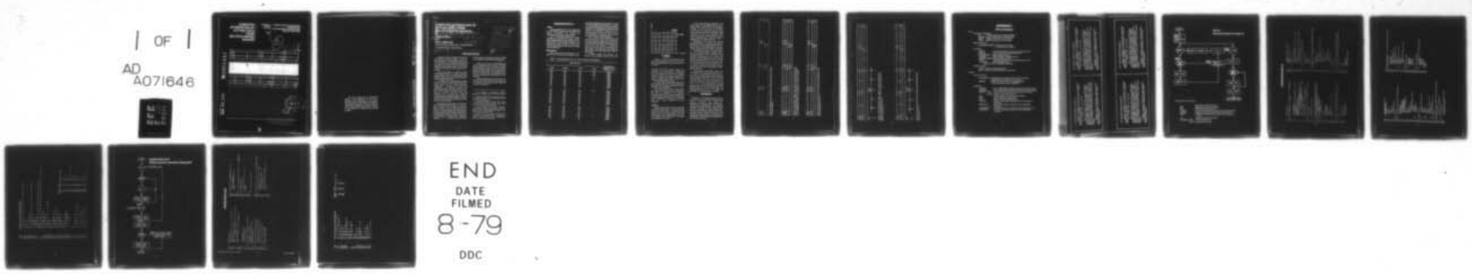
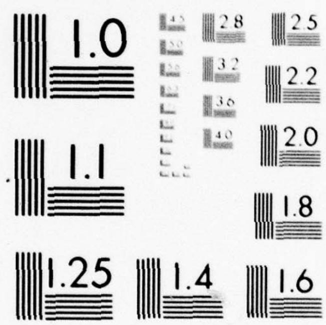
AD-A071 646 FOREST PRODUCTS LAB MADISON WIS F/G 11/12
COMPUTER OPTIMIZATION OF CUTTING YIELD FROM MULTIPLE-RIPPED BOA--ETC(U)
1978 A R STERN, K A MCDONALD
UNCLASSIFIED FSRP-FPL-318 NL

| OF |

AD
A071646



END
DATE
FILMED
8-79
DDC



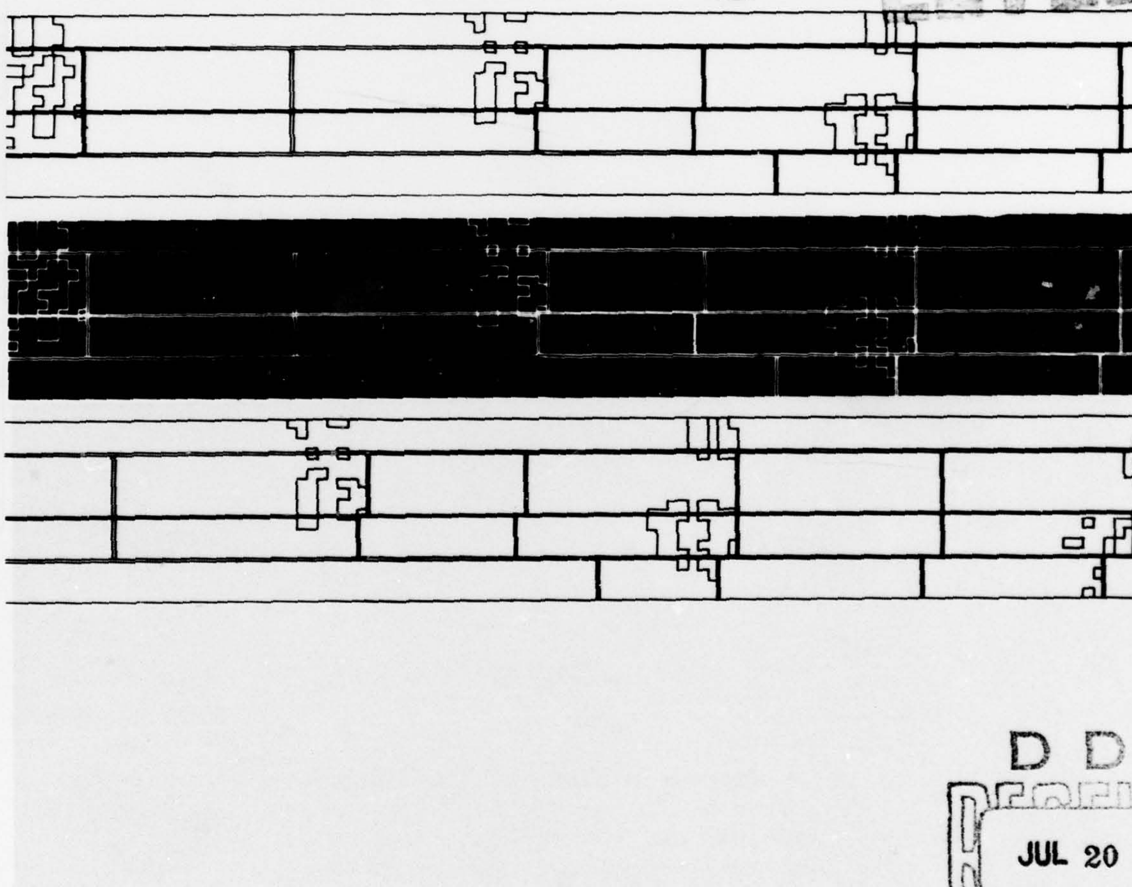
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

COMPUTER OPTIMIZATION OF CUTTING YIELD FROM MULTIPLE-RIPPED BOARDS

RESEARCH PAPER FPL 318

1978

FOREST PRODUCTS LABORATORY
FOREST SERVICE
U.S. DEPARTMENT OF AGRICULTURE
MADISON, WISCONSIN



(6) (14) FS RP-FPL-318

COMPUTER OPTIMIZATION OF CUTTING YIELD FROM MULTIPLE-RIPPED BOARDS

By

(10) Abigail R. Stern, Kent A. McDonald

and

Kent A. McDonald (12) 17p.

Forest Products Laboratory,[1] Forest Service

U.S. Department of Agriculture

(9) Forest Service research paper

INTRODUCTION

Multiple ripping of boards, followed by crosscutting to remove defects, is an operation used by both the hardwood flooring and the softwood cut-up industries. Because of the rising cost of lumber and the increasing demand on the timber supply, utilizing each board more efficiently is becoming more important.

