Memoirs of the School of Engineering, Okayama University, Vol. 21, No.1, November 1986

A Computer Program of Assembly Line Balancing Considering the Performance Rate of Each Work Station

Fumio AKAGI* and Hirokazu OSAKI*

(Received September 17, 1986)

SYNOPSIS

Assembly line balancing is to assign work elements to serial work stations so as to make the work content at each station as close as possible to one limiting cycle time or pitch time, i.e., an upper time limit over every station.

Until now, it is usually assumed that the performance rates of work stations are constant. But in practice the performance abilities of workers, machines or robots are varied by their own working conditions. Then the actual station times are different from standard ones, and consequently the line balance may diminish in many cases.

Therefore in this paper, we propose an improved balancing method, in which work elements can be assigned to the work station having the upper time limit changed by its performance rate or ability.

Further we develop the computer program of the proposed method and provide an illustrative problem and computational results.

In an application of our method to the practical problems, it is shown that the actual efficiency of the production line becomes near that planned.

---

* Department of Industrial Science

1. INTRODUCTION

The basic problem in assembly line balancing is to assign work elements to the work stations so as to balance the work content of each station under the precedence restrictions among work elements.

Various computer programs have been developed and used to solve this problem. For example, Ignall(1965)[1] and Arcus(1966)[2] summarized the important algorithms and techniques for the solution of the assembly line balancing.

They suggested some problems in more complex form for the future, which are product mix, variable performance times, tasks larger than cycle time, men of unequal abilities and so on. Two former problems, that is, variable performance times and product mix have been considered by several authors, that is, Moodie and Young (1965)[3], Ramsing and Downing (1970)[4], Kottas and Lau (1973)[5], (1981)[6], Sphicas and Silverman (1976)[7], Thomopoulos (1967)[8], (1970)[9], Macaskill (1972)[10], Dar-el and Cother (1975)[11], Dar-el and Nadivi (1981)[12], etc. For one of the other problems, that is, tasks larger than cycle time, we also presented the heuristic computer program for assembly line balancing with more than one worker in each station (Akagi et al., 1983)[13]. But the last problem, that is, the unequal ability has been less considered.

By the way, in our assembly line balancing methods until now,[14, 15, 16] we assumed that the performance rates of every work station are constant and identical. The deterministic standard times for work elements were used, and work elements were assigned to work stations so that the sum of the assigned work elements' standard times was as close as possible to one limiting cycle time or its multiples.

But in practice, the performance abilities of work stations are varied by their own working conditions. Then the actual station times are different from their standard times, and consequently the line balance may diminish in many cases.

From above reasons, we should consider the unequal ability of work station in designing the assembly line. Therefore in this paper, we propose an improved balancing method, in which work elements can be assigned to the work station having the upper time limit changed by its performance rate or ability. Further we develop the computer program of the proposed method and provide an illustrative problem and computational results.

2. PROPOSED METHOD

2.1 Notations

We use the following notations for the assembly line terminology.

Production schedule;

- Cu: limiting cycle time, reciprocal of production rate
- No: desirable number of work stations
- Ebv: operating value for efficiency of line balancing
- d: decrement parameter of Ebv

Work elements data;

- n: number of work elements
- w_k: kth work element in the element list
- t_k: work element time of w_k (by standard time)
- P_k: set of work elements preceding to w_k
- M_k: performance restrictions of w_k
- $k=1,\dots,n$: work element serial number
- $T=\sum_{k=1}^{n} t_k$: total work content per unit product
- $tmax=\max\{t_k, k=1,\dots,n\}$: maximum work element time

Work stations data;

- N: number of work stations
- R_i: performance rate of ith station (coefficient multiplied)
- D_i: set up time for ith station (constant reduced or added)
- Cu_i: limiting station time of ith station

Assignment results;

- T''_i: effective station time of ith station
- T_i: station time of ith station by standard times
- n_i: number of work elements in ith station
- w_{ij}: jth work element in ith station
- t_{ij}: work element time of w_{ij} (standard time)
- $j=1,\dots,n_i$: work element serial number in ith station
- $i=1,\dots,N$: station serial number
- C: cycle time of the line
- Eb: Efficiency of line balancing

2.2 Relations and restrictions

Two kinds of values are used for the performance ability of each work station or worker in the proposed method. One is the coefficient multiplied by the station standard time. It is defined as the rate of the normal time to the actual one and denoted by R_i in ith station. We assume that R_i is the rating value of the worker manned ith station and a particular worker's rate value is the same over every work element or any assembly of work elements.

The other is the constant reduced from or added to the station standard time. It is denoted by D_i. For example, D_i is used for considering the setup time of ith station. If the work-in-process handling time from the conveyor belt is different among stations, we can use D_i profitably.

Then there are some relations and restrictions among symbols.

$$Cu_i = (Cu - D_i) \times R_i \quad (1)$$

$$Cu = Cu_i/R_i + D_i \quad (1)'$$

$$T_i = \sum_{j=1}^{n_i} t_{ij} \quad (2)$$

where $Cu_i > T_i$ (3)

$$T''_i = T_i/R_i + D_i \quad (4)$$

when $Cu > T''_i$ (5)

$$C = \max\{T''_i, i=1,\ldots,N\} \quad (6)$$

$$Eb = \sum_{i=1}^{N} T''_i/(N \times C) \quad (7)$$

2.3 Problem formulation

One goal of assembly line balancing is to minimize the balance loss or to maximize the efficiency of line balancing as follows;

$$\min\{ N \times C - \sum_{i=1}^{N} T''_i \} \quad (8)$$

or $\max\{ Eb \} = \max\{ \sum_{i=1}^{N} T''_i/(N \times C)\}$ (9)

Here equations (8)(9) include the cycle time C, which is calculated from the assignment solution. Then in assigning procedures to get the result, we use Cu instead of C;

$$\min\{ N \times Cu - \sum_{i=1}^{N} T''_i \} \quad (10)$$

In the heuristic line balancing method to get the solution quickly and easily, work elements are assigned to the first or last work station and the assignment is continued to next stations till all the element is assigned.

