

# CUTTING TO ORDER IN THE ROUGH MILL: A SAMPLING APPROACH

*Evan D. Hamilton*

President  
Mckenzie Group  
2353 N.E. 60th Ave.  
Portland, OR 97213

*David A. Butler*

Professor  
Department of Statistics

and

*Charles C. Brunner*<sup>†</sup>

Associate Professor  
Department of Forest Products  
Oregon State University  
Corvallis, OR 97331

(Received June 2001)

## ABSTRACT

A cutting order is a list of dimension parts along with demanded quantities. The cutting-order problem is to minimize the total cost of filling a cutting order from a given lumber supply. Similar cutting-order problems arise in many industrial situations outside of forest products. This paper adapts an earlier, linear programming approach that was developed for uniform, defect-free stock materials. The adaptation presented here allows the method to handle nonuniform stock material (e.g., lumber) that contains defects that are not known in advance of cutting. The main differences are the use of a random sample to construct the linear program and the use of prices rather than cutting patterns to specify a solution. The primary result of this research is that the expected cost of filling an order under the proposed method is approximately equal to the minimum possible expected cost for sufficiently large order and sample sizes. A secondary result is a lower bound on the minimum possible expected cost. Computer simulations suggest that the proposed method is capable of attaining nearly minimal expected costs in moderately large orders.

*Keywords:* Cutting order, linear program, secondary manufacturing, knapsack problem, optimizer.

## INTRODUCTION

A rough mill converts rough-cut boards into smaller pieces, such as furniture parts or trim moldings. Because of high lumber costs, conversion efficiency is of major importance to the industry and is the subject of this paper. Although we will focus on the production of wood blanks for the furniture industry, the methods presented here are equally applicable to other industries.

Blanks are produced in different dimensions and in different quantities as specified in a cutting order (Table 1). In general, cutting orders may allow one dimension (length or width) to be random, but we will consider only cutting orders for which both are fixed. Blanks are typically categorized as *clear one-face* (C1F), *clear two-face* (C2F), or *sound two-face* (S2F). The process of filling an order by cutting boards that contain knots and other defects into blanks is called *cutting to order*, and the *cutting-order problem* is to minimize the

---

<sup>†</sup> Member of SWST.

TABLE 1. *Format of a cutting order.*

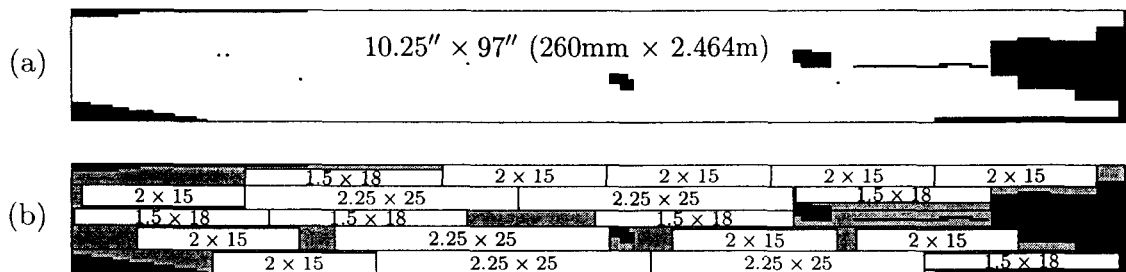
Description	Quantity required
2.00" × 15" Red Oak (C2F)	60
1.50" × 18" Red Oak (C2F)	85
2.25" × 25" Red Oak (C2F)	40

cost of producing the blanks. An alternative is *cutting to inventory*, in which a monetary value is assigned to each blank and one seeks to maximize the total value of the blanks produced from a given quantity of rough lumber. Cutting to inventory will not be considered here, though we will use artificially generated blank values to control the cutting process.

The details of the sawing process need not be described for the purposes of this research. It suffices to assume the existence of an *optimizer* that, given a specific board and specified blank values, generates sawing instructions that maximize (or approximately maximize) the total value of blanks produced from the board. In an automated sawmill, the optimizer's inputs come from a computer vision system or *scanner*, which detects the board's dimensions as well as the locations and shapes of knots and other defects. Figure 1(a) shows a sample output from a scanner. The scanned image has been reduced to a discrete grid of 0.25-in. (6.35-mm) squares, where the dark areas represent defects. Figure 1(b) shows the optimizer's output, a maximum-value cutting pattern; the light areas represent usable blanks,

the dark areas represent defects, and the gray areas represent waste.

This research builds on a linear programming (LP) approach to the cutting-order problem, due to Gilmore and Gomory (1961, 1963) (G&G). Their work applies to uniform stock materials such as paper rolls or glass sheets, where a relatively small number of different stock sizes are available for cutting and all stock objects of a given size are identical to one other. Since the stock objects are uniform, a given cutting pattern always produces the same end products. But heterogeneous materials such as rough-cut boards require individual cutting patterns because the shapes and locations of defects vary. Moreover, board defects are not usually known in advance, so cutting decisions regarding a board cannot depend upon the specific characteristics of subsequent boards. To accommodate random variation in the boards, we modify the G&G procedure in two ways. First, we use a random sample of boards rather than a single object (e.g., a glass plate of a given stock size) to generate columns for the linear program. Second, the solution is not implemented in terms of fixed cutting patterns that are to be applied to a specified number of stock objects, but rather in terms of dynamically changing the blank values (or prices) that are supplied to the optimizer. The prices are obtained from dual variables calculated in the course of solving the linear program. The order quantities specified in the linear program are updated in



$$\text{Total value: } 6.232 = (.255 \cdot 9) + (.212 \cdot 6) + (.533 \cdot 5)$$

FIG. 1. Sample of optimizer input and output: (a) scanned board; (b) a maximum-value cutting pattern.

the course of filling the order, so the linear program is solved not just once but periodically (ideally after each board is cut).

We call this approach the *LP/sampling method*, and it is intended for use with large cutting orders. The method ignores fixed costs, such as setup costs, but can accommodate variable cutting costs (e.g., each cut adds a small cost) if an optimizer is available that can find the maximum-value cutting pattern net of cutting costs.

#### LITERATURE REVIEW

The literature on cutting problems is quite large, and the review presented here is limited. For more details, see Hamilton (2001). The USDA's Forest Products Laboratory (FPL) at Madison, Wisconsin, was a pioneer in developing programs for studying lumber yields in the production of blanks. Their purpose was not to automate wood processing but to generate yield data that could improve planning at a typical operation. They include YIELD (Wodzinski and Hahm 1966); MULRIP (Hallock 1980; Gatchell et al. 1983); CROMAX (Giese and Danielson 1983); and RIPPYLD and OPTYLD (Giese and McDonald 1982). The USDA Forest Service's Northeastern Research Station has carried on this work by developing a crosscut-first simulator ROMI-CROSS (Thomas 1997) and a rip-first simulator ROMI-RIP (Thomas 1999).

Klinkhachorn et al. (1989) reported on a computer program created as part of the ALPS (Automated Lumber Processing System) project. It was designed to control the cutting of blanks with a high-powered laser that does not require through cuts. The algorithm employs various heuristics to formulate cutting patterns that maximize total yield.

Another set of programs for cutting blanks from boards is CORY (Computer Optimization of Recoverable Yield) (Brunner et al. 1989). CORY uses a heuristic decision model based on analyzing clear areas, where the objective is to identify a set of clear areas having the largest total value. A set of pre-specified

rules is used to decide among competing clear areas and to resolve conflicts arising when cuts used to extract one clear area extend into another. CORY bases its decisions on combinations of up to three clear areas. Different versions of CORY accommodate two-, three-, or four-stage processes and fixed or variable blank sizes (Anderson et al. 1992).

