Computer Program of Line Balancing, Regarding Efficiency and Number of Stations as Variables

Fumio AKAGI*, Hirokazu OSAKI*,
and Susumu KIKUCHI*

(Received February 6, 1979)

Synopsis

To assign work elements to the work stations in an assembly or manufacturing line, various computer programs have been developed and used. And it does that the number of stations or the cycle time is even given. But in practice it is desirable to obtain the assignment which shows the highest efficiency of line balancing under all possible combinations of the number of stations and the cycle time.

Therefore we propose a computer program of the assignment method in which the efficiency of line balancing, $E_{bb}$ and the number of stations, $N_N$ are regarded as variables. In this method the minimum value ($E_{Eb}$) of efficiency and the constant term ($d$) by which $E_{bb}$ is reduced are given previously. And for any combination of $E_{bb}$ ($E_{Eb} \leq E_{bb} \leq 100$) and $N_N$ ($1 \leq N_N \leq N_m$, $N_m$ calculated from $E_{Eb}$), the work elements are assigned to work stations, the precedence restrictions being used. $E_{bb}$ is reduced by $d$ from the ideal value (100) until the assignment to $N_N$ is obtained. The efficiency of the obtained assignment, $E_{bo}$ is calculated.

As $E_{bb} < E_{bo} < E_{bb} + d$, the calculation is continued until the assignment to $N_N$, which shows the maximum efficiency, is obtained. In this process $N_N$ varies from $N_m$ to 1 by 1 by 1.

* Department of Industrial Science, School of Engineering
Okayama University.
1. Introduction

The basic problem in line balancing is to assign the work elements to the work stations so as to balance the workload (the station time) at each station and to make this work content as close to the cycle time of the line as possible [1]. Various computer programs have been developed and used to solve this problem [2,3]. Generally to evaluate the assignment, the efficiency of line balancing, which is indicated by the following Eb, has been used [4].

\[
Eb = \frac{T}{N \times P} \times 100 \quad (1)
\]

where Eb = the efficiency of line balancing
T = the total work time
N = the number of stations
P = the cycle time to N stations.

As there is no slack time in the line at Eb=100%, the assignment of the work elements is ideal.

In the programs developed until now, N or P must be given. As N is given, the assignment is determined so as to minimize P. As P is given, it is done so as to minimize N. The number of stations N is determined by personnel, machines, tools, work areas, and so on. The cycle time P is restricted by the production time and the units to be produced per day. From these viewpoints the number of stations (NN), the efficiency of line balancing (Ebb), and the cycle time (PP) are regarded as variables in some restrictions. Further there is a functional relation among NN, PP and Ebb.

Therefore in this paper we propose an assignment method in which the work elements are assigned to the work stations, Ebb and NN regarded as variables. Firstly, in this method the minimum efficiency of line balancing (Eeb) and the constant term (d) by which Ebb is reduced are given. Then the maximum number of stations (Nm) is determined by Eeb and the maximum work element time (tm) from Eq.(1), and the value of NN is given as Nm. Next, the value of Ebb is given as 100 (the ideal value). Then PP is determined from Ebb and NN from Eq. (1). NN and PP being used, the work elements are assigned to the work stations according to the precedence restrictions so that the work time of each station (the station time) may be less than PP. Ebb is reduced at the constant rate (d) from 100 until the assignment to NN is obtained. And the efficiency of the obtained assignment (Ebo) is calculated. Ebo is greater than Ebb but less than Ebb+d because the maximum station time (C) is less than PP. Since then Ebb is made
exchange Ebb to Ebo.

And the calculation is continued until the assignment to NN which shows the maximum of Eb between Ebb and Ebb+d is obtained. The same procedures are repeated from Nm to 1 by 1 by 1.

As this method needs much repetition of the routine calculation, we develop the computer program.

2. Procedures for Assigning Work Elements to Work Stations

In order to obtain a favorable assignment, the sequence of the operations must be analyzed first. The precedence restrictions are used to indicate which work element must be done before others. The results are summarized by a precedence diagram[5], an arrow diagram[6], a table of functional units[7], a precedence matrix[1], and so on.

The proposed procedure for assigning the work elements to the work stations using the precedence restrictions is as follows.

Step 1. EEb and d are given.

Step 2. The theoretical maximum number of stations, Nm is calculated from

\[ Nm = \left[ \frac{T \times 100}{Tm \cdot EEb} \right] \quad (2) \]

where \( t_i \) = the operation time of work element \( i \)

\( Tm = \max t_i, \) the maximum work element time

\( T = \sum t_i, \) the total work time

\( [\cdot] \): Gaussian symbol.

