat attainable, and 2) usually they were easier to implement. The disadvantages included: 1) the sawing procedures were limited, and 2) in many situations, the simulation methods do not necessarily locate the optimum. Simulation models use heuristic methods to determine optimum solutions. Usually the operator has to inspect the results from simulation runs to obtain a result that approaches optimum. If the program seeks the optimum solution itself, it has to try all possibilities in a selected solution space and compare all results one by one, which can be very time consuming. If the program wants to arrive at an optimum solution in a reasonable period of time, it has to skip some possibilities, thus risking missing a real optimum point. This led some researchers to explore the application of mathematical programming to the subject. Although many operations research techniques such as linear programming, integer programming, dynamic programming and network techniques have been used for such related areas as log bucking and board cutting, only dynamic programming has been applied successfully to log sawing.

Dynamic programming (Bellman, 1957; Denardo, 1982), is an optimum procedure that is particularly useful for problems requiring a sequence of interrelated decisions. A sequence of decisions, which in turn results in a sequence of situations, is performed by recursive calculations to maximize overall effectiveness. By proper formulation, the log breakdown optimization problem can be defined as a dynamic programming problem, and a recursive equation can be established to find the optimum value or volume recovery of the log. When considering multi-dimensions and allowing gaps between two boards, dynamic programming (DP) can be computationaly more efficient and provide better solutions than the heuristic methods used in simulation models. It has been successfully used for the optimization of log

bucking since the early 1970's (Briggs, 1980; Glück and Koch, 1973; Pnevmaticos and Mann, 1972; Rogler and Canham, 1986). However, in the area of log sawing optimization, only a few models applying the DP algorithm have been developed, and most of them have used simplified log models (Faaland and Briggs, 1984; Geerts, 1984; Tejavibulya, 1981; Zeng, 1991).

Today, various types of optimizing systems have been used extensively in the lumber industry, and those systems have already proven themselves to be great contributors to improved lumber recovery. Most manufacturers of various log breakdown optimizers claim that their equipment is "real time" and "real shape". However, because of processing time and scanning system limitations, most current log breakdown optimization models only work on greatly simplified log shape representations, and they typically use heuristic methods rather than mathematical optimization techniques to increase recovery. The "real shape" claimed by those companies actually means that such longitudinal information as sweep and crook is measured, but cross sections are still represented by a round or elliptical shape. To squeeze the highest value from every log of any shape, optimization models capable of considering log shapes in 3-dimensional (3-D) space and log scanning systems capable of providing those true shapes have been developed in the past few years (Dasher, 1993; Green, 1993; Zeng, 1991).

3.2 Studies on Log breakdown Considering Log Internal Defects

Research considering log internal defects was conducted as early as that considering log exterior geometry. Tsolakides (1969) developed a simulation program to study the effects of three alternate sawing methods on grade and volume yields. One of the sawing methods allowed for turning of the log, while the other methods were live and cant sawing. Logs with external and internal defects were simulated using empirical data from six red oak logs. Exterior geometry and internal defects of each log were obtained by slicing the log into disks and then manually recording

information on the circumference, size and location of the internal defects. The circumference of each log was then rounded into the shape of a cylinder. When a sawing line cut through the log, internal defects were mapped on the board surfaces by calculating defects recorded in each disk. This research was methodological and did not provide a software tool.

Reynolds et al. (1969, 1970) presented a computer program (DEFECT) to simulate the log sawing process. Coordinates of the log surface and internal defects including class, size and location were collected from boards cut from a sample of real logs. Each log was reconstructed as a cylinder enclosing the real log and the internal defects were represented by collections of small three-dimensional elements. When sawing the reconstructed log, the defects were mapped on the faces of boards and represented by rectangles. The resulting boards, which included defect information on each board for each sawing pattern, were ripped and trimmed by another simulation program called YIELD to produce cut-stock parts (Wodzinski et al., 1966).

Pnevmaticos et al. (1974, 1976) used computer graphics techniques to simulate logs and the process of sawing them into lumber. The log was simulated and displayed as a cylinder or truncated cone by entering the two end diameters and the length of the log. Log positioning operations were not considered since the log representation was simple. Defect locations and dimensions were generated through a random process and displayed as rectangular solids. The process of live sawing the log was also simulated and the resulting boards and slabs were displayed.

Wagner and Taylor (1975) simulated sawing with a chipping headrig to study the effects of two alternate sawing patterns and log rotation with external and internal defects on lumber value. Data collected from ten southern pine logs were processed by the program, and the logs were theoretically sawn using predetermined sawing patterns simulating actual sawing planes produced by the saws of a chipping headrig. They concluded that all logs were found to have an optimum log rotation angle that produced the highest value lumber, and knotty logs showed value variations of as much as 43 percent due to log position.

Richards (1977) presented a simulation program in which hardwood logs were simulated as truncated cones with a taper of 0.3 inch and containing core defects and randomly located knots. The central core defect was represented by a cant 5-7/8 inches square in cross section. Each knot was simulated as a cone with its apex of 24 degree at the central axis of the log and tapering outward. The resulting boards were graded and priced by the computer. Both four-sided sawing and live sawing methods with reripping of wide boards for grade and with repeated sawing at 15 degree incremental log rotations were compared in order to study the importance of sawing methods and initial position on value yield. The results indicate that the initial rotational position on the carriage is very important in all sawing methods and that live sawing generally equals or exceeds four-sided sawing in value yield. Later, Richards et al. (1979, 1980) extended the model by including quadrant sawing, cant sawing, decision sawing, live sawing, and live sawing plus reripping for grade, as well as truncated cone shaped logs with a selection of tapers, diameters, core defect diameters, and knot patterns. The results indicated that quadrant, decision, and cant sawing gave similar lumber values. Live sawing was generally better than the other three sawing methods. Live sawing with reripping produced the highest lumber values.

A computer simulation model for sawing hardwood logs was presented by Pnevmaticos and Mouland (1978). The log was simulated as a truncated cone with solid rectangular defects either generated randomly or entered from real data. The program was used to compare live, around, and cant sawing employed when sawing sugar maple. Preliminary system evaluation showed matches between the simulated results and actually sawing results, but details of the results were not presented.

Geerts (1984) presented a mathematical solution for optimizing the sawing pattern of a log with a defect core. Both the log and defect core are assumed to be cylinders. He used a two-level approach which used similar dynamic programming algorithms for each level. The first level maximizes the breakdown of the log into flitches and the second level maximizes the breakdown of each flitch into lumber. The maximum value of each flitch, which is the output of the second level, was

returned to the first level. If the profile of log cross sections was available, the algorithm would allow for dealing with an irregular shape defined by polygons, but no taper was taken into account. He developed a computer program in which both logs and defect core were modeled as cylinders to implement his algorithm. Excluding the 3-D cutting problem and over-simplifying the log and defect models are limitations to the program.

Faaland and Briggs (1984) presented a DP formulation that simultaneously optimizes bucking, live saws logs into lumber, and edges lumber into finished dimensions. The model integrates the bucking study of Briggs (1980), the DP sawing study of Tejavibulya (1981), and a knapsack formulation for edging. Logs with taper, sweep, crook and defects are allowed by the model. Limitations to the log model are that any cross section of the log must be circular and variation along the length of the log should be in a two-dimensional plane. The model bucks a log into segments by using the DP bucking program developed by Briggs (1980), then seeks the largest cylinder inscribed inside each segment. Breakdown of the cylinder found from each segment is performed by the DP sawing model and the optimum value of each cylinder is returned to the bucking model. The edging model edges each flitch cut out of the cylinder and returns the optimum value of the flitch to the sawing model. Transforming log irregularity into a cylinder and then performing all operations on the cylinder greatly simplified calculations and significantly reduced computing time. However, these restrictions greatly increase the chance of losing value or volume.

Occena and Tanchoco (1988) developed a graphic simulator for hardwood log breakdown. The simulator uses solid-modeling principles and represents all solid objects such as the log, flitches, labs, defects, and saw blades using closed polyhedra made up of polygon paths. The sawing process was simulated by interaction operations between those solid representations via Boolean operations. Lumber grading was performed by a hardwood lumber grading program developed by Huang et al. (1987). The system was intended to be used as an analytical tool to simulate the sawing process of hardwood logs with internal defects.

Researchers at the Forest Research Institute of New Zealand developed a graphic computer simulation system for simulating breakdown of pruned logs (Garcia, 1987; Park, 1987; Todoroki, 1988, 1990). The early version was called SEESAW and only supported interactive simulation. In 1990, SEESAW became a part of a more powerful system called AUTOSAW. In AUTOSAW, logs are represented using elliptical cross sections and low quality wood around the pith is represented using polygon cross sections. It may be used either interactively or in automatic mode. In the interactive mode the user interactively controls all operation parameters such as log position and the position of every single cut from the headrig, resaw, and edger. The simulator displays results of each action on the screen. In the automatic mode, log positioning, mill parameters, sawing strategies, sawing patterns, edging and trimming options, and rough green lumber sizes are all defined in a command file. In this mode, the simulator will select a proper sawing pattern from a predefined look-up table containing 25 sawing patterns according to the instruction in the command file. This simulator has been successfully used to assist in pruned log resource evaluations and assessments of sawmill recoveries in New Zealand (Park, 1994).

Harless et al. (1991), Wagner et al. (1990) and Steele (1994) examined the effects of defect locations within logs on lumber value recovery via computer simulation. Twenty-four 12-foot long red oak logs with small-end diameters of approximately 16 inches were used as samples. Data about a log profile and its internal defects were obtained by slicing the log into 1/4-inch thick disks and then manually recording the coordinates of points on the perimeter of the disks and defects. Logs were reconstructed using three-dimensional element arrays and sawn using three sawing methods. A hardwood lumber grading system developed by Klinkhachorn, et al. (1988,1989) was used to perform lumber grading tasks. A split-plot experimental design method was used to analyze the results of simulated sawing those sample logs using live and grade sawing methods. The results showed log rotation had a significant impact on lumber value yield. The average value increase because of log rotation for each sawing method was higher than 10 percent. The results also indicated that the

live sawing method yielded the better value return compared with grade sawing methods.

Samson (1993a,1993b) developed a theoretical model to study the effect of knots in the conversion of logs into structural lumber. The model assumed logs are perfect cylinders and used circular cones emanating from the pith to represent knots. The model presented an algorithm to predict lumber grade in accordance with the ASTM D 245 standard. The model was used to study the effect of log rotation on structural grade yield considering various knot distributions and cant and live sawing methods. The results showed that for both sawing methods, log rotation had significant impacts on lumber grade yield.

Much research has been done to consider the effects of log internal defects on lumber recovery both in value and volume as discussed above. As indicated previously, many of them just intended to study the effects of log internal defects rather than provide a practical tool. Though some simulation packages have been developed at a fairly advanced level, they are either used only for hardwood log breakdown (Oceea, 1988), or have some limitations and are not suitable for the U.S. domestic market (Todoroki, 1990). A review of research over the past three decades revealed some common weaknesses of past research in development of a practical tool for log breakdown optimization considering internal defects, including (1) simplified log and defect models, (2) lack of optimization capabilities, and (3) lack of practical lumber grading capabilities in accordance with U.S. softwood lumber grading rules. This research was initiated to address those problems.

CHAPTER 4 REVIEW OF SAW3D

This research expands SAW3D's features by adding the capability of considering log internal defects when making optimum breakdown decisions. This section only gives a very brief description about SAW3D's features. Details about SAW3D are described in Zeng (1991).

SAW3D is a computer program using dynamic programming algorithms to optimize log breakdown (Zeng, 1991). It uses a polygonal cross-section representation to describe logs in a realistic 3-dimensional fashion. SAW3D uses heuristic search methods to determine the optimum positions for the log and each piece cut from the log, and uses three nested dynamic programming algorithms to decide the optimum sawing, edging, and trimming patterns. Figure 4.1 shows the three components of SAW3D.
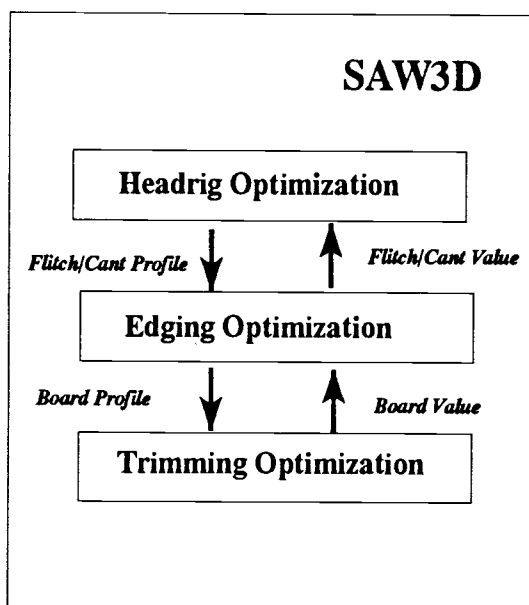
Figure 4.1. Components of SAW3D.

After receiving the log shape data, SAW3D mathematically rotates the log and skews it horizontally. Then the log is sawn into flitches or cants using a DP algorithm in an attempt to obtain the optimum value return of the log. To arrive at the optimum solution, it needs the value return from each piece cut off the log. That is done in the edging optimization component. The edging component orients the piece and then edges it into untrimmed lumber applying another DP algorithm in attempt to obtain the optimum value return of the piece. To do so, the edging component needs to know value return of each edged but untrimmed piece of lumber. That is done by the trimming optimization component. The trimming component trims all untrimmed lumber into finished dimensions and uses a third DP algorithm to look for the optimum value return of the untrimmed lumber. Information obtained at each lower level component is passed back to a higher level component so that the optimum decisions can be made at each level. Upon completion, the program outputs the optimum value of the log and optimum value of each flitch and cant along with corresponding sawing, edging and trimming patterns as well as the optimum position of the log and optimum orientation of each flitch/cant.

Features of SAW3D include:

1) handling logs of any shape in 3-dimensional space;

2) performing positioning operations on the log as well as each slab, cant and flitch cut from the log;

3) allowing multi-thickness sawing patterns;

4) allowing any mix of lumber dimensions (ALS or other, dry finished or green target, etc.);

5) allowing live and cant sawing as well as full taper and split taper sawing;

6) considering parameters such as saw kerfs, sawing variations, and wane allowances;

7) running on a PC based computer; and

8) providing both text and graphic outputs.

One unique feature that sets SAW3D apart from other similar models is the ability to represent logs in real shape using scanning data from the most recent real shape log scanning systems. Computational results using mathematically generated logs in various shapes indicated that SAW3D provided better solutions than other systems using simplified log models. However, the current version of SAW3D considers only the exterior shape of the log. Wood quality of logs is ignored.

CHAPTER 5 REVIEW OF SLGRADER

SLGRADER is an expert system for softwood lumber grading. This chapter describes the system briefly. Details about SLGRADER can be found in Zeng (1993).

The system consists of four basic components as show in Figure 5.1. The knowledge base contains grading rules based on Western Lumber Grading Rules 88 published by the Western Wood Products Association (Western Wood Products Association, 1988). The current version considers 27 grades in the Dimension, Select and Finish, and Boards categories. Defects and wood characteristics include knots, holes, checks, splits, pockets, pitch, decay, wane, etc. Since the system was planned to be linked to SAW3D, manufacturing imperfections and any defects related to lumber drying were not considered. However, the system was designed in such a way that the knowledge base can be easily refined and extended when the need arises.
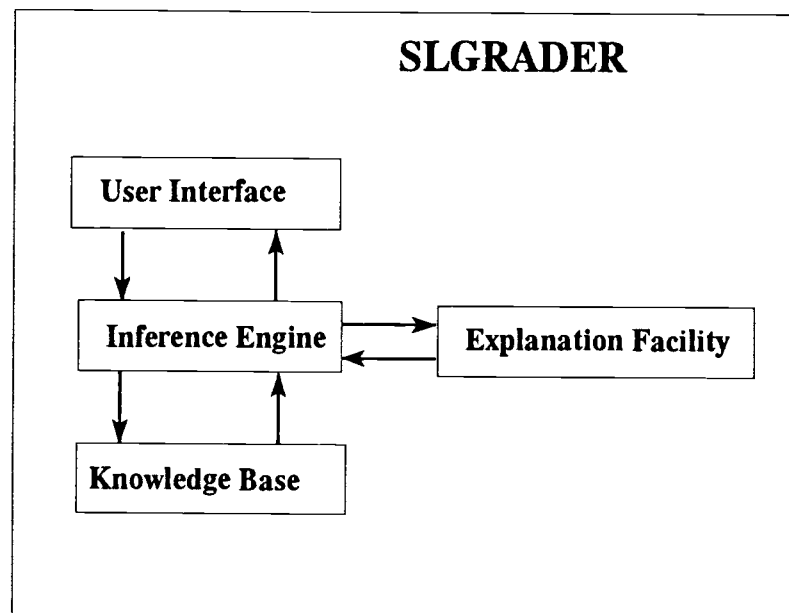
Figure 5.1. Basic Components of SLGRADER.

The user interface invites the user to enter data such as lumber size, grade category, and type, location and size of defects for each face and edge of the lumber. It also displays the resulting lumber grade along with explanations. The inference engine provides the reasoning capabilities of the system using a built-in inference mechanism. It performs deductions and inferences to arrive at conclusions. It uses the facts entered by the user to deduce new facts and then infers the lumber grade by matching the facts with the grading rules in the knowledge base. The explanation facility explains why the lumber grade was selected.

The system is menu-driven and interacts with the user through menus and a series of dialogue boxes. The input process begins with loading the knowledge base and selecting the input methods (interactively or loading a file containing all facts). If the interactive input method is selected, a series of dialogue 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 an 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 process can be repeated for any other face or edge, so if the overall grade of the lumber is desired instead of the individual face, all four sides of the lumber must to be graded. Then the system will provide the final grade for the lumber based on the grades of all four sides.

The system can be used as a stand-along instruction tool, or a component embedded into an automatic lumber grading system or a log breakdown optimization system. When used as an embedded sub-system, the knowledge base and inference engine will be the two components that provide grading functions. The user interface and explanation facility, which provide the communication channel, should be replaced by other functions linked to the calling environment.

The system was implemented using Eclipse, a knowledge-based programming toolkit (Haley Enterprise, 1992). Eclipse includes such features as rule-based

knowledge representation and a forward chaining inference engine. It has versions for MS-DOS, MS-Windows and Windows-NT. The system developed using Eclipse can be easily embedded into other systems using high level languages such as C and C++.

CHAPTER 6 SYSTEM DEVELOPMENT

SAW3DG, a computer software package designed to optimize log breakdown while considering both log exterior geometry and internal defects, were developed by modifying SAW3D and embedding SLGRADER into the modified version of SAW3D. While retaining all the features found in SAW3D, the integrated system must be able to describe defects using realistic shape representations, map those characteristics on to faces of each piece of lumber cut from the log, and determine the grade of each piece of lumber so that it can find the optimum solutions based on lumber grades.

As mentioned before, SAW3D makes the optimum sawing, edging, and trimming decisions based only on the log exterior shape. Defects are not considered. It saws a log into flitches and cants. Those flitches and cants are then edged and trimmed into final product dimensions. At this stage, it looks through a price table to find the value of a piece of lumber based only on the lumber dimensions.

To consider wood quality in addition to the log exterior geometry, the integrated system must first be able to represent those wood quality characteristics, carry those characteristics through all processing stages, and finally map those characteristics on to the face of the lumber. At this point, the system needs to know the grade of the lumber and then determines the value based on the dimension and the grade. To do so, the following features needed to be added:

1)   representing defects in realistic 3-dimensional shapes;

2)   carrying wood quality characteristics through all processing stages;

3)   mapping defects exposed on lumber surfaces; and

4)   grading rough, green lumber.

SAW3D was modified to accomplish the first three tasks listed above. Then SLGRADER was integrated with the modified version of SAW3D to do the lumber grading. The relationships between SAW3D and SLGRADER within the integrated system is shown in Figure 6.1. SAW3D processes the log through all stages to a point where the grade of a piece of lumber needs to be determined.

```
┌─────────────────────────────────────────┐
│  SAW3DG                                  │
│          ┌──────────────────┐            │
│          │ Modified SAW3D    │           │
│          └──────────────────┘            │
│                  │      ▲ Lumber grade    │
│  1. Lumber dimensions │      │             │
│  2. Sizes, types, and │      │             │
│     locations of defects ▼   │             │
│          ┌──────────────────┐            │
│          │  SLGRADER         │           │
│          └──────────────────┘            │
└─────────────────────────────────────────┘
```
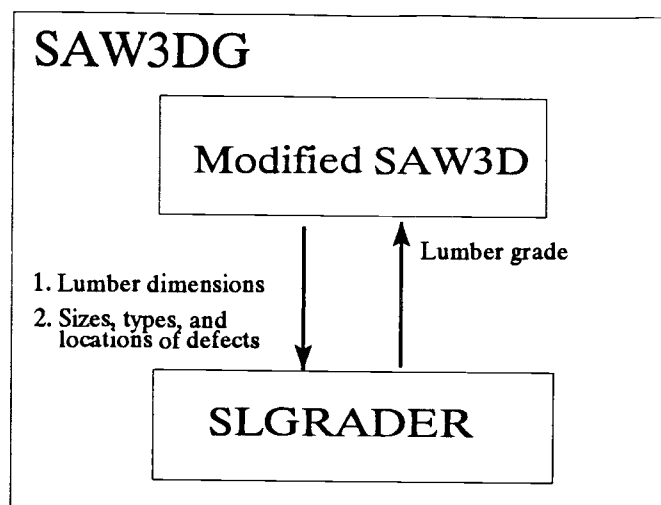
Figure 6.1. Relationship between SAW3D and SLGRADER.

It then provides SLGRADER with such information as lumber thickness, width, and length, as well as types, sizes, and locations of all defects on all faces of the lumber. SLGRADER then infers the lumber grade and returns the conclusion to SAW3D. After that, SAW3D looks up the price table associated with the lumber grade to determine value of the lumber.

6.1 Sawing Methods

Two sawing methods, live sawing and cant sawing, are allowed in SAW3D. In the live sawing method, the log is sawn from one side to the opposite side without log rotation, resulting in parallel sawing lines as shown in Figure 6.2. In cant sawing, a cant is obtained from the center portion of the log while some flitches are produced. The cant is then sawn into boards of desired finished thicknesses at a cant edger or a secondary breakdown machine. The cant sawing pattern is shown in Figure 6.3, where sawing lines on the cant are perpendicular to those produced by the headrig.
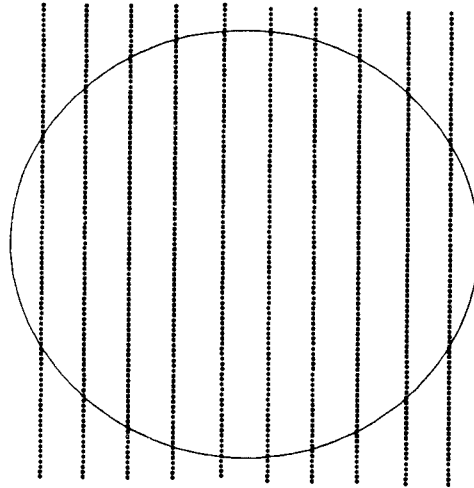
Figure 6.2. Live sawing method.
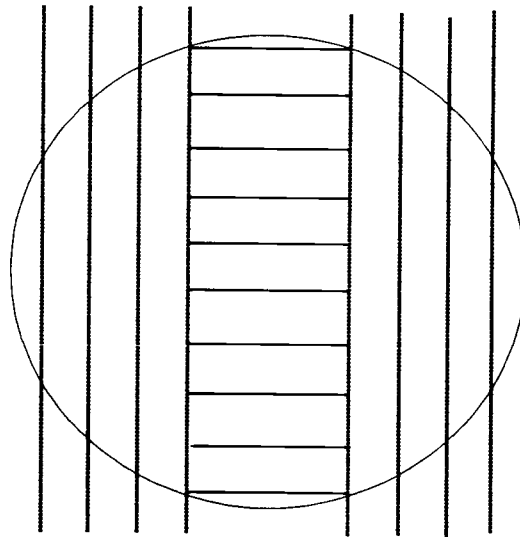
Figure 6.3. Cant sawing method.

When breaking down large logs 12 inches or larger in diameter and where the mill's major concern is to maximize profitability by recovering all of the available

highest grade lumber, the around-sawing, or grade-sawing, method is used. Figure 6.4 shows an example of the around-sawing pattern. The sawing method breaks down a log from all four sides, with sawlines perpendicular to each other and moving toward the center. This sawing method opens the log and saws the log into pieces until the face grade drops. Then the log is turned 90 degrees or 180 degrees and sawn until the face grade drops. The sawing process repeats in a similar fashion for all four sides in sequence until the remaining center portion of the log reaches a certain cant size that is to be edged or resawn.

Because it did not consider internal defects, SAW3D did not include the around-sawing method since this method depends on judgements of lumber grade. When log internal defects are considered, however, inclusion of the around-sawing method into the optimization system was required. This research developed two versions of the system. One version, SAW3DG, allows the live and cant sawing methods. Another version, SAW3DGG, allows the around-sawing method. The systems are almost identical except they use different log breakdown procedures at the headrig. Therefore, discussions on SAW3DG in all sections of this chapter are valid to both systems if there is no explicit explanation on their difference.
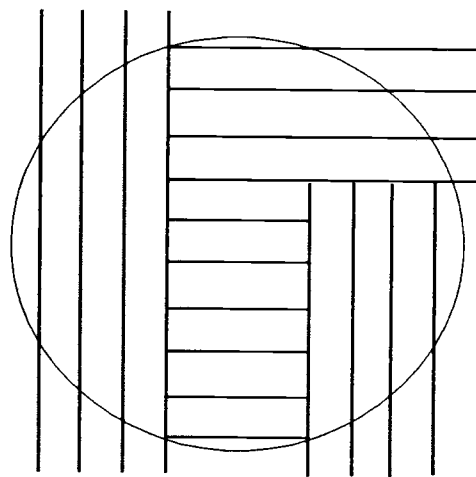
Figure 6.4. Around-sawing method.

6.2 Grading Categories

The current version of SAW3DG is intended to be used only for softwood log breakdown. Therefore, only softwood lumber grading rules are considered. Grading rules applied to softwood lumber for domestic markets are governed by the 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 of the U.S. are the Western Wood Products Association (WWPA) and the West Coast Lumber Inspection Bureau (WCLIB). SAW3DG uses the grading rules published by the WWPA as described in the Western Lumber Grading Rules 88 (1988).

Under the WWPA rules, lumber is classified into one of the following five categories based on the size and end-use: Select and Finish lumber, Boards, Dimension, Timbers, and Factory lumber and related products (Table 6.1).

Table 6.1. Grading categories

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

Each major category is further grouped into many sub-categories and each sub-category includes many grades. SLGRADER includes many of the grades in the Selects/Finish, Boards and Dimension categories. Since the focus of this research is a feasibility analysis of the methodology and prototype development of a grade sawing model, SAW3DG selects only the Commons sub-category of Boards and the Light Framing sub-category of Dimension categories that are included in SLGRADER. In addition, the grading category of Structural Joists and Planks was added into the knowledge base so that SAW3DG is able to judge the grades for dimension lumber wider than four inches. Table 6.2 show these categories and their sub-categories, and the grades used by SAW3DG. Grades followed by an asterisk are used by SAW3DG.

6.3 Defect types

Defect types included in SAW3DG are based on natural characteristics as described in the rule book. Natural characteristics and limiting provisions for each grade include wood defects and manufacturing imperfections, and are defined by their type, size, location and distribution. Tables 6.3 to 6.5 give the characteristics for each grade as described by the rule book and show the characteristics included by SLGRADER and SAW3DG. Not all characteristics listed in the rule book are considered by SAW3DG. At this development stage, the system focuses only on those characteristics that would be found in freshly sawn green lumber. Manufacturing imperfections are also not considered since this computer program does not predict manufacturing imperfections when making optimum decisions. Also, different grade categories may describe characteristics differently. In summery, the current version of SAW3DG includes the characteristics listed in Table 6.6, which are assumed to be used as input data to SAW3DG.

Table 6.2. Grading categories, sub-categories, and grades included in SLGRADER with those used by SAW3DG indicated with an asterisk.