The two steps in making better processing decisions to improve utilization of each board are to: (1) automatically locate defects, and (2) optimize sawline placement based on defect locations.

A system that automatically locates defects in lumber has been developed and is being tested at the Forest Products Laboratory [2]. Boards are scanned with ultrasound under computer control and defect location data are automatically collected. The computer program used was designed to: (1) control the scanning process, (2) store collected data on tape, (3) optimize sawline placement based on defect locations, and (4) draw the board and cutting solution on a line plotter.

The purpose of this paper is to describe RIPYLD (RIP YieLD)--that part of the computer program that optimizes sawline placement for maximum yield. RIPYLD obtains the multiple ripping and crosscutting solutions using defect location data, and is an expansion of earlier efforts to maximize cutting yields of boards using computer analyses [3], [4], [5]. In RIPYLD, any kerf width can be used and cuttings can be any length (either random or specified), and any width.

(11) 1978

RIPYLD has the option of manufacturing either specified length cuttings or random length cuttings. Up to five cutting lengths and three cutting widths can be used in the specified length option. If the random length option is chosen, three cutting widths and minimum acceptable cutting length must be specified.

Sawing variables are the maximum number of rip saws to be used on any board, and the sawkerf, which will be used in both the rip cuts and crosscuts.

1/ The Laboratory is maintained in Madison, Wisconsin, in cooperation with the University of Wisconsin.

2/ McDonald, Kent A. 1978. Lumber defect detection by ultrasonics. USDA For. Serv. Res. Pap. FPL 311. For. Prod. Lab., Madison, Wis.

3/ Wodzinski, Claudia, and Eldona Hahm. 1966. A computer program to determine yields of lumber. USDA For. Serv., For. Prod. Lab., Madison, Wis.

4/ Erickson, Bernard J., and Donald C. Markstrom. 1972. Predicting softwood cutting yields by computer. USDA For. Serv. Res. Pap. RM-98. Rocky Mountain For. Range. Exp. Sta., Fort Collins, Colo.

5/ Cornwell, Larry W., and John K. Kalita. 1977. The development of a computer program to automate the cutting of gunstock blanks. Dept. of Mathematics, Western Illinois University, Macomb, Ill.

141 700

PROGRAM RIPYLD

Input

Input parameters that must be specified for the RIPYLD program are: (a) board and defect information, (b) cutting bill requirements, and (c) sawing variables.

An X-Y coordinate system grid is superimposed on the board, and each unit grid area is designated as either defective (1) or clear (0) (fig. 1). The number of X-grids in the length, the number of Y-grids in the width, and the sizes of X-grid and Y-grid (in inches) must be specified.

Description

First, all possible combinations of rip widths that will fit within the width of the board are determined and stored. For example, if the possible rip widths are 2, 2.5, and 3 inches and there are four rip saws available, there are 3^4, 81 possible permutations of rip widths to try. However, if the board is 9 inches wide and the kerf is 0.125 inches, only 27 permutations, including kerfs, will fit within the width of the board (table 1).

Then, for each stored combination of rip widths, the board is "sawn" by the computer. The board is always ripped first, with the first rip width always positioned at the edge of the board with the lowest Y coordinate. Solutions with the first rip positioned at the other edge of the board are not considered. After ripping, the clear areas within each rip are located.

If random lengths are desired, only defects and lengths shorter than the

Table 1. -- Rip combinations of 2.0", 2.5", and 3.0" that fit in 9" wide board

Rip widths (in.)				
1st Rip	2nd Rip	3rd Rip	4th Rip	Total width (in.) (including 0.125" kerf between rips)
2.0	2.0	2.0	2.0	8.375
2.0	2.0	2.0	2.5	8.875
2.0	2.0	2.5	2.0	8.875
2.0	2.0	3.0	--	7.250
2.0	2.5	2.0	2.0	8.875
2.0	2.5	2.5	--	7.250
2.0	2.5	3.0	--	7.750
2.0	3.0	2.0	--	7.250
2.0	3.0	2.5	--	7.750
2.0	3.0	3.0	--	8.250
2.5	2.0	2.0	2.0	8.875
2.5	2.0	2.5	--	7.250
2.5	2.0	3.0	--	7.750
2.5	2.5	2.0	--	7.250
2.5	2.5	2.5	--	7.750
2.5	2.5	3.0	--	8.250
2.5	3.0	2.0	--	7.750
2.5	3.0	2.5	--	8.250
2.5	3.0	3.0	--	8.750
3.0	2.0	2.0	--	7.250
3.0	2.0	2.5	--	7.750
3.0	2.0	3.0	--	8.250
3.0	2.5	2.0	--	7.750
3.0	2.5	2.5	--	8.250
3.0	2.5	3.0	--	8.750
3.0	3.0	2.0	--	8.250
3.0	3.0	2.5	--	8.750

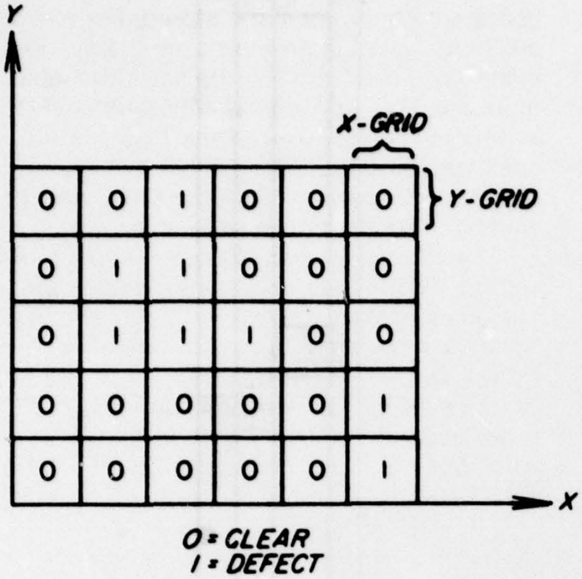


Figure 1.--In the X-Y coordinate system grid superimposed on the board, each unit grid area is designated as either defective (1) or clear (0).

specified minimum cutting length are removed by crosscutting. Otherwise, specified lengths are made by crosscutting the clear areas and removing the defects. Longest cuttings are always salvaged first even if a higher yield would result from a combination of shorter cuttings.