#### *The LP approach to the cutting-order problem*

Gilmore and Gomory (1961, 1963) showed how to formulate cutting-order problems as linear programs. In the application they addressed, a manufacturer has paper rolls of a single stock length from which an order for specified quantities of shorter rolls is to be filled. Shorter rolls are produced by cutting longer stock rolls, and the objective is to minimize the total cost of stock rolls used to fill the order. Unlimited quantities of stock rolls are available at a cost of  $c$  per roll. Suppose that the cutting order specifies  $m$  different lengths of shorter rolls in quantities  $b_i$ ,  $i = 1, 2, \dots, m$ . Cutting a stock roll produces a combination of rolls of the desired lengths and trim waste. Let  $a_{ij}$  be the number of rolls of the  $i$ th ordered length produced by using the  $j$ th cutting pattern, and assume there are  $n$  possible cutting patterns. Letting  $\mathbf{c} = (c, \dots, c)'$ ,  $\mathbf{b} = (b_1, \dots, b_m)'$ ,  $\mathbf{a}_j = (a_{1j}, \dots, a_{mj})'$ , and  $A = [\mathbf{a}_1, \dots, \mathbf{a}_n]$ , the problem can be formulated as a linear program:

$$\begin{aligned} \text{minimize:} & \quad \mathbf{c}'\mathbf{x} \\ \text{subject to:} & \quad A\mathbf{x} \geq \mathbf{b} \quad \mathbf{x} \geq 0 \quad (1) \end{aligned}$$

where  $\mathbf{x} = (x_1, \dots, x_n)'$  and  $x_j$  is the number of stock rolls cut using the  $j$ th cutting pattern. A minor modification to the formulation allows multiple stock lengths to be considered. The LP formulation usually leads to fractional elements in the optimal  $\mathbf{x}$ , which must be rounded up to obtain an approximate solution to the exact integer programming formulation. The LP approach therefore works best with larger orders, where the effects of rounding are smaller.

One difficulty with this formulation is that there is usually a very large number of possible cutting patterns, and consequently a very large number of columns in  $A$ . To surmount it, G&G showed how the revised simplex method (Murty 1976) can be used to generate columns only when they are needed by solving an auxiliary problem at each iteration. After transforming the linear program to standard form, a feasible basis  $B$  is a square matrix of linearly independent columns from  $A$  such that  $x = B^{-1}b \geq 0$ . The revised simplex method attempts to improve a current basic feasible solution by replacing one of the columns in  $B$  with a column of  $A$ ,  $a_j$ , that is not already in  $B$ . To be eligible for entry into the current basis, a column's *relative cost coefficient* must be negative, i.e.,  $c_j - \pi'a_j < 0$ , where  $\pi' = c'_B B^{-1}$  and  $c_B$  is the vector of *basic cost coefficients* (Murty 1976). In the usual pivot step, the column with the most negative relative cost coefficient is selected to enter the basis, and if no column has a negative coefficient, the algorithm terminates. In G&G's problem, the cost  $c_j = c$  is constant, so the task of finding the most negative relative cost coefficient reduces to finding a column  $a_j$  from  $A$  that maximizes  $\pi'a_j$ . The column-entry criterion can thus be evaluated by maximizing the total value of pieces (i.e., shorter rolls) cut from the stock length, where the piece values are given by the vector  $\pi$ . If this maximum value is greater than  $c$ , then the associated column enters the basis  $B$ . Otherwise, the solution cannot be improved, and the algorithm terminates. This value-maximization problem, embedded in the overall cost-minimization problem, is called the *auxiliary knapsack problem*. Using this technique, only a small subset of columns ever requires explicit consideration.

#### *LP in wood processing*

Gilmore and Gomory's LP approach is well suited to applications that involve uniform stock materials, such as the cutting of particleboard (Carnieri et al. 1994). However, difficulties arise in applying it to applications

such as furniture-blank production, where random defects must be removed from the stock material to produce the finished product. One might consider each board to be a unique stock type that is available only in a quantity of one, but this undermines the linear programming approach, which is intended for use with just a few stock types that are available in moderately large quantities. To use LP in this context, it is therefore necessary to aggregate boards in some way. In doing so, current methods no longer control the sawing of individual boards, but instead minimize raw material costs by using estimated yields and lumber prices to specify board grade mixes.

Carino and Foronda implemented a program, SELECT, that aggregated boards of a given grade by their dimensions (Carino and Foronda 1990; Foronda and Carino 1991). Their formulation assumed that all boards of a given grade and dimension have the same productive potential. The grade/size categories were then treated as separate stock sizes, each one requiring the solution of an auxiliary knapsack problem. Each knapsack problem was defined by grading rules that specified the clear-area requirements for a board of a given size and grade. Also, their formulation implicitly assumed that within a given grade of random-dimension boards, certain sizes may be selected for use while other sizes may be diverted to inventory.

An alternative approach to dealing with board heterogeneity is to use experimentally derived general yield tables to estimate how much lumber is required to produce particular quantities of blanks. This is the approach taken in the computer programs OPTIGRAMI (Martens and Nevel 1985) and ROMGOP (Suter and Calloway 1994). OPTIGRAMI minimizes the cost to produce a cutting order by determining the quantity of each grade of available lumber to use, along with the quantities of blanks to be produced from each grade. ROMGOP formulates a goal program whose goals include minimizing underproduction, overproduction, and budget-, time-, and lumber-usage overruns. Instead of yield tables,

RIP-X (Harding and Steele 1997) uses computer simulations of specified rough-mill systems and cutting orders to estimate the yields used for calculating the optimal proportions for each available lumber grade.

#### *Process-control systems in wood processing*

An automated rough mill cannot saw aggregated boards; it must cut each board using an individualized sawing pattern. Saw line placement is determined by an optimizer using blank prices to calculate maximum-value cutting patterns. When cutting to inventory, the blank prices have real-world meaning. When cutting to order they do not; they merely provide a mechanism for controlling which blanks are preferred.

We call the system that is responsible for determining price settings a *controller*. In order to cut a random sequence of boards into specified quantities of blanks using as few boards as possible, the controller takes continuously supplied information about the quantities being produced and adjusts prices accordingly. The usual objective is to produce blanks in relatively constant proportions over the entire production run. If a blank is underrepresented or overrepresented in the output, the controller can increase or decrease its price to adjust its production rate.

Dmitrovic et al. (1992) described one possible controller algorithm. It uses information obtained from the counts of blanks accumulated over a limited time horizon to adjust future prices according to an inverse hyperbolic cosine function whose parameters are based on Kalman filtering techniques. Another controller-based approach uses fuzzy logic to calculate optimizer prices (Anderson et al. 1997). One difficulty with these two methods is that they are not easy to implement, and they require a certain amount of calibration to perform well. A much simpler strategy was put forth by Thomas (1996), based on a *complex dynamic exponent* (CDE).

The CDE formulas are based entirely on the quantities remaining to be produced, except

that values for the first 35 pieces of each size are boosted somewhat so as to avoid underproduction of sizes ordered in small quantities. Unlike the two previous methods, CDE ignores the relative rates at which blanks are being produced.

#### THE LP/SAMPLING METHOD

The solution method we propose for solving cutting-order problems in wood products applications is the LP/sampling method, which extends the G&G approach to the sawing of individual boards. The LP/sampling method consists of a controller that provides the optimizer with blank prices that are used to find each board's sawing solution. It does this by using a random sample from previously cut boards to supply information regarding yield characteristics for the lumber supply and to take into account the nonuniform nature of the stock material. We assume for the purposes of this discussion that the only costs are for raw material and that only a single lumber supply is available, consisting of a single grade or an unsorted mixture of multiple grades. We also assume that boards are presented in a given sequence and must be used in the order supplied. Some extensions, such as allowing multiple lumber supplies, are discussed in Hamilton (2001).

In its broadest sense, the proposed method is essentially the same as other methods that use prices to control production. Given a cutting order and a sequence of boards, the process of filling the order consists of these steps:

1. Calculate a price (or value) for each blank size.
2. Cut the next board in the sequence at those prices (i.e., cut the board so as to maximize the total value of blanks produced).
3. Reduce the remaining order quantities by subtracting the quantities just produced.
4. If the order is not yet filled, go to (1). Otherwise, stop.

What distinguishes one method from another

is the manner in which the prices are calculated in step (1).

The linear program in the LP/sampling method is the same as the one described in the section *The LP approach to the cutting-order problem*, except that columns of  $A$  are generated from a sample of boards rather than a single piece of stock material. In the proposed method, a separate knapsack problem is solved for each board in the sample, and the results are added together to create the new column. The cost associated with each new column is the sum of the board costs in the sample. The linear program is solved in exactly the same way, although the LP solution is used differently. In G&G's formulation, an optimal solution yields a linear combination of columns, where each column represents a cutting pattern that can be applied repeatedly to the uniform stock material. In our problem, each column represents cutting patterns for boards in a sample, and these are usually different than the boards to be cut. Since the cutting patterns are thus of no immediate use, we instead use the prices that generated them.