GRADING CATEGORY	SUB-CATEGORY	GRADE
Select and Finish	Selects	B&BTR - 1&2 CLEAR
		C SELECT
		D SELECT
	Finish	SUPERIOR FINISH
		PRIME FINISH
		E FINISH
Boards	Commons	1 COMMON *
		2 COMMON *
		3 COMMON *
		4 COMMON *
		5 COMMON *
	Alternate Board Grades	SELECT MERCHANTABLE
		CONSTRUCTION
		STANDARD
		UTILITY
		ECONOMY
Dimension	Light Framing	CONSTRUCTION *
		STANDARD *
		UTILITY *
		ECONOMY *
	Structural Light Framing	SELECT STRUCTURAL
		NO. 1
		NO. 2
		NO. 3
		ECONOMY
	Stud	STUD
		ECONOMY
	Structural Joists & Planks	SELECT STRUCTURAL *
		NO. 1 *
		NO. 2 *
		NO. 3 *
		ECONOMY *

Table 6.3. Grade characteristics considered by SAW3DG for grades in the
Commons sub-category.

Characteristics	Grade				
	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 SAW3DG

N   :   Listed in the rule book but not considered by SAW3DG

N/A   :   Not listed in the rule book and not considered by SAW3DG

N/A - Y : Not listed in the rule book but considered by SAW3DG

*   :   Included in SLGRADER but not in SAW3DG

Table 6.4. Grade characteristics considered by the project for grades in the
Light Framing sub-category.

Characteristics	Grade			
	CONSTRUCTION	STANDARD	UTILITY	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 SAW3DG

N   :   Listed in the rule book but not considered by SAW3DG

N/A   :   Not listed in the rule book and not considered by SAW3DG

N/A - Y :   Not listed in the rule book but considered by SAW3DG

*   :   Included in SLGrader but not in SAW3DG

Table 6.5. Grade characteristics considered by SAW3DG for grades in
the Joists and Planks sub-category.

Characteristics	Grade				
	SELECT STRUCTURAL	NO.1	NO.2	NO.3	ECONOMY
Checks	Y	Y	Y	Y	Y
Grain	N	N	N	N	N
Knots	Y	Y	Y	Y	Y
Manufacture	N	N	N	N	N
Pitch	Y	Y	Y	Y	Y
Pockets	Y	Y	Y	Y	Y
Wane	Y	Y	Y	Y	Y
Splits	Y	Y	Y	Y	Y
Skips	N/A	N	N	N	N
Shake	N	N	N	N	N
Stain	Y	Y	Y	Y	Y
White Speck	N/A	N/A	N	N	Y
Warp	N	N	N	N	Y
Shake	N	N	N	N	N
Unsound Wood	N/A	N/A	N	N	N
Holes	Y *	Y *	Y *	Y *	Y *

Y   :   Listed in the rule book and considered by SAW3DG

N   :   Listed in the rule book but not considered by SAW3DG

N/A   :   Not listed in the rule book and not considered by SAW3DG

N/A - Y :   Not listed in the rule book but considered by SAW3DG

*   :   Included in SLGrader but not in SAW3DG

Table 6.6. Grading characteristics currently utilized by SAW3DG.

Defect	Type
Check	
Split	
Wane	
Knot	Loose
	Tight
	Red
	Black
Pocket	
Decay/Unsound Wood	
Stain	Light
	Medium
	Heavy
Pitch	Light
	Medium
	Heavy
Pith	Firm

6.4 Defect Modeling

Defect representation models used in previous research may be defined as solid-element, cylinder, mathematical, computer-array, and cross-section models. The solid-element model uses a collection of small solid three-dimensional elements or rectangular solids to represent defects. Most programs that collected defect information manually used this type of model (Harless et al., 1991; Penematicos et

al., 1974, 1976; Reynolds et al., 1969, 1970; Steele, 1994; Wagner et al., 1990). A cylinder model uses a cylinder to represent low quality wood around the pith of the log (Geerts, 1984). A mathematical model uses mathematical equations to describe defects, especially knots (Richards, 1977; Samson, 1993). A computer-array model is somewhat similar to the solid-element model. It uses a 3-dimensional array to represent a log along with internal defect information of the log. The types of defects are identified by numbers, e.g., 0 for a void element, 1 for sound wood, 2 for a knot (Allekson et al., 1980). A cross-section model uses a series of cross sections to represent defects (Todoroki, 1988, 1990). All these modeling techniques are very similar to those used for log shape modeling.

The development of log breakdown systems that consider internal defects are still in an early stage. Previous research was primarily concentrated on two aspects: knots and the low quality wood around the pith of a log. The reason that so much research was devoted to knots is perhaps that, among all defects, knots probably have the largest influence on the quality of logs and the value yield of the lumber produced from those logs. Low quality wood in the center portion of logs also has an important influence on the quality of logs and their value yields and is easy to represent with a simplified model such as a cylinder.

This research uses a polygonal cross-section model to represent all defects. There are a few good reasons to do so. First, using the same modeling technique to represent a log and all its characteristics makes the system design consistent and easier to implement because the same data structure can be used for all of them. Second, this model is more realistic than other simplified models and mathematical models. Third, a polygonal cross-section model can be easily constructed using cross-section images obtained by a scanning system. A scanning image of a cross-section can show the true shape of the log and all its nature characteristics on the cross-section. This image naturally fits in a cross-section model. Figures 6.5 shows a knot represented by a polygonal cross-section model and Figure 6.6 displays a log with several internal knots in such cross-section representations.
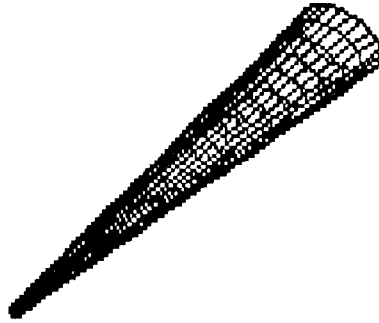
Figure 6.5. A knot model.
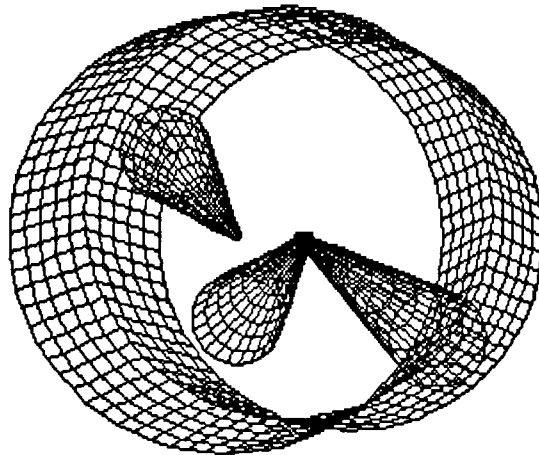
Figure 6.6. End view of a log with several knots inside.

When a sawing plane passes through a defect, a two-dimensional defect area is exposed on the surface. Some programs used a polygon to represent such a defect area while others used a rectangle. This research uses both polygons and rectangles to represent a defect area. Polygons are used to model defect areas exposed on all surfaces of unedged and untrimmed lumber so that errors resulting from using the simplified rectangles are not introduced in the early stages of log processing. Rectangles are used for defect areas mapped on four faces of the finished lumber since the lumber grading requires measurement of the defect dimensions parallel to edges of the finished lumber.

6.5 System Components

SAW3DG consists of four basic components as shown in Figure 6.7. The headrig optimization component performs the operations of log positioning such as log rotation and skewing, saws the log into slabs, flitches, and/or cants, and determines the optimum sawing patterns that optimize value yield of the log by applying the headrig dynamic programming (DP) algorithm. To do so, this component needs to know the optimum value of each flitch or cant cut from the log.

The optimum value yield of each flitch or cant produced from the log is determined by the edging optimization component. The edging optimization component receives profile information for each flitch or cant from the headrig optimization component along with profiles of defects exposed on the faces of that flitch or cant by sawing lines. It performs the positioning operation and then edges the flitch or cant into untrimmed lumber and applies the edging DP algorithm to determine the optimum edging patterns. To do so, this component needs to know the optimum value of each edged but untrimmed lumber.

The optimum value yield of the untrimmed lumber is determined by the trimming optimization component. Profile information for the untrimmed lumber and profiles of defects exposed on the edges or faces of the lumber by edging
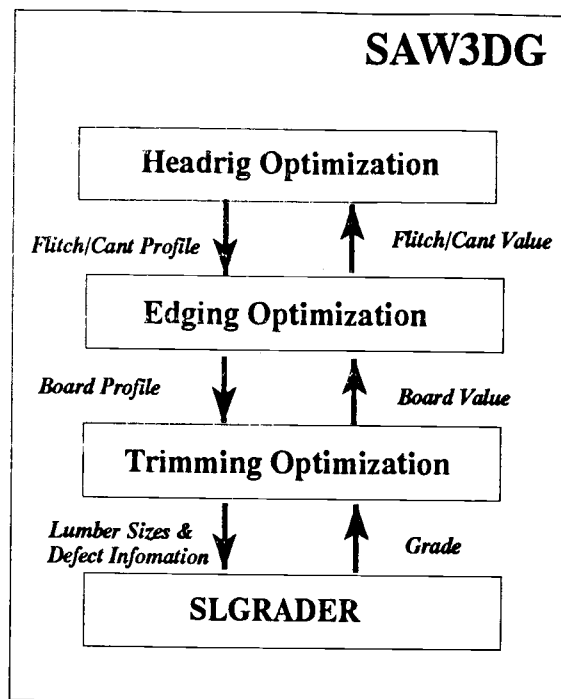
Figure 6.7. SAW3DG system components.

lines are passed to the trimming optimization component. This component then trims the lumber into finished final lengths, and passes the lumber dimensions and information about defects to SLGRADER. The value of each piece of trimmed lumber is then determined by its dimensions and grade and used by the trimming DP algorithm to decide the optimum trimming patterns.

6.5.1 Headrig Optimization

6.5.1.1 Processing Procedure for the System When Using the Live and Cant Sawing Methods

The headrig optimization component of SAW3DG determines the optimum log position and sawing patterns. Figure 6.8 shows the process of breaking down a log at the headrig. Input data include profile data of the log and its internal defects, production parameters such as saw kerfs and sawing variations, lumber price tables for various lumber grades, and control options for selecting sawing strategies and optimization levels. SAW3DG uses heuristic methods to determine the optimum log rotation angle and the optimum skewing position as described in Zeng (1991), and the headrig DP algorithm to determine the optimum sawing pattern resulting in the maximum value yield for the log at a specific position.

The process begins by rotating the log with all its internal defects. The log can be rotated through the full 360 degrees at any rotating increment angle selected by the user. Once a rotation position is fixed and the log is rotated to that position, the program will determine a skew range in which the log skew position will be searched. Skewing the log means moving it horizontally in the coordinate system. The skewing position increment is also entered by the user. Details on the rotating and skewing methods are described in Zeng (1991). In SAW3DG, all positioning operations are performed on the log and all its internal defects.

After the position of the log has been fixed, the program breaks it down into slabs, flitches, and/or cants using the headrig DP algorithm. The log breakdown process using the DP algorithm is discussed in Section 6.5.1.3. The value yield of each trial flitch or cant between two sawing lines is determined by the edging optimization component. To do so the headrig component needs to determine the two face profiles of that piece resulting from the intersection of the log with the two sawing planes and profiles of defects exposed on the two faces. The edging optimization component receives the profiles and returns the maximum value of the
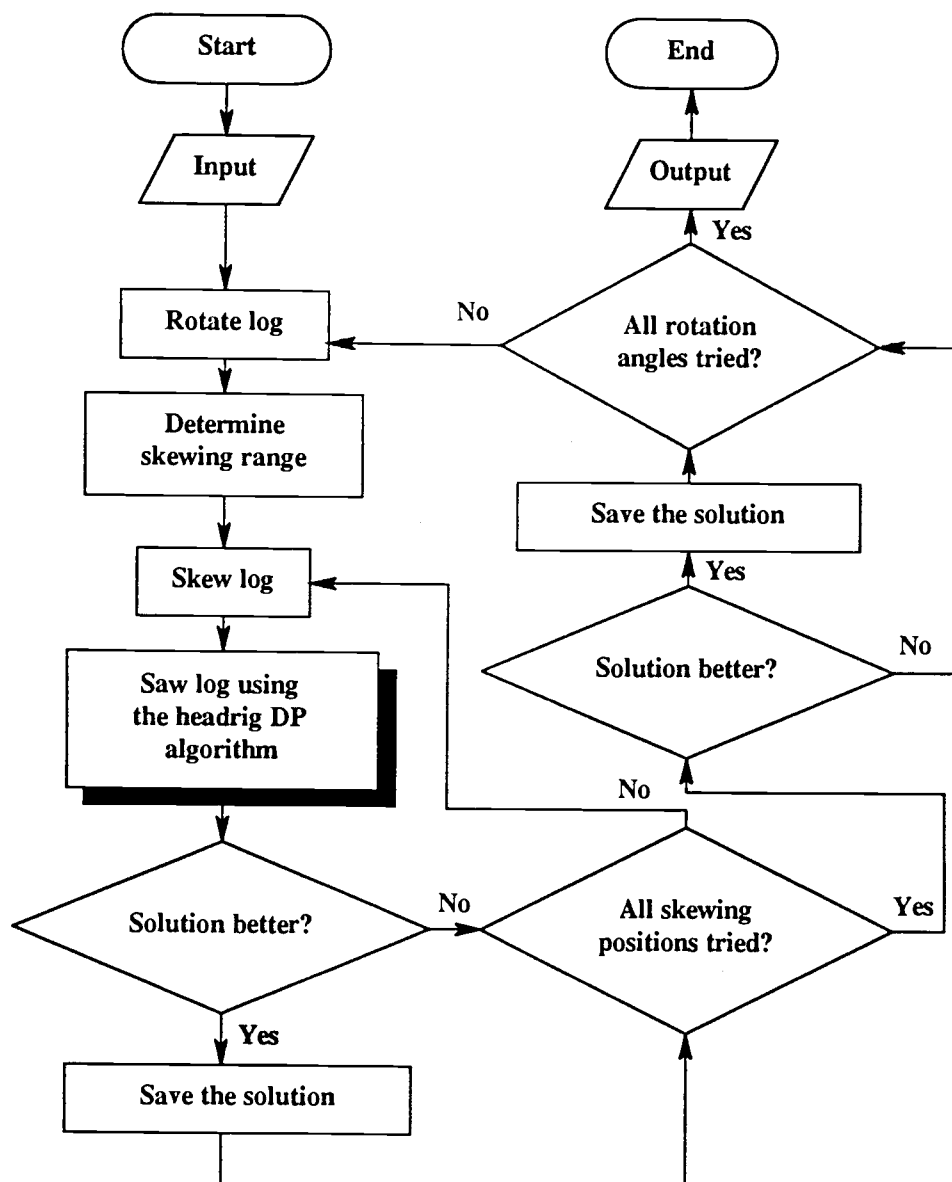
Figure 6.8. Log breakdown procedure of SAW3DG.

flitch or cant to the headrig optimization component. The value of each flitch or cant will be used by the DP algorithm of the headrig component to determine the optimum sawing pattern resulting in the maximum total value of all flitches and cants. This total value is compared with that given by the previous log position and the best one is saved along with all corresponding optimum position information and the sawing pattern. This process is repeated until all log rotation and skewing positions are enumerated and a final optimum solution is reached. Obviously, this optimum solution is only optimal within the boundaries established by the input parameters.

6.5.1.2 Processing Procedure for the System When Using the Around-sawing Method

As mentioned before, SAW3DGG is almost identical to SAW3DG, except that the log processing procedure at the headrig simulates an around-sawing method. To describe the procedure, a definition of four sides of a log is shown in Figure 6.9. Two adjacent sides are perpendicular to each other. The side index number starts from the left side of the log and increases clockwise, as in the sequence of Side 1, Side 2, Side 3, and Side 4. Two sides, Side 5, and Side 6, are the two perpendicular sides on the center cant. The breakdown process starts from Side 1. SAW3DGG uses the live sawing method and the headrig DP algorithm, which are identical to that of SAW3DG, to saw the log. SAW3DGG determines the grade of the log face every time a flitch is removed. The log face grade is the grade of the lumber that has the maximum width and length within the polygonal shape of the exposed face. If the grade is lower than that of the previous face in the same grading category, the process stops and the log is turned 90 degrees. Otherwise, if the face grade does not drop when any one of the two sawing lines has passed the pith of the log, the breakdown process continues without considering face grade any more until all the remaining portions of the log are processed. This procedure repeats for all four sides in sequence. SAW3DGG then saws the center cant starting from Side 5, using the procedure identical to that of SAW3DG using the live sawing method. After that,
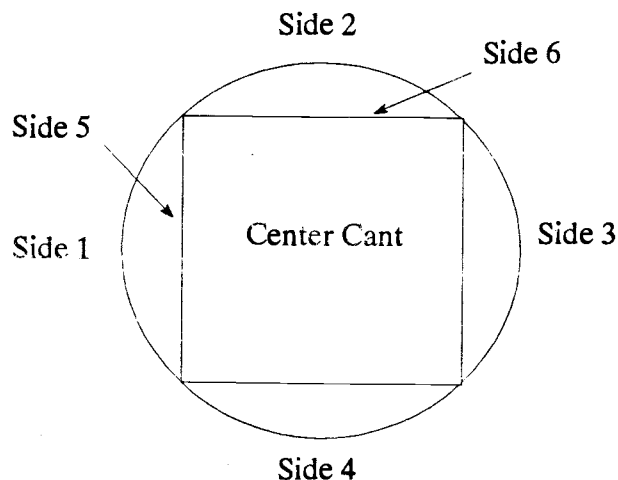
Figure 6.9. Side index of around-sawing method.

SAW3DGG turns the center cant 90 degrees to Side 6 and saws the cant again. The cant edging or resawing pattern resulting in the best value yield of the cant is saved as the final solution for the center cant. This procedure is diagrammed in Figure 6.10.

6.5.1.3 Sawing Optimization Algorithm

Once the position of a log is determined, SAW3DG uses the log-breakdown dynamic programming algorithm to find the optimum sawing pattern. The algorithm used in SAW3DG is the same as that in SAW3D, which is described in Zeng (1991) in great detail. This section only briefly explains the algorithm using a simplified equation.

The algorithm is best illustrated using a diagram shown in Figure 6.11. A saw placement can be any where (i, or j, for example) between the left and right boundaries. Saw placements at i and j result in a piece cut off the log. The value of this piece, $v(i,j)$, is to be determined by the edging optimization component. When
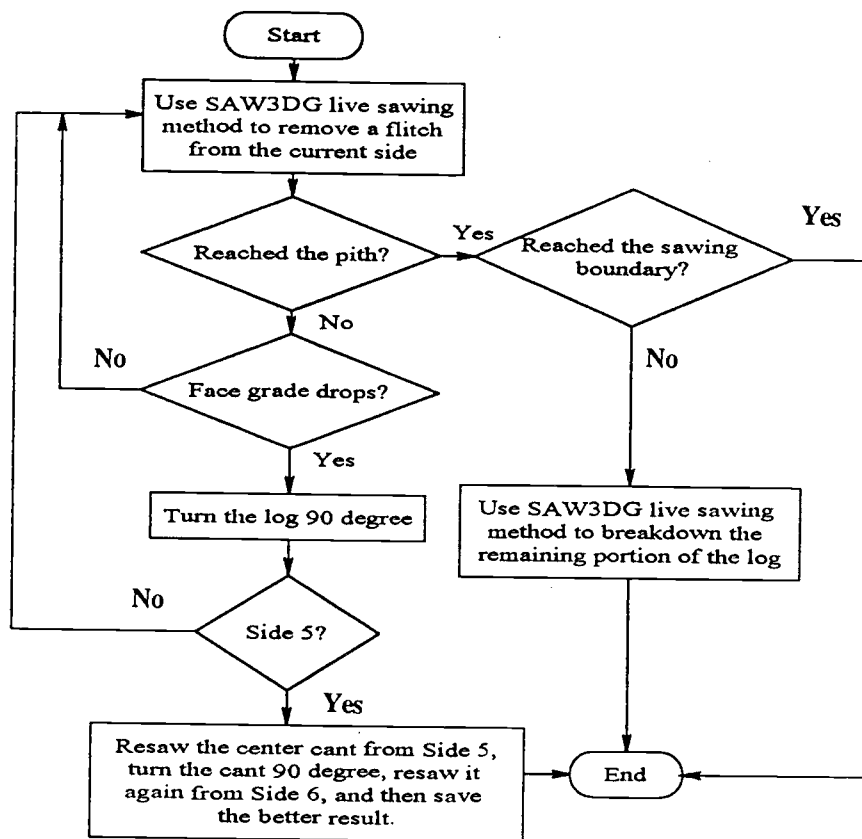
Figure 6.10. Log breakdown procedure of SAW3DGG.
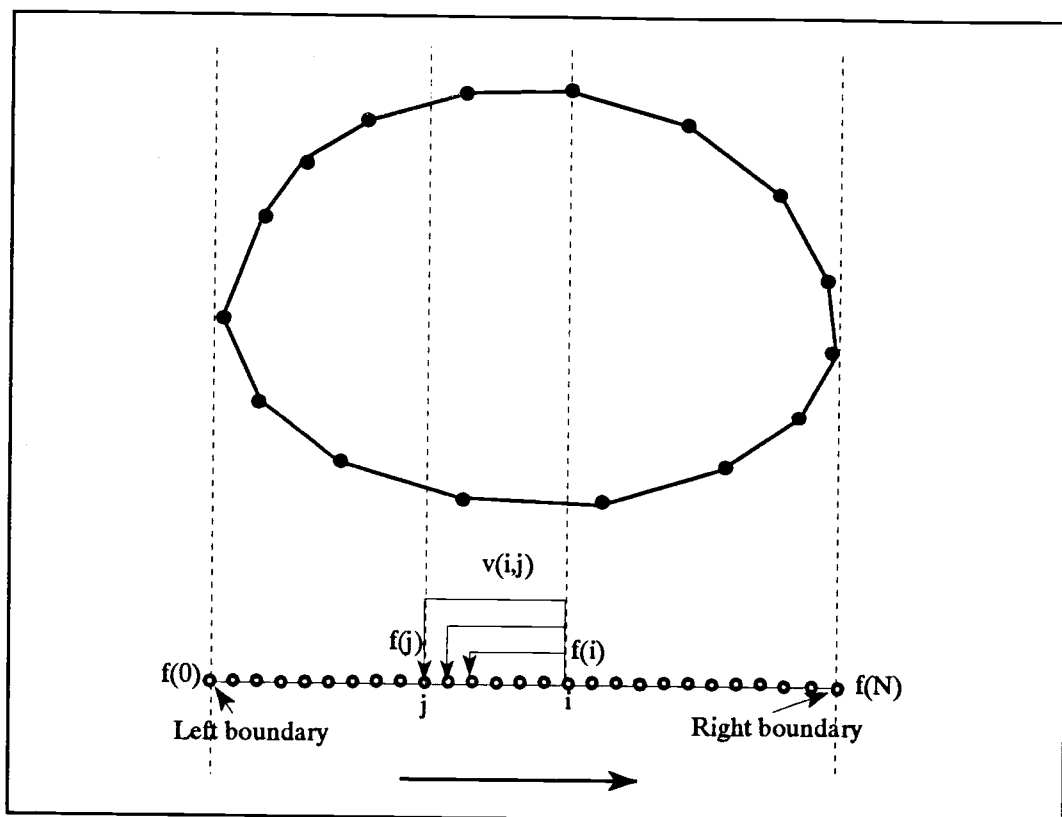
Figure 6.11. An illustration of the headrig DP algorithm.

the right sawing plane of a piece is placed at i, there are a number of choices on where to place its left sawing plane. This number is equal to the number of flitch and cant thicknesses plus the number of piece thicknesses that are too thin to be used as lumber. The objective of the algorithm is to maximize the value return from the log within the boundaries.

The algorithm can be expressed using a recursive relationship in the following manner:

f(0) = 0;

For i = 1 to N, Do:

$$f(i) \quad = \quad \max \quad \{ v(i,j) + f(j) \} \qquad\qquad 1 \le j \le m(i)$$

where:

N   =   number of possible saw placements between the left and right boundaries;

m(i)   =   number of flitch and cant thicknesses plus the number of piece thicknesses that are too thin to be used as lumber at the ith saw placement; this number may be different at various i positions, depending on the thickness from the left boundary to the i position;

i   =   a possible saw placement position between the boundaries;

j   =   a possible saw placement position between the boundaries ;

v(i,j)   =   the optimum value yield between saw placements i and j, which is determined by the edging optimization component;

f(j)   =   the optimum value yield of the portion of the log from the left boundary to saw placement j;

f(i)   =   the optimum value yield of the portion of the log from the left boundary to the current saw placement i.

The optimum value yield of the log is determined by solving this equation recursively from the left to the right boundaries. Figure 6.12 shows the flow chart of the algorithm. The interrelationship between the optimum headrig and the optimum edger components as described by the v(i,j) element of this equation, is shown in Figures 6.12 and 6.13.
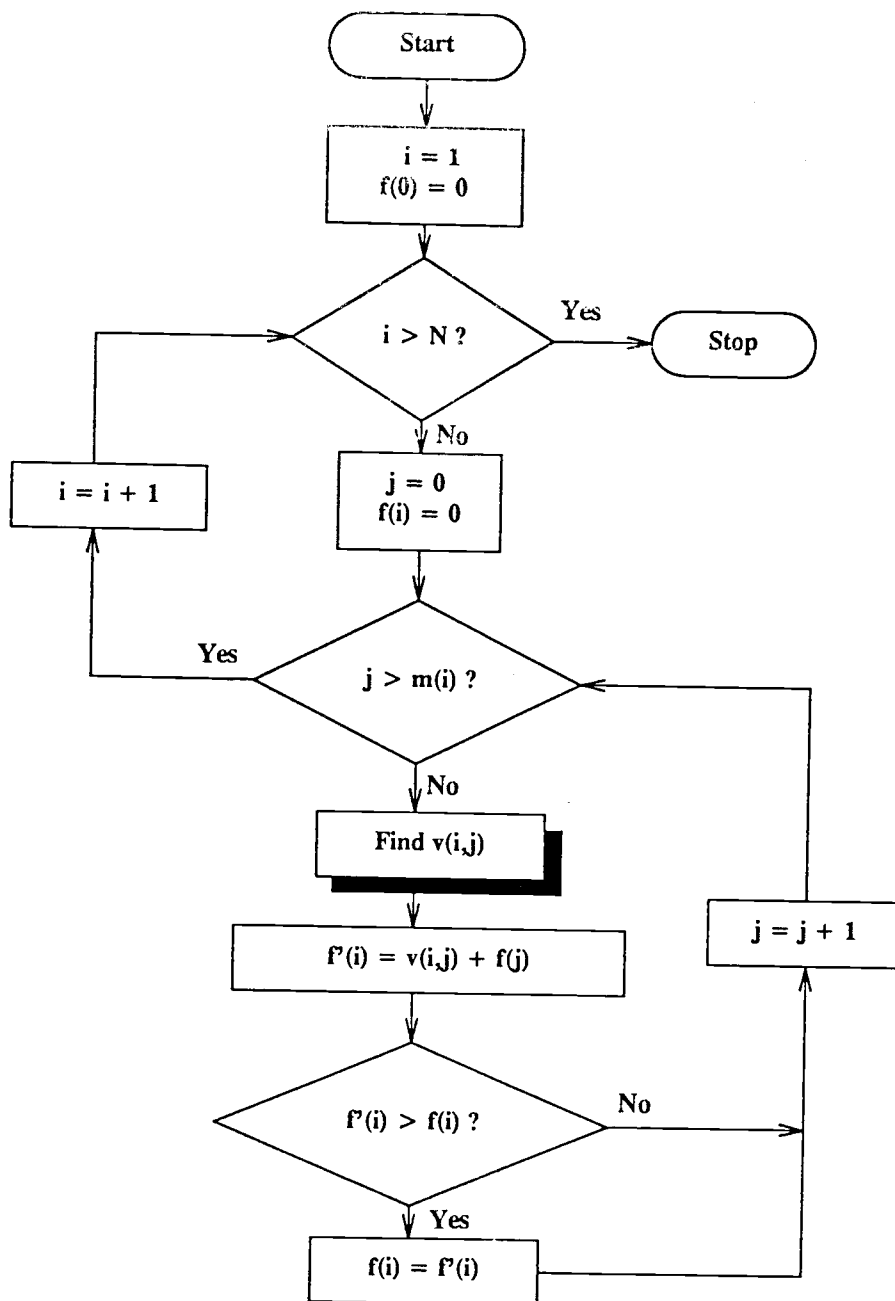
Figure 6.12. Flow chart of the headrig DP algorithm.

```
┌─────────────────────────────────────────┐
│              Find v(i, j)                │
│                                          │
│   ┌──────────────────────────────┐       │
│   │ 1. Determine profiles of the │       │
│   │    flitch/cant between i and j; │     │
│   │                              │       │
│   │ 2. Determine defects mapped  │       │
│   │    on the surfaces of the flitch or │ │
│   │    cant.                     │       │
│   └──────────────────────────────┘       │
│                   ⇓                      │
│   ┌──────────────────────────────┐       │
│   │ The Edging Component :       │       │
│   │                              │       │
│   │ Determine the optimum value  │       │
│   │ yield of the flitch/cant.    │       │
│   └──────────────────────────────┘       │
└─────────────────────────────────────────┘
```
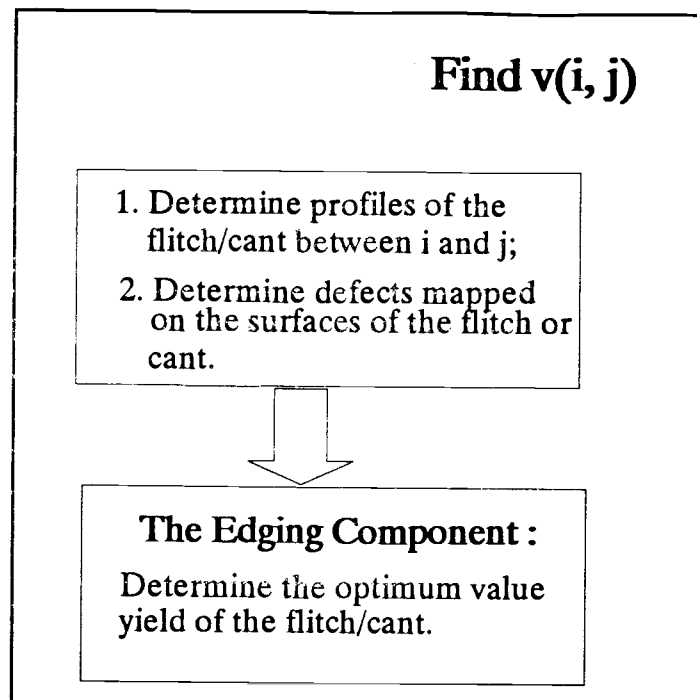
Figure 6.13. Interrelationship between the headrig and edging components.

