INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 25 (2018) 1345–1359
DOI: 10.1111/itor.12206

Improved integer programming models for simple assembly line balancing and related problems

Marcus Ritt[a] and Alysson M. Costa[b]

[a] *Instituto de Informática, Universidade Federal do Rio Grande do Sul, Av. Bento Gonçalves 9500, Porto Alegre, Brazil*
[b] *School of Mathematics and Statistics, University of Melbourne, Parkville 3010, Australia*
E-mail: marcus.ritt@inf.ufrgs.br [Ritt]; alysson.costa@unimelb.edu.au [Costa]

Received 22 January 2015; received in revised form 15 May 2015; accepted 31 July 2015

Abstract

We propose a stronger formulation of the precedence constraints and the station limits for the simple assembly line balancing problem. The linear relaxation of the improved integer program theoretically dominates all previous formulations using impulse variables, and produces solutions of significantly better quality in practice. The improved formulation can be used to strengthen related problems with similar restrictions. We demonstrate their effectiveness on the U-shaped assembly line balancing problem and on the bin packing problem with precedence constraints.

Keywords: assembly line balancing; integer linear programming; valid inequalities; precedence constraints; station limits

1. Introduction

Let $(N, \leq)$ be a weak partially ordered set of tasks with integral execution time t_i for $i \in N$. The simple assembly line balancing problem (SALBP) is to find an assignment $a : N \rightarrow S$ of the tasks to a linear sequence of stations $S = \{1, 2, \ldots, m\}$, respecting the partial order, that is, all tasks $i, j \in N$ with $i \leq j$ satisfy $a(i) \leq a(j)$. The *cycle time* of an assignment is the largest time needed to execute the tasks assigned to some station. The problem is said to be of *type 1* (SALBP-1) when the goal is to minimize the number stations for a given cycle time, and to be of *type 2* (SALBP-2) when the goal is to minimize the cycle time for a given number of stations. The decision version of both problems is NP-complete, since without precedence constraints SALBP-1 reduces to the bin packing problem, and SALBP-2 to the problem of minimizing the makespan of a schedule of the tasks on identical parallel machines.

The SALBP has been extensively studied in the literature, and there are excellent constructive and heuristic algorithms, as well as exact solution methods available (e.g., Scholl and Voß, 1996; Scholl and Klein, 1999; Fleszar and Hindi, 2003; