Erciyes Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, Sayı 9, Kasım 1991

AN INTEGER PROGRAMMING FORMULATION FOR THE LINE BALANCING PROBLEMS

Cemal ÖZGÜVEN*
Osman UNUTULMAZ**

ABSTRACT: The 0-1 programming formulations developed so far to deal with the simple assembly line balancing problems fix either the cycle time or the number of stations along the line. The integer programming formulation suggested in this paper allows simultaneous consideration of the cycle time and the number of stations. It manages to determine the line with minimum balance delay without having to fix either of them.
(PROGRAMMING-INTEGER, LINE BALANCING, PRODUCTION SCHEDULING)

INTRODUCTION

The problem considered in this paper is to determine the line with minimum balance delay for a specified range of cycle times.

The line with minimum balance delay can be determined either by solving a number of 0-1 formulations of SALBP-1 (one for each alternative cycle time) or by solving a number of 0-1 formulation of SALBP-2 (one for each atlernative number of stations) relavant to the range.

To obtain the solution in one swoop, an integer quadratic programming (IQP) formulatıon is developed. In this IQP formulation neither the cycle time nor the number of stations is fixed. In order to be able to employ the integer programming algorithms and to set the ground for making comparison to the 0-1 formulations of SALBP-1 and SALBP-2, the IQP formulation is then transformed into an integer linear programming (ILP) formulation.

* Prof.Dr., Erciyes Üniv. İktisadi ve İdari Bilimleri Fakültesi Öğr.Üy.
** Doç.Dr., Erciyes Üniv.İktisadi ve İdari Bilimleri Fakültesi Öğr.Üy.

PRELIMINARIES AND NOTATION

Here the general notation used will be presented and the additional ones will be introduced as needed.

I : task set ($I = \{1,2,.....,i,.....m\}$)
J : station set ($J = \{1,2,....j,....n\}$)
$I(j)$: Subset of tasks assigned to station j, $j \in J$
t_i : process time of tâks, i, $i \in I$

$T_j = \sum_{i \in I(j)} t_i$ (i.e., work content of station j)

$W = \sum_{i=1}^{m} t_i$ (i.e., total task processing time)

T : cycle time
$t_{min} = \min_{i \in I} \{tj\}$
$t_{max} = \max_{i \in I} \{tj\}$

$R = \{(i, h) \mid$ task i is an immediate predecessor of task h$\}$
i.e., partial order of the task set I(precedence relations)
$P(i) = \{h \in I \mid (h, i) \in R\}$ (i.e., the immediate predecessors of task i)
$P_a(i) = \{$all predecessors of i$\}$
$S(i) = \{h \in I \mid (i,h) \in R\}$ (i.e., the immediate successors of task i)
$S_a(i)$=all successors of i}
$NP_a(i)$:{the number of all predecessors of task i}
$NS_a(i)$:{the number of all successors of task i}
$K(j)$:{subset of tasks that can be assigned to station j, $j \in J$, by virtue of the task precedence relationships}
$F = \{i \in I \mid S(i) = \emptyset\}$(i.e., tasks with no successors}
n^*: the optimum number of stations needed
T^*: the optimum cycle time
T_U: the specified upper bound of the range of cycle times. (the maximum cycle time)
T_L: the specified lower bound of the range of cycle times. (the minimum cycle time)

$r=[W/T_U]^+$ i.e., the lower bound on the optimum number of stations needed (the theoretical minimum number of stations)

s=m i.e., the upper bound on the optimum number of stations needed (the theoretacal maximum number of stations)

E_i : the smallest numbered(the earliest) station to which task i can be assigned

L_i : The largest numbered (the latest) station to which task i can be assigned

$[x]^+$=the smallest integer larger than or equal to x

It should be noted that

$r \leq n^* \leq m$

$T_L \leq T^* \leq T_U$

and that T_U and T_L are specified by the decision maker under the restrictions

$T_L \geq tmax$

$$T_U \leq \frac{\text{available time per working day}}{\text{the minimum acceptable number of products per working day}}$$

THE IQP FORMULATION

In this section an integer quadratic programming (IQP) formulation, a powerful tool for handling the simple line balancing problems (BAYBARS), is presented. The decision variables are defined as follows:

$$X_{ij} = \begin{cases} 1 \text{ if task i is assigned to station j} \\ 0 \text{ otherwise} \end{cases}$$