6.5.1.4 Determining Profiles of the Piece

For the edging optimization component to determine the optimum value yield of a piece between two sawing planes, it requires information about the profiles of the piece and defects within this piece. The profile of each face of this piece, which is formed by the intersection of each sawing plane at a given X coordinate with all cross sections, is determined by the same procedure used in SAW3D. The detailed steps to determine an intersecting face of the log with a sawing plane are described in Zeng (1991). Figure 6.14 shows a flitch described using the profiles of two faces with wane on its edges.

Wane
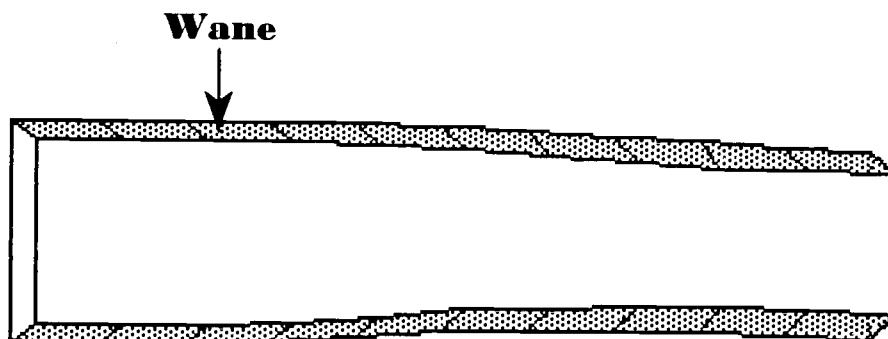
Figure 6.14. A flitch described using profiles of two faces.

6.5.1.5 Defect Mapping in Headrig Optimization Component

In addition to providing profiles of two faces of the piece to the edging component, the headrig component also needs to locate defects within the two sawing planes and their maps exposed on the two faces, and then pass all information to the edging optimization component. Figure 6.15 shows an example of a knot cut through by the two sawing planes.

When a sawing plane cuts through a defect polyhedron, a defect area is exposed on the intersecting surface of the log. This defect area may be represented by a polygon or simplified rectangle. The procedure to determine the defects mapped on the left and right surfaces of the piece involves the following steps.

Step 1:   Locate defects between the left and right saw planes, including defects intersecting with the two saw planes, as shown in Figure 6.15.

Step 2:   Determine the vertical polygons of defects resulting from the intersection of the log and sawing planes on the left and right faces. This is done by the same procedure used to determine the profile of a
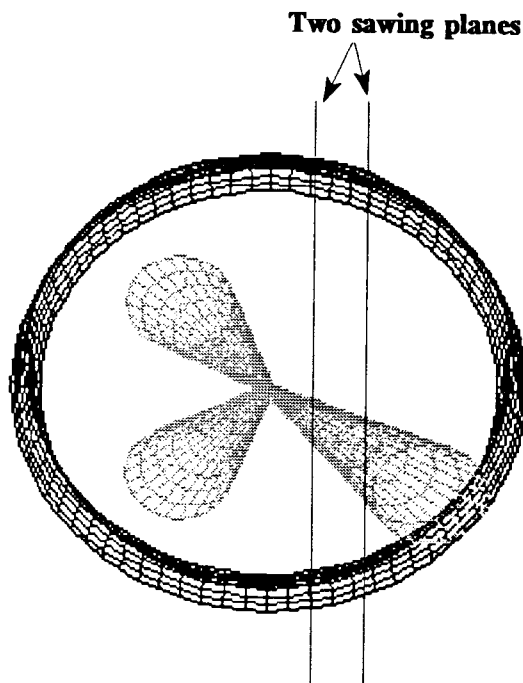
Two sawing planes

Figure 6.15. Two sawing planes cutting through a knot.

face of the piece. Figure 6.16 shows an example of a defect polygon mapped on the surface of the piece.

Step 3:   Find the largest rectangle encasing the defect polygons determined in Step 2.

The lumber grading involves measurement of defect dimensions parallel to the edges of the finished lumber, that is, only the largest rectangle is needed to measure the defects on a surface. Therefore, finding a largest rectangle of a defect polygon at this stage will save computing time at subsequent stages. However, using a rectangle to represent a defect at this early stage may introduce errors if at the
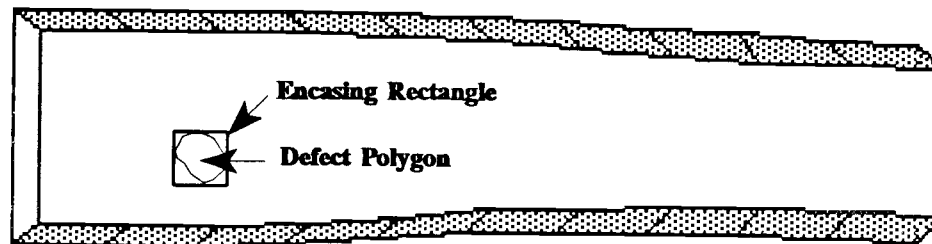
Figure 6.16. A defect polygon shape exposed on the flitch face.

subsequent stages, such as the trimmer, a cutting plane could pass through the defect. To prevent this type of error, all information on the defects, including defects within the saw planes, defect polygon maps, and simplified rectangles, is stored in an index array using data structures for polygons and rectangles, respectively, and is accessible later by the edging and trimming optimization components.

6.5.2 The Edging Optimization

Given a flitch or cant from the headrig component, the edging optimization component determines the optimum position of the piece and the optimum edging patterns. Figure 6.17 shows the process of edging a flitch or cant by this component. Required data include profile data of the piece, the defects inside it, defect polygons mapped on the two faces of the piece, production parameters such as edging saw kerfs and sawing variations, and control options for selecting positioning levels. This component uses a heuristic search method to determine the optimum positioning angle and the edging DP algorithm to determine the optimum edging pattern resulting in the
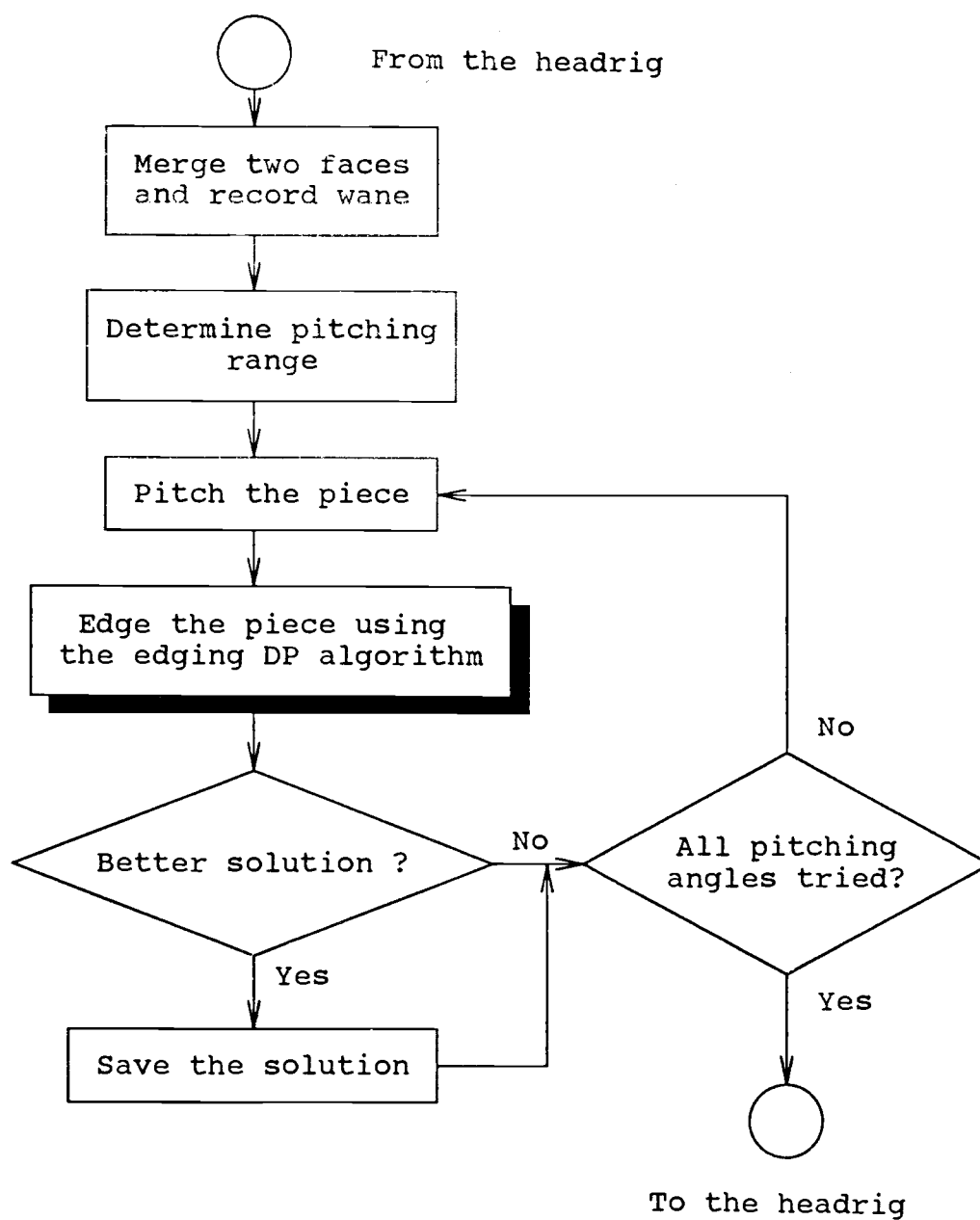
Figure 6.17. Processing procedure of the edging optimization component.

maximum value yield of the piece at a specific position. Details of the pitching operations are described in Zeng (1991).

6.5.2.1 Processing Procedure

To reduce computing time, the edging component first merges the two faces together. All subsequent operations and calculations are performed on the merged profile. Merging of the two faces and recording of full wane are followed by pitching the piece including all its internal defects and defect polygons mapped on its surfaces. Pitching refers to moving the piece vertically in the coordinate system, which is equivalent to laying down the piece on the edger and orienting it so that the edging plane is at an angle to the center line of the piece.

Using a pitching position increment entered by the user, the edger first determines a pitching range and then orients the piece to an angle within the range. Once a pitching position is fixed and the piece is oriented to that position, the edging operation then cuts the piece into untrimmed lumber. Defects exposed on the flitch or the cant are mapped to two dimensional polygons and used later by the trimming optimization component. The optimum value yield of each untrimmed piece produced by the edging component will be determined by the optimum trimmer. The value of each untrimmed piece returned from the trimming component is used by the DP algorithm of the edging component to determine the optimum edging pattern resulting in the maximum total value of the flitch or cant. This total value is compared with that given by the previous pitching position, and the best one is saved along with the corresponding optimum position information and the edging pattern. This process is repeated until all pitching positions are enumerated and the final optimum solution is reached. At last, this final optimum value is returned to the headrig component where it is used to determine the optimum sawing pattern.

6.5.2.2 Merging Two Faces and Recording Wane

As mentioned before, merging of the two faces can reduce processing time because all subsequent operations and calculations need to be performed only on a

merged face rather than two unmerged faces. To retain the full information about the profiles of the two faces, however, wane on edges of the piece need to be fully recorded. In SAW3D, wane allowances are calculated when merging two faces into one and based only on the narrowest (4 inches) and thinnest (1 inch) lumber sizes typically allowed in the industry. This simplification is reasonable when the system has no lumber grading capability. Since SAW3DG is able to grade lumber based on wane allowances on the lumber, it allows variable wane allowances. SAW3DG merges the two faces into one using the maximum and minimum Y coordinates, and records wane in four arrays of data structures to be used in the edging and trimming components. The procedure is similar to that used in SAW3D, except wane is fully recorded rather than calculated using fixed wane allowances. This is done in the following steps. Figure 6.18 shows a cross section of a flitch, which is used as an example to explain these steps.

Step 1: If any one of the four potential intersection points is not on the cross section (i.e., it is not an intersection point), eliminate the segment between this cross section and the immediately previous cross section by not adding the interval between the two sections to the length of the piece. Then go to Step 6.

Step 2: Find the merged upper and lower points, which are the maximum and minimum Y coordinates among the four points. In Figure 6.18, the merged points for this cross section will be right-upper and right-lower Y coordinates.

Step 3: Record wane in a data structure for each corner, which are upper left wane, lower-left wane, upper-right wane, and lower-right wane, respectively. Wane at each corner includes two dimensions, a vertical dimension on the wider face and a horizontal dimension on the narrower face. For example, in Figure 6.18, wane at the upper-left corner has two dimensions: the vertical dimension on the left face is the distance between Y coordinates of the upper-right point and upper-left point, while the horizontal dimension on the upper face is the
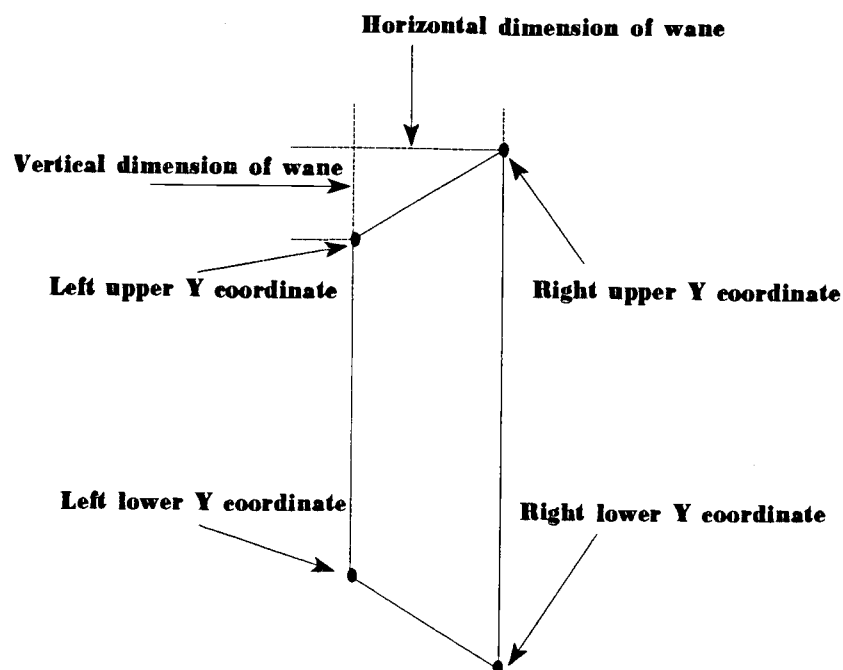
Figure 6.18. A cross section of a flitch.

distance between X coordinates of the upper-left and upper-right points. After the edging and trimming operations, wane on each face may be changed according to the positions of the edging and trimming planes. Therefore, at this stage Y and Z coordinates corresponding to wanes on each cross section are recorded so that subsequent stages can use them to determine the sizes of wane on finished lumber. In Figure 6.18, wane on the right face will be zero.

Step 4: If the distance between the merged upper and lower points is greater than or equal to the narrowest acceptable lumber width, add the interval between cross sections to the length of the piece. Otherwise, go to Step 6 if the previous length is greater than or equal to the shortest acceptable lumber length, or initialize the length (let it be zero) and go to Step 5 if the previous length is less than the shortest acceptable length.

Step 5:   Repeat Step 1 to Step 4 until all cross sections have been tried.

Step 6:   End the procedure.

6.5.2.3 Edging Optimization Algorithm

Once the position of the flitch or cant is determined, SAW3DG uses the edging dynamic programming algorithm to find the optimum edging pattern. The algorithm used in SAW3DG is the same as that in SAW3D, which is described in Zeng (1991) in great detail. This section only briefly explains the algorithm using a simplified equation.

As was the case for the DP algorithm for the headrig, the edging DP algorithm is best illustrated using the diagram shown in Figure 6.19. An edging saw plane can be any where (i or j, for example) between the lower and upper edging boundaries. Edging saws placed i and j result in a edged but untrimmed piece from the flitch or cant. Value of this untrimmed piece, u(i,j), is to be determined by the trimming optimization component. When the upper sawing plane is placed at i, there are a number of choices to place the lower edging saw plane to produce a edged piece. This number is equal to the number widths (thicknesses for a cant) plus the number of widths that are too narrow to be used as lumber. The objective of the algorithm is to maximize the value return from the flitch or cant.

The algorithm can be expressed in a recursive relationship as the following:

$f(0) = 0;$

For i = 1 to N, Do:

$$f(i) = \max \{ u(i,j) + f(j) \} \qquad 1 \leq j \leq m(i)$$

where:

N   =   number of possible edging saw placements between the lower and upper boundaries;

m(i)	=	number of possible widths (or thicknesses for a cant) plus the number of edged piece widths (thickness for a cant) that are too narrow to be used as lumber; this number may be different at various i positions, depending on the width from the lower boundary to the i the position;
i	=	a possible edging saw placement within the boundaries;
j	=	a possible edging saw placement within the boundaries;
u(i,j)	=	the optimum value yield between edging saw placements i and j, which is determined by the trimming optimization component;
f(j)	=	the optimum value yield of the portion of the unedged piece from the lower boundary to edging saw placement j;
f(i)	=	the optimum value yield of the portion of the unedged piece from the lower boundary to the current edging saw placement i.

The optimum value yield of the unedged flitch or cant is determined by solving this equation recursively from the lower to the upper boundaries. The interrelationship between the edging optimization and the trimming optimization is described by the u(i,j) element of this equation, as shown in Figures 6.19 and 6.20.

6.5.2.4 Defect Mapping in Edging Optimization Component

Defect mapping manipulations at the edger include the following two tasks:

(1) Determining defect polygons mapped on the upper and lower faces of the piece. The upper and lower faces are equivalent to the narrower faces of a flitch or the wider faces of a cant.

(2) Modifying the defect rectangles mapped on the vertical faces of the piece determined at the headrig.

Edging saw planes through defect polyhedrons expose defect areas on the upper and lower faces of the piece. The defect areas may be represented by polygons
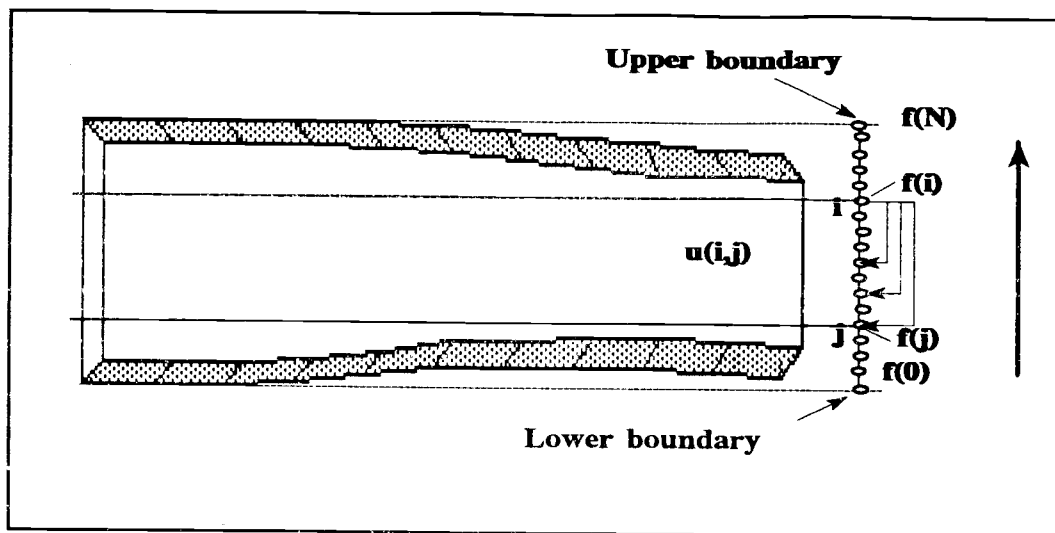
Figure 6.19. An illustration of the edging DP algorithm.
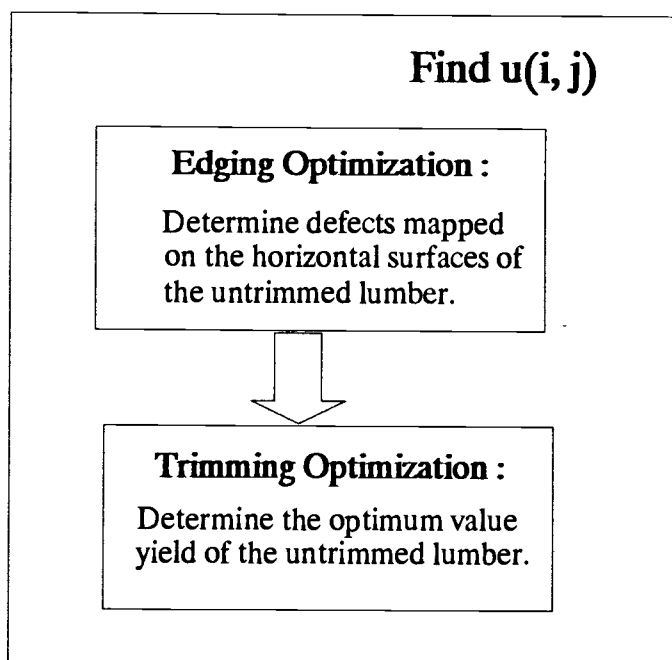
Figure 6.20. Interrelationship between the edging and trimming components.

or simplified rectangles. The procedure determining the defects mapped on the upper and lower surfaces of the piece is similar to that used at the headrig and involves the following steps.

Step 1:   Among those defects within the left and right sawing planes, locate defects between the upper and lower edging saw planes, including defects intersecting with the two edging saw planes. Figure 6.21 shows an example of an edging plane cutting through a knot.

Step 2:   Determine the intersecting horizontal polygons of defects on the upper and lower faces. This is done by a procedure similar to that used to determine the profile of a face of the piece. Here those operations are now performed on the horizontal plane instead the vertical plane.

Step 3:   Determine the left and right boundaries of the defect polygons. Figure 6.22 shows an example of a defect polygon mapped on a surface of the piece.

Step 4:   Find the largest rectangle encasing the defect polygons determined in Step 3.

Similar to the problem mentioned in section 6.5.1.5, finding a largest rectangle of a horizontal defect polygon at this stage will save computing time at the trimmer. However, using a rectangle to represent a defect at this stage may introduce some error if at the trimmer a cutting line passes through the defect. To prevent this type of error, information about the defects such as defect polygons and simplified rectangles are stored in data structures and accessible later by the trimming optimization stage.

Once the upper and lower edging saw placements are determined, the defects mapped on the left and right faces of the piece at the headrig also need to be checked. If an edging plane passes through a vertical defect rectangle, that rectangle is modified by using the edging line as one of its boundaries. This is shown in Figure 6.23. where the vertical size of the defect rectangle is modified.

If an edging plane passes through wane found at the headrig, sizes of the wane can be determined based on the position of the edging line. This is done at the
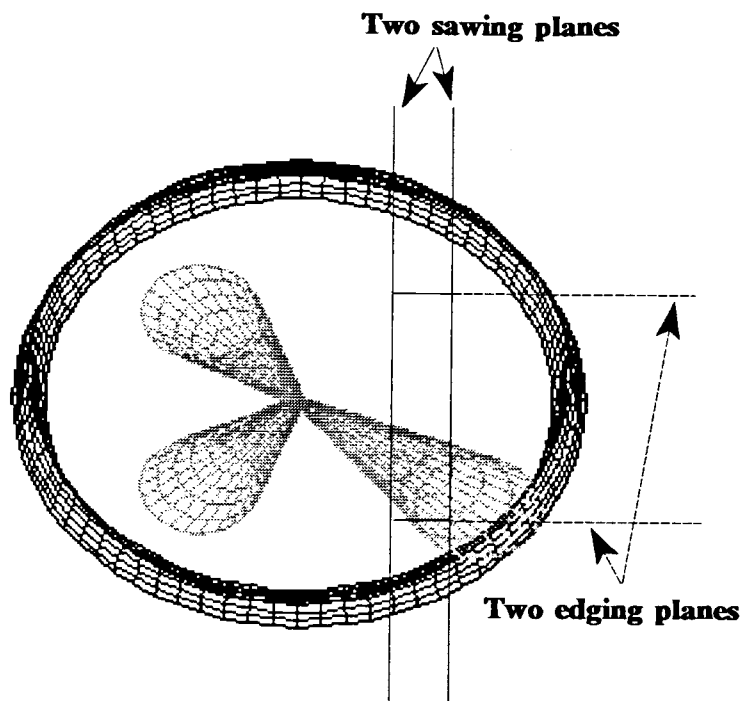
Figure 6.21. An edging plane cut through a knot.
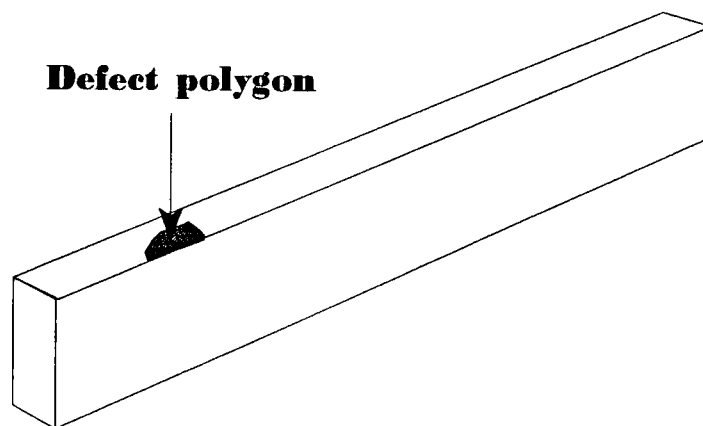
Figure 6.22. A defect polygon exposed on a narrow face of a flitch by an edging plane.
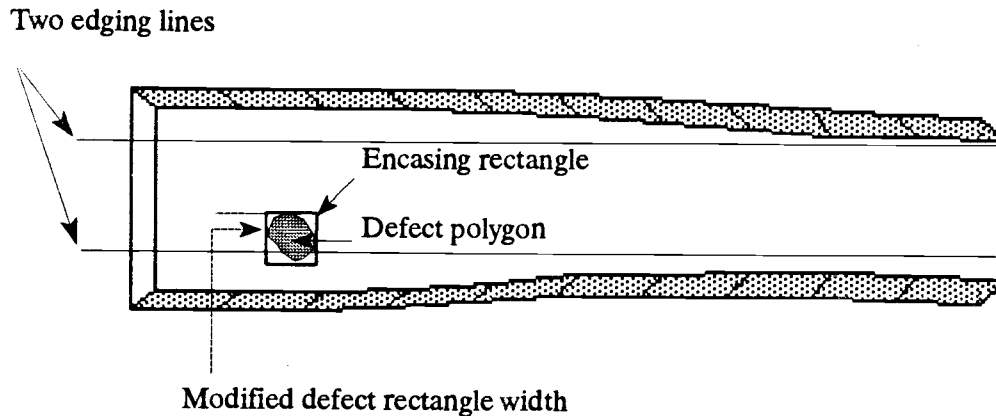
Two edging lines

Figure 6.23. An edging plane cut through a defect rectangle.

trimmer after the trimming line position is determined, since the sizes of those wane area may also depend on the trimming line positions. The edger only checks if an edging plane cuts through any wane area. If so, the edger sets a flag indicating a wane needs to be determined later by the trimmer.