The key to implementing the LP/sampling method is to properly select prices for the optimizer from the generated columns. An optimal solution consists of a linear combination of columns that equals or exceeds the specified order quantities. In the LP/sampling method we choose any column in the linear combination that has maximal weighting (there may be several) and use the prices that originally generated that column. We call these *maximal-element prices*. In practice, maximal-element prices usually differ little from optimal dual prices for the fully solved LP, but they have a tie-breaking property that is mathematically useful. See Hamilton (2001) for examples of the application of this methodology.

If one blank size is guaranteed to be produced in sufficient quantities as a byproduct of producing other blanks, the LP may calculate its price as zero, even when more of the size is still needed. For example, if a cutting order consists of a large number of large blanks and a small number of very small

blanks, the small blanks can easily be cut from board areas that are too small to accommodate large blanks. To eliminate this technical difficulty, the price of any zero-price blank can be changed to a very small positive number,  $\epsilon$ . A sufficiently small price will ensure that a blank will be included in an optimal cutting pattern only if it fits into space that would otherwise be unused.

Another potential difficulty arises when the LP for a given sample has no feasible solution; this happens when there is some blank size in the order that cannot be produced by any board in the sample. In this situation, either the order is unsuited to the lumber supply or the sample poorly represents it. In the former case, one should use a better grade of lumber; in the latter case, adding more boards to the sample should alleviate the problem.

A good sample of previously cut boards is vital for the LP/sampling method to produce good results. The most obvious way to utilize historical data is to collect data from every board that is processed from the lumber supply in question. From a database of these boards, a simple random sample (Cochran 1977) can be drawn, with or without replacement, to represent the lumber supply.

The LP/sampling method is intended mainly for use with large cutting orders, where boards are scanned just before being cut. Large orders use a long sequence of boards, which means that the composition of the sequence reflects the lumber supply as a whole. Thus, information in a random sample from the lumber supply can be reasonably applied to the boards in the sequence. This is not true if the order is very small; the sample may represent the lumber supply but not the sequence to be cut. If it is possible to scan all boards before any are cut, then there are modifications to the standard method that may be useful even with small orders (Hamilton 2001).

#### *Geometric interpretation*

For a given knapsack problem, a solution can be calculated at any price vector. Thus,

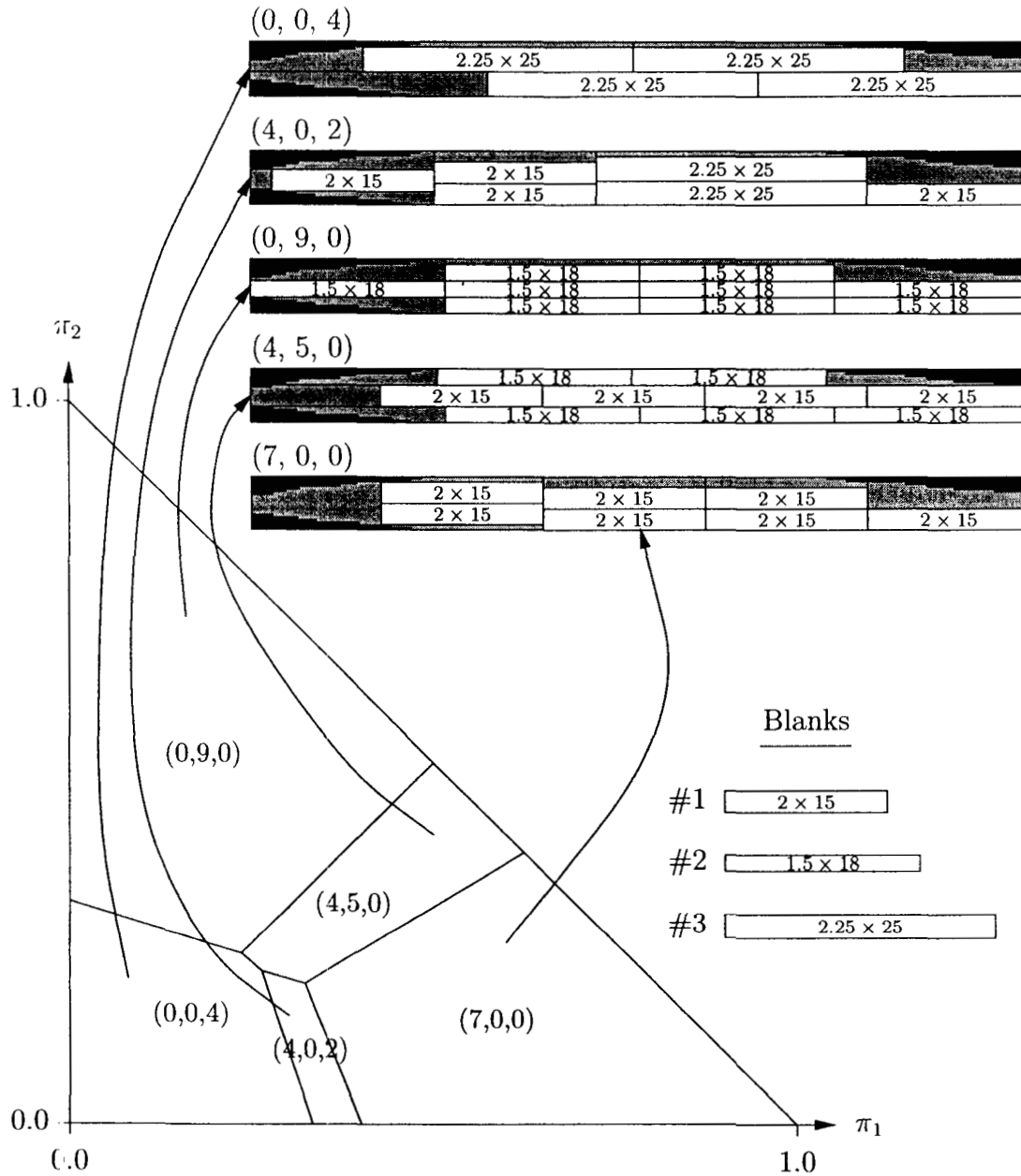


FIG. 2. Mapping price vectors to optimal cutting patterns.  $\pi_1$ ,  $\pi_2$ , and  $\pi_3 = 1 - \pi_1 - \pi_2$  are dimensionless quantities.

there is a mapping from price vectors to optimal solutions (column vectors of blank quantities) If the number of blank sizes is limited to two or three, this relationship can be diagrammed, as shown in Fig. 2. Since optimal

knapsack solutions depend only upon relative prices, the map is shown as a two-dimensional figure. The axes  $\pi_1$  and  $\pi_2$  are shown explicitly, while  $\pi_3$  is calculated as  $\pi_3 = 1 - \pi_1 - \pi_2$ . Each region is labeled by the column that