Then the objective function in ith station is as follow;

$$\min\{ Cu - T''_i \} \quad (11)$$

Equation (11) leads to the next one by substituting equations (1)', (4) and multiplying by R_i.

$$\min\{ Cu_i - T_i \}=\min\{ Cu_i - \sum_{j=1}^{n_i} t_{ij} \} \quad (12)$$

Therefore we obtain the sub set of work elements $\{w_{ij}, j=1,\ldots,n_i\}$ under the conditions (3) and (12) in every station, $i=1,\ldots,N$.

2.4 Solution procedure

The proposed procedure of assigning work elements to work stations under their own performance rates is as follows.

Step 1. Input the work elements data $\{n, w_k, t_k, P_k, M_k, k=1,\ldots,n\}$ and the decrement parameter d .

Step 2. Input the work stations data $\{No, R_i, D_i, i=1,\ldots,No\}$.

Step 3. Select one of the Branching Decision(B&D) rules among the followings.

forward* / backward*
- *largest task time rule
- *smallest task time rule
- *highest ranked positional weight rule
- *lowest flexible rate rule
- *random selecting rule

And set the ideal on the operational value for efficiency of line balancing, that is, let $Ebv=1.0$.

Step 4. Let $Ebv = Ebv - d$, and calculate the upper time limit over the line, that is, $Cu = T/(No \times Ebv)$.

Step 5. Let $R=\{w_k, k=1,\ldots,n\}$, where R is the set of still unassigned work elements, and $i=0$.

Step 6. Go to next station, that is, $i=i+1$, and set the initial data in ith station, that is, $n_i=0,\ T_i=0,\ j=0$.

Step 7. Let $j=j+1$. Using the B&D rule, select w_{ij} under the following four conditions.

(1) $w_{ij} \in R$ (unassigned work elements)

(2) $\{w_{ij} \mid P_{ij} \cap R=\phi\}$ (workable work elements with preceding relations)

(3) $\{w_{ij} \mid t_{ij} < Cu_i - T_i\}$
(assignable work elements with the bound of slack times)

(4) $\{w_{ij} \mid (M_{ij}=M_{ik}, k=1,\ldots,j-1) \cup (M_{ij}=\phi)\}$
(assignable work elements with the performance restrictions)

If no elements are selected here, return to step 6.

Step 8. Assign w_{ij} to ith station, that is, $T_i=T_i+t_{ij},\ n_i=j$, $R=R-\{w_{ij}\}$. If $R \neq \phi$, return step 7. If $R=\phi$, all work elements are assigned. Then go to next step.

Step 9. Let $N=i$. Now we get the following result; N , T_i , w_{ij} , $j=1,\ldots,n_i$, $i=1,\ldots,N$. If $N > No$, then go to next step and if $N \leqq No$, we get the possible result. Calculate the values by the following equations and remember the assignment result.

$$T''_i=T_i/R_i+D_i, \quad C=\max\{T''_i, i=1,\ldots,N\}, \quad Eb=\sum_{i=1}^{N} T''_i/(N\times C)$$

To search the better result, let $Cu=C$ and return to step 5.

Step 10. If we cannot still get the possible result of No , return to step 4. If we already got a result, we could not have the better result by diminishing the cycle time C. If we want to improve the result by the other B&D rules, change the rule and return to step 5. If we want to get the other result, return to step 3. Not doing so, we stop the procedure.

2.5 Computer program

The computer flow chart of the proposed procedure is shown in Fig. 1 and its computer program written by BASIC language is shown in Appendix 1. It is tested a NEC PC-9801 and a SHARP MZ-80B. A slight modification of the listed program may be necessary for the users of other types of microcomputer.