6.5.3 The Trimming Optimization

Given the geometric profile of the unedged lumber, the edging lines determined in the edging optimization component, the defects inside the piece and all the defect areas mapped on the four sides of the untrimmed lumber, the trimming optimization component applies the trimming dynamic programming algorithm to determine the optimum trimming decision. There are not any positioning operations involved in this stage. Details on the procedure of locating the trimming zones are described in Zeng (1991).

6.5.3.1 Processing Procedure

The process starts by locating acceptable trimming zones which are rectangles with the length equal to or larger than the shortest acceptable lumber size. The DP algorithm is then applied to each trimming zone to find the best trimming pattern on that trimming zone. Once positions of two trimming lines are decided, the algorithm checks defects within the trimming lines and modifies sizes of defect areas which are cut through by the trimming lines. Then the trimming component calculates the maximum size of each type of defect on each of the four sides of the lumber. If any edging line passes through a wane area, the trimmer will determine sizes of wane using the wane information recorded in the edging component. All that information along with lumber dimensions is sent to SLGRADER, the expert system for softwood lumber grading, to determine the lumber grade. Value of the lumber is determined by looking up the price table associated with the lumber grade returned by SLGRADER, and then the value of the finished lumber is used by the DP algorithm to determine the optimum trimming pattern resulting in the best value yield of the untrimmed lumber. Figure 6.24 shows the flow chart of the process.

6.5.3.2 Trimming Optimization Algorithm

For each trimming zone on the untrimmed lumber, SAW3DG applies the trimming DP algorithm to determine the optimum trimming pattern. Similar to the previous two DP algorithms, this DP algorithm can be described using a simplified equation in the network representation. For details about the algorithm, please refer to Zeng (1991).

The algorithm can be expressed in a recursive relationship as the following:

$f(0) = 0;$

For $i = 1$ to N, Do:

From the edger

Find trimming zones

Trim the zone using
the trimming DP algorithm

Add value of the zone
to the total value of
the untrimmed lumber

All trimming zones tried?

No

Yes

To the edger
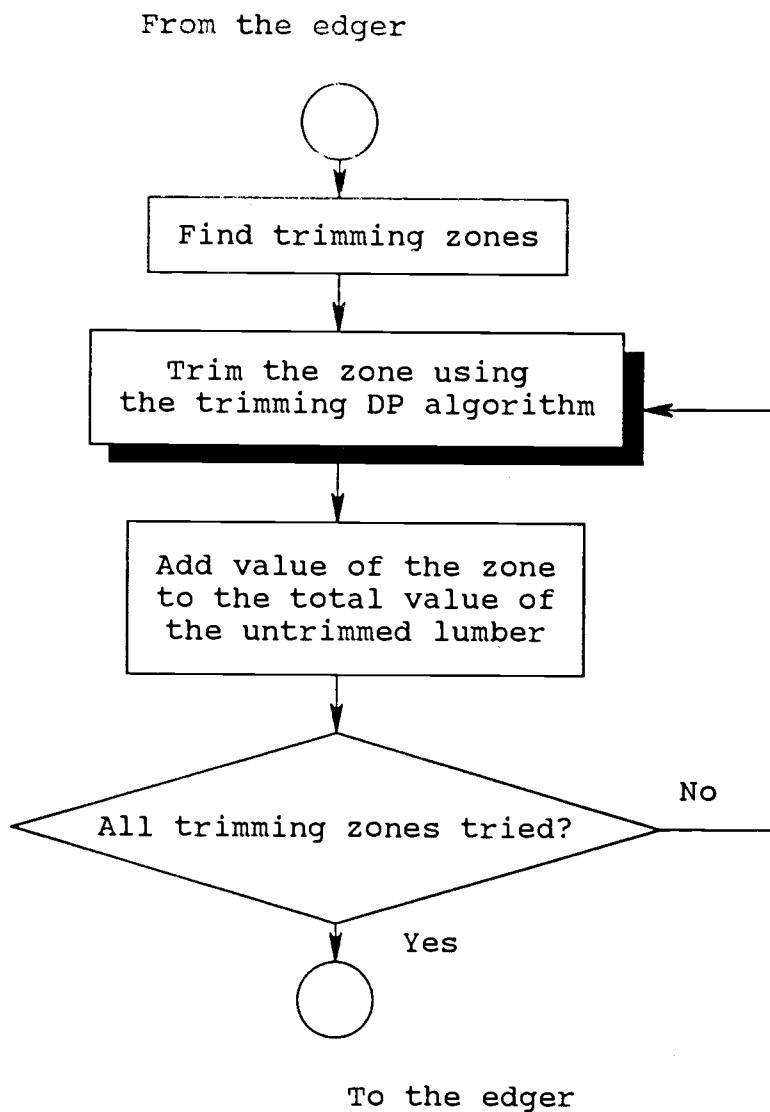
Figure 6.24. Processing procedure of the trimming component.

$$f(i) = \max \{ t(i,j) + f(i,j) \} \qquad 1 \le j \le m(i)$$

where:

N   =   number of possible trimming saw placements between the trimming boundaries;

m(i)   =   number of possible lumber lengths at the ith trimming saw placement;

i   =   trimming saw placement at ith position;

j   =   trimming saw placement at ith position;

t(i,j) =   the optimum value yield between the trimming saw placements i and j, which is determined by looking up the price table associated with the lumber grade and sizes;

f(j)   =   the optimum value yield of the portion of the trimming zone from the left trimming boundary to the trimming saw placement j;

f(i)   =   the optimum value yield of the portion of the trimming zone from the left trimming boundary to the current trimming saw placement i.

The optimum value yield of the trimming zone is determined by solving this equation recursively from the left to the right boundaries. The interrelationship between the trimming optimization component and the lumber grading expert system is described in the t(i,j) element, as shown in Figure 6.25.

6.5.3.3 Defect Mapping in Trimming Optimization Component

Defects within the untrimmed lumber, and defect polygons and their encasing rectangles mapped on the four sides of the untrimmed lumber were already found in the previous stages. However, since the trimming zone may occupy only a partial
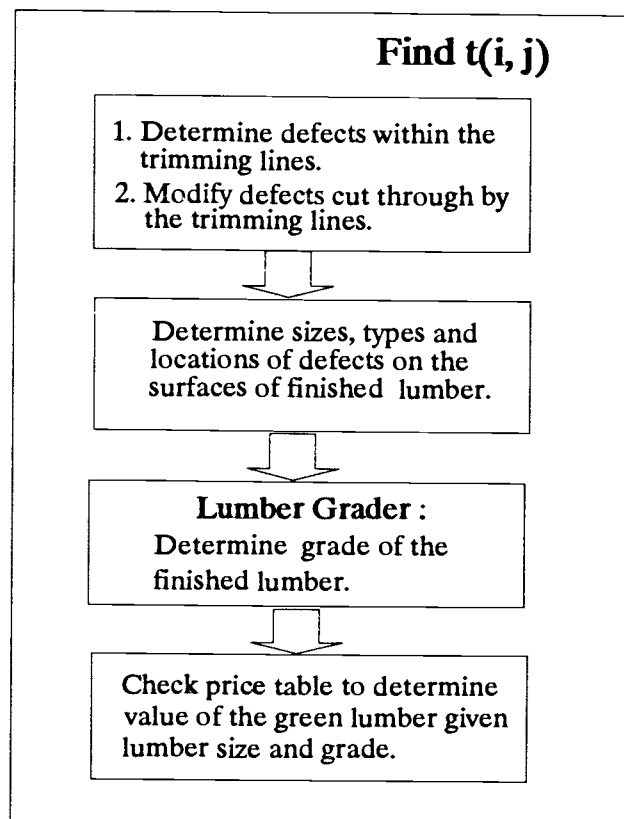
Figure 6.25. Interrelationship between the trimming optimization and SLGRADER.

portion of the untrimmed lumber, defects in the finished lumber and defect areas mapped on the surfaces may need to be recalculated for those defects cut through by the trimming lines. The procedure determining the defects in the lumber and defect areas on the surfaces involves the following steps.

Step 1:   Among the defect rectangles within the edging lines, determine the rectangles within the trimming lines, including those rectangles cut through by the trimming lines.

Step 2:   For the rectangles within the trimming lines but not intersecting with them, do nothing. Otherwise, modify the defect polygon associated with the rectangle, and then find the new encasing rectangle. For example, a defect area on the lumber in Figure 6.26 is cut through by a trimming line. The defect polygon needs to be modified accordingly, and the size of its new encasing rectangle is therefore smaller.
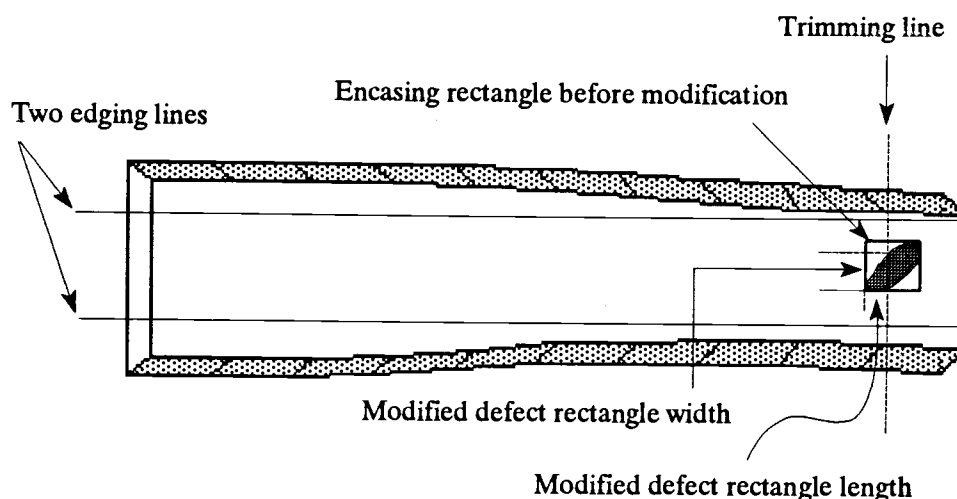
Figure 6.26. Modification of a defect area by a trimming line.

6.5.3.4 Defect Measurement

As mentioned before, all defects may be measured by dimensions parallel to edges of the finished lumber. This is why rectangles encasing defect polygons are used. In the trimming optimization component, after the two trimming saw placements are determined within the trimming DP algorithm, all the defects within the finished lumber and all the defect rectangles on the surfaces of the lumber are known. Therefore, determining defect sizes for lumber grading is straightforward as

a defect may be measured by width and length of the defect rectangle, or in some situations by thickness, i.e., depth of the defect in the lumber, as shown in Figure 6.27.
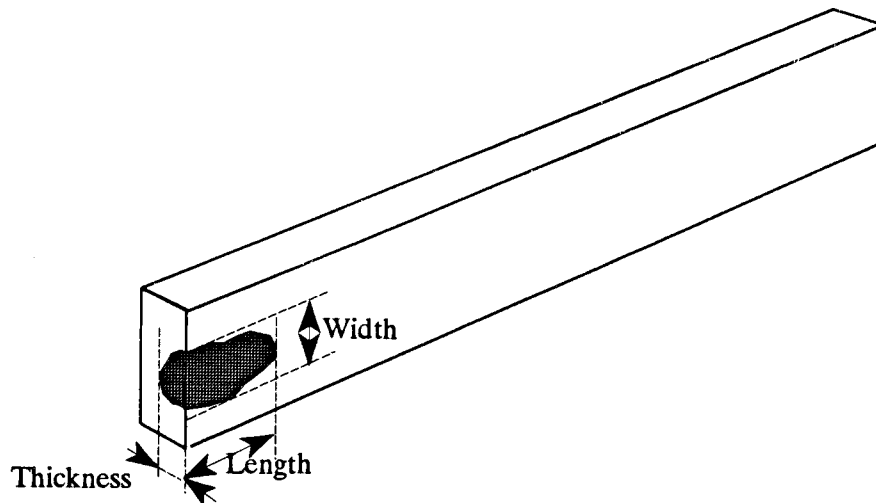
Figure 6.27. Defect measurement.

6.5.3.4.1 Defect Dimensions

Different grading categories may use different measurement approaches for the same type of defect. For example, Dimension grades need the thickness and width of wane while Board grades need the length in addition to the width and thickness of the wane. The various measurements used for defect types and grading categories are shown in Table 6.7.

The measurement process involves the following two basic steps:

1.   Determine the grading category according to the lumber sizes. Lumber one inch in thickness is in the Commons category and two inches in thickness is

Table 6.7. Measurement of Defects.

Defects	Measurements	
	Board Grades	Dimension Grades
Checks	W, L	L
Splits	L	L
Wane	T, W, L	T, W
Knots	D (W)	D (W)
Pockets	W, L	No limit
Decay(unsound wood)	*	T, W
Pith	W, L	*
Stain	W, L	No limit
Pitch	W, L	No limit

Where:

T -   thickness of the encasing defect rectangle, which is along the X direction of the coordinate system.

W -   width of the encasing defect rectangle, which is along the Y direction in the coordinate system.

L -   length of the encasing defect rectangle, which is along the Z direction of the coordinate system.

D -   diameter of knots, which is actually described by the width of the encasing rectangle.

* -   not considered in the current version of SAW3DG.

in the Dimensions category. Two-inch thick lumber with four inches of width belongs to the Light-framing category while that with a width larger than four inches belongs to the Joist and Planks category.

2. Determine the dimensions of each defect within and on the surfaces of the lumber, which are used to grade the lumber. If the results are larger than that of the previously calculated defect sizes of the same type, the previous results are replaced by the current results. The process is repeated until all the° defects are calculated and compared.

As mentioned above, measurement of each defect size is straightforward, since three dimensions, i.e., thickness, width, and length, have already been known for all defects but wane. However, knots and wane need more calculations. For knots, in addition to their dimensions, their locations are also important in determining lumber grade. For wane, previous stages record only the full wane at each cross section instead of a rectangle. Wane on the finished green lumber is calculated only after trimming saw placements are determined. Size measurements of knots and wane are described in greater detail in the following sections.

6.5.3.4.2 Measurement of Knots

When measuring a knot, in addition to width and length of its encasing rectangle as described previously, there are three more things that need to be considered. First, the knot mapped on the opposite side of the face also needs to be measured in order to calculate diameter of the knot. Second, the location of the knot needs to be determined because a knot is classified by its location as an edge knot or center knot. Third, the ratio of its width to its length needs to be known because the knot is classified by the ratio as a spike knot or regular knot.

Unlike measurement of most other defects, when calculating the diameter of a knot on one face, the diameter of the same knot exposed on the opposite face of the piece is also used. Knots are measured by an average dimension as in a line across

the width of the piece, as shown in Figure 6.28, where A is the diameter of the knot used for lumber grading.

The location of a knot is determined by the distance of its center to an edge. For knots on the wide faces, if the distance of the center of a knot to an edge is less than two thirds of the knot diameter, i.e., two thirds of the width of the encasing
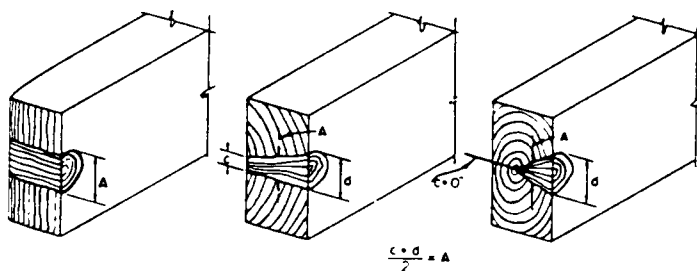
Figure 6.28. Measurement of regular knots (Source: Western Lumber Grading Rules 88).

rectangle, the knot is a center knot. Otherwise, it is an edge knot. Figure 6.29 demonstrates this. The final knot type entered into SLGRADER is determined by the knot type and its location. For example, if a knot is a loose knot as described by the scanning input data and it is an edge knot as defined above, then it is entered into SLGRADER as a loose edge knot. For knots on the narrow faces, they are always defined as edge knots.

A spike knot is determined by its ratio of its width to its length. If the width is larger than five times the length, the knot is a spike knot. Otherwise, it is a regular knot. For a spike knot, if the knot goes through both faces, i.e., the diameters of the

knot on both faces are not zero, its diameter used for grading will be the thickness of the piece as shown in Figure 6.30. If the diameter of the knot on the opposite face is zero, i.e., the knot is not exposed on the opposite face, the thickness of the knot as shown in Figure 6.31 will be used as the diameter of the knot for grading.
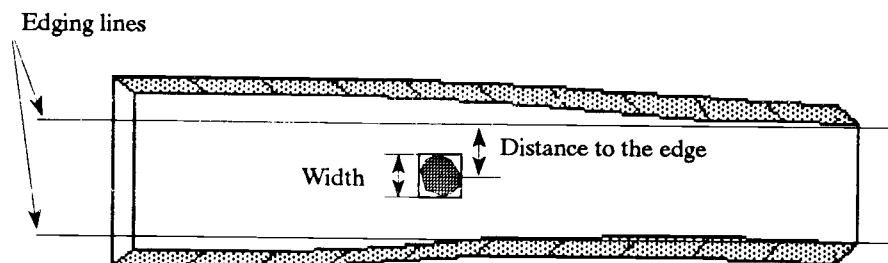
Figure 6.29 Determining knot location.
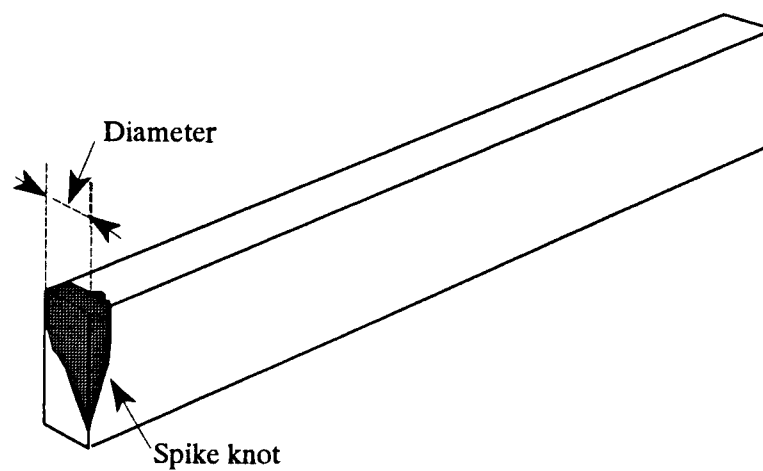
Figure 6.30. Spike knot exposed on both faces.
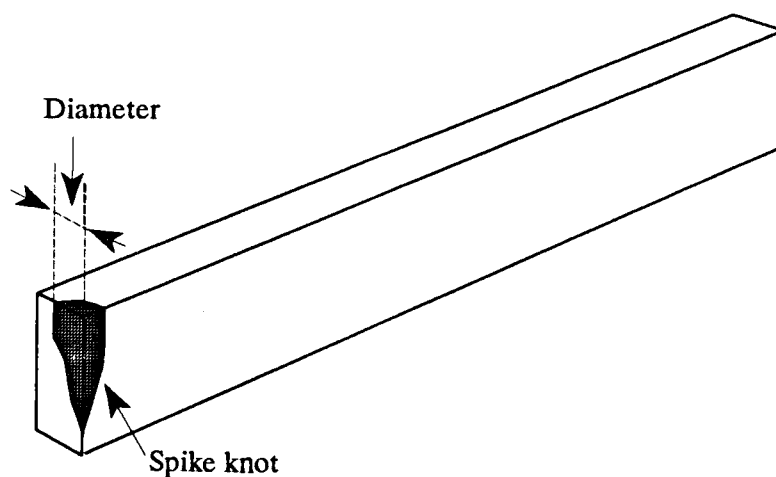
Figure 6.31. Spike knot exposed only on one face.

6.5.3.4.3 Measurement of Wane

As mentioned before, when merging two faces together at the headrig, Y coordinates of each face at every cross section are recorded so that wane can be calculated later. At the edger, if an edging line falls into a wane zone, a flag is set by the edger. In the trimmer, after the trimming saw placements are determined, the thickness, width, and length of the wane can be determined according to the edging and trimming saw placements.

Wane dimensions are determined by a calculation performed at each cross section and comparison among all related cross sections as described in the following steps.

Step 1:   Start from the first cross section.

Step 2:   If the upper edging line falls into a wane zone, i.e., its position is higher than the minimum Y coordinate of the two upper Y coordinates, determine width and thickness of wane on the upper and lower edges of the lumber, as shown in Figure 6.32.

Step 3:   If the upper edging line is not in the wane zone, find the intersecting point of the edging line and the flitch edge, add the distance between this intersecting point and the position of the last cross section to the previous wane length, and then go to Step 5.

Step 4:   If the position of the cross section is beyond a trimming line, add the distance between the trimming line and the previous cross section to the wane length. Otherwise, add the distance between this cross section and the previous cross section to the length of the wane.

Step 5:   Repeat all the steps for the lower edging line and lower edge of the piece. At each cross section, compare the thicknesses and lengths of the wane found previously and save the maximum ones.

Step 6:   At each cross section, compare the sum of the upper wane width and lower wane width with that of the previous cross section and save the maximum one.
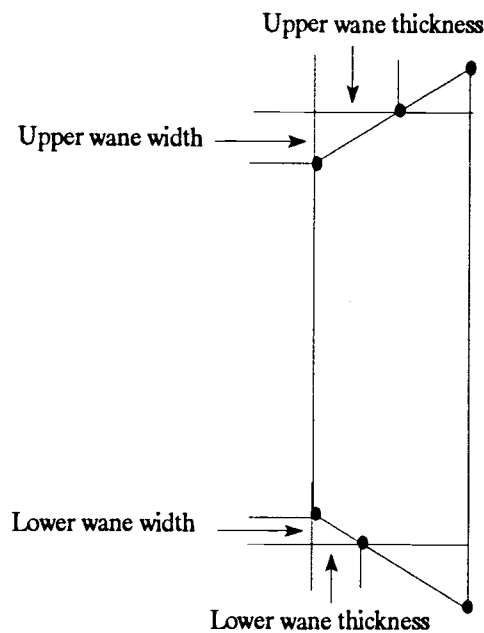
Figure 6.32. Determining the thickness and width of wane.

Step 7:   Repeat Steps 2 through 6 until all cross sections within the trimming lines are tried.

Figure 6.33 shows an example of wane dimensions on finished lumber. For this lumber, the thickness of the wane is determined by the maximum thickness among all upper and lower wane thicknesses; the width of the wane is determined by the maximum sum of the upper and lower wane widths on all cross sections that have wane within the edging lines; and the length of the wane is determined by the lower wane length since it is larger than the upper wane length.
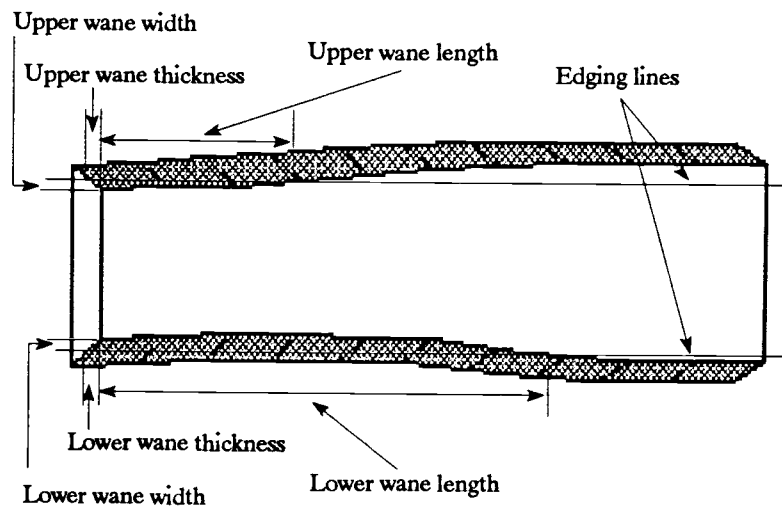
Figure 6.33. Wane dimensions on the finished lumber.

6.5.3.4.4 Measurement of Other Defects

Measurements of other defects are easier since they mainly involve only the measurement of thickness, width or length as described in Figure 6.27. The width and length of a defect are given by the dimensions of the encasing rectangle exposed on the face. Its thickness equals the thickness of the lumber if the defect is exposed

on both faces, similar to that shown in Figure 6.30. If the defect is exposed only on one face, its thickness is the depth of the defect inside the lumber.

6.5.4 Lumber Grading

Lumber grading is done in two steps. First, lumber grading is performed on each of the four surfaces of the finished lumber. SLGRADER receives data on the grading category, lumber thickness, width, and length, and type and sizes of each defect on the surface, and then determines the lumber grade of the face. Second, the overall grade of the lumber is determined based on the grades of those individual surface grades and the grading category. For the Commons category, the overall grade of the lumber is the highest grade of the two wider surfaces. For the Dimension category, the overall grade is the lowest grade of the four surfaces.

6.6 Software Implementation

Similar to SAW3D, SAW3DG is implemented using the C computer language and runs on IBM PC compatible computers. Execution time and memory consumption are two major concerns.

6.6.1 Application Platform

On a 486-66 MHz IBM PC machine, SAW3D's execution time for breakdown of a log ranges from several seconds to several hours or even days, depending upon log sizes, production parameters and optimization levels. Considering operations on log internal defects and the lumber grading process, SAW3DG could run much slower than SAW3D. Though the system is not intended to be used in real time, hours and

days to run a log are still too long for an off-line analytical tool. Also, considering all defects and the embedded SLGRADER, SAW3DG could take much more memory than SAW3D. The memory required by SAW3DG is expected to exceed the 640k conventional memory allowed by the DOS system. Considering those factors, the Windows-NT operating system (Microsoft, 1994) was selected as the development platform. There were several reasons to do so. First, a 32-bit version of SAW3DG for Windows-NT is about twice as fast as a 16-bit version. In addition, the Windows-NT version of Eclipse, which was used to develop the lumber grading expert system and embedded into SAW3DG, is also about twice as fast as its DOS or Windows version. Second, the Windows-NT system allows an application to take as much memory as is physically available, without the 640k limitation. Third, a 32-bit application developed for Windows-NT can run under the 16-bit Windows 3.1 system while enjoying the doubled execution speed, provided Win32s is installed on the 16-bit machine.