$\forall\, i \in I$ and $\forall\, j \in J$

The occurance contraints (PATTERSON and ALBRACH) guarantee that every task is assigned to a station and that each task occurs wholly within a station, since X_{ij} is 0 or 1.

$$\sum_{j=E_i}^{L_i} X_{ij} = 1 \qquad \forall\, i \in I \qquad (1)$$

The constraint (1) is satisfied if the constraint set

$$\sum_{j=E_i}^{L_{i-1}} X_{ij} \leq 1 \qquad \forall\, i \in I \qquad (2)$$

is satisfied and

$$X_{iL_j} = 1 - \left(\sum_{j=E_i}^{L_{i-1}} X_{ij} \right) \qquad \forall\, i \in I \qquad (3)$$

The cycle time constraints guarantee that the work content of every station is not greater than the cycle time.

$$\sum_{i\in K(j)} t_i X_{ij} \leq T \qquad \forall j \in J \qquad (4)$$

Next, the precedence relationships between the tasks will be considered. If $h \in S(i)$, then ask h must be assigned either to the same station as task i or to a subsequent station, never to a prior one. The constraint set (PATTERSON and ALBRACH)

$$\sum_{j=Ei}^{L_i} j(X_{ij}) \leq \sum_{j=Eh}^{L_h} j(X_{hj}) \qquad (5)$$

Where $L_i > E_h$ and $i << h$

quarantees that task h is not assigned to a station prior to the station to which task i is assigned. To prevent the violation of the precedence relationships, a constraint of type(5) must be written for each pair of tasks i and h, i.e., for each task i and its immediate successor.

Finally, the following constraints are imposed to make sure that the cycle time remains within the range specifieed by the decision maker:

$$T_L \leq T \leq T_U \qquad (6)$$

The objective is to determine T^*, n^* and the related task assignment which gives rise to the line with munimum balance delay for the specified range of cycle times. Since the balance delay of the line is a function of both the cycle time (T) and the number af stations (n) along the line, the objective function is written as

$$\text{minimize } 1 - W/nT$$

Since, 1 and W are constants, the objective function can equivalently be written as[1]

$$\text{minimize } nT \qquad (7)$$

In the final form of the objective function, the term n is replaced by an expression involving the X_{ij} variables. Before presenting the final form, it is convenient to give the following definition:

task d: the unique terminal dummy task
with $t_d=0$ and $i \ll d$ for each $i \in F$.

The number of the station to which task d is assigned indicates the number of the stations that are opened along the line. If task d is assigned to station k ($E_d = r \leq k \leq m$), then the variable X_{dk} is set equal to 1 in the expression

1 minimizing the balance delay of the line is equivalent to minimizing the total idle time along the line

$$\min \sum_{j=i}^{n} (T - T_j)$$

that is, min nT-W
and thus, min nT

$$\sum_{j=r}^{m} j(X_{dj}) \quad j \in J \qquad (8)$$

By virtue of the constraint set(1)

$$X_{dj}=0 \text{ for } j \neq k$$

Then, (8) reduces into

$$k$$

indicating the number of the stations that are opened along the line.

In (7), n is replaced by the expression (8) and the objective function is written in its final form as

$$\min \left[\sum_{j=r}^{m} jX_{dj} \right] T \qquad (9)$$

This objective function along with the constraint sets (1), (4), (5) and (6) forms the IQP formulation that serves to find the line with minimum balance delay for the specified range of cycle times.

When in this IQP formulation T is fixed at T_0, the objective function becomes

$$\min \sum_{j=r}^{m} jX_{dj} \; T_0 \qquad (9)$$

that is

$$\min \sum_{j=r}^{m} jX_{dj} \qquad (10)$$

and the constaint set (4) turns into

$$\sum_{i\in K(j)} t_i X_{ij} \leq T_0 \qquad \forall j \in J \qquad (11)$$

It must be noticed that the resulting formulation[2] (1), (5), (10) and (11) is a 0-1 programming formulation of SALBP-1 that serves to determine the line with minimum balance delay for T_0.

When in the above IQP formulation X_{dk} is set equal to 1, the objective function (9) reduces into

$$\min kT$$

that is

$$\min T \qquad (12)$$

and due to the constraint sets(1) and (5) the constraint set (4) becomes

$$\sum_{i\in K(j)} t_i X_{ij} \leq T \qquad j=1, 2, \ldots, k \qquad (13)$$

The resulting formulation (1), (5), (6), (12) and (13) is a 0-1 programming formulation of SALBP-2. It determines the line with minimum balance delay for a fixed number of stations(n_0).