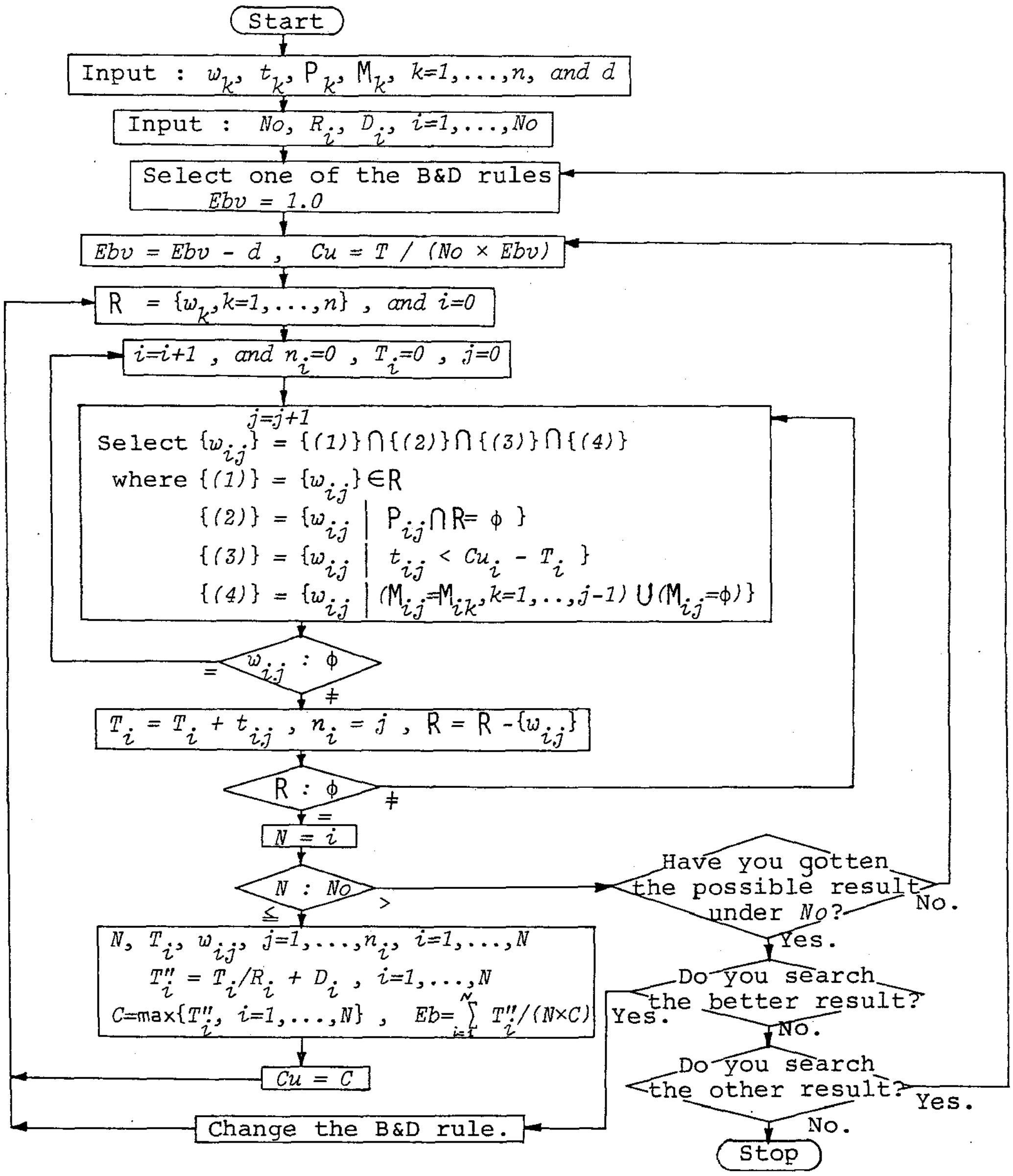


Fig.1 Flow chart of the assignment method proposed

3. APPLICATION EXAMPLE

To explain the proposed procedure, we apply it to solve the problem for assembly works of table tap concenter, whose precedence diagram is shown in Fig. 2. Time values in the diagram are standard assembly times calculated by Method Time Measurement.[17]

We intended to obtain the better line balance of this assembly work with three arbitrary workers.

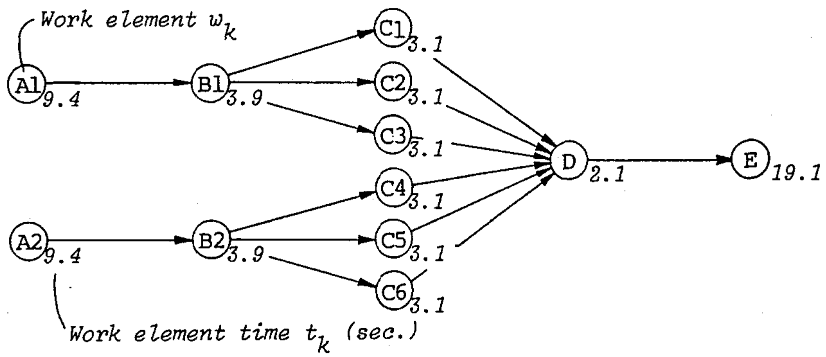


Fig.2 Precedence diagram of assembly work for table tap concenter

3.1 Result under no consideration of worker's ability

We used the proposed procedure with forward and largest task time B&D rule. At the first, we set the station data as follows;

$No=3,\ R_i=1,\ D_i=0,\ i=1,2,3,$

because each worker's ability or performance rate was unknown and every jobs could be done in the normal work area along the conveyor.

Then the assignment result under no consideration of worker's ability was shown in Fig. 3. The station standard times were T_1=22.7, T_2=22.5, T_3=21.2 sec. The efficiency of line balancing was 97.5% and its pitch time was 22.7 sec. The productivity was 159 units per hour.

A production line manning three arbitrary workers was designed on basis of the assignment result and then the production started under unpaced working condition. At the same time, the station cycle times were measured. The average cycle times of 5 cycles were shown in Fig. 3 with the standard as the form of pitch diagram. T'_i(1-5) shows the average time from 1st to 5th cycle in ith station.

As the result, T'_1(1-5)=29.8, T'_2(1-5)=18.0, T'_3(1-5)=22.6 sec., the practical cycle time was 29.8 sec. and Eb was 78.7%. The practical productivity was 121 units per hour. The line balance decreased from 97.5%(planned) to 78.7%(measured) by 18.8%. The productivity dropped from 159 units per hour to 121 units/hour by 38 units/hour.

Then we would revise the line balance to improve the productivity. We would assign the work elements to three work stations considering the average time of each station in next section.

3.2 Result under considering the performance rate of each work station

The performance rates of each worker were calculated from the result of Fig. 3, that is, the ratio of the station standard time to the average time measured.

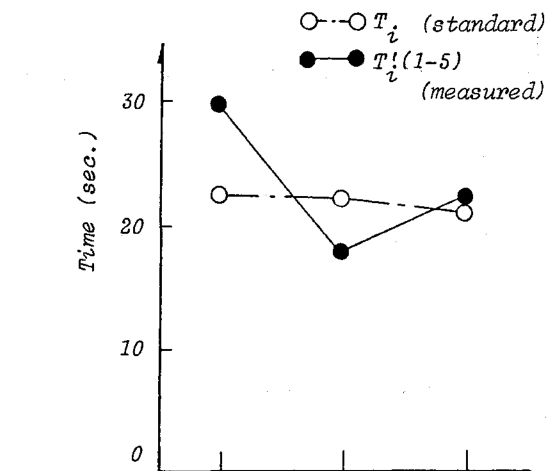
Then we revised the station data as follows;

$R_i = T_i / T'_i(1\text{-}5)$, $i = 1,2,3$, ($R_1 = 0.762$, $R_2 = 1.25$, $R_3 = 0.938$)

and assigned work elements to three work stations considering their performance rates. The obtained result was shown in Fig. 4 and Appendix 2. Some work elements were exchanged between 1st and 2nd station. The estimated station times were $T''_1 = 21.6$, $T''_2 = 23.0$, $T''_3 = 22.6$. C was 23.0 sec. and Eb was 97.2% in the planned line.

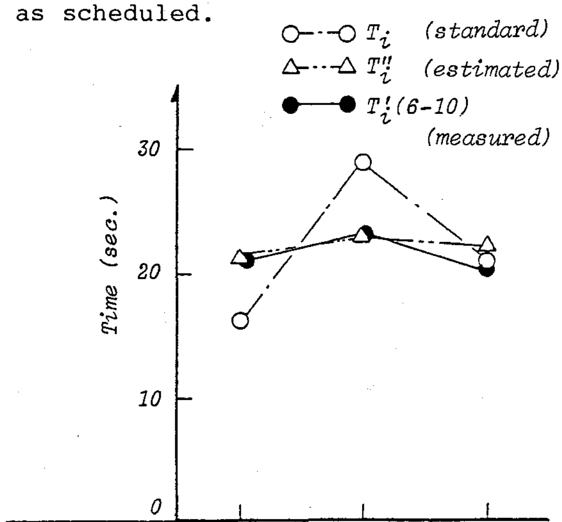
The assembly works continued under the rebalanced line. The average cycle time of next 5 cycles were shown in Fig. 4 with the estimated plan.

As the result, $T'_1(6\text{-}10) = 21.2$, $T'_2(6\text{-}10) = 23.6$, $T'_3(6\text{-}10) = 20.4$ sec., C was 23.6 sec. and Eb was 92.1%. The assembly works from 6th to 10th cycle were nearly proceeded as scheduled.



Station i	1	2	3
Assigned work element w_{ij}	A1 A2 B1	B2 C1 | C6	D E

Fig.3 Pitch diagram(1st 5 cycles)



Station i	1	2	3
Assigned work element w_{ij}	A1 B1 C1	A2 B2 C2 | C6	D E

Fig.4 Pitch diagram(next 5 cycles)

4. SAMPLE RUN ON NEC PC-9801

To explain the usage of the program, we apply it to solve the problem of the section 3.2. Once we load the program using the floppy disk system, we are ready to run it. By typing in "RUN" and hitting the "ENTER" key, the program is self-documenting. The series of questions and answers, and computer outputs are shown in Appendix 2.

At the first step, we enter work elements data from key board. Computer displays the list of input data. If they are correct, go to next stage and computer calculates the ranked positional weights for each work element. Then the full work element list is displayed.

We are now ready to balance the assembly line for this product. At the second step, we input the work stations data, that is, performance rates of each work station. In this example of Appendix 2, input as follows; No=3, R_1=0.762, D_1=0, R_2=1.25, D_2=0, R_3=0.938, D_3=0. Next let d=0.05 and select Forward Largest Task Time Rule as the B&D rule.

Then the computer outputs the assignment result.

REFERENCES

(1) E. J. Ignall : A review of assembly line balancing, The Journal of Industrial Engineering, 16(1965), 244.

(2) A. L. Arcus : COMSOAL—A computer method of sequencing operations for assembly lines, International Journal of Production Research, 4(1966), 259.

(3) C. L. Moodie, and H. H. Young: A heuristic method of assembly line balancing for assumptions of constant or variable work element times, The Journal of Industrial Engineering, 16(1965), 23.

(4) K. Ramsing, and R. Downing : Assembly line balancing with variable element times, Industrial Engineering, 2(1970), 41.

(5) J. F. Kottas, and H. Lau : A cost-oriented approach to stochastic line balancing, AIIE Transactions, 5(1973), 164.

(6) J. F. Kottas, and H. Lau : A stochastic line balancing procedure, International Journal of Production Research, 19(1981), 177.

(7) G. P. Sphicas, and F. N. Silverman : Deterministic equivalents for stochastic assembly line balancing, AIIE Transactions, 8(1976),280.

(8) N. T. Thomopoulos : Line balancing-sequencing for mixed-model assembly, Management Science, 14(1967), B-59.

(9) N. T. Thomopoulos : Mixed model line balancing with smoothed

station assignments, Management Science, 16(1970), 593.