6.6.2 Structure of the Program

The construction of the software follows the procedures described in previous sections. The program consists of functional modules. The basic structure of SAW3D was retained while modules for defect manipulations were added into SAW3DG. In addition, minor changes were made so that the logic of the program was more readable. For details about the SAW3D structure, please refer to Zeng (1991). Figures 6.34 to 6.36 show the hierarchy charts of SAW3DG, which execute from top to bottom and left to right. Modules in the charts are conceptual and functional rather than real program modules, since the real modules may not be that clear because of coding optimization considerations.
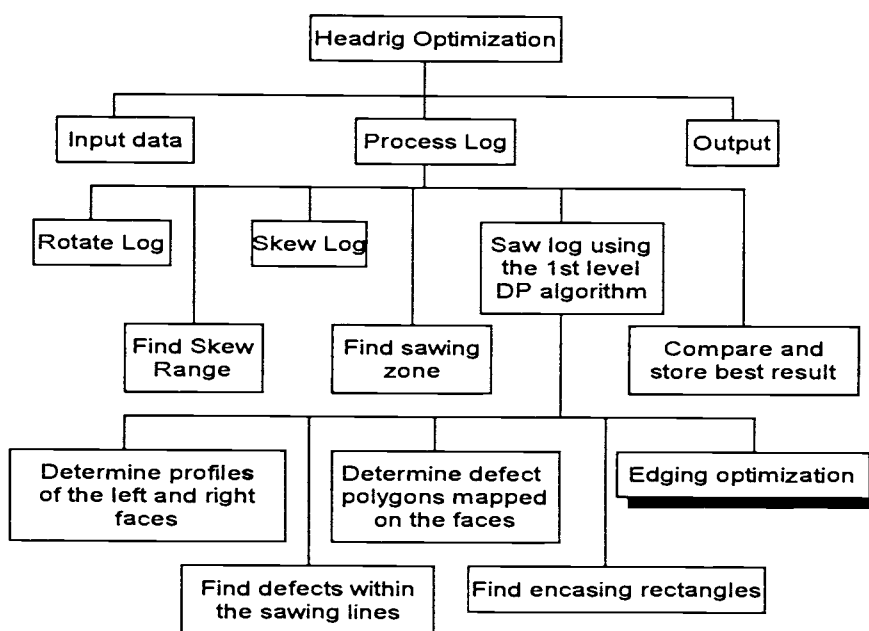
Figure 6.34. Hierarchy chart of the headrig optimization component.
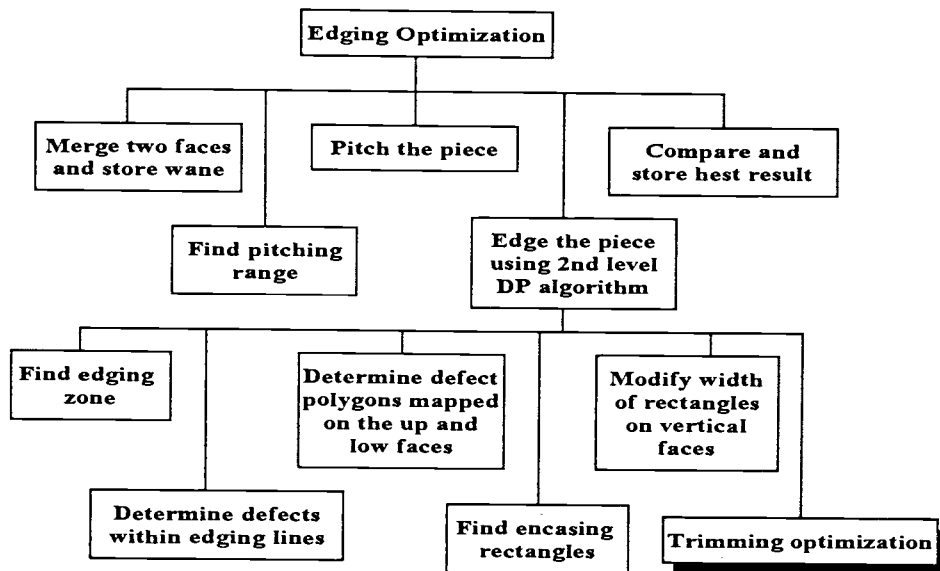
Figure 6.35. Hierarchy chart of the edging optimization component.
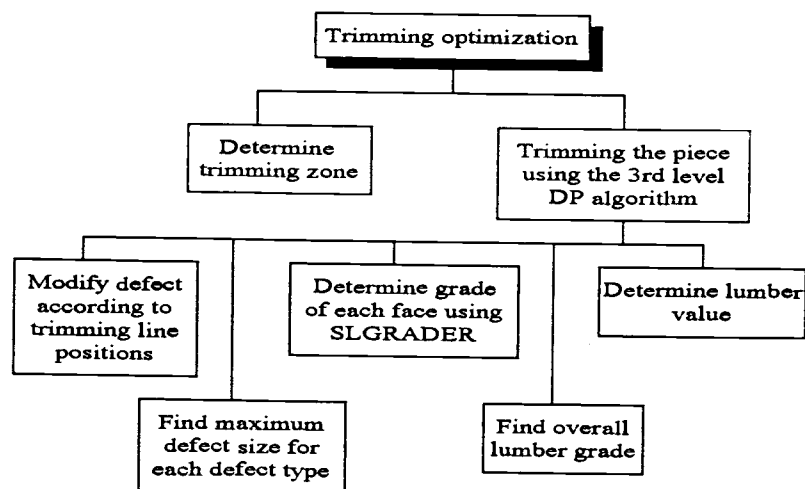
Figure 6.36. Hierarchy chart of the trimming optimization component.

6.6.3 Embedding The Expert System

The softwood lumber grading expert system, SLGRADER, was developed using the Windows-NT version of Eclipse.   As mentioned before, Eclipse is a commercial rule-based programming tool from The Haley Enterprise (The Haley Enterprise, 1991).   Eclipse itself was implemented using the C language and has excellent integration capability. Eclipse toolkits are provided with run-time libraries and source code necessary to construct applications which embed Eclipse.

SLGRADER is a stand-along expert system with its own user interface. When used with SAW3DG, only its core components, including its knowledge base containing the selected grading rules and the run-time portion of Eclipse, need to be embedded. Embedding SLGRADER into SAW3DG involved the following steps.

Step 1:   Develop the knowledge base containing selected grading rules by using the stand-along version of SLGRADER. Grading rules in each grading category are contained in individual files, such as commons.clp, lframe.clp, and jp.clp (Zeng, 1993), which are in the Eclipse source file format. Those files are loaded into SLGRADER and then saved as a single binary file containing all those categories, for instance, a file named "lfcomjk.bin".

Step 2:   Initialize Eclipse by calling init_eclipse(), a runtime routine of Eclipse. This is done at the beginning of SAW3DG in file saw3dgnt.c.

Step 3:   Load the knowledge base in the binary file created in Step 1 by calling bload(), a runtime routine of Eclipse. This is also done at the beginning of SAW3DG in file saw3dgnt.c.

Step 4:   Once lumber sizes and types as well as sizes and locations of all defects are known at the end of trimming process, a set of Eclipse's runtime routines, such as find_relation(), assert_begin_relation(), add_symbol(), assert_element(), and assert_end(), etc., are called to assert those facts into the working memory so that the SLGRADER can perform grading task according to the facts.

Step 5:   After that, another set of Eclipse's runtime routines are called to obtain the grade of the lumber based on the asserted facts.

All of Eclipse's runtime routines, except init_eclipse() and bload(), are included in one source file named astfact.c.

6.6.4 Input and Output Files of SAW3DG

SAW3DG starts running by looking up an initialization file named "saw3dg.ini". Names of all input and output files along with number of runs are specified in this initialization file. Figure 6.37 shows a portion of an initialization file. In the file, each section starts with a line in [] symbols. The first two sections

```
[StartRun]
StartAt  = 0
[EndRun]
EndAt    = 2

[Run0]
input    =      inputl.in
log      =      e22new2.log
defect   =      knots.in
textout  =      e22n2klr.out
grfout   =      e22n2klr.grf
[Run1]
input    =      inputl.in
log      =      h22new2.log
defect   =      knots.in
textout  =      h22n2klr.out
grfout   =      h22n2klr.grf
[Run2]
input    =      inputl.in
log      =      s22new2.log
defect   =      knots.in
textout  =      s22n2klr.out
grfout   =      s22n2klr.grf
[Run3]
input    =      inputl.in
log      =      e22new2.log
defect   =      knots.in
textout  =      e22n2wl.out
grfout   =      e22n2wl.grf
  .        .      .
  .        .      .
  .        .      .
```

Figure 6.37. Format of the initialization file.

tell the section numbers that SAW3DG should run through. For example, in this file
SAW3DG should run from sections Run0 to Run2. Each Run# section provides the
names of all input and output files that this section should use. There are five files
used in each section, including input, log scanning data, defect scanning data, text
output, and graphic output files. For instance, section Run0 uses inputl.in as an input

file, e22new2.log as a log scanning data file, knots.in as a defect scanning data file, e22n2klr.out as a text output file, and e22n2klr.grf as a graphic output file. Each file is described below.

The input file requires the same data as SAW3D (Zeng, 1991), including log and piece positioning options, sawing method selection, production parameters, optimum solution control options, and lumber sizes, plus additional lumber price tables for the different lumber grades. While SAW3D has a graphic user interface to help the user edit the input file, no graphic user interface has been developed for the current version of SAW3DG.

Log scanning data file contains log scanning data. The data are supposed to be provided by a log profile scanning system or mathematically generated using a computer program utility. This file uses the same format as that used by SAW3D, as explained in Table 6.8.

Defect scanning data file format is basically the same as that of the log scanning data file, except that the header contains additional information, because logs and defects use the same polygonal cross-section model representations, . Similar to the scanning data for the log, the defect scanning data are supposed to be provided by a log internal defect scanning system or mathematically generated by a computer program utility. Table 6.9 describes the file format.

The text output file contains results of the breakdown of the log in ASCII file format, including optimum sawing, edging, and trimming patterns; optimum value yields; and optimum positions for the log and each flitch cut from the log plus the grade of each resulting piece of lumber. Appendix A gives the output from sawing an S-twisted shape log 20 inches in diameter and 10 feet long with four knots in the log.

The graphic output file contains data to be used for graphically drawing the log sawing, flitch and cant edging, and trimming patterns on the screen. A commercial graphic program, called AcroSpin (Parker, 1989), is used to draw the output. AcroSpin is a computer program which is able to rotate, translate (pan over), and scale (enlarge or shrink) 3-dimensional wire frame objects. It reads a description of

Table 6.8 File format of log scanning data.

X01	X02	X03		
$X_{1,1}$	......	$X_{1,j}$	......	$X_{1,N}$
$X_{2,1}$	......	$X_{2,j}$	......	$X_{2,N}$
......	......	......	......	......
$X_{i,1}$	......	$X_{i,j}$	......	$X_{i,N}$
......	......	......	......	......
$X_{M-1,1}$	......	$X_{M-1,j}$	......	$X_{M-1,N}$
$X_{M,1}$	......	$X_{M,j}$	......	$X_{M,N}$
$Y_{1,1}$	......	$Y_{1,j}$	......	$Y_{1,N}$
$Y_{2,1}$	......	$Y_{2,j}$	......	$Y_{2,N}$
......	......	......	......	......
$Y_{i,1}$	......	$Y_{i,j}$	......	$Y_{i,N}$
......	......	......	......	......
$Y_{M-1,1}$	......	$Y_{M-1,j}$	......	$Y_{M-1,N}$
$Y_{M,1}$	......	$Y_{M,j}$	......	$Y_{M,N}$

Where:

X01 = M, number of cross sections;

X02 = N, number of scanning points on each cross section;

X03 = interval between cross sections (inches);

$X_{i,j}$ = X coordinate of the jth scanning point on ith cross section;

$Y_{i,j}$ = Y coordinate of the jth scanning point on ith cross section.

Table 6.9. File format of defect scanning data.

X00				
......	....	....	....	......
Z_k	T_k	$X_{k,0,1}$	$X_{k,0,2}$	$X_{k,0,3}$
$X_{k,1,1}$	....	$X_{k,1,j}$	......	$X_{k,1,N(k)}$
$X_{k,2,1}$	......	$X_{k,2,j}$	......	$X_{k,2,N(k)}$
......	......	......	......	......
$X_{k,i,1}$	......	$X_{k,i,j}$	......	$X_{k,i,N(k)}$
......	......	......	......	......
$X_{k,M(k),1}$	......	$X_{k,M(k),j}$	......	$X_{k,M(k),N(k)}$
$Y_{k,1,1}$	......	$Y_{k,1,j}$	......	$Y_{k,1,N(k)}$
$Y_{k,2,1}$	......	$Y_{k,2,j}$	......	$Y_{k,2,N(k)}$
......	......	......	......	......
$Y_{k,i,1}$	......	$Y_{k,i,j}$	......	$Y_{k,i,N(k)}$
......	......	......	......	......
$Y_{k,M(k),1}$	......	$Y_{k,M(k),j}$	......	$Y_{k,M(k),N(k)}$
......	......	......	......	......

Where:

X00 = number of defects in this file;

Z_k = Z coordinate of the first cross section of the kth defect;

T_k = a number indicating type of the kth defect;

$X_{k,0,1}$ = M(k), number of cross sections of the kth defect;

$X_{k,0,2}$ = N(k), number of scanning points on each cross section of the kth defect;

$X_{k,0,3}$ = interval between cross sections (inches) of the kth defect;

$X_{k,i,j}$ = X coordinate of the jth scanning point on ith cross section of the kth defect;

$Y_{k,i,j}$ = Y coordinate of the jth scanning point on ith cross section of the kth defect.

the wire frame object from an ASCII data file and then displays the object on the screen. Such operations as rotation and scaling are done interactively by the user once the object is on the screen. The graphic output file of SAW3DG contains profile coordinates of the log, its internal defects, and each flitch or cant cut from the log, as well as all sawing, edging, and trimming placements. A set of computer program utilities was developed to convert the graphic output file into AcroSpin files, and describe the attributes of the object and proper actions by using AcroSpin commands. The current version of SAW3DG has an integrated user interface that allows the user to show a log or a piece cut from the log with all the optimum patterns on the screen, without concern for how the process was done. The user only needs to select a graphic output file to use, and specify which object (the log or a certain piece between two sawing lines) should be displayed. The integrated environment will automatically covert the graphic file into an AcroSpin file and then invoke AcroSpin to show the object. Figures 6.38 to 6.40 show the results from sawing an S-twisted shaped log.
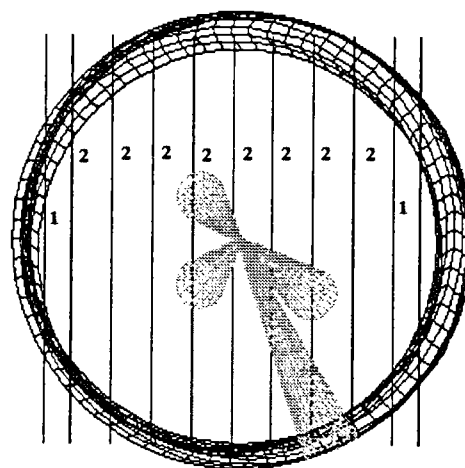
Figure 6.38. A plot of the optimum sawing sequence for an S-twisted shaped log 20 inches in diameter and 10 feet long. There are four knots in the log distributed along the longitudinal direction of the log.
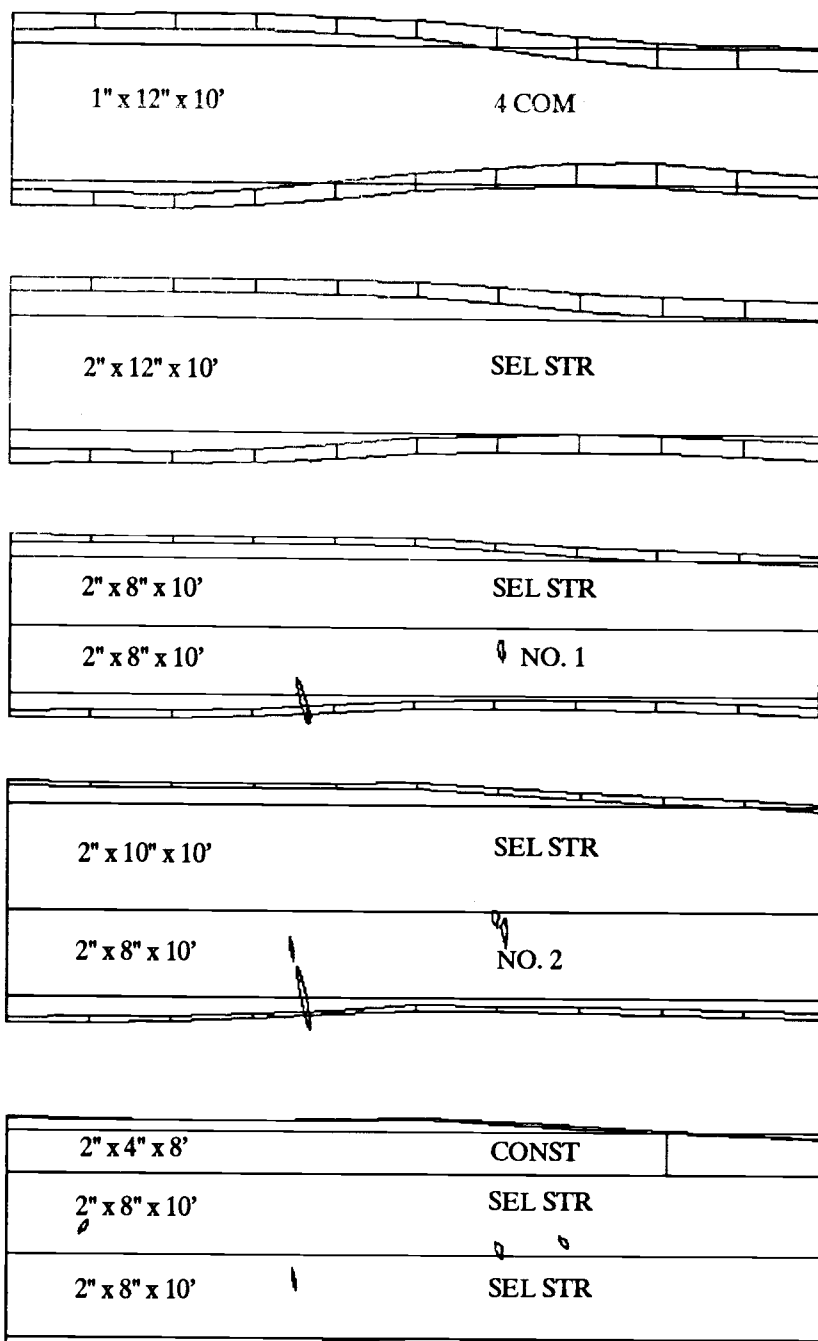
Figure 6.39. Plots of the optimum edging and trimming patterns for the five flitches sawn from the right portion of the log in Figure 6.38.
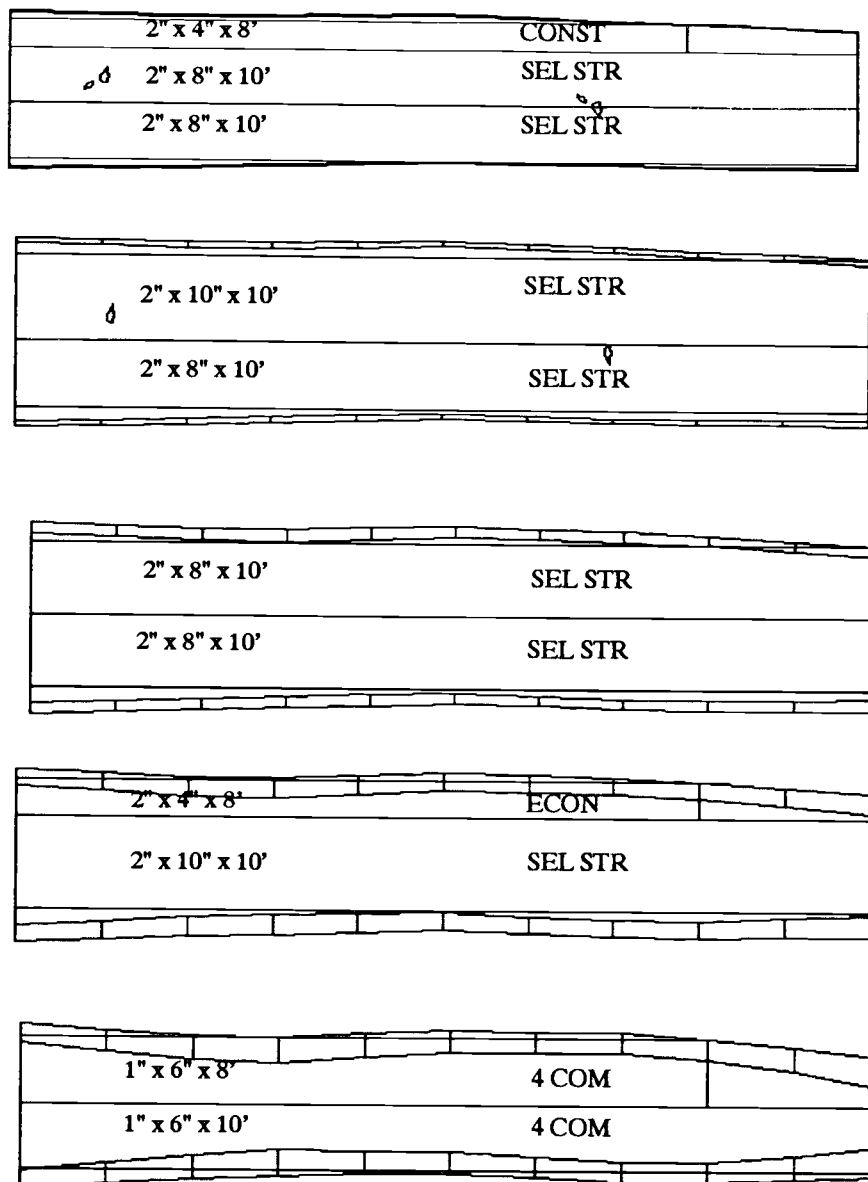
Figure 6.40. Plots of the optimum edging and trimming patterns for the five flitches sawn from the left portion of the log in Figure 6.38.

6.6.5 Input and Output Files of SAW3DGG

The input files for SAW3DGG are exactly the same as those used for SAW3DG. The output files include text and graphic files similar to those for SAW3DG. Since SAW3DGG breaks down the log around the log from different sides, all output information is organized according to its side index. Figure 6.41 shows the optimum sawing pattern resulting from breaking down a horn-down shaped log 20 inches in diameter and 10 feet long. Appendix B gives the text output.
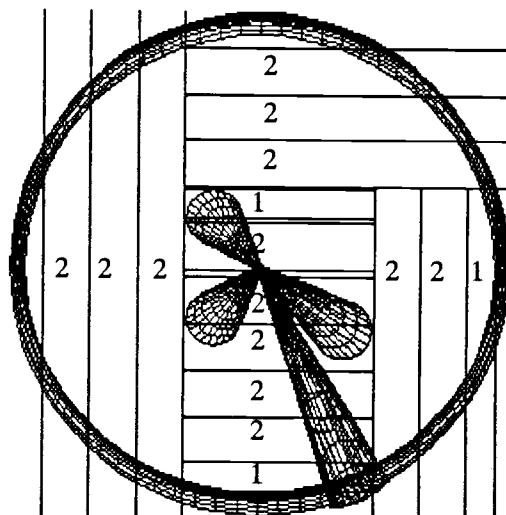
Figure 6.41. A plot of the optimum sawing pattern resulting from breaking down a horn-down shaped log 20 inches in diameter and 10 feet long using the around sawing.

CHAPTER 7 COMPUTATIONAL RESULTS

The system provides a research tool to study the effects of such factors as sawing methods, log shapes and internal defect distributions on lumber recovery. However, at present the focus of this research is to develop a prototype using the proposed methodology rather than conduct extensive studies on lumber recovery. Therefore, only a limited number of logs and internal defects were used to evaluate the system. The purpose of the system evaluation was to verify that the system works properly and provide guidelines for more extensive and topic-specific tests in the future.

The system assumes that the scanning data are in a series of cross sections providing 3-dimensional log shape and internal defect information as given by a scanning system. Lacking such scanning data at present, logs in various shapes with various defect types and distributions were mathematically generated to evaluate the programs.

7.1 Log Samples

Three log shapes were used: ellipsoid, horn-down shaped, and 3-dimensional S-twisted shaped logs. Figure 7.1 shows those log shapes along with internal knots within the logs that will be discussed later. Four logs of each shape were generated to provide combinations of log shapes and sizes as shown in Tables 7.1 to 7.3. To represent the majority of log shapes that can be expected to occur in practice rather than extremely irregular shapes, very small rates of taper, sweep, and crook were used. Also, using these log samples helped in the evaluation of the system because it was easier to see if the outputs were as expected compared to common sense and research results from other programs. All logs have 1 inch of taper per 8 feet of the length. Horn-down shaped logs have sweep in the center of their lengths, with the
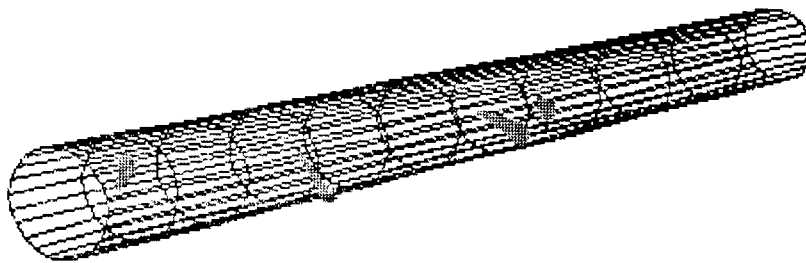
The Ellipsoid Log
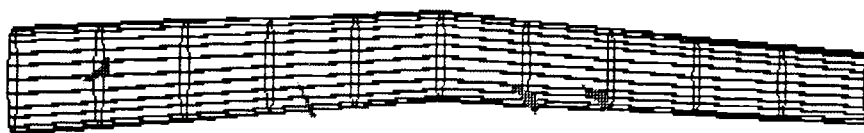
The Horn-down Shaped Log
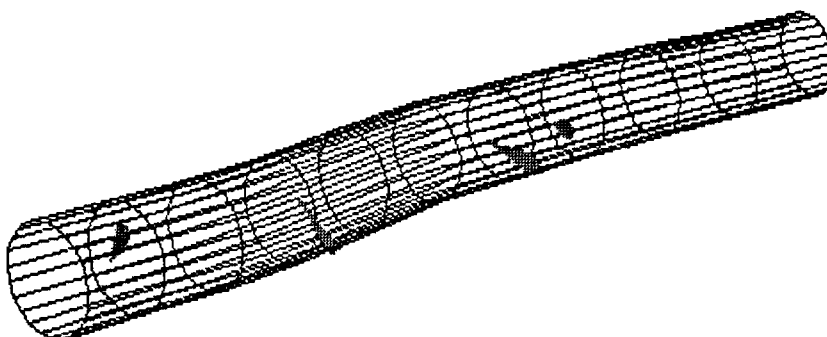
The S-twisted Shaped Log

Figure 7.1. Three log shapes used to evaluate the system.

Table 7.1. Parameters of S-twisted shaped logs.

	Log number			
	S11	S21	S12	S22
Small end diameter	10"	20"	10"	20"
Length	10'	10'	20'	20'
Taper	1"/8'	1"/8'	1"/8'	1"/8'
Scanning degree	5	5	5	5
Interval between cross sections	1'	1'	1'	1'
Crook	1"/8'	1"/8'	1"/8'	1"/8'

Table 7.2 Parameters of horn-down shaped logs.

	Log number			
	H11	H21	H12	H22
Small end diameter	10"	20"	10"	20"
Large end dia.	11.25"	21.25"	12.5"	22.5"
Length	10'	10'	20'	20'
Scanning degree	5	5	5	5
Interval between cross sections	1'	1'	1'	1'
Sweep position from the large end	5'	5'	10'	10'
Amount of sweep	0.625"	0.625"	1.25"	1.25"

Table 7.3 Parameters of ellipsoid shaped logs.

