CIVIL ENGINEERS
POCKET COMPUTER MONTHLY

BEAMS - SIMPLE AND CONTINUOUS - PART III

Our July issue completes our three-month series on the subject of beams-providing a basis for design of simple and continuous beams, including shear and moment diagrams.

PROGRAM-BEAM-MOVING LOADS

The following program permits rapid solution for the problem of a simple uniform-loaded beam which also supports a pattern of moving concentrated loads. The loads maintain a constant dimensional relationship to each other as they move across the beam.

Standard textbooks provide a rule permitting determination of the optimum position for the load pattern, as follows:

At maximum moment, the centerline of the beam lies midway between the resultant and the next nearest large concentrated load; the maximum moment will occur at this large load. Using this rule, the designer may promptly proceed to a solution. However the rule assumes a weightless beam. The self-weight of the beam, plus any other uniform loadings, will affect the location of the optimum position. We know of no rule which accounts for the combination of uniform load plus a pattern of moving concentrated loads.

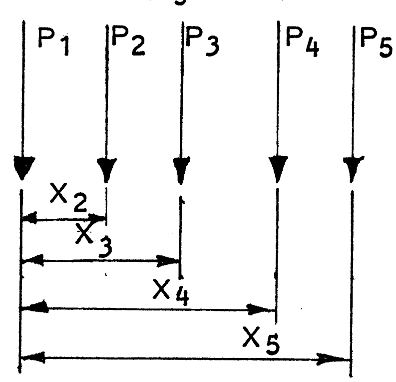
In seeking to solve the problem using the PC-1 and PC-2 computers, we elected to initially place the pattern of loads such that the resultant is located on beam centerline; at this point, by definition, the reactions are equal, and the moment should be near maximum.

The program then moves the load pattern in increments, first one way, then the other, following the rule requiring that incrementing must continue until the moment ceases to increase. Moments are compared and the greater is reported, along with other information defining the location of the load pattern, etc. The designer should account for the possibility that an unsymmetrical pattern of loads can be reversed in orientation.

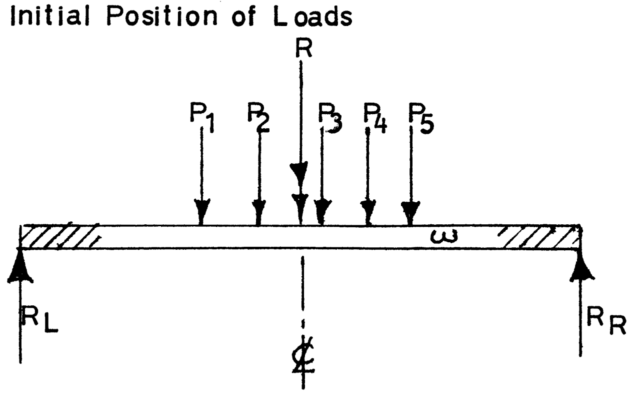
Because of memory limitations, the PC-1 program will accomodate a pattern of up to ten loads. However, the PC-2 program is limited by the DIM statement which, for parameters A1 and A2, allows up to 99 loads. You may, of course, change this dimension in accordance with the memory capacity of your PC-2.

Note that, during entry of loads, it is mandatory that the loads be entered from left to right, with corresponding distance from the left-hand load. Of course, the distance X_1 for the initial load (left-hand) will always be zero; the program does not require entry of X_1. Thus, you will enter the first two loads, P_1 and P_2, before entry of X_2.