THE ILP FORMULATION

The ILP formulation in which neither T nor n is fixed can be derived by defining some additional variables and constraints and by making some modifications to the IQP formulation (1), (4), (5), (6) and (9).

No matter what the cycle time is, at least $r=[w/T_U]^+$ stations (i.e., stations j=1, 2, ...r) must be opened altogether. The R_j variables take care of the fact that the additional stations may be opened sequentially, starting from the r+1'th station up to the m'th one.

$$R_j = \begin{cases} T & \text{if the j'th station with cycle time T in opened} \\ 0 & \text{otherwise} \end{cases}$$

2 Relation (6) becomes redundant.

$j = r+1,\ r+2,\ \ldots,\ m$

The R_j variables make it possible to convert the constraint set (4) into the following from:

$$\sum_{i \in K(j)} t_i X_{ij} \leq T \qquad j = 1, 2, \ldots r$$

$$\sum_{i \in K(j)} t_i X_{ij} \leq R \qquad j = r+1, r+2, \ldots, m \qquad (14)$$

To prevent the R_j variables taking on values other than 0 or T

$$L_j = 0, 1 \qquad j = r+1,\ r+2, \ldots,\ m$$

variables are introduced and the following constraints are imposed

$$\begin{aligned} R_j &\leq M(1-L_j) && j = r+1,\ r+2, \ldots,\ m \\ R_j &\geq T-ML_j && j = r+1,\ r+2, \ldots,\ m \\ R_j &\leq T && j = r+1,\ r+2, \ldots,\ m \end{aligned} \qquad (15)$$

where M is a large number.

An additional set of constraints is needed to make sure that after the first r stations are opened a new station cannot be opened unless the previous station has been opened:

$$R_{j+1} \leq R_j \qquad j = r+1,\ r+2, \ldots,\ m-1 \qquad (16)$$

Finally, the objective function, now in linear form, is expressed as

$$\text{minimize} \quad rT + \sum_{j=r+1}^{m} R_j \qquad (17)$$

The resulting formulation (1), (5), (6), (14), (15), (16) and (17) is the ILP formulation suggested in this paper. It is the linearized version of the IQP formulation developed above.

EXAMPLE: A LINE BALANCING PROBLEM

The minimum acceptable number of products within an eight hour shift is 235 units. Six different tasks are involved. The relevant information is presented below.

Table 1: Standard Process Times and The Precedence Relationships

i	T_i(seconds)	P(i)
1	83	-
2	30	-
3	60	1
4	85	1,2
5	45	3,4
6	70	4

The decision maker has specified the lower and the upper bounds of the range of cycle times within which he desires to determine the line with minimum balance delay as R_L=88 seconds and R_U=120 seconds respectively:

$$T_L = 88 > t_{max} = 85$$
$$T_U = 120 < 8x60x60/235$$

No matter what the cycle time is at least

$$r = [83+30+60+85+45+70/120]^+ = 4$$

stations must be opened along the line. The theoretical minimum number of stations is r=4.

The theoretical maximum number of stations is taken to be equal to the number of the tasks: s=m=6.

THE PROBLEM VARIABLES

The variables that will be used in the ILP formulation are defined as follows:

$$X_{ij} = \begin{cases} 1 & \text{if task i is assigned to station j} \\ 0 & \text{otherwise} \end{cases}$$

$i = 1, 2, 3, 4, 5, 6$
$j = 1, 2, 3, 4, 5, 6$

$$R_j = \begin{cases} 1 & \text{if teh j'th station with cycle time T is opened} \\ 0 & \text{otherwise} \end{cases}$$

$j = 5, 6$

$L_j = 0,1$ $\qquad$ $j = 5,6$

T: cycle time

Considerable amount of reduction in the number of X_{ij} variables can be achieved by determining the infeasible stations for task i, i.e., the stations to which task i cannot be assigned.

THE FEASIBLE AND INFLASIBLE STATIONS

When the precedence relationships, the minimum number of stations (r) and the maximum number of stations (m) are taken into consideration, it becomes clear that task i cannot be assigned to some of the stations.

The infeasible stations are determined by finding the earliest and the latest stations to which task i can be assigned (E_i and L_i). The stations prior to E_i and subsequent to L_i are the infeasible ones for task i.