	Log number			
	E11	E21	E12	E22
Long small end dia.	10"	20"	10"	20"
Short small end dia.	9"	18"	9"	18"
Long large end dia.	11.25"	21.25"	12.5"	22.5"
Short large end dia.	10.25"	19.25"	11.5"	20.5"
Length	10"	10"	20"	20"
Scanning degree	5	5	5	5
Interval between cross sections	1'	1'	1'	1'
Crook	0	0	0	0

amount of sweep depending on the loglength. All S-twisted shaped logs have crook at a point 1/4 along the length and crook at a point 3/4 of the length but in the opposite direction to the first crook. Both locations of crook are not in the same plane so that an S-twisted shape is formed. The actual amount of crook depends on the point where the maximum amount of crook is present as well as the length of the log since a one inch offset from the center axis of the log per 8 feet of length was used to generate the crook.

7.2 Defect Samples

Defects in various types, sizes, and locations were mathmetically generated. Though the systems are able to consider many defect types, not every defect type was tested. There are several reasons for this. First, at present the development and evaluation are focused on the most frequently present defects that have the dominant

effect on lumber recovery, that is, knots. Second, some defect types are only considered by a few grading categories. For instance, pith, pitch, and stain are only considered for 1 inch thick boards. These defect types were not tested at this stage since they have no effect on 2-inch thick lumber. Third, some defect types may be much more difficult to detect than knots and research on detecting such defect types in logs have not been successful. Stain is such a type. Considering these factors, only various types of knots, checks, pockets, and decay were tested and the most effort was concentrated on the knots. Table 7.4 describes the sizes, locations, and shapes of four knots mathematically generated to evaluate SAW3DG, while Figure 7.2 shows an end-view of a S-twisted shaped log with these four knots. These knots are distributed along the longitudinal direction of each log sample as shown in Figure 7.1. Figures 7.3 to 7.5 show a check, a pocket, and decay used to test SAW3DG, respectively.

Table 7.4. Parameters of the knots shown in Figures 7.2 and 7.3.

Knot number	Knot 1	Knot 2	Knot 3	Knot 4
Knot type	Tight-black	Loose-black	Tight-red	Loose-black
X Angle	-30	-30	45	45
Y Angle	30	-30	-60	-30
Z Angle	15	15	45	45
Height	10"	80"	40"	70"
Maximum dia.	2.6"	2.6"	4.0"	2.5"
Minimum dia.	0.2"	0.2"	0.2"	0.2"
Length	4.0"	4.0"	6.0"	3.2"
Scanning degrees	15	15	15	15
Interval between cross sections	0.4"	0.4"	0.5"	0.4"

Where:

X Angle = Angle between knot axis and Y-Z plane of the coordinate system. Positive: knot axis rotates to the right side of the Y-Z plane. Negative: knot axis rotates to the left side of the Y-Z plane.

Y Angle = Angle between knot axis and X-Z plane of the coordinate system. Positive: knot axis rotates upwards from the X-Z plane. Negative: knot axis rotates downwards from the X-Z plane.

Z Angle = Angle between knot axis and X-Y plane of the coordinate system. Positive: knot axis rotates toward the small-end of the log. Negative: knot axis rotates toward the large-end of the log.

Height = Distance between the starting point of the knot and the large-end of the log.

Length = Distance between the first and the last cross sections of the knot.
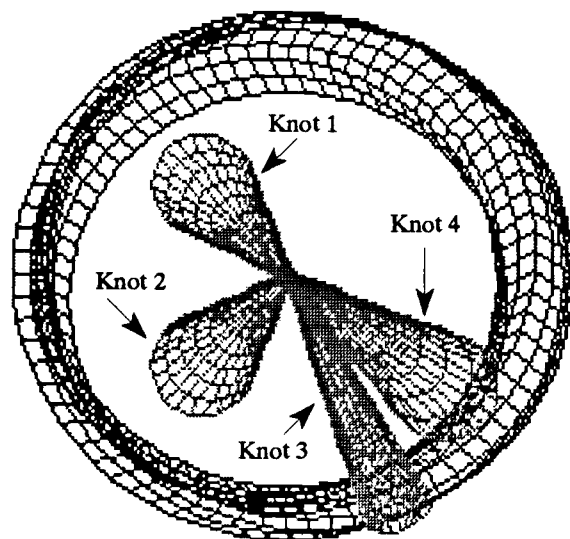
Figure 7.2 An end-view of an S-twisted shape log 10 inches in diameter and 10 feet long with four internal knots. Note that a long knot, knot 3, which represents a branch sticking out of the log, was used to see whether the system works properly in this situation.
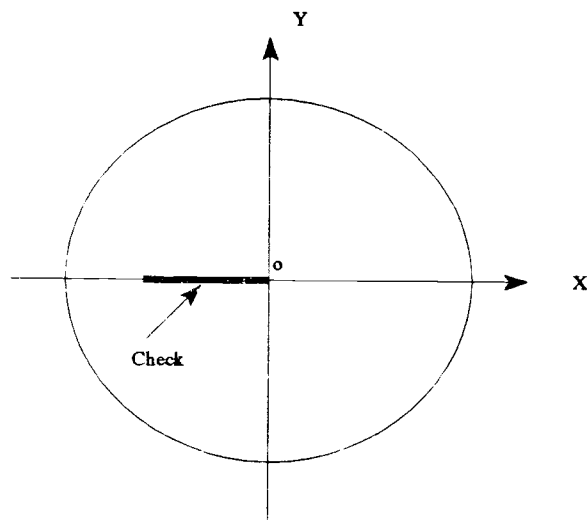
Figure 7.3. The check used to test the system. It is 0.08 inch thick (Y direction), 2.5 inches wide (X direction), and 2 feet long (Z direction) starting from the largest cross section of the log. There are 24 scanning points on each cross section (every 15 degrees), and the interval between cross sections is 0.5 inch.
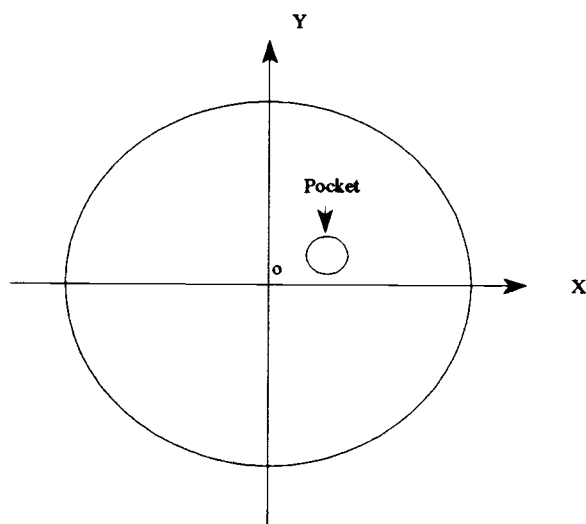
Figure 7.4. A pocket used to test the system. It has a cylinder shape 1 inch in diameter and 1.5 inches long. It starts 80 inches from the bottom of the log. The center of its cross sections offset 1.5 inches from the X axis and 1 inch from the Y axis of the coordinate system. There are 24 scanning points on each cross section (every 15 degrees), and the interval between cross sections is 0.5 inch.
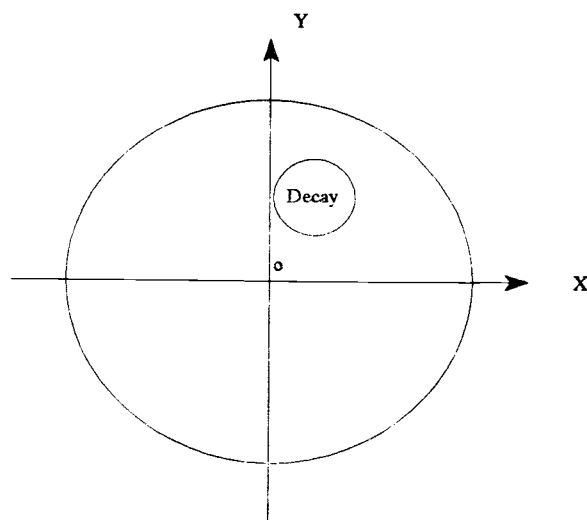
Figure 7.5. The decay used to test the system. It has a cylinder shape 2 inches in diameter and 1 inch long. It starts 50 inches from the bottom of the log. The center of its cross sections offset 1 inch from the X axis and 2 inches from the Y axis of the coordinate system. There are 24 scanning points on each cross section (every 15 degrees), and the interval between cross sections is 0.5 inch.

7.3 Production Parameters and Lumber Price

The system allows variable production parameters such as saw placement increments, saw kerfs and variations, rotation angle increment, skew position increment, and a mix of product sizes. Previous research concerning only 3-dimensional log shapes has already found that those parameters have significant effects on lumber recovery. It is expected that the same is true when log internal defects are considered. An exhaustive test concerning all possible combinations of those parameters, log shapes and defect distributions may be necessary to obtain solid conclusions in this regard. Unfortunately, at the current development stage it is impossible to do such an exhaustive test because of time limitations. Therefore, for preliminary system evaluation purposes, only representative parameters were chosen, as shown in Table 7.5.

Table 7.5. Production parameters used to test the system.

Parameters	Value
Saw placement increment (all saws)	1/8 inch
Saw kerf (all saws)	1/8 inch
Saw variation (all saws)	1/8 inch
Log rotation increment	5 degrees
Log skewing positions	minimum, middle, maximum
Flitch/cant pitching positions	minimum, middle, maximum
Nominal lumber size (inches)	1, 2, 4, 6, 8, 10, 12
Final green lumber size (inches)	1, 1.75, 3.75, 5.75, 7.625, 9.625, 11.625
Lumber length (feet)	8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30

Lumber prices were obtained from Crow's Market Report (May 13, 1994) for green Douglas-fir, in the Portland, Oregon area, which were based on a given grade range such as Standard & Better of the Framing category. To evaluate the effect of different grades and internal defects, certain dollar differences were assigned to different grade levels. For example, starting from a base grade, such as Standard for dimension lumber, 50 dollars per MBF were added to or deducted from each higher or lower grade level to project lumber prices in different grades.

7.4 Effects of Log Rotation

Log rotation has a significant impact on value yield. Figure 7.6 shows the results from sawing an S-twisted shape log 10 inches in diameter and 10 feet long while the log was rotated every 5 degrees in the range of 0 to 355 degrees. The
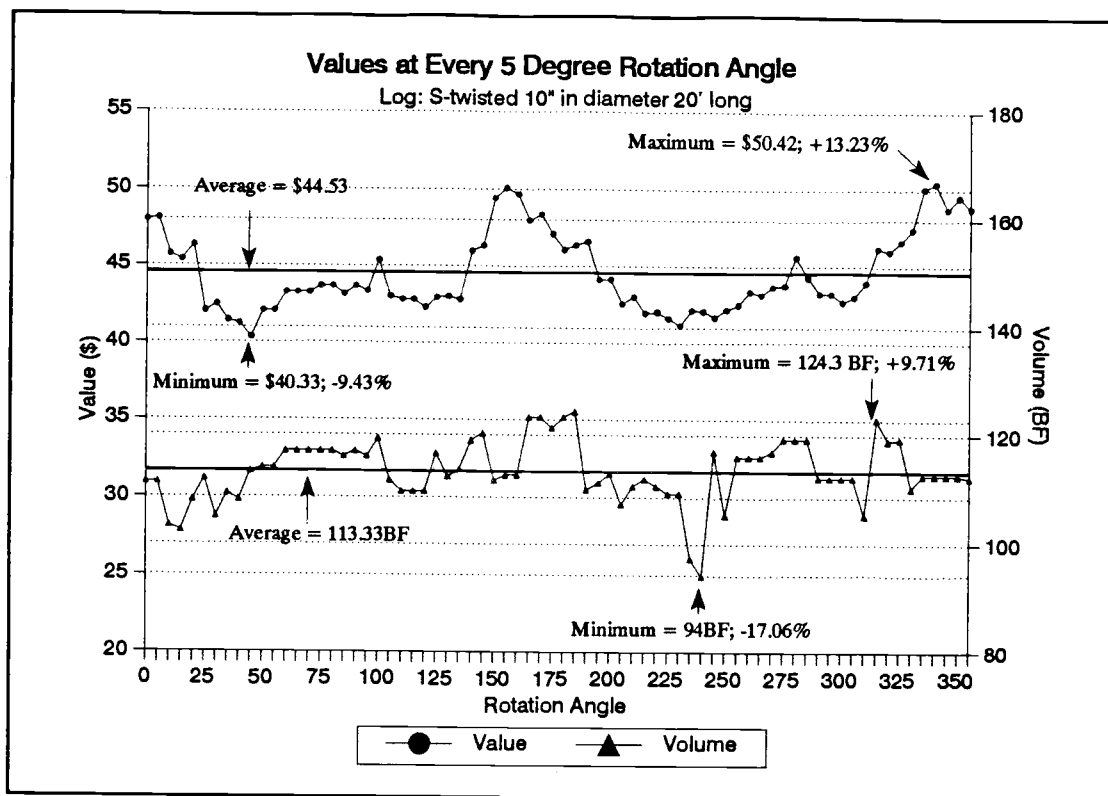
Figure 7.6. Value and volume effects of log rotation angle on an S-twisted shaped log 10 inches in diameter and 20 feet long containing four knots.

fluctuating curves indicate that value and volume recovery are significantly different at the various rotation positions. At the best rotation angle, the maximum value yield is 13.23% higher than the average value at all rotation positions; whereas at the worst rotation position, the minimum value yield is 9.43% lower than the average value. The results also show that high value yield does not mean the volume recovery is also high compared to other positions. There seems to be certain rotation ranges that are better than other ranges. For instance, in Figure 7.6 the average value yield in each of the rotation ranges from 0 to 20 degrees, 140 to 200 degrees, and 315 to 355 degrees is much higher than the total average value yield in the entire rotation range from 0 to 355 degrees.

Similar yield patterns hold true for the other logs. Figure 7.7 represents the value increases of the optimum value yields at the best rotation angles compared to the value yields without log rotation for all logs tested. Care must be taken when viewing this figure. Smaller logs showed higher percentage increases. Though large logs may have a small percentage increase, they may have a larger dollar increase. For instance, the horn-down shaped log 10 inches by 10 feet showed an increase of 8.3%, while the horn-down shaped log 20 inches by 20 feet only had a 1.7 percent gain. However, the 10-inch log's total revenue was only increased $1.72 while the 20-inch log's total revenue was increased $3.64. Also, Figure 7.7 shows only the value increases when log rotation is allowed versus values without log positioning operations. While the percentage gain may not be high when a log's original position is not bad, the gain at the optimum rotation angle may be much
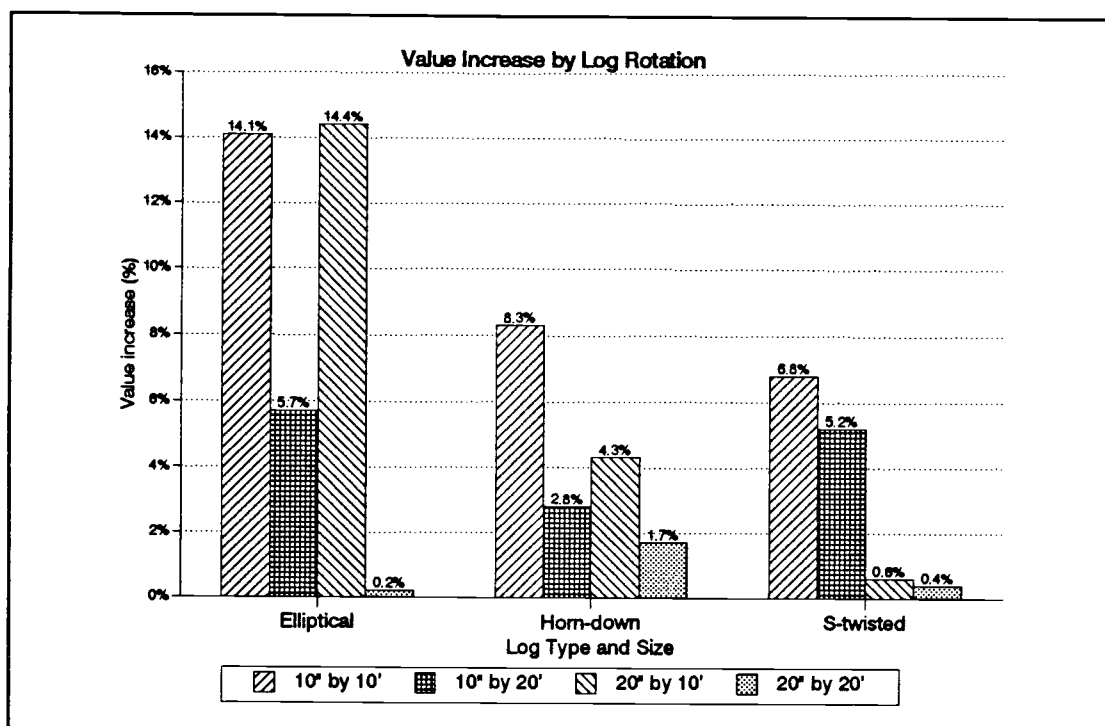
Figure 7.7. Value increases for each of the twelve log samples when log rotation is allowed versus values when no positioning operations are allowed .

higher than the average value at all rotation angles or the value at the worst position. For instance, compared to the value at the 0 rotation angle, that is, the log's original position without rotation, the S-twisted shaped log 10 inches by 20 feet showed a 5.2 percent increase in Figure 7.7. However, Figure 7.6 shows that when compared to the average value and the worst value, the same log at the optimum rotation angle had a 13.23% gain and a 25% gain, respectively.

7.5 Effects of Log Skew

Log skewing means orienting the log horizontally at the headrig. Figure 7.8 shows the effect of log skewing without any other positioning operations. The results
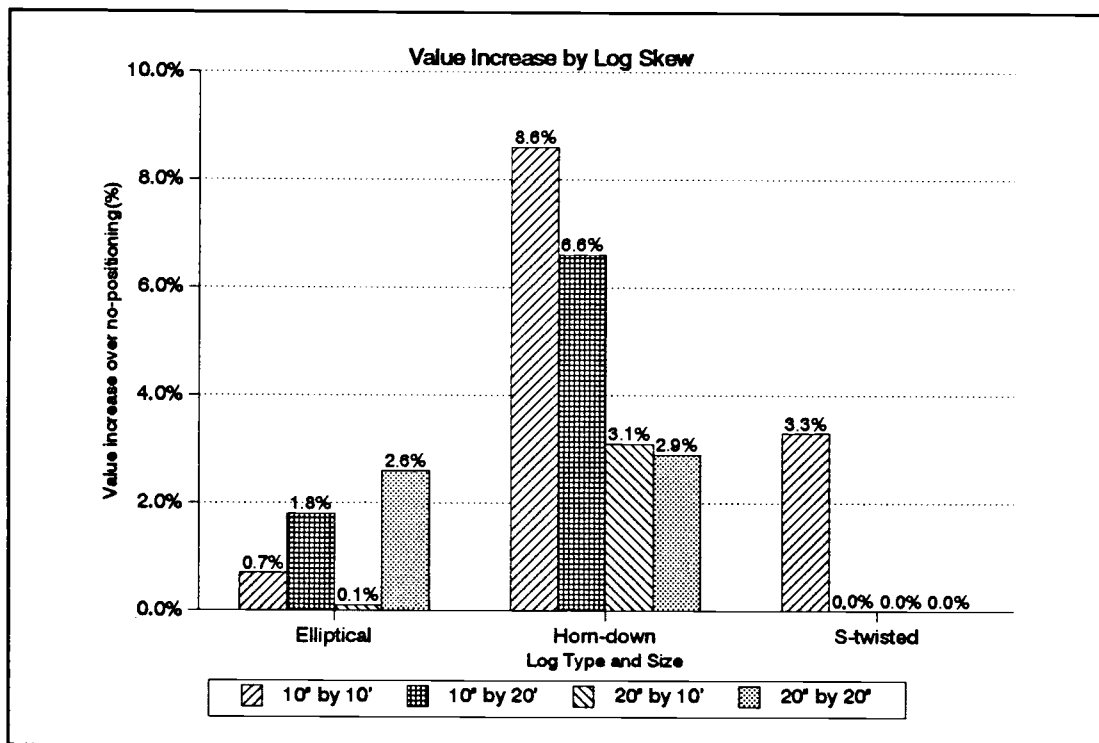
Figure 7.8. Value increases resulting from the log skewing operation only for each of the twelve log samples.

showed that the value increases were not as significant as that provided by log rotation. Some logs (S12, S21, S22) had the best value yields when the logs were not skewed at all. The results are not surprising given the logs have little taper, crook, or sweep.

7.6 Effects of Flitch/Cant Pitch

Each piece cut from the log was pitched to find the best position of the piece. This is equivalent to orienting a flitch at a board edger or a cant at a cant edger. Some logs gained significant value increases while others did not gain any increase at all (Figure 7.9). Again, the effect of piece pitch was not as significant as the log rotation. This can also be explained by the small amount of log irregularity.
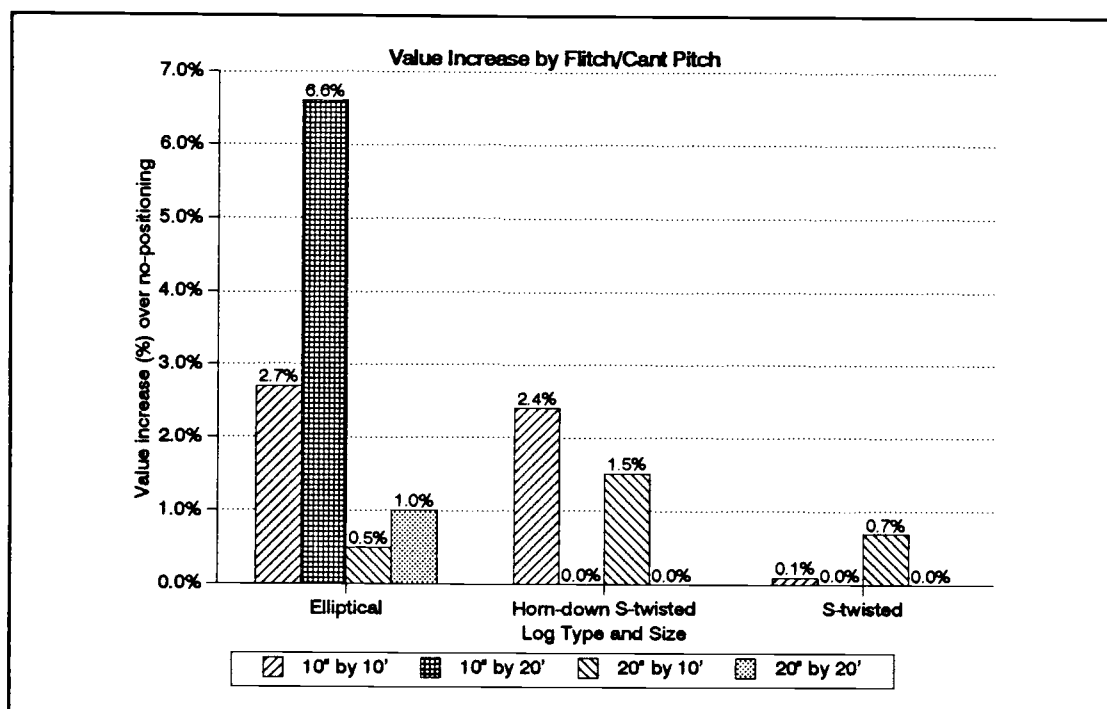
Figure 7.9. Value increases resulting from the flitch/cant orientation operation (pitching) only for each of the twelve log samples.

7.7 Effects of the Combination of Rotation, Skew, and Pitch

When all positioning operations are performed, that is, all possible combinations of rotation, skew and pitch positions are searched, there may be higher value gains. However, the computing time will be much longer. Figure 7.10 shows the results from sawing the three types of logs 10 inches by 10 feet while all positioning operations were applied. It shows that among all positioning operations, rotation usually resulted in the highest value return compared to log skewing and flitch/cant pitching if positioning operations were applied individually. When the combination of all possible positioning operations was performed, the result was higher than the rotation-only orientation.
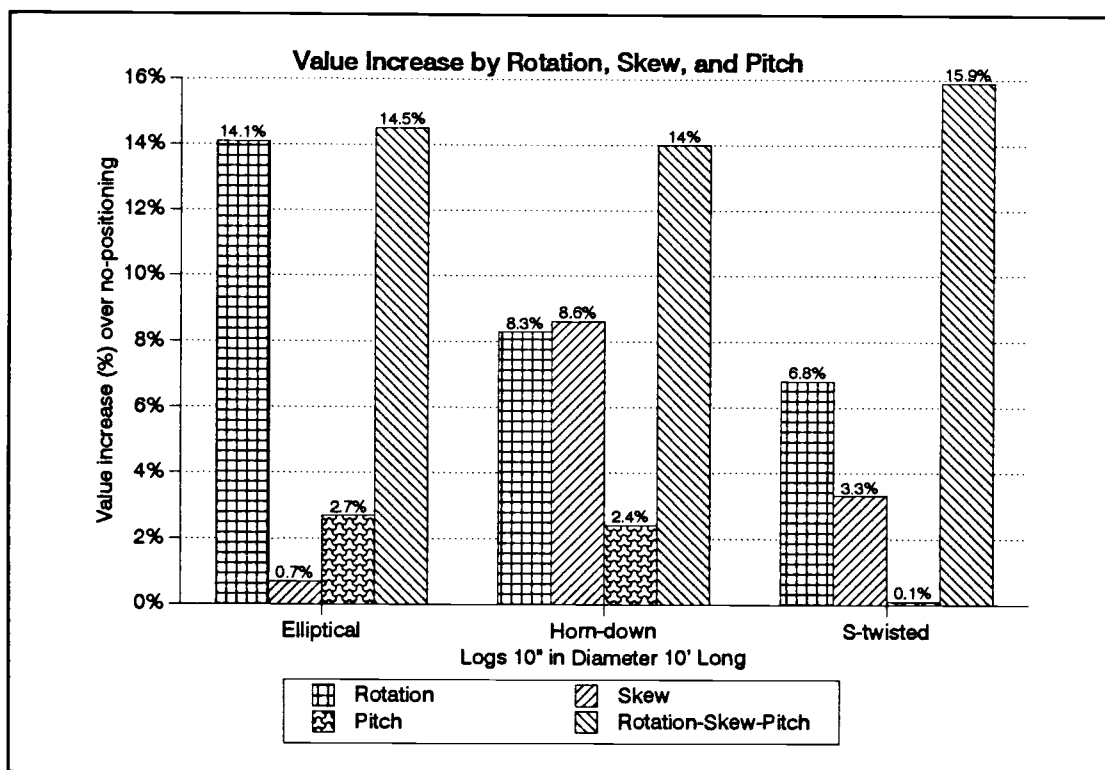
Figure 7.10. Value increases resulting from the combination of all positioning operations for the log samples 10 inches in diameter and 10 feet long.