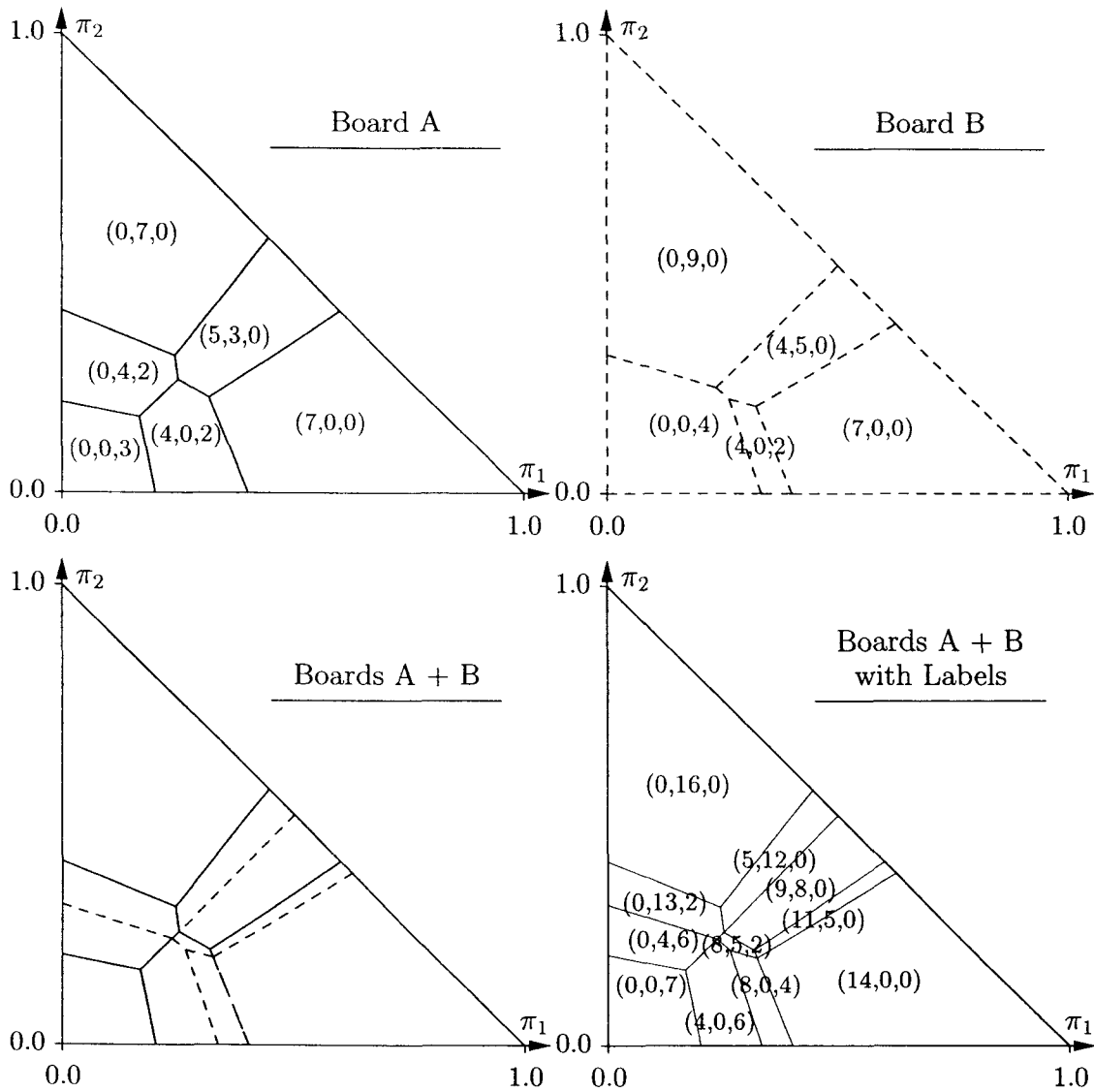


FIG. 3. Additive property of individual maps.

is generated by price vectors in that region, and the corresponding cutting pattern is identified by a connecting arrow. Note that each region is a closed convex set; this is a general property of such maps.

The knapsack solution for a set of boards is simply the sum of the knapsack solutions for the individual boards. In a similar vein, maps for individual boards can be overlaid to obtain a map of the aggregate. Figure 3 shows an

example of how the maps for two boards can be combined into a single map. The boundaries of each map appear unchanged in the combined map, and columns are added together in the obvious way. This process can be extended to any number of boards. Figure 4 maps a sample of twenty boards; the enlarged area has radius 0.005.

Suppose that we want to cut equal quantities of the three blanks, and that the only infor-



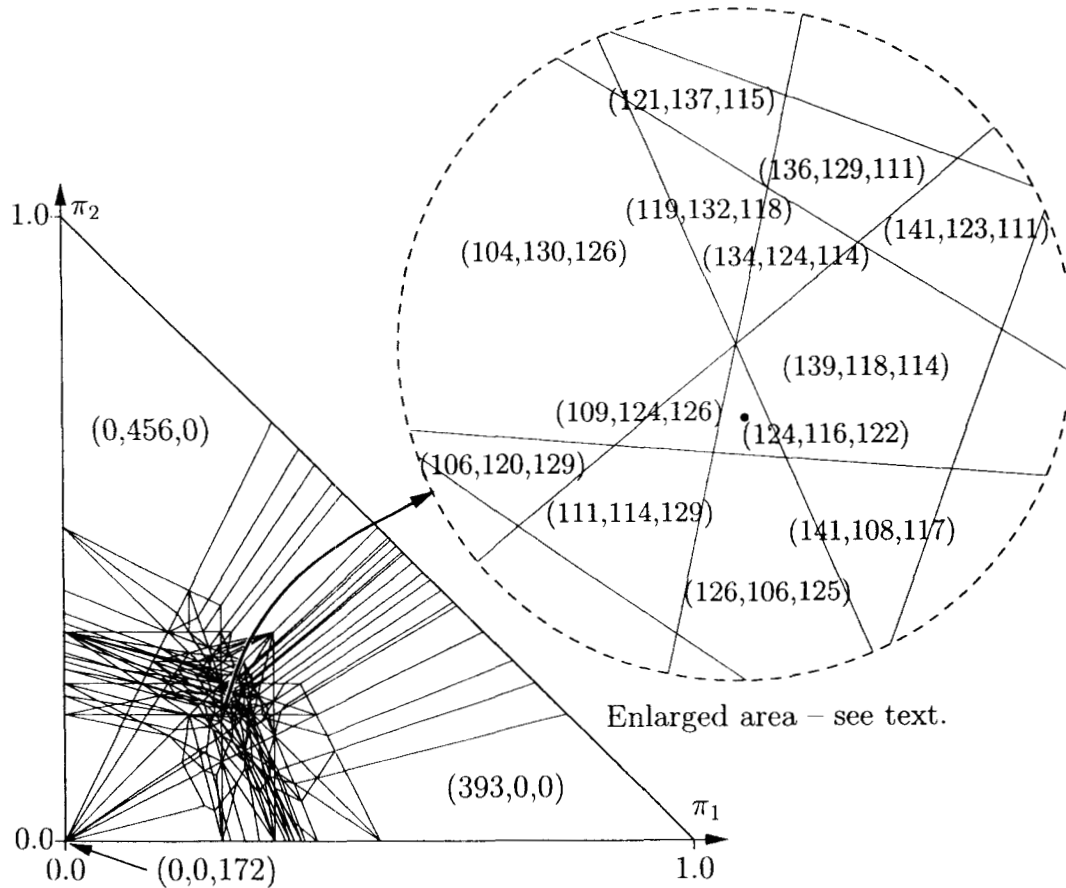


FIG. 4. Aggregate map for a sample of twenty boards.

mation we have is a very large copy of this map, with all regions labeled. We could start by finding a region in which the three blank quantities are nearly equal. The region labeled (124, 116, 122) in the enlarged area of Fig. 4 best fits our needs. The center of this region, identified by the dot, is  $(\pi_1, \pi_2, \pi_3) = (0.2555, 0.2117, 0.5329)$ . Cutting all 581 boards in the data set at this price vector produces total blank counts of (2803, 2509, 2694). These are fairly close to equal proportions, which was the goal, and even closer to the proportions in (124, 116, 122). Figure 5 diagrams this process.

The preceding example showed how a random sample can be used to estimate production quantities at a given price vector. This constitutes one half of the LP/sampling meth-

od. The other half, namely the LP, provides an efficient way for locating particular price vectors on the map. In effect, the process of building a map is the process of finding all of the columns of the LP. This process is much too slow. In contrast, the column-generating method explicitly finds only a small subset of columns. For example, in solving the problem considered above, fewer than 2% of the LP's 953 columns were generated.