The number of tasks that must be performed after task i is $NS_a(i)$. The total number of tasks excluding task i is m-1. So, $(m-1)-NS_a(i)$ indicates the total number of the tasks that may be done before task i. If $(m-1)-NS_a(i)$ tasks may be done before task i and if at most m stations can be opened, then the latest station that task i can be assigned to is the $m-NS_a(i)$'th one:

$L_i = m-NSa(i)$ $\qquad$ $i=1, 2, 3, 4, 5, 6$

The stations opened after the L_i'th station are the infeasible stations for task i.

The sum-total of the process time of task i and its predecessors can be expressed as

$$V_i = t_i + \sum_{h \in P_a(i)} t_h \qquad i = 1, 2, 3, 4, 5, 6$$

Adopting the formula suggested in (TALBOT and PATTERSON) the earliest station to which task i can be assigned is determined as follows[3]

$$E_i = [V_i/120]^+ \qquad i = 1, 2, 3, 4, 5, 6$$

The stations opened before the E_i'th one are the infeasible stations for task i.

The steps taken to determine the feasible stations for each task may be presented in tabular form. For the sample problem Table 2 is formed.

Table 2: Determination of The Feasible Stations

i	$P_a(i)$	V_i	$S_a(i)$	$NS_a(i)$	E_i	L_i	The Feasible Stations
1	-	83	3,4,5,6	4	1	2	1,2
2	-	30	4, 5, 6	3	1	3	1,2,3
3	1	143	5	1	2	5	2,3,4,5
4	1,2	198	56	2	2	4	2,3,4
5	1,2,3,4	303	-	0	3	6	3,4,5,6
6	1,2,4	268	-	0	3	6	3,4,5,6

If the last column of Table 2 is used for the ILP formulation, the number of the X_{ij} variables will be reduced from 6x6=36 to 20.

3 The maximum cycle time of 120 seconds is employed in the formula, because the range of feasible stations becomes wider than the ranges that would be obtained for the shorter cycle times.

THE ILP FORMULATION OF THE SAMPLE PROBLEM

Using the last column of Table 2, the ILP formulation (1), (5), (6), (14), (15), (16) and (17) can be written as

$$\text{minimize } 4T + R_5 + R_6 \qquad (17)$$

subject to

$$
\begin{aligned}
&X_{11}+X_{12} = 1\\
&X_{21}+X_{22}+X_{23} = 1\\
&X_{32}+X_{33}+X_{34}+X_{35} = 1\\
&X_{42}+X_{43}+X_{44} = 1\\
&X_{53}+X_{54}+X_{55}+X_{56} = 1\\
&X_{63}+X_{64}+X_{65}+X_{66} = 1
\end{aligned}
\qquad (1)
$$

$$
\begin{aligned}
&X_{21}+2X_{22}+3X_{23} \le 2X_{42}+3X_{43}+4X_{44}\\
&2X_{32}+3X_{33}+4X_{34}+5X_{35} \le 3X_{53}+4X_{54}+5X_{55}+6X_{56}\\
&2X_{42}+3X_{43}+4X_{44} \le 3X_{53}+4X_{54}+5X_{55}+6X_{56}\\
&2X_{42}+3X_{43}+4X_{44} \le 3X_{63}+4X_{64}+5X_{65}+6X_{66}
\end{aligned}
\qquad (5)
$$

$$
\begin{aligned}
&T \le 120\\
&T \le 88
\end{aligned}
\qquad (6)
$$

$$
\begin{aligned}
&83X_{11}+30X_{21} \le T\\
&83X_{12}+30X_{22}+60X_{32}+85X_{42} \le T\\
&30X_{23}+60X_{33}+85X_{43}+45X_{53}+70X_{63} \le T\\
&60X_{34}+85X_{44}+45X_{54}+70X_{64} \le T\\
&60X_{35}+45X_{55}+70X_{65} \le R_5\\
&45X_{56}+70X_{66} \le R_6
\end{aligned}
\qquad (14)
$$

$$
\begin{aligned}
&R_5 \le 1{,}000(1-L_5) \qquad && R_6 \le 1{,}000(1-L_6)\\
&R_5 \ge T-1{,}000\ L_5 && R_6 \ge T-1{,}000\ L_6\\
&R_5 \le T && R_6 \le T
\end{aligned}
\qquad (15)
$$

$$R_6 \le R_5 \qquad (16)$$

This is an ILP formulation with 25 variables and 25 constraints. If the constraint set (1) is replaced by two sets written in terms of the relations (2) and (3), six X_{ij} variables are eliminated from the constraint sets(5) and (14), and the ILP formulation of the sample problem in its final form with 19 variables and 25 constraints is obtained.