7.8 Effects of True Log Shape

Compared with systems using simplified log shapes, the Best Opening Face System (BOF) for instance, SAW3DG should provide better solutions since it considers 3-D log shape and internal defects. Figure 7.11 compares SAW3DG vs BOF. Logs were 10 inches in diameter and 10 feet long in the three shapes. To compare SAW3DG with BOF, BOF was used to saw logs in the same sizes as the log samples. However, BOF assumes all logs have perfect truncated cone shapes and are defect free. The results were "called BOF projected values" in Figure 7.11 and consistently overestimate what can actually be recovered. Therefore, the sawing patterns that resulted from the BOF solutions were then used as input to SAW3DG in order to saw the twelve logs while using the true log shapes and defects. In this case,
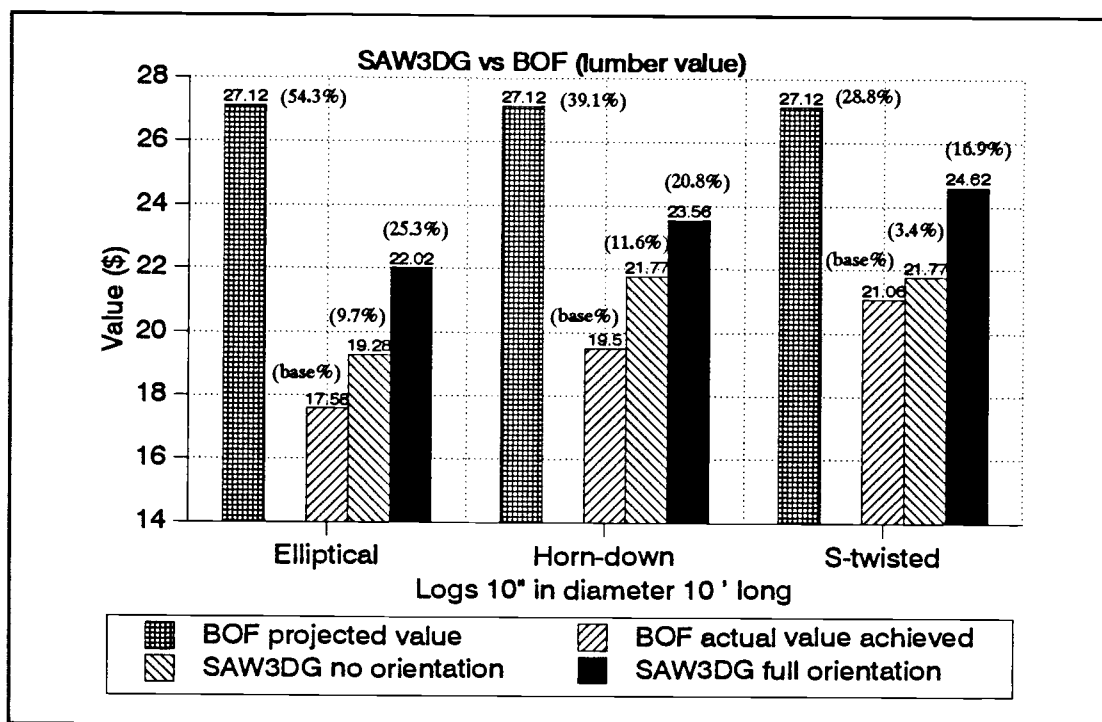
Figure 7.11. Effects of real shape and defects on lumber value comparing SAW3DG vs BOF.

SAW3DG did not use its own log breakdown DP algorithm, but instead used the sawing patterns generated by BOF. Other than that, SAW3DG still used its own edging and trimming algorithms. The results, called "BOF actual value achieved" in Figure 7.11, showed decreases of 54.3% for the ellipsoid log, 39.1% for the horn-down shaped log, and 28.8% for the S-twisted shaped log. After that, SAW3DG used its own log breakdown DP algorithm to saw the logs in the true shapes containing defects, but it did not perform any positioning operations. The value results, as indicated by "SAW3DG no orientation" in Figure 7.11, were higher than the BOF actual value achieved. When the combination of all positioning operations was considered by SAW3DG, the value results were even higher than the BOF actual values, as indicated by "SAW3DG full orientation" in Figure 7.11.

7.9 Effects of Log Internal Defects

All preceding discussions were based on logs with the four sample knots. Figure 7.12 shows value yields of logs in the same shapes and sizes but with different defects: 1) without any defects except wane; 2) with knots; and 3) with a check, decay and a pocket in addition to the four knots. As the wood quality of the logs decreased from containing no defects to all the defects, the value yields also decreased.

SAW3DG is superior to systems that use 3-dimensional log shape but ignore log internal defects. To compare SAW3DG with SAW3D, SAW3D was used to saw the logs using the true log shapes but ignoring any internal defects. The value results are indicated by "SAW3D projected values" in Figure 7.13. Then the sawing patterns generated by SAW3D were used as input by SAW3DG to saw the logs while considering internal defects. In this case, SAW3DG used sawing patterns provided by the SAW3D solutions instead of its own sawing DP algorithm. The value results, as indicated by "SAW3D actual values", decreased by 15.8% for the ellipsoid log, 20.7% for the horn-down shaped log, and 20.9% for the S-twisted shaped log,
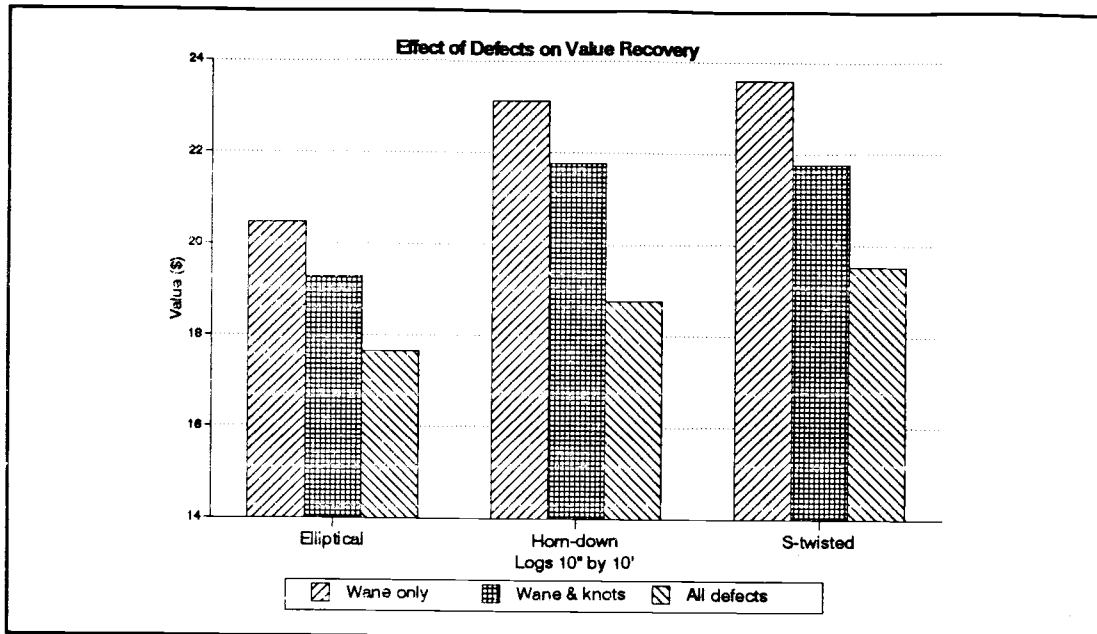
Figure 7.12. Effect of internal defects on value recovery.
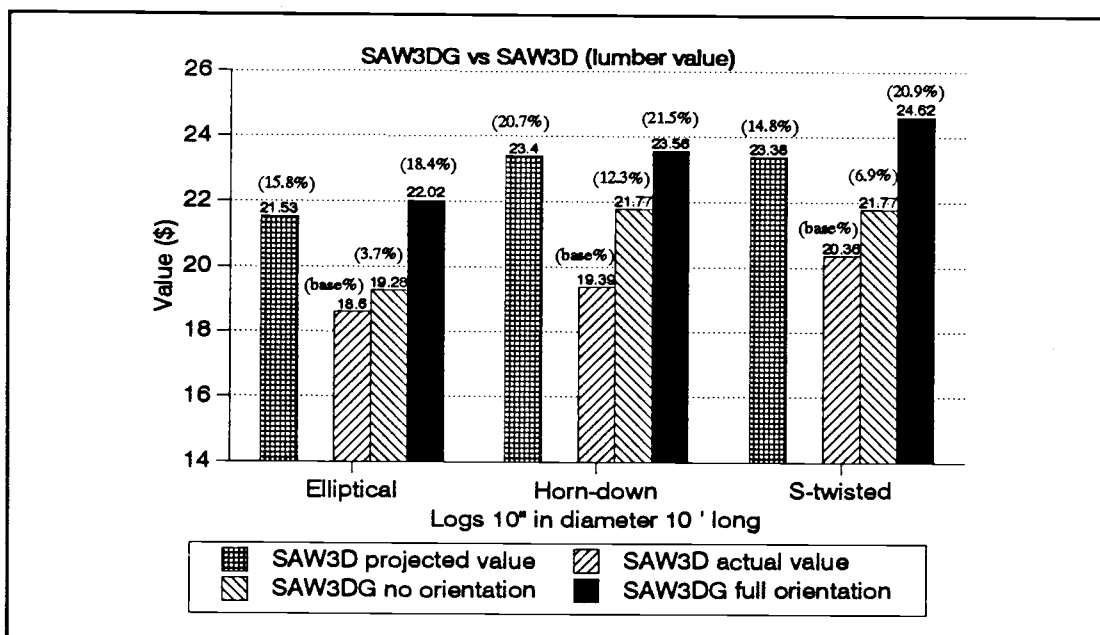
Figure 7.13. Effects of defects on lumber value comparing SAW3DG vs SAW3D.

respectively. When SAW3DG's sawing DP algorithm was used, the value results increased even without performing any positioning operations, as indicated by "SAW3DG no orientation" in Figure 7.13. When all positioning operations were used, value results were much higher than the SAW3D actual values, as indicated by "SAW3DG full orientation" in Figure 7.13.

7.10 Effects of Sawing Method

SAW3DG allows live, cant, and around sawing as described before. In addition to these three basic sawing methods, the system also allows multi-thickness sawing patterns. The multi-thickness sawing is similar to live sawing, with the exception of allowing any thickness in any sequence instead of requiring all 2-inch thick lumber stacked between jacket boards as is typically found in sawmills.

The comparison between different sawing methods indicated that generally live sawing resulted in a higher value yield than cant sawing. Among all the twelve sample logs with the four knots, the live sawing method had higher value yields for all but the logs 10 inch in diameter and 10 feet long in ellipsoid and S-twisted shapes.

Multi-thickness sawing, as a special case of live sawing, resulted in higher value recovery than any other sawing method. For instance, when the multi-thickness sawing method allowing one and two-inch thicknesses in any sequence was applied, higher value yields were obtained than for live sawing for all the sample logs (Figure 7.14). This is expected since the multi-thickness sawing allowed more thickness options while searching for the optimum solution.

Around sawing resulted in significant value yield decreases for all logs compared to the live sawing method (Figure 7.15). The around-sawing method focused on maximizing grade recovery while searching for the optimum solution, resulting in possibly higher grade pieces but lower total value recovery of the log. When sawing the log around, some portion of the log was wasted because the size of that portion was smaller than the thinnest acceptable board.
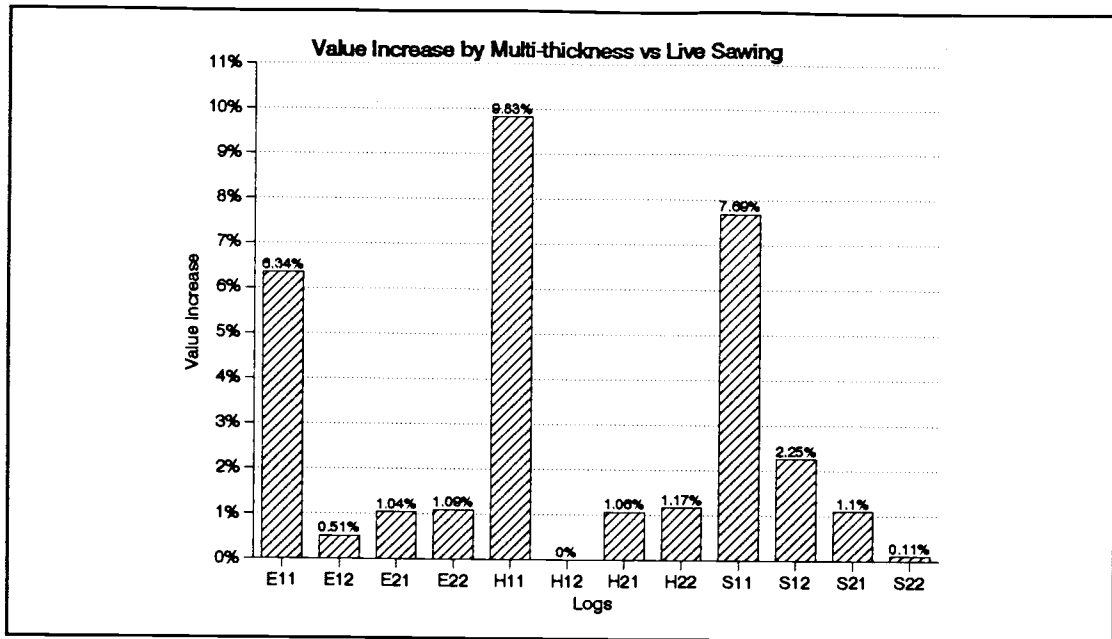
Figure 7.14. Value yield increase resulting from the multi-thickness sawing method allowing one and two-inch thick lumber in any sequence compared to live sawing.
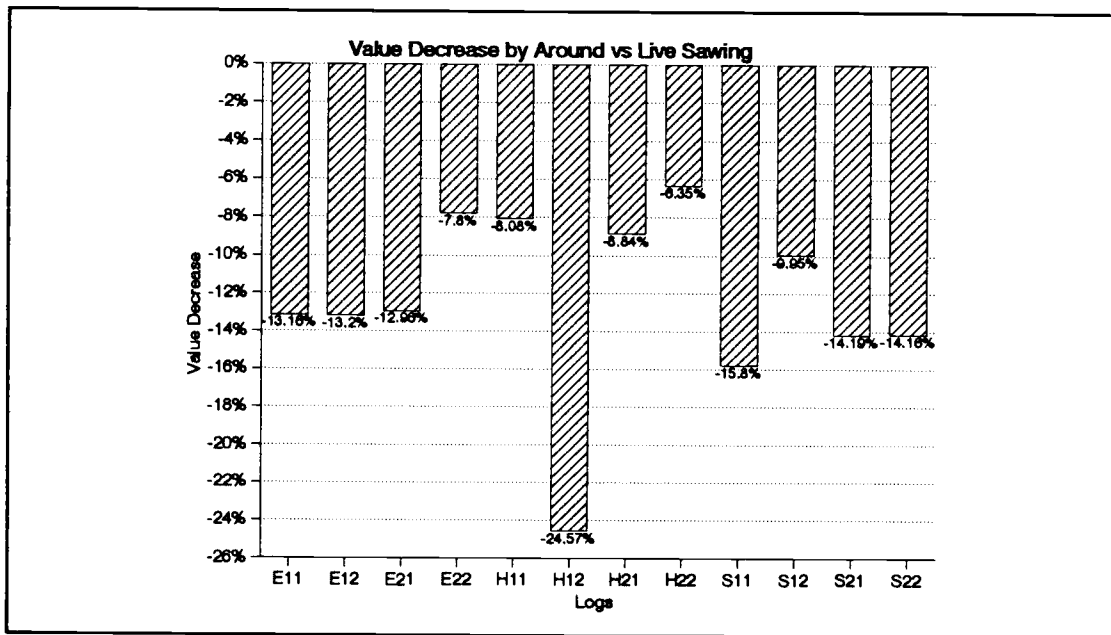
Figure 7.15. Value yield decrease by around sawing vs live sawing.

7.11 Value Yield vs Volume Recovery

All discussions so far are based on lumber value. Traditionally, lumber volume recovery was widely used to evaluate the efficiency of a sawmill. However, optimizing lumber volume recovery could be misleading when the major concern of a sawmill is to maximize profit, because the optimum volume recovery does not necessarily mean the optimum value. Take the results from breaking down an S-twisted shaped log 10 inches in diameter and 20 feet long as an example (Figure 7.6). Volume recoveries at the rotation angles from 60 to 100 degrees are larger than the average volume recovery, but values in the same rotation range are smaller than the average value. Similarly, values at the rotation angles from 330 to 355 degrees are larger than the average value, whereas volume recoveries in the same rotation range are smaller than the average volume recovery. The optimum value occurs at 340 degrees where the volume recovery is smaller than the average.

Obviously, the lower the lumber volume recovery is, the larger the chip volume will be. When considering maximizing profit, value yields from chips should also be taken into account. Assuming a chip price is of 100 dollars per ton (oven dry), the following conversions are used to estimate chip value (Nielson et al., 1985):

1 cubic feet of solid wood /SWE = volumetric unit of chips

1 volumetric unit = basic density * SWE / 2000 = oven dry tons

Where:

SWE = solid wood equivalent = 72 cubic feet / volumetric unit

basic density = 28.08 lb / cubic feet

Considering chip value, the total value of a log should be the sum of the values from lumber and chips, provided sawdust is ignored. Figure 7.16 shows the same information as in Figure 7.6, plus additional information on chip value and total value at each rotation angle. The total values exhibit the same trends as the lumber

values, indicating that chip value does not change the sawing solution and the dominate factor determining the optimum solution is the lumber value. Therefore, all conclusions based solely on lumber value still hold true when chip value is also considered.

Compared to BOF and SAW3D, SAW3DG may or may not result in higher lumber volume recovery, but always provides a higher value regardless of chip value. For instance, breaking down logs 10 inches in diameter and 10 feet long shows that BOF (using real shape and defects) resulted in smaller lumber values as discussed previously (Figure 7.11), but higher lumber volume recoveries (Figure 7.17). The total value of each log after considering chip value, however, is still larger when using SAW3DG (Figure 7.18). Similarly, lumber volume recovery may be higher when using SAW3D (horn-down shaped log in Figure 7.19), but SAW3DG always provided higher value regardless of chip value (Figures 7.13 and 7.20).
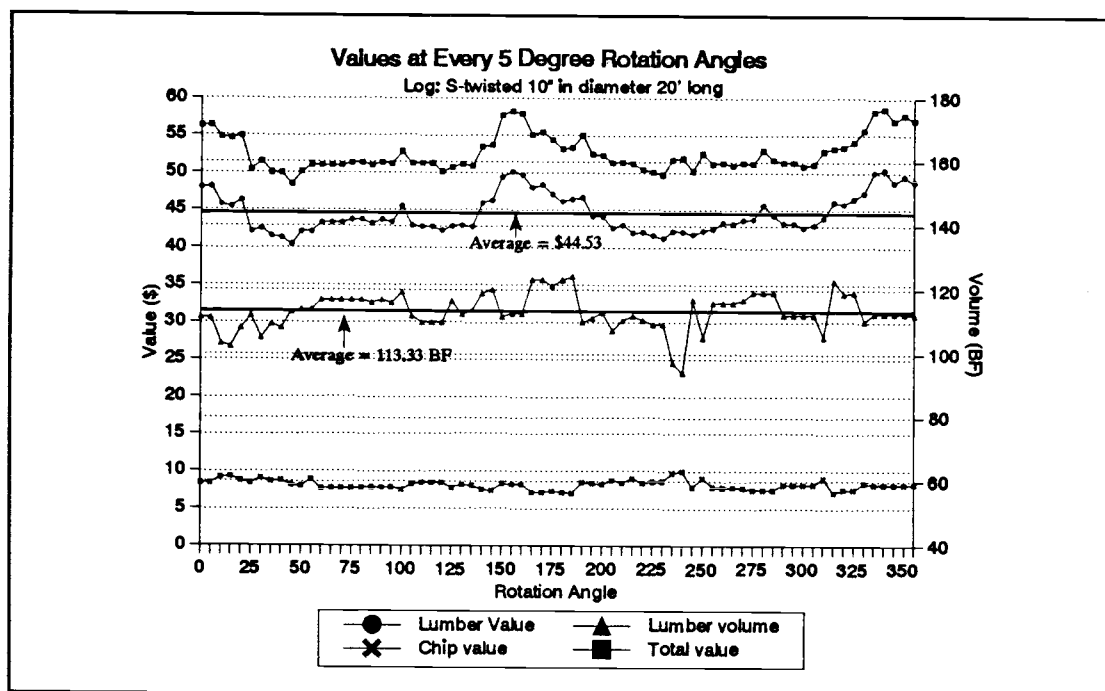
Figure 7.16. Chip value effects on total value yields when breaking down an S-twisted shaped log 10 inches in diameter and 20 feet long containing four knots.
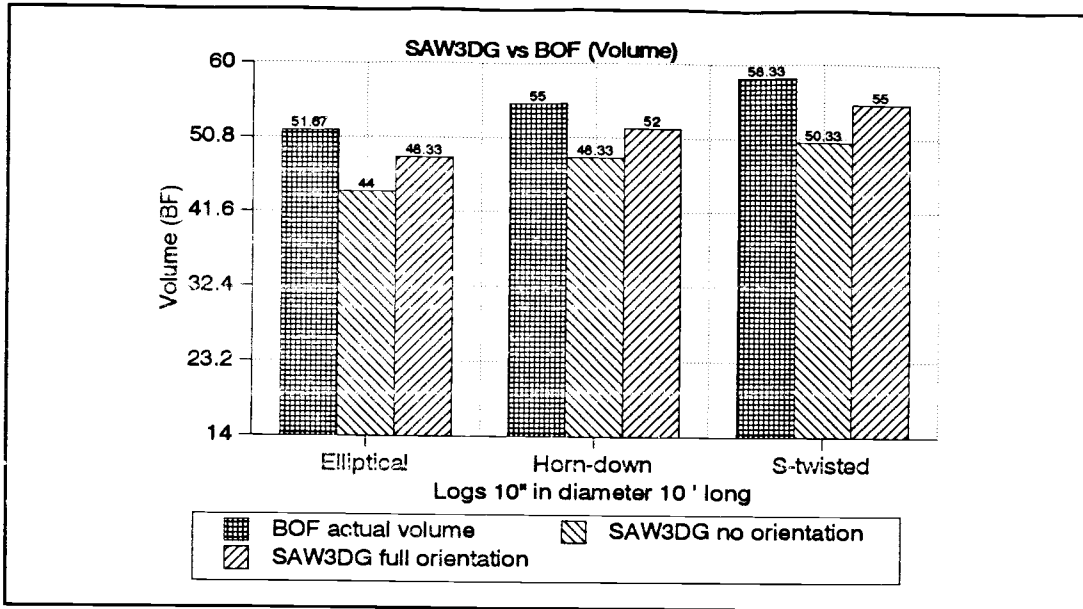
Figure 7.17. Effect of real shape and defects on lumber volume recovery comparing SAW3DG vs BOF.
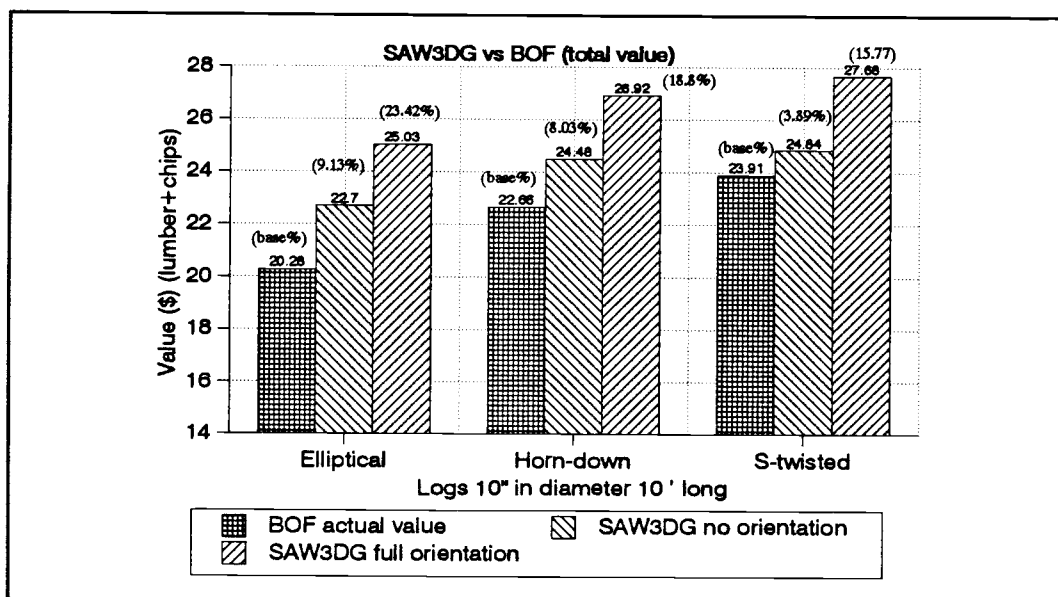
Figure 7.18. Total log value yield after considering chip value comparing SAW3DG vs BOF.
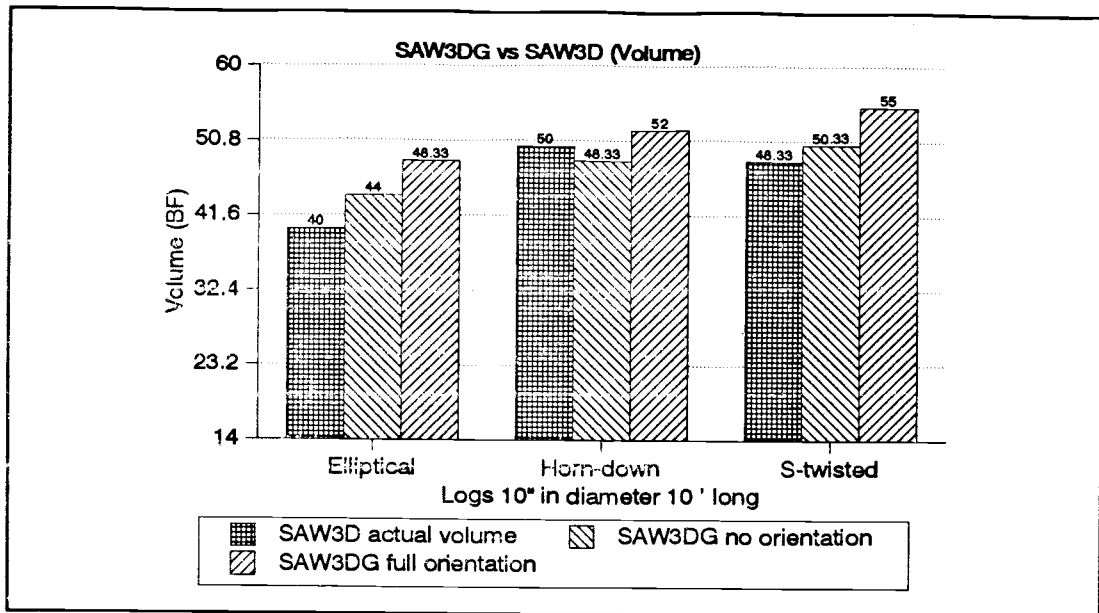
Figure 7.19. Effect of real shape and defects on lumber volume recovery comparing SAW3DG vs SAW3D.
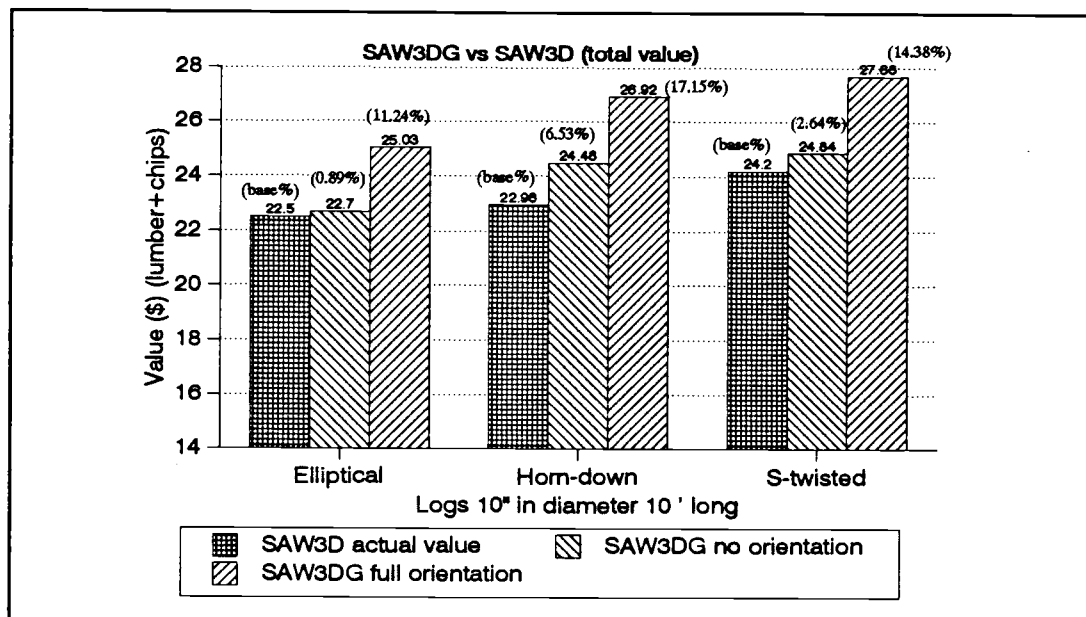
Figure 7.20. Total log value yield after considering chip value comparing SAW3DG vs SAW3D.