(10)J. L. C. Macaskill : Production-line balances for mixed-model lines, Management Science, 19(1972), 423.

(11)E. M. Dar-el, and R. F. Cother: Assembly line sequencing for model mix, International Journal of Production Research, 13(1975), 463.

(12)E. M. Dar-el, and A. Nadivi: A mixed-model sequencing application, International Journal of Production Research, 19(1981), 69.

(13)F. Akagi, H. Osaki, and S. Kikuchi : A method for assembly line balancing with more than one worker in each station, International Journal of Production Research, 21(1983), 755.

(14)F. Akagi, H. Osaki, and S. Kikuchi : Computer program of line balancing regarding efficiency and number of stations as variables, Memoirs School Eng., Okayama Univ., 13(1979), 83.

(15)F. Akagi, H. Osaki, and S. Kikuchi : The method of analysis of assembly work based on the fastener method, Bulletin of the JSME, 23(1980), 1670.

(16)F. Akagi, H. Osaki, and S. Kikuchi: 1981, Computer program of line balancing under the multiple workers in each station, Memoirs School Eng., Okayama University, 15(1981), 21.

(17)H. B. Maynard:METHOD-TIME MEASUREMENT(in Japanese),GIHOUDOU(1958).

APPENDIX 1. Program list.

```
1100 REM***Enter Work Element's Data*******************
1110 PRINT"Enter work element's data from key board."
1120 INPUT" How many work element (n)";N
1130 DIM WK$(N),TK(N)
1140 DIM PK(N,N),PK$(N,N),MK(N)
1150 DIM WD(N),WG(N)
1160 FOR K=1 TO N
1170 INPUT"Name of element (Wk)";WK$(K)
1180 INPUT"Time of element (tk)";TK(K)
1190 INPUT"Constraints No. (Mk)";MK(K)
1200 INPUT" No. of pre works";PK(K,0)
1210 IF PK(K,0)=0 THEN 1250
1220 FOR B=1 TO PK(K,0)
1230 INPUT"Name of pre work (pk)";PK$(K,B)
1240 NEXT B
1250 PRINT:NEXT K
1300 REM***********************************************
1310 REM*****DIMENSION SECTION*************************
1320 REM***********************************************
1330 DIM TI(N),NI(N),WI(N,10)
1400 DIM BB$(5)
1500 REM***Display Input Data**************************
1510 PRINT:PRINT"  Work Elements List"
1520 PRINT"  k Wk        tk      Mk     Pk"
1530 PRINT"--------------------------------"
1540 T=0
1550 TM=0
1560 FOR K=1 TO N
1570 T=T+TK(K)
1580 IF TM<=TK(K) THEN TM=TK(K)
1590 PRINT"(";K;")";WK$(K);TAB(10);TK(K);TAB(18);MK(K);TAB(24)
1600 IF PK(K,0)=0 THEN 1720
1610 FOR B=1 TO PK(K,0)
1620 PRINT PK$(K,B);" ";
1630 FOR KD=1 TO N
1640 IF WK$(KD)=PK$(K,B) THEN 1680
1650 GOTO 1700
1660 PRINT" Precedence relation is error.":STOP
1680 PK(K,B)=KD
1690 KD=N
1700 NEXT KD
1710 NEXT B
1720 PRINT:NEXT K
1740 PRINT:PRINT" T=";T;"  tmax=";TM
1750 INPUT"If you want to calculate DATA, hit b.";B$
1760 IF B$<>"b" THEN STOP
```

```
2000 REM***Calculation of work element's DATA********
2010 DIM SK(N,N),UK(N),IU(N),VK(N),IV(N)
2020 REM***Get the succeeding elements Sk************
2030 FOR K=1 TO N:SK(K,0)=0:NEXT K
2060 FOR K=1 TO N
2070 IF PK(K,0)=0 THEN 2110
2080 FOR B=1 TO PK(K,0)
2090 KD=PK(K,B):SK(KD,0)=SK(KD,0)+1:SK(KD,SK(KD,0))=K
2100 NEXT B
2110 NEXT K
2200 REM***Calculate ranked positional weights*********
2210 FOR K=1 TO N:WD(K)=SK(K,0):VK(K)=TK(K)
2220 UK(K)=SK(K,0):NEXT K
2260 FOR K=1 TO N
2270 IF WD(K)>0 THEN 2320
2280 NEXT K
2290 GOTO 3000
2320 FOR K=1 TO N
2330 IF WD(K)=<0 THEN 2750
2340 FOR B=1 TO SK(K,0)
2350 FOR KD=1 TO N
2360 IF WD(KD)=<0 THEN 2380
2370 IF SK(K,B)=KD THEN 2750
2380 NEXT KD
2390 NEXT B
2460 FOR B=1 TO SK(K,0)
2470 FOR KD=1 TO N
2480 IF SK(K,B)<>KD THEN 2720
2490 VK(K)=VK(K)+TK(KD)
2500 IF UK(KD)=0 THEN 2720
2510 FOR L=1 TO UK(KD)
2520 FOR M=1 TO UK(K)
2530 IF SK(KD,L)=SK(K,M) THEN 2710
2540 NEXT M
2650 UK(K)=UK(K)+1
2660 SK(K,UK(K))=SK(KD,L)
2670 VK(K)=VK(K)+TK(SK(KD,L))
2680 GOTO 2710
2710 NEXT L
2720 NEXT KD
2730 NEXT B
2740 WD(K)=-1
2750 NEXT K
2760 GOTO 2260
```

```
3000 REM***Calculate inverse ranked positional weights*
3010 FOR K=1 TO N
3020 WD(K)=PK(K,0)
3030 IV(K)=TK(K)
3040 IU(K)=PK(K,0)
3050 NEXT K
3060 FOR K=1 TO N
3070 IF WD(K)>0 THEN 3120
3080 NEXT K
3090 GOTO 4000
3120 FOR K=1 TO N
3130 IF WD(K)=<0 THEN 3440
3140 FOR B=1 TO PK(K,0)
3150 FOR KD=1 TO N
3160 IF WD(KD)=<0 THEN 3230
3170 IF PK(K,B)<>KD THEN 3230
3220 GOTO 3440
3230 NEXT KD
3240 NEXT B
3250 FOR B=1 TO PK(K,0)
3260 FOR KD=1 TO N
3270 IF PK(K,B)<>KD THEN 3410
3280 IV(K)=IV(K)+TK(KD)
3290 IF IU(KD)=0 THEN 3410
3300 FOR L=1 TO IU(KD)
3310 FOR M=1 TO IU(K)
3320 IF PK(KD,L)<>PK(K,M) THEN 3360
3350 GOTO 3400
3360 NEXT M
3370 IU(K)=IU(K)+1
3380 PK(K,IU(K))=PK(KD,L)
3390 IV(K)=IV(K)+TK(PK(KD,L))
3400 NEXT L
3410 NEXT KD
3420 NEXT B
3430 WD(K)=-1
3440 NEXT K
3450 GOTO 3060
```

```
4000 REM***Display full element's DATA*****************
4010 UK(0)=0
4020 FOR K=1 TO N
4030 UK(0)=UK(0)+UK(K)
4040 NEXT K
4050 PRINT
4060 PRINT
4070 PRINT"   Work elements list"
4080 PRINT" k  Wk        tk     Mk  Pk              Sk";
4090 PRINT"             Vk     IVk    Uk   IUk"
4100 PRINT"----------------------------------------";
4110 PRINT"----------------------------------"
4120 FOR K=1 TO N
4130 PRINT"(";K;")";WK$(K);TAB(10);TK(K);TAB(18);MK(K);
4140 IF PK(K,0)=0 THEN 4180
4150 PRINT TAB(22);:FOR B=1 TO PK(K,0)
4160 PRINT WK$(PK(K,B));" ";
4170 NEXT B
4180 IF SK(K,0)=0THEN 4220
4190 PRINT TAB(35);:FOR B=1 TO SK(K,0)
4200 PRINT WK$(SK(K,B));" ";
4210 NEXT B
4220 PRINT TAB(47);VK(K);TAB(53);IV(K);TAB(61);UK(K);
4230 PRINT TAB(66);IU(K)
4240 NEXT K
4250 PRINT"----------------------------------------";
4260 PRINT"-----------------------------"
4270 B=N*(N-1)/2
4280 B=(B-UK(0))/B
4290 PRINT"  T=";T;"  tmax=";TM;"  F=";B
4300 INPUT" If you want to assign elements, hit B.";B$
4310 IF B$="b" THEN 5000