Loading Pattern



Initial Position of Loads



PROGRAM/BEAM-MOVING LOADS

Program Listing PC-2

```
10   "A"LPRINT "BEAM-MOVING
     LOADS": LPRINT "** ** **
     ** ** **":LF 2
100  CLEAR:DIM A1(99),A2(99):WAIT 60
110  INPUT "SPAN(FT)=";L
115  LPRINT "SPAN(FT)=":LF-1:
     LPRINT L
120  INPUT "NO. OF CONC. LOADS= ";N
125  LPRINT "NO. CONC. LOADS= ":
     LF-1:LPRINT N
130  INPUT "SIZE OF INCR.(FT)= ";D
132  LPRINT "SIZE INCR.(FT)= ":LF-1:
     LPRINT D
135  INPUT "UNIF. LOAD(K/FT)= ";W:
     LPRINT "UNIF.LD(K/FT)= ":LF-1:
     LPRINT W:LF 1
137  IF W=0 LET W=1E-12
138  IF N=0 LET A1(1)=1E-12:A2(1)=0:
     A=A1(1):GOTO 190
140  REM ENTER CONCENTRATED
     LOADS FROM LEFT TO RIGHT
144  FOR I=1 TO N
145  PRINT "CONC.LOAD #";I
150  INPUT "LOAD,(K)= ";A1(I):IF
     I=1 LET A2(I)=0:GOTO 170
155  PRINT "DISTANCE #";I
160  INPUT "DIST.(FT)=";A2(I)
165  IF A2(I)>L LPRINT "ERROR-LOAD
     PATTERN EXCEEDS SPAN":END
170  A=A+A1(I):B=B+A1(I)*A2(I)
180  NEXT I
185  LPRINT " LOAD      DIST."
187  FOR I=1 TO N:LPRINT A1(I);
     "   ";A2(I):NEXT I
190  C=B/A
191  LPRINT"- - - - - - - -
     - - - - - - - - - - -":LPRINT
     "SUM,LOADS=":LF-1:LPRINT A+W*L
195  LPRINT "DIST,LEFT LOAD
     TO RESULTANT=":LF-1:LPRINT
     INT (10*C+.5)/10
200  E=A-A*C/L+W*L/2
210  F=A-A*(A2(I)-C)/L+W*L/2
225  TT=0:O=L/2-C-D:IF AQ=1 LET
     O=O+D*2
240  X=0:IF AQ=0 LET O=O+D:GOTO 246
245  O=O-D
246  IF O+A2(N)<L/2 LET O=L/2-A2(N):AR=1
247  IF O+A2(N)>L GOSUB 1500:
     AO=1:GOTO 310
248  IF O>L/2 LETO=L/2:AP=1
249  IF O<0 GOSUB 1500:AO=1:
     GOTO 310
250  Q=(L-O-C)/L*A+W*L/2
252  RR=O
260  S=Q
265  T=S*O-O^2*W/2
267  S=S-W*O-A1(X+1)
268  IF S<=0 THEN 273
269  GOTO 278
270  CD=W*(A2(X+1)-A2(X)):IF CD<
     S LET T=T+S*(A2(X+1)-A2(X))-
     (A2(X+1)-A2(X))^2*W/2
271  IF CD<S LET S=S-W*(A2(X+1)-
     A2(X))-A1(X+1):RR=RR+A2(X+1)-A2(X):
     GOTO 273
272  GOTO 276
273  IF S<=0 IF (T<=TT)+(AP=1)THEN 300
274  IF S<=0 LET OP=O:RS=RR:
     TT=T:QQ=Q:GOTO 240
275  GOTO 278
276  RR=RR+S/W:T=T+S^2/W/2:IF
     (T<=TT)+(AP=1) THEN 300
277  OP=O:RS=RR:TT=T:QQ=Q:GOTO 240
278  X=X+1:GOTO 270
300  IF AQ=1 THEN 310
305  AQ=1:TU=TT: OQ=OP: RT=RS:
     QR=QQ:IF AR=0 THEN 225
310  IF TU>TT LET TT=TU:RS=RT:
     OP=OQ:QQ=QR
1000 LF 2:LPRINT "RESULTS- - -":LF 1
1002 LPRINT "MAX. LT.REAC.=":LF-1:
     LPRINT INT (10*E+.5)/10
1003 LPRINT "MAX. RT. REAC.=":LF-1:
     LPRINT INT (10*F+.5)/10
1004 IF AO=1 LF 3:END
1005 LF 1:LPRINT "WHERE MOMENT
     IS MAXIMUM:"
1006 LPRINT "DIST.TO LT.-HAND
     LOAD=":LF-1:LPRINT INT (10*OP+.5)/10
1007 LPRINT "DIST.TO PT. OF MAX.
     MOMENT=":LF-1:LPRINT INT
     (10*RS+.5)/10
1010 LPRINT "MAX. MOMENT=":
     LF-1:LPRINT INT (100*TT+.5)/100
1020 LPRINT "LEFT REAC=":LF-1:
     LPRINT INT (QQ*10+.5)/10
1030 LPRINT "RGHT REAC=":LF-1:
     LPRINT INT ((A-QQ+W*L)*10+.5)/10:
     LF3:END
1500 LF 2:LPRINT "DUE TO WIDE
       LOAD PATTERN, MAXIMUM
     MOMENT NOT FOUND":RETURN
```