For each clear cutting found and cut out, surface area of the cutting is calculated. Surface areas of cuttings are summed to obtain the total yield of the board.

After total yield of clear cuttings from the board for a rip combination is calculated, the yield is compared to the previous maximum yield. If the new yield is greater, it is stored as the new maximum. The new yield is also compared to the previous minimum yield and, if less, becomes the new minimum.

Output

Output from RIPYLD contains complete information about both the maximum and minimum yield solutions. Included are the percent yield of clear cuttings from the board, the rip width combination, the crosscut locations, and a piece tally if the specified length option is used.

At the Forest Products Laboratory, the same computer (Harris 6024) that is used to collect defect information from the Defectoscope [2] is used to control a line plotter. The minimum or maximum solution is plotted, including the outline of the board, defect locations, rip cuts, and crosscuts. Alternatively, the output could be directed to computer controlled saws, stored on tape, or displayed on a TV screen or printer.

Examples of the plots with RIPYLD solutions are shown in figures 2 through 6. A 90-inch long, 9-inch wide board with the defects found by the Defectoscope, was outlined on a data grid 0.5 inch by 0.5 inch (fig.2).

The board was "sawn" with a 0.125-inch kerf, into random-length cuttings with a minimum length of 10 inches. RIPYLD chose between rip widths of 2", 2.5", and 3". The optimum yield of 80.84 percent was achieved with a rip combination of 2", 2.5", 2", 2" (fig. 3). The minimum solution with a 65.73 percent yield was from a rip combination of 2", 2", 3" (fig. 4). There was not enough room for another rip at the top of the board, so 1.625" was not utilized.

The same board was again "sawn" with a 0.125-inch kerf and combinations of 2", 2.5", and 3" rip widths (figs. 5, 6). However, here the specified length option was used with a choice of 50", 40", 30", 20", and 10" cuttings. Piece tallies are included on the plots. The optimum solution (fig. 5) was a 2.5", 2", 2", 2" rip combination with 54.80 percent yield. The minimum solution of 44.94 percent yield (fig. 6) was found with a rip width combination of 2", 2", 3". Again, the top 1.625" of the board was not utilized.

SUMMARY

RIPYLD is a computer program that optimizes the cutting yield from multiple-ripped boards. Decisions are based on automatically collected defect information, cutting bill requirements, and sawing variables. The yield of clear cuttings from a board is calculated for every possible permutation of specified rip widths and both the maximum and minimum percent yield solutions are saved. Solutions include rip cut and crosscut locations as well as the percent yield of clear cuttings.

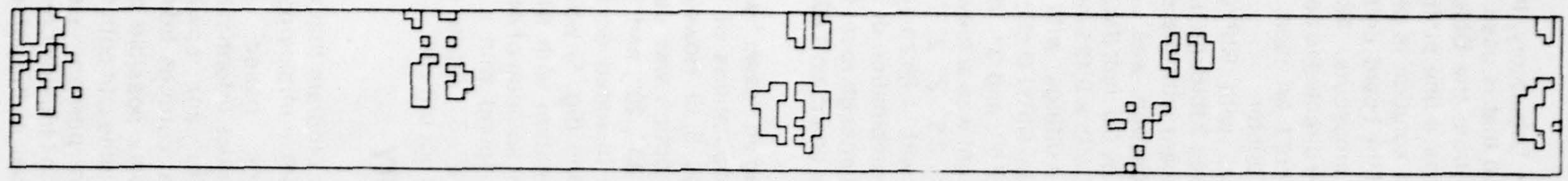


Figure 2.–Defects were outlined by the Defectoscope on a grid of 0.5 inch by 0.5 inch.

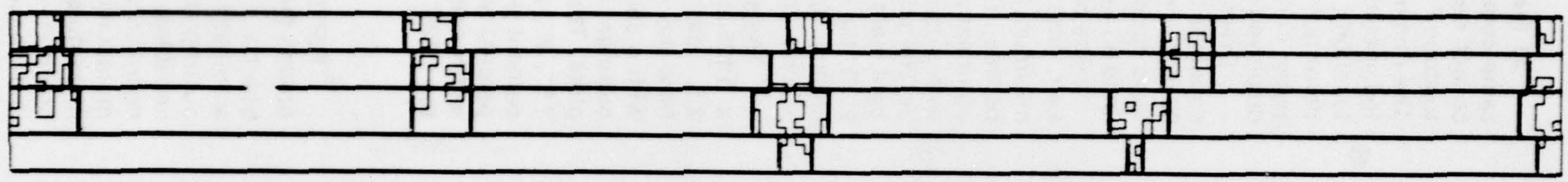


Figure 3.–The board in figure 2 was "sawn" by the computer, into random-length cuttings, 10-inch minimum. This optimum yield of 81% was achieved with a rip combination of 2", 2.5", 2", and 2".