4320 STOP
```

```
5000 REM***Balancing method 3*************************
5010 PRINT"Line balancing with considering performance";
5020 PRINT"rates of work stations."
5100 PRINT:REM***Input No, Ri, Di,i=1,...,No***********
5110 INPUT" Set No. No=";NO
5120 DIM RI(N),DI(N)
5200 PRINT:PRINT" Enter performance rate & set time."
5210 FOR I=1 TO NO
5220 PRINT:PRINT" Performance rate R(";I;")";
5230 INPUT RI(I)
5240 PRINT"             Set time D(";I;")";
5250 INPUT DI(I)
5260 NEXT I
5420 REM***Enter d************************************
5430 PRINT
5440 PRINT"Enter the constant d which you decrease Ebv."
5450 INPUT" d =";D
5500 REM***Assign work elements to work stations*******
5510 GOSUB 9700
5520 L1=0
5530 EB=1
5540 EB=EB-D
5550 PU=T/NO/EB
5560 IF PU>TM GOTO 5580
5570 GOTO 5540
5580 GOSUB 7000
5590 IF NS > NO THEN 5700
5600 L1=L1+1:GOSUB 6800
5610 PU=P
5620 IF PU =< TM THEN STOP
5630 GOTO 5580
5700 IF L1 < 1 THEN 5540
5710 PRINT"We can't get better result by this B&B rule."
5720 PRINT"Do you want to search the better result?"
5730 INPUT"   y or n";B$
5740 IF B$="n" THEN 5820
5750 PRINT"Select one of the following two."
5760 PRINT" 1.Search better result by other B&B rule."
5770 PRINT" 2.Search another result by other B&B rule."
5780 INPUT"  Input choice  ";B$
5790 IF VAL(B$)=2 THEN 5510
5800 IF VAL(B$)<>1 THEN 5750
5810 GOSUB 9700:GOTO 5580
5820 STOP
```

```
6800 REM***Print out the result************************
6810 PRINT:PRINT"B&B rule= ";BB$;"  ";BB$(BB)
6820 PRINT"   No =";NO;" Pu =";PU
6830 PRINT" i   Ti'(Ti)     Wij    tij   Ri Di"
6840 P=0
6850 TT=0
6860 FOR I0=1 TO NS
6870 I=I0
6880 IF BB$="b" THEN I=NS-I0+1
6890 TD=TI(I)/RI(I)+DI(I):TT=TT+TD:IF P<TD THEN P=TD
6900 PRINT I0;" ";TD;"(";TI(I);")";TAB(30);RI(I);DI(I)
6910 FOR J=1 TO NI(I):B=WI(I,J)
6915 PRINT TAB(17);WK$(B);TAB(23);TK(B):NEXT J
6920 NEXT I0
6930 EB=TT/NS/P
6940 PRINT"Pitch=";P;"   Eb=";EB
6950 RETURN

7000 REM***SUBROUTINE(Assigning Method 4)**************
7020 IF BB$<>"f" THEN 7070
7030 FOR K=1 TO N
7040 WD(K)=PK(K,0)
7050 NEXT K
7060 GOTO 7100
7070 FOR K=1 TO N
7080 WD(K)=SK(K,0)
7090 NEXT K
7100 R=0
7110 I=0
7150 I=I+1:IF I > NO THEN 7400
7160 NI(I)=0
7170 TI(I)=0
7180 MR=0
7190 PI=(PU-DI(I))*RI(I)
7200 GOSUB 8500
7210 IF A<=0 THEN PRINT" Wk DATA ERROR!":STOP
7220 GOSUB 8600
7230 IF A<=0 GOTO 7150
7240 GOSUB 8700
7250 IF A<=0 GOTO 7150
7260 ON BB GOSUB 9010,9100,9200,9300,9500
7300 TI(I)=TI(I)+TK(WA)
7310 NI(I)=NI(I)+1
7320 J=NI(I)
7330 WI(I,NI(I))=WA
7340 WD(WA)=-1:R=R+1
7350 PI=PI-TK(WA)
7351 PI=PI*10000:PI=INT(PI)/10000
7360 IF MK(WA)>0 THEN MR=MK(WA)
7370 GOSUB 8900
7380 IF R < N THEN 7200
7400 NS=I
7410 RETURN

8500 REM***Select {Wa} with precedence relations*******
8510 A=0
8520 FOR K=1 TO N
8530 IF WD(K)<>0 THEN 8560
8540 A=A+1
8550 WG(A)=K
8560 NEXT K:RETURN
8600 REM***Select {Wa} with time restrictions**********
8610 C=0
8620 FOR B=1 TO A
8630 IF TK(WG(B))>=PI GOTO 8660
8640 C=C+1
8650 WG(C)=WG(B)
8660 NEXT B
8670 A=C
8680 RETURN
8700 REM***Select {Wa} with performance restrictions***
8710 IF MR=0 THEN RETURN
8720 C=0
8730 FOR B=1 TO A
8740 IF MK(WG(B))<=0 GOTO 8760
8750 IF MR<>MK(WG(B)) GOTO 8780
8760 C=C+1
8770 WG(C)=WG(B)
8780 NEXT B
8790 A=C
8800 RETURN

8900 REM***Remove Wa in Pk,k=1,...,n***********
8910 FOR K=1 TO N
8920 IF WD(K)<=0 THEN 8960
8925 IF BB$<>"f" THEN 8945
8930 FOR B=1 TO PK(K,0)
8935 IF PK(K,B)=WA THEN WD(K)=WD(K)-1
8940 NEXT B:GOTO 8960
8945 FOR B=1 TO SK(K,0)
8950 IF SK(K,B)=WA THEN WD(K)=WD(K)-1
8955 NEXT B
8960 NEXT K
8970 RETURN

9000 REM***Branching Rules********************
9010 REM*****Largest time*********************
9020 WA=WG(1)
9030 C=TK(WA)
9040 FOR B=1 TO A
9050 IF TK(WG(B))>C THEN WA=WG(B):C=TK(WA)
9060 NEXT B
9070 RETURN
9100 REM*****Smallest time********************
9110 WA=WG(1)
9120 C=TK(WA)
9130 FOR B=1 TO A
9140 IF TK(WG(B))<C THEN WA=WG(B):C=TK(WA)
9150 NEXT B
9160 RETURN
9200 REM*****Random select********************
9210 C=INT((A-1)*RND(1)+1)
9220 WA=WG(C)
9230 RETURN
9300 REM*****Highest PW**********************
9310 IF BB$="b" GOTO 9410
9320 WA=WG(1)
9330 C=VK(WA)
9340 FOR B=1 TO A
9350 IF VK(WG(B))<C THEN WA=WG(B):C=VK(WA)
9360 NEXT B
9370 RETURN
9400 REM*****Inverse HPW*********************
9410 WA=WG(1)
9420 C=IV(WA)
9430 FOR B=1 TO A
9440 IF IV(WG(B))<C THEN WA=WG(B):C=IV(WA)
9450 NEXT B
9460 RETURN
9500 REM***Lowest flexible********************
9510 IF BB$="b" GOTO 9600
9520 WA=WG(1)
9530 C=UK(WA)
9540 FOR B=1 TO A
9550 IF UK(WG(B))<C THEN WA=WG(B):C=UK(WA)
9560 NEXT B
9570 RETURN
9600 REM*****Inverse lowest flexible**********
9610 WA=WG(1)
9620 C=IU(WA)
9630 FOR B=1 TO A
9640 IF IU(WG(B))<C THEN WA=WG(B):C=IU(WA)
9650 NEXT B
9660 RETURN
9700 REM***Select assignment methods**********
9710 PRINT:REM***Forward or Backward**********
9720 PRINT"Select Forward or Backward Assignment."
9730 INPUT"  Hit f or b ";BB$
9740 IF (BB$<>"f")*(BB$<>"b") THEN 9730
9750 PRINT:REM***Select one of B&B rules*********
9760 PRINT"Select one of the following B&B rules."
9770 PRINT" 1  Largest task time rule."
9775 BB$(1)="Largest task time rule."
9780 PRINT" 2  Smallest task time rule."
9785 BB$(2)="Smallest task time rule."
9790 PRINT" 3  Random selecting rule."
9795 BB$(3)="Random selecting rule."
9800 PRINT" 4  Highest ranked positional weight rule."
9805 BB$(4)="Highest ranked positional weight rule."
9810 PRINT" 5  Lowest flexible rate rule."
9815 BB$(5)="Lowest flexible rate rule."
9820 INPUT" Input choice ";B$
9830 BB=VAL(B$)
9840 IF(BB>0)*(BB<6) GOTO 9860
9850 PRINT" Not select.":GOTO 9760
9860 RETURN
```