#### THEORY

The LP/sampling method is supported by two theoretical results. The first gives a lower bound on the cost of filling a cutting order; the second states that the LP/sampling method comes close, in some circumstances, to

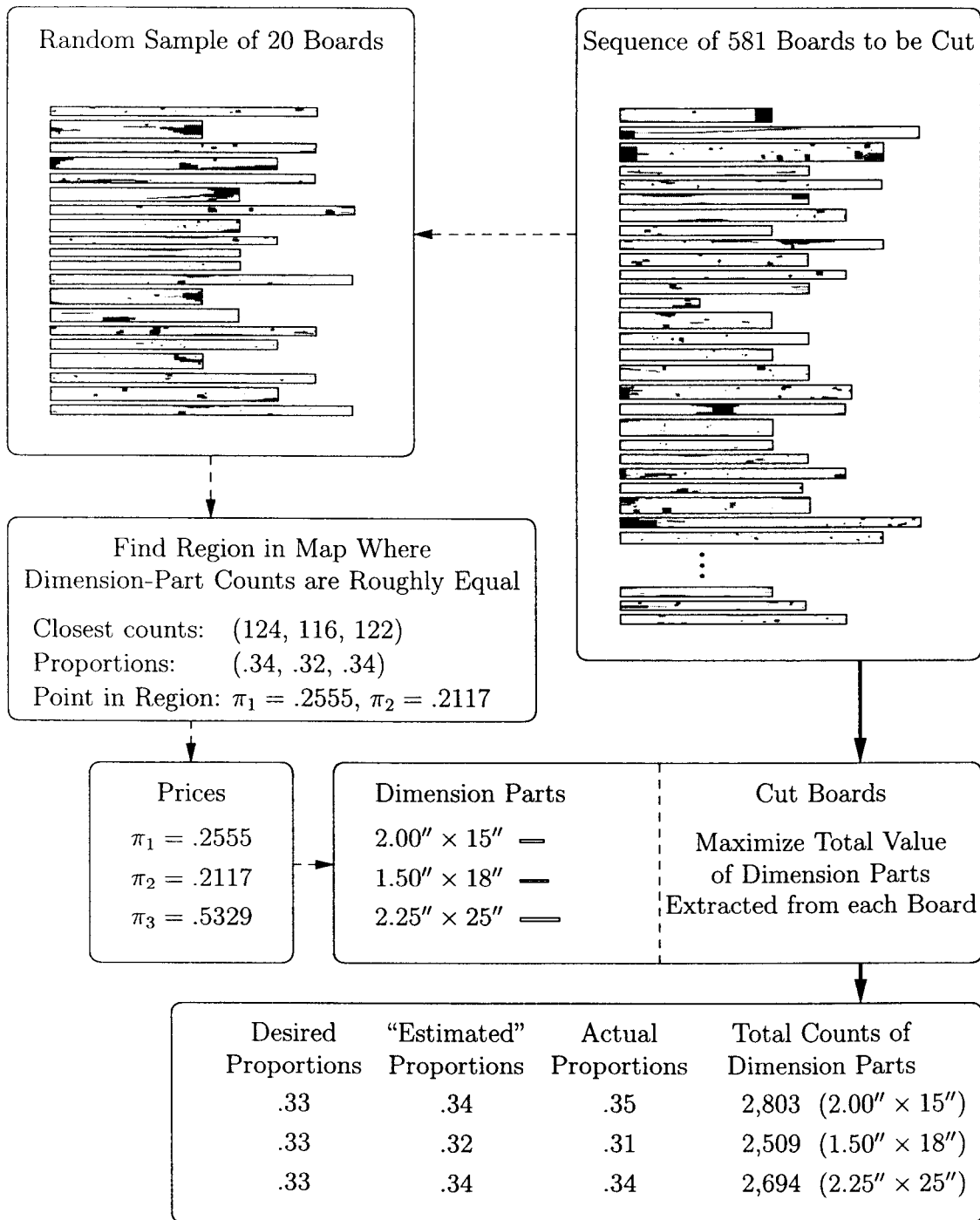


FIG. 5. Strategy to cut equal proportions of blanks.

achieving that lower bound. These results are expressed via three functions:  $f$ , the minimum cost of filling an order;  $g$ , a lower bound on the cost of filling an order; and  $h$ , the cost of filling an order using the LP/sampling method. Proofs of the results presented in this section may be found in Hamilton (2001).

In order to define  $f$ ,  $g$ , and  $h$ , it is necessary to represent in some way the boards that are used to produce blanks. In our formulation, we describe each individual board in terms of what it can produce. Suppose that the cutting order specifies quantities for  $M$  different blank sizes. A *yield vector* is defined as an  $M \times 1$  vector of nonnegative integers and represents quantities of blanks obtained from cutting up a given board in a particular way. The productive capacity of a given board is represented by the set of  $\mathcal{M}$  ordered pairs  $(\mathbf{a}, c)$ , where  $\mathbf{a}$  is a yield vector,  $c$  is its associated cost, and  $\mathcal{M}$  is the number of ways that the board can be cut into blanks. This set is called a *board type*, and it represents a group of boards that have the same cost and produce identical blank-size counts when using identical blank values, but not necessarily with identical sawing solutions.

A *board class* is defined as a finite collection of board types, and is used to model a lumber supply. A board class of  $\mathcal{N}$  board types can be expressed as the set  $A$  of all matrix pairs  $(A, C)$ , where  $A$  is an  $M \times \mathcal{N}$  matrix,  $C$  is a  $1 \times \mathcal{N}$  row vector, the  $n$ th column of  $A$  is a yield vector from the  $n$ th board type, and the  $n$ th element of  $C$  is its associated cost. The number of elements in  $A$  is  $N = \prod_{n=1}^{\mathcal{N}} \mathcal{M}_n$  (i.e., the number of combinations consisting of one yield vector from each of the  $\mathcal{N}$  board types). For convenience, we usually express  $A$  as a set of  $N$  indexed elements:

$$A = \{(A_1, C_1), \dots, (A_N, C_N)\}. \quad (2)$$

Each  $A_j$  can be viewed as a cutting specification for each board in the board class. We are assuming that the only costs are for raw material, i.e., that costs do not depend upon the cutting pattern. Thus, every yield vector for a given board type has the same cost, which im-

plies that all the  $C_j$ 's are equal. Hence, we will drop the index on  $C$  in the following.

In the cutting-order problem, boards from a given lumber supply are supplied in the form of an infinite sequence. Any sequence of boards from the lumber supply can be represented by a sequence of board types from the class. We thus represent a sequence of board types from a board class with  $\mathcal{N}$  board types by a sequence  $\{\mathbf{y}_\ell\} = \{\mathbf{y}_1, \mathbf{y}_2, \dots\}$ , where each  $\mathbf{y}_\ell$  is an  $\mathcal{N} \times 1$  vector consisting entirely of 0's, except for a single element, which is set to 1.

*The exact formulation:  $f(\{\mathbf{y}_\ell\}, \mathbf{b})$*

Suppose that  $\{\mathbf{y}_\ell\}$  represents a sequence of boards from a lumber supply. Let  $\mathbf{b} \geq \mathbf{0}$  be an  $M \times 1$  vector of order quantities, where  $b_i$  is the quantity for the  $i$ th blank, and at least one  $b_i > 0$ . It is convenient to assume no integer restrictions on  $\mathbf{b}$ , even though in reality blanks are produced only in nonnegative-integer quantities. The exact formulation of the cutting-order problem for a single board class is written as follows:

$$f(\{\mathbf{y}_\ell\}, \mathbf{b}) = \min_{L=1,2,\dots} \{f_L(\{\mathbf{y}_\ell\}, \mathbf{b})\} \quad (3)$$

where

$$f_L(\{\mathbf{y}_\ell\}, \mathbf{b}) = \min_{x_{\ell j} \in \{0,1\}} \left\{ \sum_{\ell=1}^L \sum_{j=1}^{\mathcal{N}} x_{\ell j} C_j \mathbf{y}_\ell : \sum_{\ell=1}^L \sum_{j=1}^{\mathcal{N}} x_{\ell j} A_j \mathbf{y}_\ell \geq \mathbf{b}, \sum_{j=1}^{\mathcal{N}} x_{\ell j} = 1, \ell = 1, \dots, L \right\} \quad (4)$$

and  $f_L(\{\mathbf{y}_\ell\}, \mathbf{b}) = \infty$  if there is no feasible solution.  $f_L$  represents the cost of the best solution(s) that can be obtained from the first  $L$  boards in the sequence. We also define

$$\hat{L}(\{\mathbf{y}_\ell\}, \mathbf{b}) = \min_{L=1,2,\dots} \{L: f_L(\{\mathbf{y}_\ell\}, \mathbf{b}) = f(\{\mathbf{y}_\ell\}, \mathbf{b}) < \infty\} \quad (5)$$

which is the length of the shortest initial sub-

sequence of  $\{y_\ell\}$  from which an optimal solution can be obtained, and

$$\hat{r}(\{y_\ell\}, \mathbf{b}) = \sum_{\ell=1}^{\hat{L}(\{y_\ell\}, \mathbf{b})} y_\ell \quad (6)$$

which represents the number of boards of each type used in an optimal solution of length  $\hat{L}(\{y_\ell\}, \mathbf{b})$ .