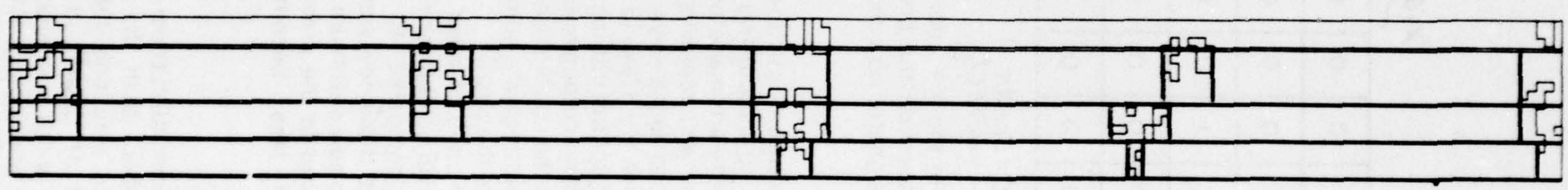
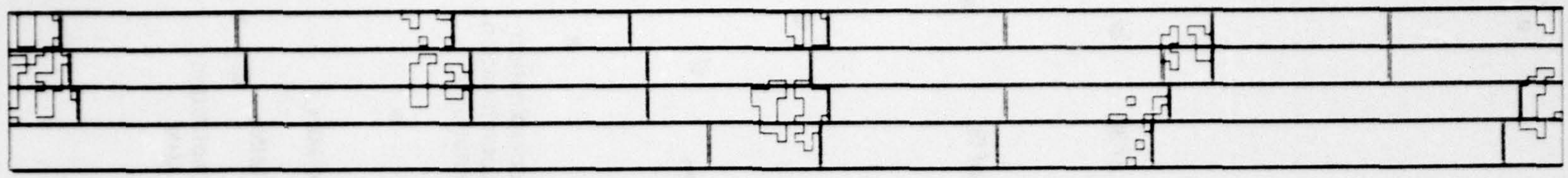
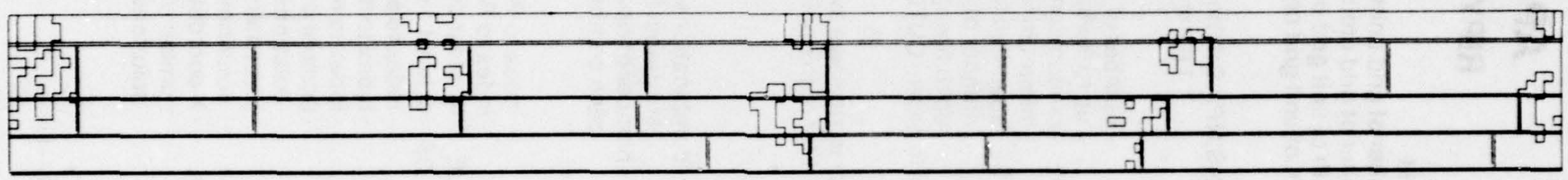


Figure 4.–Minimum solution for the same board yielded 65%, with a rip combination of 2", 2", 3".



WIDTH	LENGTH 50"	40"	30"	20"	10"
2	0	0	0	2	10
2.5	0	1	0	1	1
3	0	0	0	0	0

Figure 5.--The same board "sawn" again, this time with specified-length cuttings ranging from 50" to 10", yielded an optimum of 55% from rips of 2.5", 2", 2", 2".



WIDTH	LENGTH 50"	40"	30"	20"	10"
2	0	1	0	2	4
2.5	0	0	0	0	0
3	0	0	0	0	4

Figure 6.--The minimum solution for the same conditions was 45% from rips of 2", 2", 3".

APPENDIX I

RIPYLD Variables

Input

Dimensions of data grid

NP - number of grid units in the board length
NSCANS - number of grid units in the board width
XGRID - length of unit grid on X axis (inches)
YGRID - width of unit grid on Y axis (inches)

Defect information

BOARD(NSCANS,NP) = 0 if the grid unit is clear
= 1 if the grid unit is a defect

Cutting bill

NWIDTH - number of rip widths to choose from (maximum of 3)
WIDTH(3) - up to 3 widths can be specified (inches)
RANDOM=TRUE - random length cuttings
SAWMIN - minimum length acceptable cutting
RANDOM=FALSE - specified length cuttings
NLEN - number of specified cutting lengths (maximum of 5)
CUTLEN(5) - up to 5 lengths (inches)
(Must be in order: CUTLEN(1)-maximum)

Sawing variables

NSAW - number of rip saws available
KERF - sawkerf for both ripping and crosscutting

Output

Rip combinations

RIPCOM(81,4) - combinations of rip widths that fit in the board width.
(Maximum 81 combinations, 4 rip saws)
NRIP(81) - number of rips that will fit in the board width for
each combination stored in RIPCOM

Solutions

MINCOM - index to RIPCOM and NRIP of the lowest yield combination
MAXCOM - index to RIPCOM and NRIP of the highest yield combination.
REJECT = TRUE - no clear cuttings can be found for any rip combination.
= FALSE - at least 1 clear cutting is found
ACT - index to solution of current rip combination
MAX - index to maximum yield solution
MIN - index to minimum yield solution
YIELD(3) - percent of clear area of the board for ACT, MAX, MIN
solutions
PIECE(5,3,3) - piece tally (5 lengths, 3 widths,) for ACT, MAX, MIN
solutions
CROSS(150,3) - X-coordinates of crosscuts for ACT, MAX, MIN solutions
NXCUT(4,3) number of crosscuts in each rip for ACT, MAX, MIN
solutions

LD

U.S. Forest Products Laboratory.

Computer optimization of cutting yield from multiple-ripped boards, by Abigail R. Stern and Kent A. McDonald. Madison, Wis., FPL, 1978.
13 p. (USDA For. Res. Pap. FPL 318).

RIPYLD (RIP YieLD) is the part of a lumber-defect-locating computer program that is designed to optimize sawline placement for maximum cutting yield. RIPYLD obtains multiple ripping and crosscutting solutions using defect locating data, and is an expansion of earlier efforts to maximize cutting yields of boards using computer analysis.

KEYWORDS: Computer optimization; lumber processing, yield, automation, sawing, remanufacturing; software; secondary processing.

U.S. Forest Products Laboratory.

Computer optimization of cutting yield from multiple-ripped boards, by Abigail R. Stern and Kent A. McDonald. Madison, Wis., FPL, 1978.
13 p. (USDA For. Res. Pap. FPL 318).

RIPYLD (RIP YieLD) is the part of a lumber-defect-locating computer program that is designed to optimize sawline placement for maximum cutting yield. RIPYLD obtains multiple ripping and crosscutting solutions using defect locating data, and is an expansion of earlier efforts to maximize cutting yields of boards using computer analysis.

KEYWORDS: Computer optimization; lumber processing, yield, automation, sawing, remanufacturing; software; secondary processing.

U.S. Forest Products Laboratory.

Computer optimization of cutting yield from multiple-ripped boards, by Abigail R. Stern and Kent A. McDonald. Madison, Wis., FPL, 1978.
13 p. (USDA For. Res. Pap. FPL 318).

RIPYLD (RIP YieLD) is the part of a lumber-defect-locating computer program that is designed to optimize sawline placement for maximum cutting yield. RIPYLD obtains multiple ripping and crosscutting solutions using defect locating data, and is an expansion of earlier efforts to maximize cutting yields of boards using computer analysis.