APPENDIX 2. Result of sample run.

```
RUN
Enter work element's data from key board.
 How many work element (n)? 12
Name of element (Wk)? A1
Time of element (tk)? 9.4
Constraints No. (Mk)? 0
 No. of pre works? 0

Name of element (Wk)? A2
Time of element (tk)? 9.4
Constraints No. (Mk)? 0
 No. of pre works? 0

Name of element (Wk)? B1
Time of element (tk)? 3.9
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? A1

Name of element (Wk)? B2
Time of element (tk)? 3.9
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? A2

Name of element (Wk)? C1
Time of element (tk)? 3.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? B1

Name of element (Wk)? C2
Time of element (tk)? 3.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? B1

Name of element (Wk)? C3
Time of element (tk)? 3.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? B1

Name of element (Wk)? C4
Time of element (tk)? 3.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? B2

Name of element (Wk)? C5
Time of element (tk)? 3.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? B2

Name of element (Wk)? C6
Time of element (tk)? 3.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? B2

Name of element (Wk)? D
Time of element (tk)? 2.1
Constraints No. (Mk)? 0
 No. of pre works? 6
Name of pre work (pk)? C1
Name of pre work (pk)? C2
Name of pre work (pk)? C3
Name of pre work (pk)? C4
Name of pre work (pk)? C5
Name of pre work (pk)? C6

Name of element (Wk)? E
Time of element (tk)? 19.1
Constraints No. (Mk)? 0
 No. of pre works? 1
Name of pre work (pk)? D
```