THE SOLUTION

The ILP formulation with 19 variables and 25 constraints is solved within 9.083 minutes in the Vest Vestel 64 computer by employing program package LINDO that is based on the branch and bound algorithm. The following solution is obtained:

$$4T+R_5+R_6=450 \qquad X_{11}=X_{22}=X_{32}=X_{43}=X_{64}=X_{55}=1$$
$$T=R_5=90,\ L_6=1$$

This solution indicates that the optimum number of stations is five ($n^*=5$) and the optimum cycle time is 90 seconds ($T^*=90$) for the specified range of cycle times with $R_L=88$ and $R_U=120$ seconds. When five stations with cycle time of 90 seconds is opened and the tasks are assigned accordingly, the line with minimum balance delay for the specified range is formed. The balance delay of this line is 1-(373/5x90)=0.1711.

COMPARISON TO THE 0-1 FORMULATIONS

The same result can be obtained by the 0-1 programming formulations. This can be done either by solving four 0-1 formulations of SALBP-1 (one for each alternative cycle time relavant to the range), or by solving two 0-1 formulations of SALBP-2 (one for four stations and one for five stations)[4]

Four 0-1 formulations of SALBP-1 are formed in terms of the relations (1), (5), and (11). The objective function

$$\text{minimize} \quad 1{,}000(X_{56}+X_{66})+100(X_{55}+X_{65})+10(X_{54}+X_{64})$$

is employed.

Two 0-1 formulations of SALBP-2 are formed according to the relations (1), (5), (6), (12), and (13). In all of the SALBP-1 and SALBP-2 formulations relation (1) is replaced by the relations (2) and (3) as it is done in the ILP formulation.

The six 0-1 formulations are solved under exactly the same conditions. The computational result are presented below.

---

4 The 0-1 formulation of SALBP-2 for six stations has a trivial solution.

Table 3: The Computational Results[5]

Formulation	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
ILP	-	-	19	25	450	46	488	9.08
SALBP-1	90	-	14	16	450	15	154	2.16
SALBP-1	105	-	14	16	525	37	263	3.25
SALBP-1	113	-	14	16	452	17	99	1.22
SALBP-1	115	-	14	16	460	21	102	1.53
SALBP-2	-	4	10	16	452	18	129	1.61
SALBP-2	-	5	13	17	450	74	510	6.33

The sum-total of the computation times is 8.16 minutes for the four 0-1 formulations of SALBP-1 and 7.94 minutes for the two 0-1 formulations of SALBP-2. Both of the totals are about one minute shorter than the 9.08 minutes of the ILP formulation.

However, when the 0-1 formulations of SALBP-1 are used, the determination of the feasible alternative cycle length becomes extremely cumbersome, if not impossible, as the problem size increases.

One does not experience such a complication in employing the 0-1 formulations of SALBP-2. Yet, when the formulation, loading and compulation times are considered altogether, the total time spent for the 0-1 formulations will exceed that of the ILP formulation.

CONCLUSION

The ILP fromulation recommended here enables one to determine the line with minimum balance delay (and the related task assignment) for a specified range of cycle times without fixing either the cycle time or the number of stations. This ILP formulation seems to be more efficient than the 0-1 formulations in terms of the total time required to attain the solution.

---

5 (1) To

(2) n_o

(3) variables

(4) constraints

(5) n*T*

(6) branches

(7) pivots

(8) Vest Vestel 64 compatation time, in minutes

REFERENCES

[1] Baybars, İlker, "A Survey of Exact Algorithms For Simple Assembly Line Balancing Problems" **Management Science**, Volume 32, Number, 8, Pages 909-932, August 1986.

[2] Talbot, F.Brian and Patterson, James H., "An Integer Programming Algorithm With Network Cuts For Solving The Assembly Line Balancing Problem" **Management Science**, volume 30, Number 1, January 1984.

[3] Patterson J.H. and Albrach J.J., "Assembly Line Balancing: 0-1 Programming With Fibonacci Search "**Operations Research**, 23(1975).