Therefore the possible range of NN is from Nm to 1, that is, \( 1 < NN < Nm \). For any NN, the following procedures are applied.

Step 3. At the first time NN is given as Nm. After the second time NN is down from Nm by 1, that is, NN=NN-1. Go to next step.

When NN equals 1, stop the procedure.

Step 4. PP to NN and Ebb is calculated by

\[ PP = \frac{T}{NN} \times \frac{100}{EBb} \quad . \quad (3) \]

Ebb is reduced by d from 100 until an assignment to NN is obtained. To any combination (NN,PP), go to next step.

Step 5. At the first time, select the work elements which don't have the precedence elements.

After the second time, add to them the work elements which have only the assigned ones as the precedence ones. Go to next step.

Step 6. Select the work elements, which is less than the remaining cycle time (the slack time) of the station, from among the work elements
selected at step 5. Then go to step 7.

If no work element can be selected here, proceed to the next station and repeat the same step.

If the last station has been examined, go to step 8.

Step 7. Assign the work element having the maximum work time in those elements selected at step 6 to the station. Subtract the work time from the remaining cycle time and eliminate the assigned work element. Then return step 5.

Step 8. When all the elements has been assigned to all the stations, an assignment has been obtained.

The efficiency of the obtained assignment \((Ebo)\) is calculated from

\[
Ebo = \frac{T}{NN \times C} \times 100
\]  

(4)

where \(C\) is the maximum station time of the obtained assignment for \(NN\) and \(PP\)

\(C < PP\).

Because of \(Ebb < Ebo \leq Ebb + d\), since then we search the assignment which shows the maximum efficiency. That is, let \(Ebb = Ebo\) and \(PP = C\). To the new combination of \((NN, PP)\), return step 5.

If one or more work elements remain and an assignment to \(NN\) hadn't been obtained, return step 4 and reduce \(Ebb\) by \(d\), that is, \(Ebb = Ebb - d\). If one or more work elements remain and an assignment to \(NN\) had been ever obtained, return step 3 and reduce \(NN\) by 1.

3. Program

This program is written in Fortran IV and is the form of the subroutine.

The subroutine name is CPLB.

\[
\text{SUBROUTINE CPLB}(Eeb,D,Nwork,Name,Time,Na0,Kindp,Namep,Nstart)
\]

3.1. Usage

The work time and the precedence restrictions among work elements are provided from the table of functional units or the arrow diagram.

3.1.1 Argument List in the case of Table of Functional Units

<table>
<thead>
<tr>
<th>ARGUMENT</th>
<th>I/O</th>
<th>TYPE</th>
<th>SIZE</th>
<th>DEFINITION</th>
</tr>
</thead>
<tbody>
<tr>
<td>EEB</td>
<td>I</td>
<td>REAL</td>
<td>1</td>
<td>Eeb, the minimum efficiency of line balancing</td>
</tr>
<tr>
<td>D</td>
<td>I</td>
<td>REAL</td>
<td>1</td>
<td>d, the constant term by which Ebb is reduced</td>
</tr>
<tr>
<td>NWORK</td>
<td>I</td>
<td>INTEGER</td>
<td>1</td>
<td>number of work elements</td>
</tr>
<tr>
<td>NAME</td>
<td>I</td>
<td>CHARACTER</td>
<td>50X5</td>
<td>name of work element (5A4)</td>
</tr>
</tbody>
</table>
ARGUMENT I/O TYPE SIZE DEFINITION
TIME I REAL 50 \( t_i \), time of work element
NA0 I INTEGER 50 work element's number
KINDP I INTEGER 50 number of parts in work element
NAMEP I INTEGER 50 part's number in work element
NSTART I INTEGER 1 0

where \( C < D < EEB \leq 100 \), \( NWORK \leq 50 \), \( NAME \leq 20 \) characters, \( KINDP \leq 10 \)

3.1.2 Argument List in the case of Arrow Diagram
ARGUMENT I/O TYPE SIZE DEFINITION
EEB I REAL 1 \( EEB \), the minimum efficiency of line balancing
D I REAL 1 \( d \), the constant term by which \( EEB \) is reduced
NWORK I INTEGER 1 number of work elements
NAME I CHARACTER 50X5 name of work element (5A4)
TIME I REAL 50 \( t_i \), time of work element
NA0 I INTEGER 50 preceding event number
KINDP I INTEGER 50 succeeding event number
NAMEP I INTEGER 50 not use in this case
NSTART I INTEGER 1 source node number

where \( 0 < D < EEB \leq 100 \), \( NWORK \leq 50 \), \( NAME \leq 20 \) characters.