KEYWORDS: Computer optimization; lumber processing, yield, automation, sawing, remanufacturing; software; secondary processing.

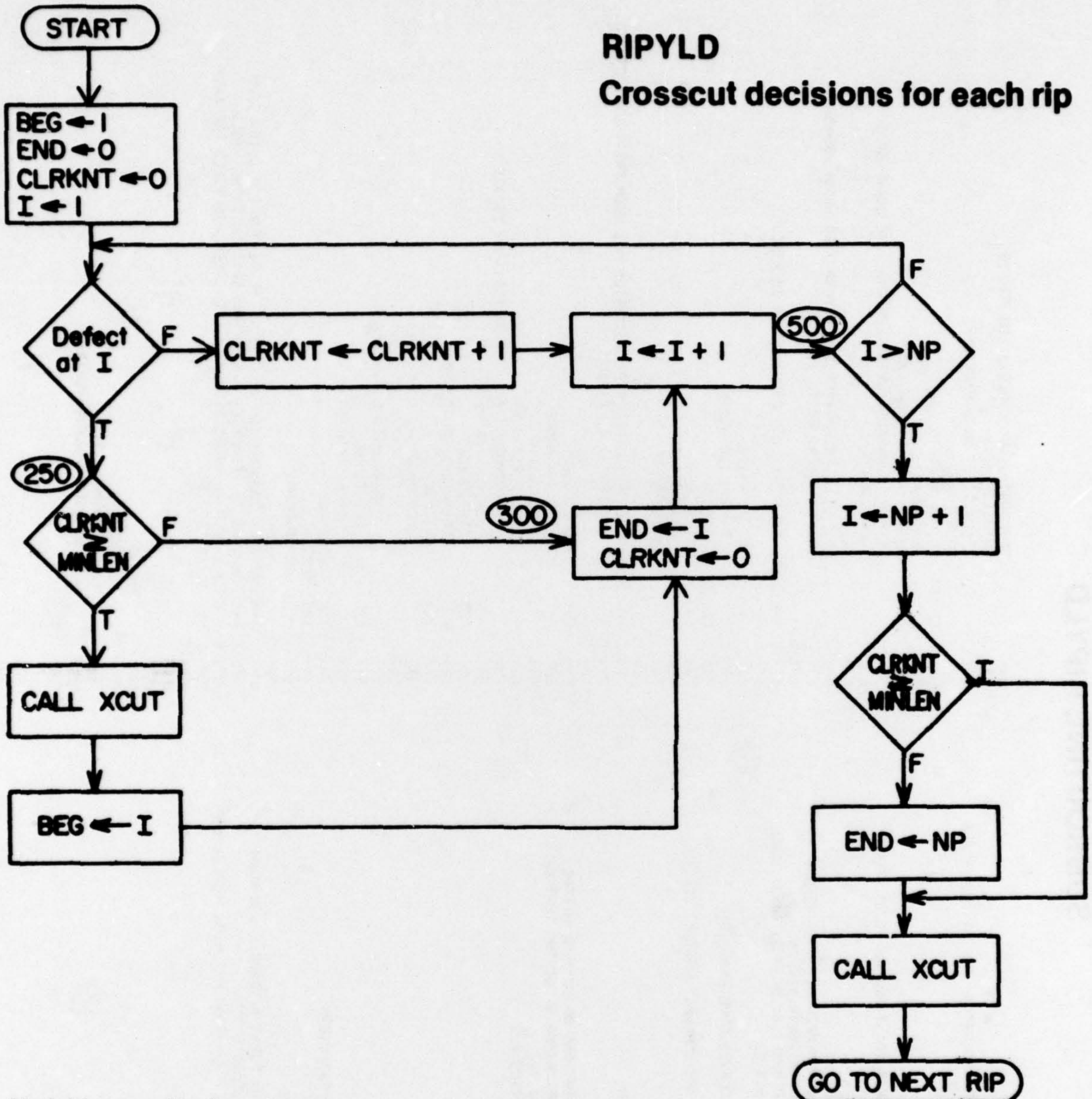
U.S. Forest Products Laboratory.

Computer optimization of cutting yield from multiple-ripped boards, by Abigail R. Stern and Kent A. McDonald. Madison, Wis., FPL, 1978.
13 p. (USDA For. Res. Pap. FPL 318).

RIPYLD (RIP YieLD) is the part of a lumber-defect-locating computer program that is designed to optimize sawline placement for maximum cutting yield. RIPYLD obtains multiple ripping and crosscutting solutions using defect locating data, and is an expansion of earlier efforts to maximize cutting yields of boards using computer analysis.

KEYWORDS: Computer optimization; lumber processing, yield, automation, sawing, remanufacturing; software; secondary processing.

RIPYLD
Crosscut decisions for each rip



Variables used in flowcharts

I	- present grid position on X axis
BEG	- beginning of defect (grid number)
END	- last defect grid encountered
CLRKNT	- number of clear grids encountered since last defect grid
MINLEN	- number of grid units in the minimum cutting length
XCUT	- subroutine to store crosscut locations and to calculate yield
NP	- number of X-grids in the board
RANDOM = TRUE	- random length option
= FALSE	- specified length option