7.12 Effects of Positioning, Log Sizes, and Internal Defects On Computing Time

The current version of SAW3DG was developed as an analytical tool rather than an on-line optimization system. Therefore, execution speed was not a major concern when constructing the software package. On a Pentimum 90MHZ IBM compatible computer, the twelve logs have execution times ranging from several seconds to several days. Table 7.6 shows execution times associated with breaking down two S-twisted shaped logs. One log is 10 inches in diameter and 10 feet long (S11), and another is 20 inches in diameter and 20 feet long (S22). When sawing the logs using SAW3D, that is, assuming no internal defects, the execution time to find the solution for S11 was 0.5 second if not positioning operations were performed. When sawing the same log containing four knots using SAW3DG, the execution time was 6 minutes and 1.3 seconds, 722.6 times as slow. If the log size is larger, the execution time is also longer. S22 for instance, consumed 3.2 seconds when using SAW3D and 5 hours 29 minutes and 45.1 seconds when using SAW3DG, even without performing any positioning operations. Another factor that affects execution time is the positioning operation. Positioning operations need to spend more time to find the best solutions, including the best orientation. For example, SAW3DG spent 6 minutes 1.3 seconds to find the optimum solution for S11 without performing any positioning operations. If full positioning operations were performed, that is, a combination of rotation, skew, and pitch, SAW3DG spent 54 hours 45 minutes and 5 seconds to find the optimum solution and the best positions.

Table 7.6. Computer execution times of breaking down two S-twisted logs (times in hours:minutes:seconds).

Log number	Model	No positioning	Rotation only	Skew only	Pitch only	Full positioning
S11	SAW3D	0:0:0.5	0:0:31	0:0:1.2	0:0:1	0:4:16.8
	SAW3DG	0:6:1.3	7:5:39.4	0:11:37.1	0:17:50.2	54:45:5
S22	SAW3D	0:0:3.2	0:3:35.5	0:0:10.1	0:0:9	0:43:1.3
	SAW3DG	5:29:45.1	125:7:33.8	19:7:24.5	23:4:25.7	not available

CHAPTER 8 CONCLUSIONS AND FUTURE RESEARCH

In an attempt to approach the ideal program for log breakdown optimization, SAW3DG considered complete log information, true log shape and wood quality, while searching for optimum solutions. Compared to existing programs using simplified log shapes or using 3-dimensional log shape but ignoring internal defects, SAW3DG provides better solutions resulting in the recovery of higher valued lumber.

SAW3DG was developed by modifying SAW3D, a log breakdown optimization program that considers only 3-dimensional log shape and is written in the C computer language. It integrated the modified SAW3D with SLGRADER, an expert system for softwood lumber grading. The methodology of integrating an expert system and a conventional program makes the system easy to maintain and modify if grading rule expansions or changes are required.

This system provides an analytical tool to study the effects of various factors on lumber recovery, such as log shape, internal defects, positioning operations on the log and each piece cut from the log, production parameters, scanning system capabilities, and market conditions. Though many existing systems also claim to be used for similar purposes, including SAW3D, SAW3DG is superior because it is based on 3-dimensional shapes of both log geometry and internal defects.

In addition to accepting manually digitized log and internal defect information or data from a scanning system, a set of programs was also developed to allow the user to mathematically generate logs containing internal defects. The generated log, as well as internal defects, can be in any size and shape in 3-dimensional space as defined by the user. The internal defects can be located anywhere and in any direction. The capabilities of allowing the user to define logs makes SAW3DG a very useful and flexible analytical tool.

Twelve mathematically generated logs in various sizes and shapes were used to test the system. The logs also contained several defects. Observations made by breaking down the logs using SAW3DG, SAW3D, and BOF include:

1. Compared to BOF, which assumes logs are perfect truncated cones without any defects, SAW3DG provided better sawing, edging, and trimming patterns resulting in higher value recovery.

2. Compared to SAW3D, which is better than BOF because it considers the true shape of the log but not internal defects, SAW3DG made sawing ,edging, and trimming decisions not only based on the true shapes, but also log internal defects and therefore resulted in higher value recovery than SAW3D's solutions.

3. Among all sawing methods, including live, cant, around, and multi-thickness sawing, multi-thickness resulted in the highest value recovery, followed by live sawing. Around sawing resulted in the lowest value recovery.

4. Log positioning significantly affects value recovery. Among all positioning operations, including log rotation, skewing, and flitch/cant pitching, rotation resulted in the most significant value increases in most cases. When combinations of all positioning operations were applied, the highest value increases were achieved.

5. The maximum value recovery did not necessarily mean the volume recovery was also maximum. Lower lumber volume recovery means a higher portion of the log is converted to chips. When chip value was considered, the total value return of the log became the lumber value plus chip value. In this case, the optimum solutions made without considering chip value were still optimum. That is, lumber value was the dominate factor affecting the decisions that resulted in the optimum total value recovery.

Care should be taken when making any solid, statistically meaningful conclusions. The observations discussed above were based on only the twelve log samples and selected defects. It seems that results were sensitive to each individual log shape, size, and defect distributions. Therefore, at present the conclusions should not be extended to explain general log breakdown problems.

The current version of SAW3DG needs to be further improved in the following areas:

1. Considering more grading categories. At present only Light Framing and Structural Joist and Planks sub-categories of Dimension and Commons of the Boards

sub-category of Board grading categories are included in the system. More grading sub-categories and categories may need to be included for the system to be applicable to the lumber products using these grading rules.

2. Refining the grading rule implementations in the knowledge base. For example, knot positions as related to each other may have an impact on the lumber grade. However, the current grading rules in the knowledge base does not consider such information.

3. Including growth rings. Growth rings play an important role in lumber grading. The lumber grading component of SAW3DG is able to determine lumber grade based on slope of grain, but growth rings are not represented in the current version of SAW3DG.

4. Improving execution speed. On a Pentium 90MHZ IBM compatible computer, logs tested so far have execution times ranging from tens of seconds to several days, depending upon log size, shape, number of internal defects as well as positioning operations.

5. Evaluating the system using more logs in a variety of shapes and sizes containing more defects in a number of distributions. The current version of SAW3DG was evaluated using only a limited number of logs and internal defects. The results observed by sawing a limited number of logs just serves as a guideline for further analysis on log breakdown optimization but cannot be used to draw any statistically significant conclusions.

6. Developing a graphic user interface. A graphic user interface has been developed for SAW3D, whereas no such graphic user interface is developed for SAW3DG. The user interface for SAW3D needs to be modified so that features for SAW3DG can be handled, such as allowing price tables based on different grade categories and integrating utilities for generating logs and internal defects.

BIBLIOGRAPHY

1. Adkins, W.K., D.B. Richards, D.W. Lewis, and E.H. Bulgrin. 1980. Programs for computer simulation of hardwood log sawing. USDA Forest Service Res. Paper FPL-357. Forest Products Laboratory, Madison, WI.

2. Airth, J.M., and W.W. Calvert. 1973a. Adapting computer simulation techniques to the sawmill. Canadian Forest Industries 93(9):48-52.

3. Airth,J.M., and W.W. Calvert. 1973b. Computer simulation of log sawing. Info. Rept. OP-X-66. Forintek Canada Corp., Vancouver, B.C. Canada.

4. Alleckson, T.D., H.B. Sanders, A.J. Koivo, and T.J. Williams. 1980. Studies on the optimum production of lumber by computer positioning of logs in sawmills. Purdue Lab. for Applied Industrial Control Rept. No. 79. Purdue University, West Lafayette, IN.

5. Anderson, R.B., and H.W. Reynolds. 1981. Simulated sawing of squares: a tool to improve wood utilization. USDA Forest Service Research Paper NE-473. Northeastern Experiment Station, Upper Darby, PA.

6. Aune, J.E. 1982. Application and benefits of simulation models in the sawmilling industry. Pages 9-18 in Proceeding of the Process Control In The Forest Products Industry Symposium. Society of Wood Science and Technology, Madison, WI.

7. Bellman, R.E. 1957. Dynamic programming. Princeton University Press, Princeton, N.J.

8. Bricka, V. 1990. Growth ring pattern consideration in computer sawing simulation. Unpublished M.S. thesis. Oregon State University, Corvallis, OR.

9. Briggs, D.G., 1980. A dynamic programming approach to optimizing stem conversion. Unpublished Ph.D. dissertation. Univ. of Washington, Seattle, WA.

10. Burger, P., and D. Gillies. 1989. Interactive computer graphics. Addison-Wesley Publishing Company, Inc, Reading, MA.

11. Chang, S.J. 1989. Economic feasibility analysis of the NMR fast imaging scanner. Pages VII-1 to VII-6 in Proceedings of the Third International Conference On Scanning Technology In Sawmilling. Miller-Freeman Publications, San Francisco, CA.

12. Chassen, L.H. 1987. Application of "snapshot" scanning and optimization to log bucking. Pages 39-47 in Proceedings of the Log Bucking Technology Symposium. Special Publication No. SP-29. Forintek Canada Corp., Vancouver, B.C., Canada.

13. Cooney, T. 1987. Bucking decisions: computer programs help loggers increase revenues. Journal of Forestry 85(1):13-14.

14. Cummins, L.K., and D.D. Culbertson. 1972. Sawmill simulation model for maximizing log yield values. Forest Products Journal 22(10):34-40.

15. Dasher, B. 1993. 3D log scanning: the next generation. Pages I-1 to I-13 in Proceedings of the Fifth International Conference On Scanning Technology & Process Control For the Wood Products Industry. Miller-Freeman Publications, San Francisco, CA.

16. Denardo, E.V. 1982. Dynamic programming - models and applications. Prentice-Hall, Inc., Englewood Cliffs, N.J.

17. Faaland, B., and D. Briggs. 1984. Log bucking and lumber manufacturing using dynamic programming. Management Science 30(2):245-257.

18. Funt, B.V., and E.C. Bryan. 1987. Detection of internal log defects by automatic interpretation of computer tomography images. Forest Products Journal 37(1):56-62.

19. Garcia, O. 1987. SEESAW: A Visual sawing simulator. Part II: The SEESAW computer program. FRI Bulletin No. 128, Ministry of Forestry, New Zealand.

20. Geerts, J.M. 1984. Mathematical solution for optimising the sawing pattern of a log given its dimensions and its defect core. New Zealand Journal of Forestry Science 14(1):124-34.

21. Gilmore, P.C., and R.E. Gomory. 1966. The theory and computation of knapsack functions. Operations Research 14(6):1045-1074.

22. Glück, P., and W. Kock. 1973. Die Optimale Rohholzaustormung. Centralblatt fur des gesampte forstwesen 90(4): 193-228.

23. Green, J. C. 1993. 3-D log scanning and optimization. Pages II-1 to II-16 in Proceedings of the Fifth International Conference On Scanning Technology & Process Control For the Wood Products Industry. Miller-Freeman Publications, San Francisco, CA.

24. Haley Enterprise. 1992. Eclipse reference manual. The Haley Enterprise, Inc., Sewickley, PA.

25. Hallock, H., and D.W. Lewis. 1971. Increasing softwood dimension yield from small logs -- best opening face. USDA Forest Service Res. Paper FPL-166. Forest Products Laboratory, Madison, WI.

26. Hallock, H., A.R. Stern, and D.W. Lewis. 1976. Is there a "best" sawing method? USDA Forest Service Res. Paper FPL-280. Forest Products Laboratory, Madison, WI.

27. Harless, T.E.G., F.G. Wagner, P.H.Steele, F.W. Taylor, V. Yadama, and C.W. McMillin. 1991. Methodology for locating defects within hardwood logs and determining their impact on lumber-value yield. Forest Products Journal 41(4):25-30.

28. Leach, H.A. 1973. Computer program for log bucking and sawing, user's manual. Unpublished manual. Carrol-Hatch Ltd., Vancouver, B.C., Canada.

29. Leach, H.A. 1979. SAWSIM: sawmill simulation program. H.A. Leach & Co., Ltd. Vancouver, B.C., Canada.

30. Leach, H. A. and Company Ltd. 1990. SAWSIM. Users guide. Howard A. Leach, Ltd., Vancouver, B.C., Canada.

31. Leban, J.M., F. Colin, and F. Houllier. 1990. SIMQUA - a wood quality simulation software program. SIMQUA user's manual. French Wood Product Quality Station (INRA). Champenoux, France.

32. Lewis, D.W. 1985a. Sawmill simulation and the best opening face system: a user's guide. USDA Forest Service Gen. Tech. Report FPL-48. Forest Products Laboratory, Madison, WI.

33. Lewis, D.W. 1985b. Best opening face system for sweepy, eccentric logs: a user's guide. USDA Forest Service Gen. Tech. Report FPL-49. Forest Products Lab., Madison, WI.

34. Lewis, D.W., and H. Hallock. 1974. Best opening face programme. Australian Forest Industries Journal 11:21-30.

35. McAdoo, J.C. 1969. Computer simulation of small-log mill processing. Forest Products Journal 19(4):34-35.

36. Microsoft Corp., 1994. Microsoft Windows NT system guide. Microsoft Corp. Redmond, WA.

37. Nakata, K. 1986. Simulation of softwood-log sawing. Journal of The Hokkaido Forest Products Research Institute 410(3):15-22.

38. Nielson, R.W., J. Dobie, and D.M. Wright. 1985. Conversion factors for the forest products industry in western Canada. Special Publication No. SP-24R. Forintek Canada Corp. Vancouver, B.C., Canada.

39. Occena, L.G. 1992. Hardwood log breakdown decision automation. Wood and Fiber Science 24(2):181-188.

40. Occena, L.G., and J.M.A. Tanchoco. 1988. Computer graphics simulation of hardwood log sawing. Forest Products Journal 38(10):72-76.

41. Park, J. C. 1987. SEESAW: A Visual sawing simulator. Part I: data, methods, and program evaluation. FRI Bulletin No. 128. Ministry of Forestry, New Zealand.

42. Park, J.C. 1994. Evaluating pruned sawlog quality and assessing sawmill recoveries in New Zealand. Forest Products Journal 44(4):43-52.

43. Parker, D.B. 1989. Acrospin user's guide. Acrobits, Redwood City, CA.

44. Peter, R., and J.H. Bamping. 1962. Theoretical sawing of pine logs. Forest Products Journal 12(11):549-557.

45. Peter, R.K. 1967. Influence of sawing methods on lumber grade yield from yellow-poplar. Forest Products Journal 17(11):19-24.

46. Pnevmaticos, S.M., P.E. Dress, and F.R. Stocker. 1974. Log and sawing simulation through computer graphics. Forest Products Journal 24(3):53-55.

47. Pnevmaticos, S.M., H.G. Lama, and M. R. Milot. 1976. Application of computer graphics in simulating sawmilling operations. Pages 1-13 in Proceedings of 9th Annual Simulation Symposium. IEEE, Tampa, FL.

48. Pnevmaticos, S. M., and S. H. Mann. 1972. Dynamic programming in tree bucking. Forest Products Journal 22(2):26-30.

49. Pnevmaticos, S.M., and P. Mouland. 1978. Hardwood sawing simulation techniques. Forest Products Journal 28(4):51-55.

50. Priasukmana, S. 1984. Sawing of sweepy logs using a live-sawing simulation model. Unpublished M.S. thesis. Univ. of Washington, Seattle, WA.

51. Reynolds, H.W. 1970. Sawmill simulation: data instructions and computer programs. USDA Forest Service Res. Paper NE-152. Northeastern Forest Exp. Station, Upper Darby, PA.

52. Reynolds, H.W., and C.J. Gatchell. 1969. Sawmill simulation: concepts and computer use. USDA Forest Service Res. Note NE-100. Northeastern Forest Experiment Station, Upper Darby, PA.

53. Richards, D.B. 1973. Hardwood lumber yield by various simulated sawing methods. Forest Products Journal 23(10):50-58.

54. Richards, D.B. 1977. Value yield from simulated hardwood log sawing. Forest Products Journal 27(12):47-50.

55. Richards, D.B., W.K. Adkins, H. Hallock, and E.H. Bulgrin. 1979. Simulation of hardwood log sawing. USDA Forest Service Res. Paper FPL-355. Forest Products Laboratory, Madison, WI.

56. Richards, D.B., W.K. Adkins, H. Hallock, and E.H. Bulgrin. 1980. Lumber values from computerized simulation of hardwood log sawing. USDA Forest Service Res. Paper FPL-356. Forest Products Laboratory, Madison, WI.

57. Rickford, E.N. 1989. Scanning for true shape. Pages II-1 to II-40 in Third International Conference on Scanning Technology in Sawmilling. Miller Freeman Publications, San Francisco, CA.

58. Rogers, D.F., and J.A. Adams. 1990. Mathematical elements for computer graphics, 2/ed. McGraw-Hill Publishing Company, New York, NY.

59. Rogler, R. K., and H. O. Canham. 1986. An optimal log bucking program for microcomputers. The Compiler 4(2):27-30.

60. Samson, M., 1993a. Method for assessing the effect of knots in the conversion of logs into structral lumber. Wood and Fiber Science 25(3):298-304.

61. Samson, M., 1993b. Modelling of knots in logs. Wood Science and Technology 27:429-437.

62. Scaramella, G. 1987. Stem optimizing: a critical link in sawmill production control. Pages 49-63 in Proceedings of the Log Bucking Technology Symposium. Special Publication No. SP-29. Forintek Canada Corp., Vancouver, B.C., Canada.

63. Steele, P.H., T.E.G. Harless, F.G. Wagner, L. Kumar, and F.W. Taylor. 1994. Increased lumber value from optimum orientation of internal defects with

respect to sawing pattern in hardwood sawlogs. Forest Products Journal 44(3):69-72.

64. Sullivan, P. 1987. Bucking optimization with "real shape" scanning. Pages 64-75 in Proceedings of the Log Bucking Technology Symposium. Special Publication No. SP-29. Forintek Canada Corp., Vancouver, B.C., Canada.

65. Tejavibulya, S. 1981. Dynamic programming sawing models for optimizing lumber recovery. Unpublished Ph.D. Dissertation. University of Washington, Seattle, WA.

66. Thomlinson, W.W. 1987. True-shape scanning: Lloyd Controls' optimum log value system. Pages 5-15 in Proceedings of the Log Bucking Technology Symposium. Special Publication No. SP-29. Forintek Canada Corp., Vancouver, B.C., Canada.

67. Tochigi, T., K. Onose, H. Satoh, and K. Sakatsume. 1983. A computer simulation of grain patterns on sawn surfaces. Mokuzai Gakkaishi 29(12):845-852.

68. Todoroki, C.L. 1988. Seesaw: a visual sawing simulator, as developed in version 3.0. New Zealand Journal of Forestry Science 18(1):116-123.

69. Todoroki, C.L. 1990. Autosaw system for sawing simulation. New Zealand Journal of Forestry Science 20(3):332-348.

70. Tsolakides, J.A. 1969. A simulation model for log yield study. Forest Products Journal 19(7):21-26.

71. Wagner, F.G., T.E.G. Harless, P.H. Steele, F.W. Taylor, V. Yadama, and C.W. McMillin. 1990. Potential benefits of internal-log scanning. Pages 77 to 88 in Proceedings of Process Control/Production Management of Wood Products: Technology for the '90s. Forest Products Research Society.

72. Wagner, F.G., and F.W. Taylor. 1975. Simulated sawing with a chipping headrig. Forest Products Journal 25(10):24-28.

73. Wagner, F.G., F.W. Taylor, P. Steele, and P.E.G. Harless. 1989. Benefit of internal log scanning. Pages V-1 to V-17 in Proceedings of the Third International Conference On Scanning Technology In Sawmilling. Miller-Freeman Publications, San Francisco, CA.

74. Western Wood Products Association. 1988. Western lumber grading rule 88. Western Wood Products Association, Portland, OR.

75. Williston, E.M. 1981. Small log sawmills. Miller-Freeman Publications, San Francisco, CA.

76. Wodzinski, C., and E. Hahn. 1966. A computer program to determine yields of lumber. USDA Forest Service Misc. Pub. Forest Products Lab., Madison, Wis..

77. Zeng, Y. 1991. Log breakdown using dynamic programming and 3-D log shape. M.S. thesis. Oregon State University, Corvallis, OR.

78. Zeng, Y. 1993. An expert system for softwood lumber grading. M.S. thesis. Oregon State University, Corvallis, OR.

79. Zheng, Y., F.G. Wagner, P.H. Steele, and Z. Ji. 1989. Two-dimensional geometric theory for maximizing lumber yield from logs. Wood and Fiber Science 21(1):91-100.

APPENDICES

Appendix A. The text output from sawing a S-twisted shaped log 20 inches in diameter and 10 feet long.

log scanning data file: s21new3.log

Rotation angle   = 0.00
Optimum skew angle   = 0.00

saws placement position :
8.875   (1in) 7.625
        edging positions:
        5.749 (12in) -6.126
        trimming positions:
        0.000 (10ft) 120.000
        Grade: 4_COMMON

        edging value = $3.50

saws placement position :
7.625   (2in) 5.625
        edging positions:
        9.633 (4in) 5.633
        don't cut
        edging positions:
        5.633 (12in) -6.242
        trimming positions:
        0.000 (10ft) 120.000
        Grade: SEL-STR-JK

        edging value = $9.60

saws placement position :
5.625   (2in) 3.625
        edging positions:
        7.984 (8in) 0.109
        trimming positions:
        0.000 (10ft) 120.000
        Grade: SEL-STR-JK

        edging positions:
        0.109 (8in) -7.766

trimming positions:
0.000 (10ft) 120.000
Grade: NO.1-JK

edging value = $12.80

saws placement position :
3.625 (2in) 1.625
edging positions:
9.127 (10in) -0.748
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging positions:
-0.748 (8in) -8.623
trimming positions:
0.000 (10ft) 120.000
Grade: NO.2-JK

edging value = $13.60

saws placement position :
1.625 (2in) -0.375
edging positions:
9.892 (4in) 5.892
trimming positions:
0.000 (8ft) 96.000
Grade: CONST

edging positions:
5.892 (8in) -1.983
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging positions:
-1.983 (8in) -9.858
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $15.39

saws placement position :
-0.375 (2in) -2.375
    edging positions:
    10.017 (4in) 6.017
    trimming positions:
    0.000 (8ft) 96.000
    Grade: CONST

    edging positions:
    6.017 (8in) -1.858
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging positions:
    -1.858 (8in) -9.733
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging value = $15.39

saws placement position :
-2.375 (2in) -4.375
    edging positions:
    9.015 (10in) -0.860
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging positions:
    -0.860 (8in) -8.735
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging value = $14.53

saws placement position :
-4.375 (2in) -6.375
    edging positions:
    8.273 (8in) 0.398

trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging positions:
0.398 (8in) -7.477
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $13.07

saws placement position :
-6.375 (2in) -8.375
edging positions:
8.236 (4in) 4.236
trimming positions:
0.000 (8ft) 96.000
Grade: ECONOMY

edging positions:
4.236 (10in) -5.639
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $8.71

saws placement position :
-8.375 (1in) -9.625
edging positions:
6.380 (6in) 0.380
trimming positions:
0.000 (8ft) 96.000
Grade: 4_COMMON

edging positions:
0.380 (6in) -5.620
trimming positions:
0.000 (10ft) 120.000
Grade: 4_COMMON

edging value = $2.92

sawing sequence = 1 2 2 2 2 2 2 2 2 1
Optimum lumber recovery in value = $109.51
Lumber tally = 238.333333 (BF)
Real lumber volume = 16.808160 (cu. ft.)
Log volume = 24.623   (cub. ft.)
Chips volume = 7.815   (cub. ft.)
Chips value = $10.854
Optimum value of the log = $120.367
CPU time = 0:35:16.09

Appendix B. **The text output from sawing a horn-down shaped log 20 inches in diameter and 10 feet long using the around-sawing.**

log scanning data file: h21new2.log
defect scanning data file: knots.in

Rotation angle   = 0.00

Side Index: 0
Optimum skew angle   = 0.00

saws placement position :
-3.125 (2in) -5.125
        edging positions:
        8.482 (6in) 2.482
        trimming positions:
        0.000 (10ft) 120.000
        Grade: SEL-STR-JK

        edging positions:
        1.982 (4in) -2.018
        trimming positions:
        0.000 (10ft) 120.000
        Grade: CONST

        edging positions:
        -2.268 (6in) -8.268
        trimming positions:
        0.000 (10ft) 120.000
        Grade: SEL-STR-JK

        edging value = $12.30

saws placement position :
-5.125 (2in) -7.125
        edging positions:
        6.952 (8in) -0.923
        trimming positions:
        0.000 (10ft) 120.000
        Grade: SEL-STR-JK

        edging positions:
        -0.923 (6in) -6.923

trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $11.23

saws placement position :
-7.125 (2in) -9.125
    edging positions:
    8.007 (4in) 4.007
    don't cut
    edging positions:
    4.007 (8in) -3.868
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging value = $6.53

sawing sequence = 2 2 2
Optimum value of this side = $30.07
lumber tally = 63.333333

Side Index: 1
Optimum skew angle   = 0.00

saws placement position :
-3.437 (2in) -5.437
    edging positions:
    8.375 (10in) -1.500
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging value = $8.00

saws placement position :
-5.437 (2in) -7.437
    edging positions:
    9.125 (4in) 5.125
    don't cut
    edging positions:
    5.125 (8in) -2.750

trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $6.53

saws placement position :
-7.437 (2in) -9.437
edging positions:
7.125 (4in) 3.125
trimming positions:
0.000 (8ft) 96.000
Grade: ECONOMY

edging positions:
3.125 (6in) -2.875
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $5.41

sawing sequence = 2 2 2
Optimum value of this side = $19.95
lumber tally = 45.333333

Side Index: 2
Optimum skew angle   = 0.00

saws placement position :
-4.875 (2in) -6.875
edging positions:
7.188 (10in) -2.687
trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $8.00

saws placement position :
-6.875 (2in) -8.875
edging positions:
4.563 (8in) -3.312

trimming positions:
0.000 (10ft) 120.000
Grade: SEL-STR-JK

edging value = $6.53

saws placement position :
-8.875 (1in) -10.125
    edging positions:
    4.563 (8in) -3.312
    trimming positions:
    0.000 (8ft) 96.000
    Grade: 4_COMMON

    edging value = $1.76

sawing sequence = 2 2 1
Optimum value of this side = $16.29
lumber tally = 35.333333

Side Index: 3
Optimum skew angle   = 0.00

saws placement position :
3.375 (1in) 2.125
    edging positions:
    3.125 (8in) -4.750
    trimming positions:
    0.000 (10ft) 120.000
    Grade: 2_COMMON

    edging value = $2.87

saws placement position :
1.875 (2in) -0.125
    edging positions:
    3.125 (8in) -4.750
    trimming positions:
    0.000 (10ft) 120.000
    Grade: SEL-STR-JK

    edging value = $6.53

saws placement position :
-0.375 (2in) -2.375

edging positions:
3.125 (8in) -4.750
trimming positions:
0.000 (10ft) 120.000
Grade: NO.2-JK

edging value = $5.60

saws placement position :
-2.375 (2in) -4.375
edging positions:
3.125 (8in) -4.750
trimming positions:
0.000 (10ft) 120.000
Grade: NO.2-JK

edging value = $5.60

saws placement position :
-4.375 (2in) -6.375
edging positions:
3.125 (4in) -0.875
trimming positions:
0.000 (10ft) 120.000
Grade: CONST

edging value = $2.90

saws placement position :
-6.375 (2in) -8.375
edging positions:
3.125 (8in) -4.750
trimming positions:
0.000 (10ft) 120.000
Grade: NO.1-JK

edging value = $6.27

saws placement position :
-8.375 (1in) -9.625
edging positions:
3.125 (8in) -4.750

trimming positions:
0.000 (10ft) 120.000
Grade: 4_COMMON

edging value = $2.20

sawing sequence = 1 2 2 2 2 2 1
Optimum value of this side = $31.97
lumber tally = 73.333333
Optimum value of the log for grade sawing = $98.28
Total tally = 217.333333
CPU time = 0:15:52.15