3.2. Suggestion on Using
Subroutine GRPUNT or GRPARR, PRINT, and MAXGRP are used in CPLB. GRPUNT is used to select work elements which don't have the precedence works using the table of functional units. GRPARR to do so using the arrow diagram. PRINT to print out the results of the assignment. And MAXGRP to select the work element having the maximum work time from the selected work elements.

Program list is shown in Table 1.

4. Example

The assembly work of a small electric switch is used as an example to execute the program. The work has been analyzed and divided into work elements. The table of the functional units and the list of the arrow diagram have been developed.

Either the table or the list is used as the input data of the precedence restrictions among the work elements. The computer outputs are the same.
The data given from the table of the functional units are shown in Table 2. And the data given from the list of the arrow diagram are shown in Table 3. The computer outputs of this example are shown in Table 4.

References

Table 1. Program Listing

SUBROUTINE CPLB(EEB, D, NWORK, NAME, TIME, NA0, KINDP, NAMEP, NSTART)
DIMENSION NAME(50, 5), TIME(50), NA0(50), KINDP(51), NAMEP(50, 10)
DIMENSION NGROUP(50), NSGRP(50), NSGRPL(50), NAOS(50), NA0SS(50)
DIMENSION CYCLET(50), NKELMT(50), NSTELM(50, 50)
BIG=0.0
SUM=0.0
DO 100 I=1, NWORK
SUM=SUM+TIME(I)
IF(TIME(I) .LE. BIG) GO TO 100
BIG=TIME(I)
100 CONTINUE
STN=SUM/EEB/BIG*100.0
NSTN=STN
90 CONTINUE
IF(NSTN .LE. 0) GO TO 91
EB=1.0
KDOWN=0
THEAN=SUM/FLOAT(NSTN)
55 CONTINUE
TMAX=THEAN/EB
50 CONTINUE
TMIN=THEAN*(2.0-1.0/EB)
DO 101 I=1, NWORK
NAOS(I)=NA0(I)
NA0SS(I)=NA0(I)
101 CONTINUE
DO 110 I=1, NSTN
CYCLET(I)=3.0
NKELMT(I)=0
110 CONTINUE
K=0
NUMGRP=0
NPLACE=0
MAXET=51
KINDP(51)=NSTART
10 CONTINUE
K=K+1
IF(K .GT. NSTN) GO TO 30
L=0
20 CONTINUE
IF(NSTART .LE. 0) GO TO 1
CALL GRPARR(NWORK, NUMGRP, NGROUP, NAOS, NA0SS, KINDP, NA0, MAXET)
GO TO 2
1 CONTINUE
CALL GRPUNT(NWORK, NUMGRP, NGROUP, NAOS, NA0SS, KINDP, NAMEP)
2 CONTINUE
DO 250 I=1, NUMGRP
NUM=NGROUP(I)
IF(NAOS(NUM) .LE. 0) GO TO 250
IF(TIME(NUM) .LE. 0.0) GO TO 61
250 CONTINUE
K1=0
TREST = TMAX - CYCLET(K)
NTREST = TREST * 10000 + 0
TREST = NTREST / 10000
DO 300 1 = I, NUMGRP
   K2 = NGROUP(I)
   IF(NA0S(K2) .EQ. 0) GO TO 300
   IF(TREST .LE. TIME(K2)) GO TO 300
   KI = KI + 1
   I = SGRP(K1) = NGROUP(I)
300 CONTINUE
   IF(K1 .EQ. 0) GO TO 10
   MAXET = MAXGRP(K1, SGRP, TIME)
   CYCLET(K) = CYCLET(K) + TIME(MAXET)
   NKEMLT(K) = NKEMLT(K) + 1
   L = NKEMLT(K)
   NSTELM(K+1) = MAXET
   GO TO 60
61 PAXET = NUM
60 CONTINUE
   NA0S(MAXET) = 0
   NPLACE = NPLACE + 1
   GO TO 20
30 CONTINUE
   IF(NPLACE .GE. NWORK) GO TO 41
   IF(KDOWN .NE. 0) GO TO 40
   ES = EU - (D/100.0)
   GO TO 55
41 CONTINUE
   CALL PKINT(SUM, NSTN, CYCLET, NKEMLT, NSTELM, NA07, NAME, TIME, EU, MAXET,
               & TMIN, TMEAN)
   KDOWN = 1
   GO TO 50
40 CONTINUE
   WRITE(*,6800)
   6800 FORMAT(6, 'DIVIDE THE WORK ELEMENT OR REDUCE THE STATION')
   NSTN = NSTN - 1
   GO TO 90
91 CONTINUE
RETURN
END