SUBROUTINE RIPYLD

```
      SUBROUTINE RIPYLD
C
C *** PLACES THE RIP CUTS ON A BOARD TO OBTAIN THE MAXIMUM YIELD
C *** OF CLEAR CUTTINGS.  (RIP YIELD)
C
C
      IMPLICIT INTEGER(A-Z)
      REAL KERF,SUM,WIDTH,XGRID,YGRID,YDIST,SAWMIN,YIELD,AREA,
     1     CLRYLD,CROSS
      LOGICAL REJECT,FULL,MATCH
      DIMENSION COUNT(4),CYCLE(4),WX(4)
      COMMON /SM/ MAX,MAXCOM,MIN,MINCOM,NRIP(81),NSAW,NSCANS,
     .           NWIDTH,REJECT,SAWMIN,YGRID,YIELD(3)
      COMMON /SMX/ CROSS(150,3),RIPCOM(81,4),NP,NXCUT(4,3),XGRID,
     .             WIDTH(3),PIECE(5,3,3)
      COMMON /SRMX/ KERF
      COMMON /MX/ACTIVE,AVAIL,BEG,CLRYLD,COMB,END,I,RIP
C
C ***IF THE BOARD IS SHORTER THAN THE SMALLEST CUTTING LENGTH,
C *** REJECT THE BOARD.
C
      REJECT = .TRUE.
      IF(NP*XGRID.LT.SAWMIN) RETURN
C
C
C *** CALCULATE ALL POSSIBLE PERMUTATIONS OF CUTTING WIDTHS.
C *** STORE ALL UNIQUE ORDERED COMBINAIONS OF WIDTHS THAT WILL
C *** FIT IN THE BOARD IN RIPCOM(81,4).
C *** MAXIMUM NUMBER OF CUTTING WIDTHS = 3
C *** MAXIMUM NUMBER OF RIP SAWS = 4
C *** MAXIMUM PERMUTATIONS = 81
C
C *** INITIALIZE
C
      NPERM=NWIDTH**NSAW
      CYCLE(1)=NPERM/NWIDTH
      DO 110 J=1,NSAW
        WX(J)=1
        COUNT(J)=0
        IF(J.NE.1)CYCLE(J)=CYCLE(J-1)/NWIDTH
  110 CONTINUE
      COMB=0
C
C *** FOR EACH POSSIBLE COMBINATION, FIRST DETERMINE HOW MANY
C *** OF THE RIPS WILL FIT IN THE BOARD WIDTH.
C *** SECOND, CHECK TO SEE IF THE COMBINATION HAS BEEN PREVIOUSLY
C *** STORED.
C
      DO 140 PERM=1,NPERM
        SUM=0.
        FULL=.FALSE.
        DO 130 J=1,NSAW
          COUNT(J)=COUNT(J)+1
          IF(COUNT(J).LE.CYCLE(J))GO TO 120
          WX(J)=WX(J)+1
          IF(WX(J).GT.NWIDTH)WX(J)=1
          COUNT(J)=1
  120     IF(FULL) GO TO 130
          TEMP=WX(J)
          SUM=SUM+WIDTH(TEMP)+KERF
          IF(SUM.LE.NSCANS*YGRID+KERF.AND.J.NE.NSAW) GO TO 130
          FULL=.TRUE.
          NR=J-1
          IF(SUM.LE.NSCANS*YGRID+KERF.AND.J.EQ.NSAW) NR=NSAW
          IF(COMB.NE.0) GO TO 123
C
C *** STORE THE FIRST RIP COMBINATION IN RIPCOM.
C
          DO 122 K=1,4
            RIPCOM(1,K)=WX(K)
  122     CONTINUE
          NRIP(1)=NR
          COMB=1
          GO TO 130
C
C *** DETERMINE IF THE NEW RIP COMBINAION HAS BEEN PREVIOUSLY STORED.
C *** IF NOT, STORE IT IN RIPCOM.
C
  123     NCOMB=COMB
          DO 128 I=1,NCOMB
            MATCH=.TRUE.
            DO 125 RIP=1,NR
              IF(RIPCOM(I,RIP).EQ.WX(RIP)) GO TO 125
              MATCH=.FALSE.
  125       CONTINUE
            IF(MATCH) GO TO 130
  128     CONTINUE
              COMB=COMB+1
              DO 126 K=1,4
                RIPCOM(COMB,K)=WX(K)
  126         CONTINUE
              NRIP(COMB)=NR
  130   CONTINUE
  140 CONTINUE
        NCOMB=COMB
C
C *** FOR EACH COMBINATION IN RIPCOM, PLACE THE RIP CUTS, SCAN
C *** FOR DEFECTS, PLACE CROSSCUTS AND CALCULATE THE YIELD.
C *** CALL XCUT TO STORE CROSSCUTS AND CALCULATE YIELD FOR EACH
C *** CLEAR CUTTING.
C
      ACTIVE=1
      MAX=2
      MIN=3
      YIELD(3)=-1.
      YIELD(2)=.001
      AREA=NP*NSCANS*XGRID*YGRID
```