PROGRAM/BEAM-MOVING LOADS

Program Listing PC-1

It should be noted that the following program completely fills the PC-1's memory.

```
11Ø  "A" PRINT "BEAM-M⊥LOAD":PRINT
     " ":CLEAR:INPUT "SPAN(FT)=";
     L: PRINT "SPAN=":PRINT L
12Ø  INPUT "#CONC⊥LOADS=";N:PRINT
     "#CONC⊥DS=":PRINT N: USING
     "#####.##"
13Ø  INPUT "INCR(FT)=";D:PRINT
     "INCR (FT)=":PRINT D:INPUT
     "UNIF LD=";W: PRINT "UNIF
     LD=":PRINT W:PRINT" "
136  IF N<>Ø PRINT " LOAD,P
       X,FT": PRINT " "
137  IF W=Ø LET W=1E-12
138  IF N=Ø LET A(27)=1E-12:A(37)=Ø:
     A=A(27):GOTO 19Ø
15Ø  FOR I=28 TO N+27:J=I+11:INPUT
     "LOAD(K)=";A(I):IF I=28LET
     A(J)=Ø: GOTO 17Ø
```

If the width of the load pattern exceeds the span, an error message is given.

```
16Ø  INPUT "DIST.(FT)=";A(J):IF
     A(J)>L PRINT "ERROR":END
17Ø  A=A+A(I):B=B+A(I)*A(J):PRINT
     A(I); A(J):NEXT I:PRINT " "

19Ø  C=B/A:E=A-A*C/L+W*L/2:F=A-A*
     (A(J)-C)/L+W*L/2:X=A+W*L:PRINT
     "SUM,LOADS=":PRINT X
195  C=INT (1Ø*C+.5)/1Ø:PRINT "DIST:
     LEFT LOAD TO RES=";C
225  B=Ø:V=Ø:O=L/2-C-D:IF G=1
     LET O=O+2*D
24Ø  X=28:Y=39:J=38:IF G=Ø LET
     O=O+D: GOTO 244
242  O=O-D
244  IF O+A(N+38)<L/2 LET O=L/2-A
     (N+38):B=1
```

If the pattern of loads is too wide relative to the span, such that an optimum (maximum moment) position cannot be found, an error message is given.

```
245  IF O+A(N+38)>L PRINT "ERROR":END
247  IF O>L/2 LET O=L/2:A(51)=1
248  IF O<Ø PRINT "ERROR":END
25Ø  Q=(L-O-C)/L*A+W*L/2:P=O:S=Q:
     T=SO-OOW/2:S=S-WO-A(X):IF
     S<=Ø THEN 273
267  GOTO 298
268  A(52)=W*(A(Y)-A(J)):IF A(52)<S LET
     T=T+S*(A(Y)-A(J))-(A(Y)-A(J))∧2*W/2
271  IF A(52)<S LET S=S-W*(A(Y)-A(J))-A(X):
     P=P+(A(Y)-A(J)):GOTO 273
272  GOTO 286
273  IF S<=Ø IF T<=V THEN 3ØØ
274  IF S<=Ø IF A(51)=1 THEN 3ØØ
284  IF S<=Ø LET H=O:R=P:V=T:I=Q:
     GOTO 24Ø
285  GOTO 298
286  P=P+S/W:T=T+SS/W/2:IF T<=V THEN 3ØØ
288  IF A(51)=1 THEN 3ØØ
289  H=O:R=P:V=T:I=Q:GOTO 24Ø
298  X=X+1:Y=Y+1:J=J+1:GOTO 268
3ØØ  IF G=1 THEN 31Ø
3Ø5  G=1:Z=V:K=H:U=R:M=I:IF B=Ø THEN 225
31Ø  IF Z>V LET V=Z:R=U:H=K:I=M
32Ø  PRINT "MAX REACS,L/R":PRINT E:
     PRINT F
33Ø  PRINT "DIST TO LEFT LD=":PRINT
     H:PRINT "TO MAX MOM=":PRINT
     R:PRINT "MAX MOM=":PRINT V
```