C ** SELECT ASSIGNABLE WORK ELEMENT FROM TABLE OF FUNCTIONAL UNITS **
SUBROUTINE GRPUNT(NWORK, NUMGRP, NGROUP, NA0S, NA0S5, KINDP, NAMEP)
DIMENSION NGROUP(50), NA0S(50), NA0S5(50), KINDP(51), NAMEP(50, 10)
DO 200 I = 1, NWORK
   IF(NA0S(I) .EQ. 0) GO TO 200
   KP = KINDP(I)
   DO 210 J = 1, KP
   DO 211 11 = 1, NWORK
      IF(NA0S(11) .EQ. 0) GO TO 211
      IF(NAMEP(I, J) .EQ. NA0S(11)) GO TO 200
211 CONTINUE
210 CONTINUE
   NUMGRP = NUMGRP + 1
   NGROUP(NUMGRP) = I
   NA0S(I) = 0
200 CONTINUE
RETURN
END
**Computer Program of Line Balancing, CPLB**

**C**

**SELECT ASSIGNABLE WORK ELEMENT FROM ARROW DIAGRAM**

**SUBROUTINE GRPARR (NWORK, NUMGRP, NGROUP, NA05, NA05S, NBACK, NFRONT, MAXET)**

**DIMENSION** NGROUP (50), NA05 (50), NA05S (50), NBACK (51), NFRONT (50)

DO 100 I = 1, NWORK

IF (NA05 (I) .LE. 0) GO TO 100

IF (NBACK (I) .EQ. NBACK (MAXET)) GO TO 300

CONTINUE

DO 200 I = 1, NWORK

IF (NA05S (I) .LE. 0) GO TO 200

IF (NFRONT (I) .NE. NBACK (MAXET)) GO TO 200

NUMGRP = NUMGRP + 1

NGROUP (NUMGRP) = I

NA05S (I) = 0

CONTINUE

300 CONTINUE

RETURN

END

**C**

**PRINT OUT THE SOLUTION OF ASSIGNMENT**

**SUBROUTINE PRINT (SUM, NSTN, CYCLET, NKLMT, NSTELM,)**

**& NAME (N0), TIME (50)**

**DIMENSION** CYCLET (50), NKLMT (50), NSTELM (50, 50), NA05 (50),

**& NAME (50, 10), TIME (50)**

WRITE (6, 6100) NSTN, EB, SUM, TIME, TMAX, TMIN, TMEAN

6100 FORMAT (1H ,//, /30X, "THE INITIAL DATA",/10X, "NUM. OF STATION =",

1 15, 5X, EB = "", F5, 3, 5X, SUM = "", F10, 5, 2X, TMEAN = "", F10, 5, 2X,

2 TMAX = "", F10, 5, 2X, TMIN = "", F10, 5, "", 10X, "STATION =", 3X,

3 "CYCLE TIME =", 5X, "WORK ELEMENT =")

BIG = 0.0

DO 600 I = 1, NSTN

WRITE (6, 6200) I, CYCLET (I), NKLMT (I)

6200 FORMAT (10X, I5, 3X, F10, 5, I5)

L = NKLMT (I)

DO 610 J = 1, L

J2 = NSTELM (I, J)

WRITE (6, 6300) NA05 (J2), NAME (J2, I), J3 = 1, 4, TIME (J2)

6300 FORMAT (30X, I5, 5X, 4A4, F10, 4)

610 CONTINUE

IF (CYCLET (I) .LE. BIG) GO TO 600

BIG = CYCLET (I)

600 CONTINUE

EB = SUM / (NSTN * BIG)

AG = FLOAT (NSTN) * BIG / SUM - 1.0

WRITE (6, 6400) BIG, EB, AG

6400 FORMAT (1H ,/20X, "TMAX = ", F10, 5, 5X, "EB = ", F5, 3, 5X, "AG = ", F5, 3)

TMAX = BIG

RETURN

END

FUNCTION MAXGRP (NUMBER, NSET, DATA)