Work Elements List

k Wk	tk	Mk	Pk
(1)A1	9.4	0	
(2)A2	9.4	0	
(3)B1	3.9	0	A1
(4)B2	3.9	0	A2
(5)C1	3.1	0	B1
(6)C2	3.1	0	B1
(7)C3	3.1	0	B1
(8)C4	3.1	0	B2
(9)C5	3.1	0	B2
(10)C6	3.1	0	B2
(11)D	2.1	0	C1 C2 C3 C4 C5 C6
(12)E	19.1	0	D

```
T= 66.4   tmax= 19.1
If you want to calculate DATA, hit b.? b
```

Work elements list

k	Wk	tk	Mk	Pk	Sk	Vk	IVk	Uk	IUk
(1)	A1	9.4	0		B1	43.8	9.4	6	0
(2)	A2	9.4	0		B2	43.8	9.4	6	0
(3)	B1	3.9	0	A1	C1 C2 C3	34.4	13.3	5	1
(4)	B2	3.9	0	A2	C4 C5 C6	34.4	13.3	5	1
(5)	C1	3.1	0	B1	D	24.3	16.4	2	2
(6)	C2	3.1	0	B1	D	24.3	16.4	2	2
(7)	C3	3.1	0	B1	D	24.3	16.4	2	2
(8)	C4	3.1	0	B2	D	24.3	16.4	2	2
(9)	C5	3.1	0	B2	D	24.3	16.4	2	2
(10)	C6	3.1	0	B2	D	24.3	16.4	2	2
(11)	D	2.1	0	C1 C2 C3 C4 C5 C6	E	21.2	47.3	1	10
(12)	E	19.1	0	D		19.1	66.4	0	11

T= 66.4 tmax= 19.1 F= .469697
If you want to assign elements, hit B.? B

Line balancing with considering performancerates of work stations.

Set No. No=? 3

Enter performance rate & set time.

Performance rate R(1)? 0.762
Set time D(1)? 0

Performance rate R(2)? 1.25
Set time D(2)? 0

Performance rate R(3)? 0.938
Set time D(3)? 0

Enter the constant d which you decrease Ebv.
d =? 0.05

Select Forward or Backward Assignment.
Hit f or b ? f

Select one of the following B&B rules.
1 Largest task time rule.
2 Smallest task time rule.
3 Random selecting rule.
4 Highest ranked positional weight rule.
5 Lowest flexible rate rule.
Input choice ? 1

B&B rule= f Largest task time rule.
No = 3 Pu = 23.2982

i	Ti'(Ti)	Wij	tij	Ri	Di
1	21.5223 (16.4)			.762	0
		A1	9.4		
		B1	3.9		
		C1	3.1		
2	23.04 (28.8)			1.25	0
		A2	9.4		
		B2	3.9		
		C2	3.1		
		C3	3.1		
		C4	3.1		
		C5	3.1		
		C6	3.1		
3	22.6013 (21.2)			.938	0
		D	2.1		
		E	19.1		

Pitch= 23.04 Eb= .971695
We can't get better result by this B&B rule.
Do you want to search the better result?
y or n? n