The reader will note the frequent use of PRINT " " which advances the paper tape. Also note the difference between the zeros which are slashed; and the O's which are not.

In order to obtain the left reaction at the optimum position, press I ENTER.

Note: Civil Engineers Pocket Computer Monthly supports Radio Shack's PC-1 and PC-2 (Sharp PC-1211 and PC-1500). We believe our software will be helpful to civil engineers who have other equipment.

PROGRAM - RECTANGULAR WOOD MEMBERS-COMBINED AXIAL COMPRESSION AND FLEXURAL LOADING - X and Y DIRECTIONS - SIDE LOADING AND ECCENTRICITY - PC-1

The following program has been rewritten for PC-1 because the Errata in the June Issue and the PC-2 version in the April, 1983 Issue, did not provide a fully correct, workable program. This PC-1 program will not accomodate eccentric loads. Also, all end restraint coefficients are assumed to be unity (1).

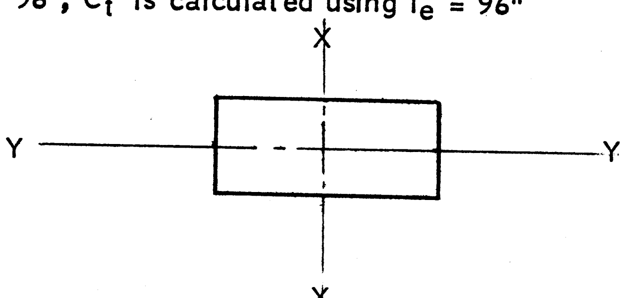
Ref: "National Design Specification for Wood Construction," 1977 Edition, National Forest Products Association, Washington, D. C.

This combined axial and bending stress type of problem is a trial-and-error design process, lengthy and time consuming without the aid of a computer.

If a truss compression chord, 2 x 4 or smaller, (in combined axial load and flexural bending) is subjected to bending such that the edge to which the plywood is attached is in flexural compression, and if the truss is not used in "wet service conditions", and if the wood is seasoned to 19% moisture content or less, use C_T (buckling stiffness factor):

$$C_t = 1 + .002 * l_e$$

where L_e = effective buckling length used in the design of the chord for compressive loading, inches. For l_e values greater than 96", C_t is calculated using l_e = 96"



References herein to the x-x axis and the y-y axis are intended to pertain to the "strong" way and the "weak" way, resp., of a rectangular section. See diagram, above.

Parameter Designations:

Input:

I = Actual Member Depth (greater Dimn.), in.
J = Actual Member Thickness, in.
K = Unbraced Length, x-x, ft.
L = Unbraced Length, y-y, ft.
M = F_b = Allowable Stress, psi
N = F_c = Allowable Compressive Stress, psi
O = E = Modulus of Elasticity, psi
P = Axial Load on Member, lb.
Q = Load Factor, Transverse Load
R = Load Factor, Axial Load
S = Bending Moment, Max, x direction, ft-lb
T = Bending Moment, Max, y direction, ft-lb

Determine whether member being designed is a truss compression chord, and whether member dimensions qualify for modification using the Buckling Stiffness Factor.