The exact formulation implicitly assumes that all board information is available before processing commences, but usually a board is not scanned until just before it is to be cut. Thus, the exact formulation may be an overly high standard for comparison.

*The lower-bound function:  $g(\mathbf{r}, \mathbf{b})$*

The lower-bound function,  $g(\mathbf{r}, \mathbf{b})$ , uses a single vector  $\mathbf{r}$  as its argument, instead of a sequence,  $\{y_\ell\}$ , of individual board types.  $\mathbf{r}$  is an  $\mathcal{N} \times 1$  nonnegative real vector and represents the relative proportions of board types within a sample from the board class.  $g(\mathbf{r}, \mathbf{b})$  is defined as

$$g(\mathbf{r}, \mathbf{b}) = \min_{\mathbf{x} \geq 0} \left\{ \sum_{j=1}^N x_j C_j \mathbf{r} : \sum_{j=1}^N x_j A_j \mathbf{r} \geq \mathbf{b} \right\}. \quad (7)$$

The matrix  $A_j$  can be interpreted as a *cutting policy*, which specifies that whenever the  $n$ th board type is encountered on the production line, it should be cut so as to produce the yield vector in the  $n$ th column of  $A_j$ . An optimal value for  $\mathbf{x}$  can then be viewed as specifying how often each policy  $A_j$  should be put into effect.

Similarities between  $g(\mathbf{r}, \mathbf{b})$  and the exact formulation become evident when  $\mathbf{r} = \hat{r}(\{y_\ell\}, \mathbf{b})$ . In that case, the respective objective functions look very similar:

$$\begin{aligned} g(\hat{r}(\{y_\ell\}, \mathbf{b}), \mathbf{b}) &= \sum_{\ell=1}^L \sum_{j=1}^N x_{\ell j} C_j y_\ell \\ f(\{y_\ell\}, \mathbf{b}) &= \sum_{\ell=1}^L \sum_{j=1}^N x_{\ell j} C_j y_\ell \end{aligned} \quad (8)$$

where  $L = \hat{L}(\{y_\ell\}, \mathbf{b})$ . The difference is in the variables. The  $\mathbf{x}$  that appears in the definition

of  $g$  is a single, nonnegative vector with no integer restrictions, whereas the  $x_\ell$ 's that appear in the definition of  $f$  are integer-valued unit vectors. It can be shown that

$$g(\hat{r}(\{y_\ell\}, \mathbf{b}), \mathbf{b}) \leq f(\{y_\ell\}, \mathbf{b}) \quad (9)$$

so that  $g(\hat{r}(\{y_\ell\}, \mathbf{b}), \mathbf{b})$  is a lower bound for  $f(\{y_\ell\}, \mathbf{b})$ .

The above result deals with a fixed board sequence,  $\{y_\ell\}$ . If we consider instead a sequence  $\{Y_\ell\}$  of random vectors and assume that  $Y_1, Y_2, \dots$  are independent and identically distributed, with  $E[Y_1] = \boldsymbol{\theta}$ , a similar lower bound holds:

$$g(\boldsymbol{\theta}, \mathbf{b}) \leq E[f(\{Y_\ell\}, \mathbf{b})]. \quad (10)$$

Moreover, this lower bound is nearly attainable for large orders:

$$\lim_{\langle \mathbf{b} \rangle \rightarrow \infty} \frac{E[f(\{Y_\ell\}, \mathbf{b})]}{g(\boldsymbol{\theta}, \mathbf{b})} = 1, \quad (11)$$

where  $\langle \mathbf{b} \rangle = \max\{b_1, b_2, \dots, b_m\}$ . Since  $g(\boldsymbol{\theta}, \mathbf{b})$  can be calculated (as the optimal value of an LP), it is a useful surrogate for the usually unknowable value of  $E[f(\{Y_\ell\}, \mathbf{b})]$ . Using  $g(\boldsymbol{\theta}, \mathbf{b})$ , we can assess the performance of any cutting-order algorithm, not just the LP/sampling method.

*The LP/sampling method:  $h(\{y_\ell\}, \mathbf{r}, \mathbf{b})$*

A mathematical definition of  $h(\{y_\ell\}, \mathbf{r}, \mathbf{b})$ , the cost of filling a cutting order using the LP/sampling method, is complex because cases involving one or more blank sizes that are given zero prices before their counts are fulfilled must be included. Due to space limitations, we omit a precise mathematical definition of  $h(\{y_\ell\}, \mathbf{r}, \mathbf{b})$  here, referring the reader who is interested in the details of the formulation to Hamilton (2001). Instead, we informally define  $h(\{y_\ell\}, \mathbf{r}, \mathbf{b})$  as the maximum possible cost to fill a cutting order under the LP/sampling method. The need to specify the "maximum possible cost" arises because  $h$  must have a unique value for every board sequence and sample, yet the LP/sampling method can produce different results, depending on how ties

are broken in the event of multiple optimal solutions in the LP's or knapsack problems.

The theoretical justification for the LP/sampling method is given in the following result:

$$\begin{aligned} & \lim_{\substack{(\mathbf{b}) \rightarrow \infty \\ n \rightarrow \infty}} \frac{E[h(\{Y_i\}, R_n, \mathbf{b})]}{g(\boldsymbol{\theta}, \mathbf{b})} \\ &= \lim_{(\mathbf{b}) \rightarrow \infty} \frac{E[f(\{Y_i\}, \mathbf{b})]}{g(\boldsymbol{\theta}, \mathbf{b})} \\ &= \lim_{\substack{(\mathbf{b}) \rightarrow \infty \\ n \rightarrow \infty}} \frac{E[h(\{Y_i\}, R_n, \mathbf{b})]}{E[f(\{Y_i\}, \mathbf{b})]} = \mathbf{1}, \quad (12) \end{aligned}$$

where  $R_n$  is the proportion of boards of each type in a sample of size  $n$  and  $\lim_{n \rightarrow \infty} R_n = \boldsymbol{\theta}$  with probability 1. The last condition simply says that the sample used in the LP/sampling method should adequately represent the board class as a whole. This condition would be satisfied by a simple random sample of sufficient size.

The left-hand limit above says that the expected cost under the LP/sampling method divided by the lower bound tends towards 1. The middle limit follows easily when the lower bound holds, for in that case we have  $g(\boldsymbol{\theta}, \mathbf{b}) \leq E[f(\{Y_i\}, \mathbf{b})] \leq E[h(\{Y_i\}, R_n, \mathbf{b})]$ . In fact, the limits hold under more general conditions in which  $g(\boldsymbol{\theta}, \mathbf{b})$  is not necessarily a lower bound.

The right-hand limit is the main result of this paper. It says that for large order quantities and large sample sizes, the expected cost obtained under the LP/sampling method deviates by only a small percentage from the true minimum. In this context, we say that costs under the LP/sampling method are approximately optimal. As is usual for asymptotic results, however, it doesn't say how large the order quantities and sample sizes must be to obtain a reasonable approximation. Furthermore, since there is no practical way to determine the value of  $f(\{y_i\}, \mathbf{b})$  for all but the simplest cases, direct comparisons between  $h$  and  $f$  are usually impossible.