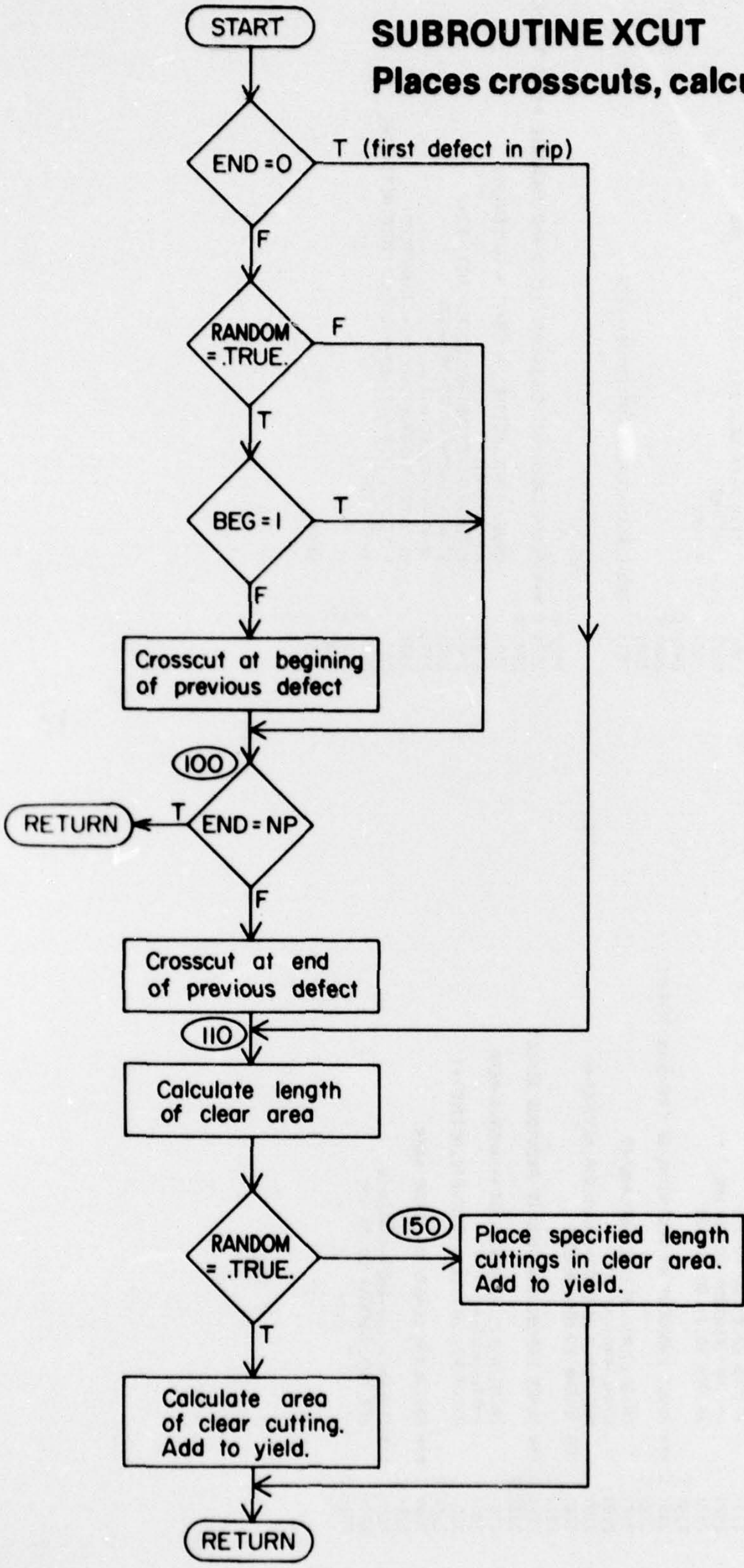
```
      MINLEN=(SAWMIN/XGRID)+.005
      DO 600 COMB=1.NCOMB
        AVAIL=1
        CLRYLD=0.
        YDIST=0.
        YLOW=1
        DO 150 I=1.5
          DO 150 J=1.3
          PIECE(I.J.ACTIVE)=0
  150   CONTINUE
C
C *** PLACE CROSSCUTS AND CALCULATE YIELD FOR EACH RIP.
C
        NR=NRIP(COMB)
        DO 550 RIP=1.NR
          NXCUT(RIP.ACTIVE)=0
          TEMP=RIPCOM(COMB.RIP)
          YDIST=YDIST+WIDTH(TEMP)
          YHI=IFIX((YDIST/YGRID)+.99)
          BEG = 1
          END = 0
          CLRKNT = 0
          DO 500 I=1.NP
            DO 200 J=YLOW.YHI
              IF(BOARD(J.I).EQ.0) GO TO 200
              GO TO 250
  200       CONTINUE
            CLRKNT=CLRKNT+1
            GO TO 500
C
C *** DEFECT FOUND
C
  250       IF(CLRKNT.LT.MINLEN) GO TO 300
            CALL XCUT
            BEG = I
  300       END = I
            CLRKNT = 0
  500     CONTINUE
          I=NP+1
          IF(CLRKNT.LT.MINLEN) END = NP
          CALL XCUT
  580     YDIST=YDIST+KERF
          YLOW=IFIX((YDIST/YGRID)+1.01)
  550   CONTINUE
C
C *** CALCULATE ( YIELD.  COMPARE EACH SOLUTION WITH THE PREVIOUS
C *** MINIMUM AND MAXIMUM.
C
        YIELD(ACTIVE)=(CLRYLD/AREA)*100.
C
C *** TEST FOR NEW MAXIMUM.
C
        IF(YIELD(ACTIVE).LT.YIELD(MAX))GO TO 590
        IF(.NOT.REJECT) GO TO 585
C
C *** FIRST SOLUTION FOUND.  INITIALIZE BOTH MINIMUM AND MAXIMUM
C *** SOLUTION ARRAYS.
C
        REJECT=.FALSE.
        MINCOM=COMB
        YIELD(MIN)=YIELD(ACTIVE)
        DO 581 I=1.150
          CROSS(I.MIN)=CROSS(I.ACTIVE)
  581   CONTINUE
        DO 582 I=1.4
          NXCUT(I.MIN)=NXCUT(I.ACTIVE)
  582   CONTINUE
        DO 583 I=1.5
          DO 583 J=1.3
            PIECE(I.J.MIN)=PIECE(I.J.ACTIVE)
  583   CONTINUE
C
C *** STORE NEW MAXIMUM.
C
  585   MAXCOM=COMB
        TEMP=ACTIVE
        ACTIVE=MAX
        MAX=TEMP
        GO TO 600
C
C
C *** TEST FOR NEW MINIMUM.
C
  590   IF(YIELD(ACTIVE).GT.YIELD(MIN)) GO TO 600
        MINCOM=COMB
        TEMP=ACTIVE
        ACTIVE=MIN
        MIN=TEMP
  600 CONTINUE
      RETURN
      END
```

```
SYMBOL        REFERENCED AT LINES (MINUS MEANS SYMBOL DEFINED.
 NAME            EXCLUDING SUBPROGRAM CALLS AND EQUIVALENCE)

ACTIVE          17 -102  116  123  156  160  168  170  173  177  183 -184  191  193 -194
AREA             8 -107  156
AVAIL           17 -110
BEG             17 -127 -142
BOARD          132
CLRKNT        -129 -135  140 -144  147
CLRYLD           8   17 -111  156
COMB            17  -42   65  -73   79  -88   90   92   95 -109  121  124  167  182  192
COUNT           11  -39  -53   54  -57
CROSS            8   14 -170
CYCLE           11  -36  -40   54
END             17 -128 -143 -147
FULL            10  -51   58  -62
I               17  -80   83 -114  116 -130  132  142  143 -146 -169  170 -172  173 -175  177
IFIX           126  150
J              -37   38   39   40  -52   53   54   55   56   57   59   61   63   64 -115  116 -131  132 -176  177
K              -69   70  -89   90
KERF             8   16   60   61   64  149
MATCH           10  -81  -84   86
MAX             12 -103  160  184 -185
MAXCOM          12 -182
MIN             12 -104  168  170  173  177  191  194 -195
MINCOM          12 -167 -192
MINLEN        -108  140  147
MX              17
NCOMB          -79   80  -95  109
NP              14   23  107  130  146  147
NPERM          -35   36   49
NR             -63  -64   72   82   92 -121  122
NRIP            12  -72  -92  121
NSAW            12   35   37   52   61   64
NSCANS          12   61   64  107
NWIDTH          12   35   36   40   56
NXCUT           14 -123 -173
PERM           -49
PIECE           14 -116 -177
REJECT          10   12  -22  161 -166
RIP             17  -82   83 -122  123  124
RIPCOM          14  -70   83  -90  124
RIPYLD           1
SAWMIN           0   12   23  108
SM              12
SMX             14
SPMX            16
SUM              8  -50  -60   61   64
TEMP           -59   60 -124  125 -183  185 -193  195
WIDTH            8   14   60  125
WX              11  -38  -55   56   59   70   83   90
XCUT           141  148
XGRID            8   14   23  107  108
YDIST            8 -112 -125  126 -149  150
YGRID            8   12   61   64  107  126  130
YHI           -126  131
YIELD            8   12 -105 -106 -156  160 -168  191
YLOW          -113  131 -150
```

```
STATEMENT  DEFINED
 NUMBER    AT LINE   REFERENCED AT LINES

   110        41        37
   120        58        54
   122        71        69
   123        79        65
   125        85        82   83
   126        91        89
   128        87        80
   130        93        52   58   61   74   86
   140        94        49
   150       117       114  115
   200       134       131  132
   250       140       133
   300       143       140
   500       145       130  136
   550       151       122
   580       149
   581       171       169
   582       174       172
   583       178       175  176
   585       182       161
   590       191       160
   600       196       109  186  191
```