Program Listing PC-1:

This program uses a modified GCP format.

```
1Ø    "A" CLEAR
2Ø    BEEP 4:PAUSE "*****GCP*****":
      PAUSE "INPUT PARAMS:
      I-Z"
25    BEEP 2:INPUT "# INPUT PARAMS?
      (12 MAX)";A(53)
3Ø    I$="I":J$="J":K$="K"
31    L$="L":M$="M":N$="N":O$="O":
      P$="P":Q$="Q":R$="R":S$="S":
      T$="T"
35    FOR A=9 TO 8+A(53)
4Ø    A$(A+26)=A$(A):NEXT A
5Ø    PRINT " ":PRINT " ":PRINT
      " "
67    A(54)=ØGOTO 7Ø
69    "C" PAUSE "GCP-RERUN":A(54)=1:
      FOR A=9 TO 8+A(53):A$(A)=A$(A+26):
      NEXT A
7Ø    BEEP 2:FOR A=9 TO A(53)+8:
      PAUSE "INPUT":PAUSE A$(A);".
      . ";A$(A);" . . ";A$(A);". .";A$(A):
      INPUT A(A): NEXT A
75    IF A(54)=1 THEN 9Ø
```

```
90   PAUSE "CALCULATION"
100  INPUT "C.CHORD?Y/N";H$:
     IF H$="N" THEN 112
110  H$="N":IF I< =4 IF J<=2 LET
     H$="Y"
112  IF Q=0 LET Q=1
114  IF R=0 LET R=1
120  G=K*12/I:F=L*12/J:C=1
130  E=G:IF F>G LET E=F
140  IF E>50 PRINT "ERROR":BEEP
     4:END
```

Calculate Buckling Stiffness Factor $(C_T)=1+.002*l_e)$ It is assumed that, for this case, the plywood will be fastened to the narrow edge of the chord; no lateral bending will be involved. Therefore the bending moment in the y direction must equal zero.

```
150  IF T<>0 THEN 170
160  IF H$="Y" LET C=1+.002*12*K:
     IF K>8 LET C=1+.002*96
```

Calculate K and J values

```
170  A=.671 *√(O/N):A=A*√C
180  G=(G-11)/(A-11)
183  IF G< 0 LET G=0
186  IF G>1 LET G=1
190  F=(F-11)/(A-11)
193  IF F<0 LET F=0
196  IF F>1 LET F=1
```

Calculate F_c

```
200  IF E>=A LET Y=.3*O/E/E
210  IF E>11 IF E<A LET Y=N*(1-
     (E/A)^4/3)
215  IF Y=0 LET Y=N
```

Calculate f_c and Interaction Number for Compression modified by Axial Load Factor

```
220  B=P/I/J/R/Y
```

Modify Interaction Number for Bending in x and y directions

```
230  D=S*12*6/J/I/I/Q/(M-G*B*Y)
240  D=D+T*12*6/J/J/I/Q/(M-F*B*Y)
250  A=B+D
260  PRINT "INT #=";A:END
```

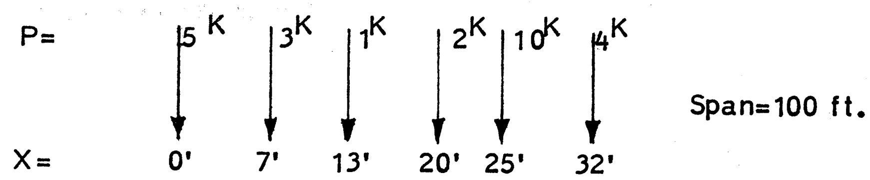
In DEF Mode:

For initial run, Press Shift A.
For subsequent runs, Press Shift C.

WORKED-OUT EXAMPLES

BEAM WITH MOVING LOADS/PC-2

Pattern of Moving Loads



RUN #1

Assume beam is weightless. Uniform Load = zero. Use an increment of 0.1 ft.

```
BEAM-MOVING LOADS
** ** ** ** ** **


SPAN(FT)=          100
NO.CONC.LOADS=     6
SIZE INCR.(FT)=0.1
UNIF.LD(K/FT)=     0

LOAD      DIST.
 5         0
 3         7
 1         13
 2         20
 10         25
 4         32
-------------------
SUM,LOADS=         25
DIST,LEFT LOAD TO
RESULTANT=      18.1

RESULTS---

MAX.LT.REAC=   20.5
MAX.RT.REAC=   21.5

WHERE MOMENT IS
MAXIMUM:
DIST.TO LT.-HAND
LOAD=            28.4
DIST.TO PT.OF MAX.
MOMENT=          53.4
MAX.MOMENT=  513.49
LEFT REAC=       13.4
RGHT REAC=       11.6
```

Note that the optimum position for maximum moment occurs at 28.4 ft., and that the maximum moment is at the 10K load, although the 2K load lies nearer to the resultant. The centerline of the beam bisects the space between the resultant and the 10K load.