Fortunately, the lower-bound result provides a way to make an indirect comparison. When

the parameters of the lumber supply are known, as they are in the controlled environment of a computer simulation, the value of  $g(\boldsymbol{\theta}, \mathbf{b})$  can be calculated (by applying the method described in the Section *The LP Sampling Method* to the entire board data set, not just a sample). The value  $E[h(\{Y_i\}, R_n, \mathbf{b})]$  can then be estimated by averaging the observed costs for the LP/sampling method for a number of random sequences and samples. To the extent  $E[h(\{Y_i\}, R_n, \mathbf{b})]$  is close to  $g(\boldsymbol{\theta}, \mathbf{b})$ , the lower bound, it follows that  $E[h(\{Y_i\}, R_n, \mathbf{b})]$  is close to  $E[f(\{Y_i\}, \mathbf{b})]$ , the "gold standard." Since  $g(\boldsymbol{\theta}, \mathbf{b})$  is a proportionally tight bound in larger orders, it is a practical evaluation tool, as will become evident in the simulation results of the next section.

#### SIMULATION RESULTS

We can use simulation to examine how well the LP/sampling method performs, employing the lower bound,  $g(\boldsymbol{\theta}, \mathbf{b})$ , as our standard of comparison, since to be near the lower bound is to be near the expected minimum cost of filling a given cutting order. The results that follow also compare the LP/sampling method with the previously described CDE algorithm, since that algorithm is easy to implement.

The computer simulations presented here utilize lumber data that originated with the U.S. Department of Agriculture's Forest Products Laboratory in Madison, Wisconsin (McDonald et al. 1981, 1983). The data represent two thicknesses of No. 1 Shop grade ponderosa pine. Each board is defined by its length and width and a list of defects, all at a resolution of 0.25 in. (6.35 mm). The original data differentiate defects according to type (sound knot, unsound knot, etc.) and identify the face on which they appear. In the simulations, however, all defects are treated as being equally bad, and it is assumed that blanks must be C2F. All boards are assumed to have equal cost per unit area. A summary of the data is given in Table 2.

The simulations used a two-stage, rip-first cutting process in which there were no limits

TABLE 2. Summary of ponderosa pine data.

Data set	5/4	6/4
Nominal thickness in inches (mm)	1.25 (32)	1.50 (38)
Total board area in ft <sup>2</sup> (m <sup>2</sup> )	6,940.8 (644.82)	2,035.8 (189.13)
Avg. area per board in ft <sup>2</sup> (m <sup>2</sup> )	18.1 (1.68)	19.0 (1.77)
Number of boards	384	107
Avg. clear area per board (%)	88.9	90.4

on the number of cuts per stage. The knapsack problems were solved using a dynamic programming algorithm that is loosely based on Hahn's algorithm (Hahn 1968). Because this

algorithm makes no allowance for kerf, it is somewhat unrealistic.

The cutting order used in the simulations was obtained by Carino and Foronda (1990) from an Alabama cabinet manufacturer. It is shown in Table 3. Although the order was intended to be cut from a hardwood species, we applied it to the ponderosa pine data. The order was cut from a random sequence of boards drawn, with replacement, from the data for 5/4 (32-mm-thick) lumber. The random samples of boards required by the LP/sampling method were selected from the 6/4 (38-mm-thick) lumber.

For this order, the lower bound, calculated from the 5/4 data and valuing boards arbitrarily at \$1 per square foot, was \$5,649. This corresponds to an area yield of about 80%. Since yield and cost are inversely related, 80% is an approximate upper bound on the expected yield.

Results for several different sample sizes are displayed as box plots (Ramsey and Schaffer 1997) in Fig. 6. Each point in a given box plot represents the cost of filling the order from a random sequence of boards, using a random sample of boards to calculate prices for the optimizer. Because the lower bound function is a bound on an expected or average value, it is possible for individual points to fall below it, as was the case for  $n = 8$ . The box's centerline represents the mean, and the top and bottom represent the upper- and lower-quartile boundaries, respectively. The simulations for each sample size were repeated 24 times, with a new sample and a new board sequence being drawn each time. In the figure, the vertical axis indicates the percentage by which the ob-

TABLE 3. Carino and Foronda's cutting order.

ID	Width in inches	(mm)	Length in inches	(m)	Quantity
1	1.75	(44.5)	10	(0.254)	303
2	1.75	(44.5)	13	(0.330)	931
3	1.75	(44.5)	16	(0.406)	582
4	1.75	(44.5)	18.5	(0.470)	260
5	1.75	(44.5)	19	(0.483)	136
6	1.75	(44.5)	21.5	(0.546)	109
7	1.75	(44.5)	22	(0.559)	428
8	1.75	(44.5)	24	(0.610)	622
9	1.75	(44.5)	25	(0.635)	64
10	1.75	(44.5)	28	(0.711)	926
11	1.75	(44.5)	30	(0.762)	170
12	1.75	(44.5)	34	(0.864)	118
13	1.75	(44.5)	37	(0.940)	128
14	1.75	(44.5)	43	(1.092)	12
15	1.75	(44.5)	46	(1.168)	133
16	1.75	(44.5)	79.5	(2.019)	36
17	1.75	(44.5)	91.5	(2.324)	15
18	1.8125	(46.0)	16.5	(0.419)	198
19	1.8125	(46.0)	18.5	(0.470)	455
20	2.1875	(55.6)	10	(0.254)	576
21	2.1875	(55.6)	13	(0.330)	234
22	2.1875	(55.6)	16	(0.406)	290
23	2.1875	(55.6)	22	(0.559)	1,133
24	2.1875	(55.6)	26.75	(0.679)	360
25	2.1875	(55.6)	28	(0.711)	1,255
26	2.25	(57.2)	30	(0.762)	120
27	3.5	(88.9)	6	(0.152)	1,171
28	3.5	(88.9)	16	(0.406)	445
29	3.5	(88.9)	22	(0.559)	273
30	3.5	(88.9)	28	(0.711)	1,124
31	5.375	(136.5)	13	(0.330)	173
32	5.375	(136.5)	14.5	(0.368)	145
33	5.375	(136.5)	16	(0.406)	166
34	5.375	(136.5)	19	(0.483)	249
35	5.375	(136.5)	22	(0.559)	22

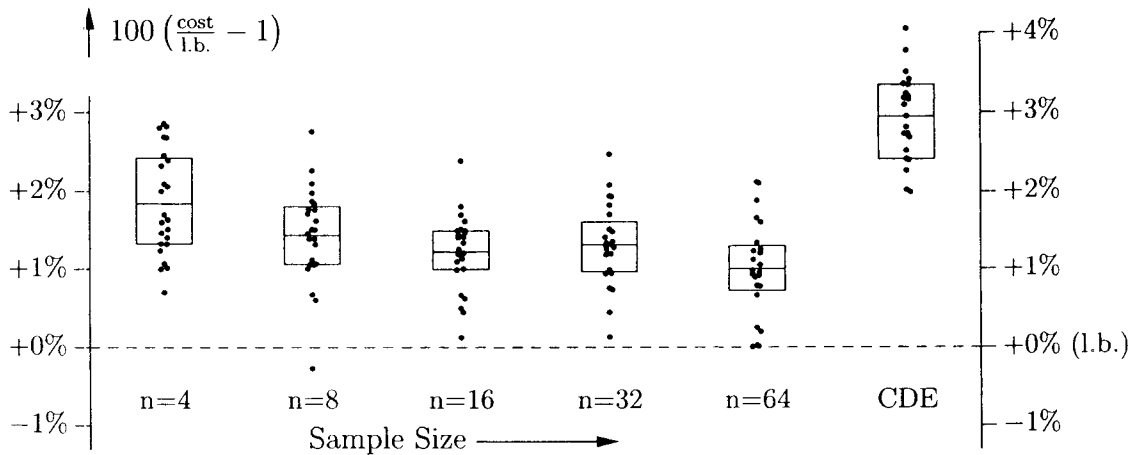


FIG. 6. Cost vs. sample size for Carino and Foronda's cutting order.

served cost exceeds the lower bound. The trend shows average costs declining as sample size increases, leveling off somewhere around 1% or so above the lower bound. It is easy to confuse the lower bound with the true minimum. But the true minimum is almost certainly higher than the lower bound, which would imply that the observed costs are slightly closer to the true minimum than the graphs suggest.

Figure 6 also shows results for CDE. Compared to the LP/sampling method, CDE yields a consistently higher cost, though the average difference in all cases is less than 2%.