SUBROUTINE XCUT

Places crosscuts, calculates cutting yields



SUBROUTINE XCUT

```
      SUBROUTINE XCUT
      IMPLICIT INTEGER(A-Z)
      REAL CLRYLD,KERF,XGRID,CROSS,CLRLEN,START,CUTLEN,WIDTH
      LOGICAL RANDOM
      COMMON /SX/ CUTLEN(5),NLEN,RANDOM
      COMMON /SMX/ CROSS(150,3),RIPCOM(81,4),NP,NXCUT(4,3),XGRID,
                 WIDTH(3),PIECE(5,3,3)
      COMMON /SRMX/ KERF
      COMMON /MX/ ACTIVE,AVAIL,BEG,CLRYLD,COMB,END,I,RIP
C
C
C
      IF(END.EQ.0) GO TO 110
      IF(.NOT.RANDOM) GO TO 100
      IF(BEG.EQ.1) GO TO 100
C
C *** PLACE CROSSCUT AT BEGINNING OF PREVIOUS DEFECT
C
      CROSS(AVAIL,ACTIVE)=BEG*XGRID
      AVAIL=AVAIL+1
      NXCUT(RIP,ACTIVE)=NXCUT(RIP,ACTIVE)+1
  100 IF(END.EQ.NP) RETURN
C
C *** PLACE CROSSCUT AT END OF PREVIOUS DEFECT
C
      CROSS(AVAIL,ACTIVE)=(END+1)*XGRID-KERF
      AVAIL=AVAIL+1
      NXCUT(RIP,ACTIVE)=NXCUT(RIP,ACTIVE)+1
C
C *** CALCULATE LENGTH OF CLEAR AREA.
C
  110 CLRLEN = ((I-END)-1)*XGRID
      IF(.NOT.RANDOM) GO TO 150
C
C *** CALCULATE YIELD FOR RANDOM LENGTH CUTTING
C
      TEMP=RIPCOM(COMB,RIP)
      CLRYLD=CLRYLD+CLRLEN*WIDTH(TEMP)
      RETURN
C
C
C *** CALCULATE SPECIFIED LENGTH CUTTINGS TO FIT IN CLEAR AREA.
C
  150 START=(END+1)*XGRID
  200 DO 250 J=1,NLEN
        JS=J
        IF(CLRLEN.GE.CUTLEN(J)) GO TO 300
  250 CONTINUE
      RETURN
C
C
C
  300 CLRLEN=CLRLEN-CUTLEN(JS)-KERF
C
C *** PLACE CROSSCUT, CALCULATE YIELD AND INCREASE PIECE TALLY.
C
      CROSS(AVAIL,ACTIVE) = START + CUTLEN(JS)
      AVAIL=AVAIL+1
      NXCUT(RIP,ACTIVE)=NXCUT(RIP,ACTIVE)+1
      START=START+CUTLEN(JS)+KERF
      TEMP=RIPCOM(COMB,RIP)
      CLRYLD=CLRYLD+CUTLEN(JS)*WIDTH(TEMP)
      PIECE(JS,TEMP,ACTIVE)=PIECE(JS,TEMP,ACTIVE)+1
      GO TO 200
C
C
      END$
```

SYMBOL NAME	REFERENCED AT LINES (MINUS MEANS SYMBOL DEFINED, EXCLUDING SUBPROGRAM CALLS AND EQUIVALENCE)
ACTIVE	207 217 219 224 226 255 257 261
AVAIL	207 217 -218 224 -225 255 -256
BEG	207 213 217
CLRLEN	201 -230 236 245 -251
CLRYLD	201 207 -236 -260
COMB	207 235 259
CROSS	201 204 -217 -224 -255
CUTLEN	201 203 245 251 255 258 260
END	207 211 220 224 230 242
I	207 230
J	-243 244 245
JS	-244 251 255 258 260 261
KERF	201 206 224 251 258
MX	207
NLEN	203 243
NP	204 220
NXCUT	204 -219 -226 -257
PIECE	204 -261
RANDOM	202 203 212 231
RIP	207 219 226 235 257 259
RIPCOM	204 235 259
SMX	204
SRMX	206
START	201 -242 255 -258
SX	203
TEMP	-235 236 -259 260 261
WIDTH	201 204 236 260
XCUT	199
XGRID	201 204 217 224 230 242

STATEMENT NUMBER	DEFINED AT LINE	REFERENCED AT LINES
100	220	212 213
110	230	211
150	242	231
200	243	262
250	246	243
300	251	245