**DIMENSION** NSET (50), DATA (50)

BIG = 0.0

DO 100 I = 1, NUMBER

NUM = NSET (I)

IF (DATA (NUM) .LE. BIG) GO TO 100

MAXGRP = NUMBER

BIG = DATA (NUM)

100 CONTINUE

RETURN
Table 2. Given Data from Table of Functional Units

<table>
<thead>
<tr>
<th>NAME(i,j)</th>
<th>TIME(i)</th>
<th>NA0(i)</th>
<th>KINDP(i)</th>
<th>NAMEP(i,k)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE</td>
<td>32.3</td>
<td>1</td>
<td>7</td>
<td>101 802 803 804 805 306 407</td>
</tr>
<tr>
<td>GUIDE</td>
<td>15.3</td>
<td>2</td>
<td>4</td>
<td>802 408 309 810</td>
</tr>
<tr>
<td>BUTTON</td>
<td>5.8</td>
<td>3</td>
<td>3</td>
<td>811 2 803</td>
</tr>
<tr>
<td>TERMINAL 1</td>
<td>12.6</td>
<td>4</td>
<td>3</td>
<td>112 313 805</td>
</tr>
<tr>
<td>TERMINAL 2</td>
<td>12.6</td>
<td>5</td>
<td>3</td>
<td>112 313 805</td>
</tr>
<tr>
<td>TERMINAL 3</td>
<td>12.6</td>
<td>6</td>
<td>3</td>
<td>112 313 805</td>
</tr>
</tbody>
</table>

NSTART = 0

Table 3. Given Data from List of Arrow Diagram

<table>
<thead>
<tr>
<th>NAME(i,j)</th>
<th>TIME(i)</th>
<th>NA0(i)</th>
<th>KINDP(i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASE</td>
<td>32.3</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>GUIDE</td>
<td>15.3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>BUTTON</td>
<td>5.8</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>TERMINAL 1</td>
<td>12.6</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>TERMINAL 2</td>
<td>12.6</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>TERMINAL 3</td>
<td>12.6</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>DUMMY</td>
<td>0.0</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>DUMMY</td>
<td>0.0</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>DUMMY</td>
<td>0.0</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

NSTART = 1

** Arrow Diagram of the Small Electric Switch**

BASE (1) (32.3 D.M.)

<table>
<thead>
<tr>
<th>GUIDE (2) (15.3 D.M.)</th>
<th>BUTTON (3) (5.8 D.M.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>TERMINAL 1 (4) (12.6 D.M.)</td>
<td>DUMMY (0.0 D.M.)</td>
</tr>
<tr>
<td>TERMINAL 2 (5) (12.6 D.M.)</td>
<td>DUMMY (0.0 D.M.)</td>
</tr>
<tr>
<td>TERMINAL 3 (6) (12.6 D.M.)</td>
<td>DUMMY (0.0 D.M.)</td>
</tr>
</tbody>
</table>
### Table 4. Computer Output

Table 4 shows the computer output for different scenarios with varying numbers of stations and cycle times. The table includes the number of stations, cycle times, work elements, and their corresponding TMAX and TMIN values for each scenario.

#### **The Initial Data**

<table>
<thead>
<tr>
<th>NUM. OF STATION</th>
<th>CYCLE TIME</th>
<th>WORK ELEMENT</th>
<th>TMAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>32.3000</td>
<td>BASE</td>
<td>32.3000</td>
</tr>
</tbody>
</table>

#### **The Initial Data**

<table>
<thead>
<tr>
<th>NUM. OF STATION</th>
<th>CYCLE TIME</th>
<th>WORK ELEMENT</th>
<th>TMAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>44.9000</td>
<td>BASE</td>
<td>32.3000</td>
</tr>
</tbody>
</table>

#### **The Initial Data**

<table>
<thead>
<tr>
<th>NUM. OF STATION</th>
<th>CYCLE TIME</th>
<th>WORK ELEMENT</th>
<th>TMAX</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>91.2000</td>
<td>BASE</td>
<td>32.3000</td>
</tr>
</tbody>
</table>

---

*1 Nm = NN, *2 Eb/100, *3 T = \( \frac{1}{n} \), *4 PP, *5 tm = \( \text{max} \),

*6 C

*7 Eb/100 = T/(NN X C),

*8 AG = (NN X C - T)/ T.