A statistical analysis of the data shown in Fig. 6 was done by conducting pairwise *t*-tests on the means. The mean cost of filling the order using CDE was significantly different than any of the LP/sampling means, with all *p*-values < 0.001. Similarly the mean of the LP/sampling method for  $n = 4$  was significantly different than for any other value of  $n$ , with *p*-values ranging from < 0.001 ( $n = 64$ ) to 0.035 ( $n = 8$ ). The only other statistically significant difference in means was between the LP/sampling method for  $n = 8$  and  $n = 64$ , with a *p*-value of 0.017.

It must be emphasized that the scope of inference of the preceding statistical analysis is very narrow. It is limited to a single cutting order and two small lumber data sets. The LP/

sampling method outperforms CDE for these data, but may not do so for other cutting orders or lumber having different characteristics. Further research is underway that investigates the effects of cutting order and lumber characteristics on the performance of the LP/sampling method.

Because of space limitations, only a few simulation results are presented here. Further results, including an examination of the effects of order size (area and number of blanks), sawing configuration (two-stage, three-stage, rip-first, etc.), species, random price deviations, nonindependent board sequences, and the use of heuristics to solve the knapsack problems, can be found in Hamilton (2001).

#### CONCLUSIONS

This paper has established a lower bound on the expected cost of filling a cutting order from an independent sequence of boards. No matter what method is used to fill the cutting order, the expected cost can never be less than the lower bound, and thus the lower bound provides an absolute standard against which any method can be evaluated. The paper has also described a method for filling a cutting order (the LP/sampling method) that in computer simulations produces costs that are consistently close to the lower bound when the

cutting order is large. The two results reinforce each other. On one hand, the lower bound is used to demonstrate that the LP/sampling method can do a good job of minimizing costs. On the other hand, the LP/sampling method shows that the lower bound is often nearly attainable and is therefore a useful benchmark.

Both computing the lower bound and carrying out the LP/sampling method require substantial amounts of computation. In practice, it is probably not feasible to calculate new prices before cutting each board. Perhaps the best approach is to make initial price calculations prior to cutting, then to update them as often as possible. Boards are then cut at whatever prices are in effect until they are updated again. Thus, it is not a matter of whether or not the LP/sampling method is possible to implement, but a matter of how frequently prices can be updated and the impact on costs of not updating them for each new board. Fortunately, the computational difficulties are not as bad as they first seem. In principle, prices should not change much in a large order until near the end of processing. Also, there is some evidence (Hamilton 2001) that prices can wander a bit from the “correct” values without greatly affecting costs. Thus, there is reason to believe that the LP/sampling method can be successfully adapted to actual use.

Further research should include more realistic simulations, where boards must be processed every few seconds without waiting for new prices to be calculated. Another avenue needing more research is the handling of heuristics in the auxiliary knapsack problems. When knapsack problems are solved only approximately, the LP can terminate prematurely without generating a good set of prices. This problem corrects itself somewhat during the process of filling the order, but it is still preferable to have better solutions early in the process.

#### REFERENCES

- ANDERSON, J. D., C. C. BRUNNER, AND A. G. MARISTANY. 1992. Effect of sawing stages on fixed-width, fixed-

length dimension yields. *Forest Prod. J.* 42(11/12):74–78.

———, ———, AND S. U. RANDHAWA. 1997. Design and implementation of a fuzzy logic controller for wood parts recovery in rough mills. *Int. J. Flexible Autom. Int. Mfg.* 4(3–4):255–271.

BRUNNER, C. C., M. S. WHITE, F. M. LAMB, AND J. G. SCHROEDER. 1989. CORY: A computer program for determining dimension stock yields. *Forest Prod. J.* 39(2): 23–24.

CARINO, H. F., AND S. U. FORONDA. 1990. SELECT: A model for minimizing blank costs in hardwood furniture manufacturing. *Forest Prod. J.* 40(5):21–26.

CARNIERI, C., G. A. MENDOZA, AND L. GAVINHO. 1994. Optimal cutting of lumber and particleboards into dimension parts: Some algorithms and solution procedures. *Wood Fiber Sci.* 26(1):131–141.

COCHRAN, W. G. 1977. *Sampling techniques*. John Wiley & Sons, New York, NY.

DMITROVIC, B., D. WOLF, AND R. H. C. HUSSON. 1992. A standard model of the placement problem aimed at resolving a cutting stock problem. *IEEE Trans. Industry Appl.* 28(6):1406–1415.

FORONDA, S. U., AND H. F. CARINO. 1991. A heuristic approach to the lumber allocation problem in hardwood dimension and furniture manufacturing. *European J. Operational Res.* 54:151–162.

GATCHELL, C. J., R. B. ANDERSON, AND P. A. ARAMAN. 1983. Effect of gang-ripping width on CIF yields from No. 2 Common oak lumber. *Forest Prod. J.* 33(6):43–48.

GIESE, P. J., AND K. A. McDONALD. 1982. OPTYLD—A multiple rip-first computer program to maximize cutting yields. Research Paper FPL-412. USDA Forest Serv., Forest Products Lab, Madison, WI.

———, AND J. D. DANIELSON. 1983. CROMAX—A crosscut-first computer simulation program to determine cutting yield. Gen. Tech. Rep. FPL-38. USDA Forest Serv., Forest Products Lab, Madison, WI.

GILMORE, P. C., AND R. E. GOMORY. 1961. A linear programming approach to the cutting stock problem. *Operations Res.* 9:849–859.

———, AND ———. 1963. A linear programming approach to the cutting stock problem—Part II. *Operations Res.* 11:863–888.

HAHN, S. G. 1968. On the optimal cutting of defective sheets. *Operations Res.* 16(6):1100–1114.

HALLOCK, H. 1980. Cutting yields from standard hardwood lumber grades when gang ripping. Research Paper FPL-370. USDA Forest Serv., Forest Products Lab, Madison, WI.

HAMILTON, E. D. 2001. A linear programming and sampling approach to the cutting-order problem. Ph.D. thesis, Oregon State University, Corvallis, OR.

HARDING, O. V., AND P. H. STEELE. 1997. RIP-X: Decision software to compare crosscut-first and rip-first



- rough mill systems. *Wood Sci. Technol.* 31(5):367–381.
- KLINKHACHORN, P., J. P. FRANKLIN, C. W. McMILLIN, AND H. A. HUBER. 1989. ALPS: Yield optimization cutting program. *Forest Prod. J.* 39(3):53–56.
- MARTENS, D. G., AND R. L. NEVEL, JR. 1985. OPTIGRAMI: Optimum lumber grade mix program for hardwood dimension parts. Tech. Rep. NE-563. USDA Forest Service, Northeastern Forest Experiment Station, Princeton, WV.
- MCDONALD, K. A., P. J. GIESE, AND R. O. WOODFIN. 1981. 5/4 ponderosa pine shop cutting yields. Research Paper FPL-394. USDA Forest Service.
- , ———, AND ———. 1983. Maximum cutting yields for 6/4 ponderosa pine shop lumber. Research Paper FPL-437. USDA Forest Service, Forest Products Lab, Madison, WI.
- MURTY, K. G. 1976. *Linear and combinatorial programming*. Wiley, New York, NY.
- RAMSEY, F. L., AND D. W. SCHAFER. 1997. *The statistical sleuth: A course in methods of data analysis*. Duxbury Press, Pacific Grove, CA.
- SUTER, JR., W. C., AND J. A. CALLOWAY. 1994. Rough mill policies and practices examined by a multiple-criteria goal program called ROMGOP. *Forest Prod. J.* 44(10):19–28.
- THOMAS, R. E. 1996. Prioritizing parts from cutting bills when gang-ripping first. *Forest Prod. J.* 46(10):61–66.
- . 1997. ROMI-CROSS: ROugh Mill CROSScut-first simulator. Tech. Rep. NE-229. USDA Forest Service, Northeastern Forest Experiment Station, Princeton, WV.
- . 1999. ROMI-RIP Ver. 2: ROugh Mill RIP-first simulator. Tech. Rep. NE-259. USDA Forest Service, Northeastern Forest Experiment Station, Princeton, WV.
- WODZINSKI, C., AND E. HAHM. 1966. A computer program to determine yields of lumber. Tech. Rep. 66-009. USDA Forest Service, Forest Products Lab, Madison, WI.