RUN #2

Now rerun the same problem with uniform load of 2K/ft.

```
BEAM-MOVING LOADS
** ** ** ** ** **


SPAN(FT)=          100
NO.CONC.LOADS=     6
SIZE INCR.(FT)=0.1
UNIF.LD(K/FT)=     2

LOAD      DIST.
 5         0
 3         7
 1         13
 2         20
 10         25
 4         32
-----------------
SUM,LOADS=        225
DIST,LEFT LOAD TO
RESULTANT=       18.1

RESULTS---

MAX.LT.REAC=  120.5
MAX.RT.REAC=  121.5

WHERE MOMENT IS
MAXIMUM:
DIST.TO LT.-HAND
LOAD=              25.7
DIST.TO PT.OF MAX.
MOMENT=            50.7
MAX.MOMENT=  3011.1
LEFT REAC=      114.1
RGHT REAC=        111
```

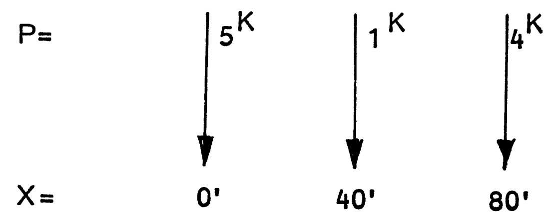
Note that the optimum position shifts to the left by 2.7 ft., compared with RUN #1.

RUN #3

To illustrate the problem encountered when the load pattern is wide relative to the span, the following is provided.

```
BEAM-MOVING LOADS
** ** ** ** ** **


SPAN(FT)=          100
NO.CONC.LOADS=       3
SIZE INCR.(FT)=     .1
UNIF.LD(K/FT)=       0

LOAD      DIST.
 5          0
 1         40
 4         80
----------------
SUM,LOADS=          10
DIST,LEFT LOAD TO
RESULTANT=          36


DUE TO WIDE   LOAD
PATTERN, MAXIMUM
MOMENT NOT FOUND


RESULTS---

MAX.LT.REAC=     6.4
MAX.RT.REAC=     5.6
```

Pattern of Moving Loads



Span = 100 ft.

The program was unable to find the optimum moment because the moving load pattern encountered the end of the beam, without finding an optimum load position.

As a final comment, you may be frustrated by the length of run-time, especially using the PC-1. The run-time is affected by 1) the number of loads, and 2) the size of the increment. We suggest you use an increment no greater than 2 or 3 feet because of potential accuracy problems. However, if the increment is less than 1, the run-time (especially for PC-1) may be rather long. For PC-2, run-times are typically less than 30 seconds; however for PC-1, they may exceed 3-5 minutes. The PC-2's run-time will not be intolerably long even if an increment of 0.1 is used, as in the above examples.

RECTANGULAR WOOD MEMBERS-COMBINED AXIAL COMPRESSION AND FLEXURAL LOADING - X AND Y DIRECTIONS - SIDE LOADING AND ECCENTRICITY - PC-1

This program runs in essentially the same way as the GCP program for PC-2 provided in the April Issue. However, because the end restraint coefficients are assigned unity (1), the output for the two illustrative examples in the April issue, is altered.

In the example on page 12 (Analysis of Top Chord) the interaction number changes from 0.9999 to 1.0776; this is because the end restraint coefficients, X and Y directions are assigned the values, 1 and 1 instead of 0.65 and 1. In the Wood Pole examples on pages 14 and 15, both end coefficients changed from 0.8 to 1, and the revised answers are as follows: .5975 becomes .6613, and .7629 becomes .8571.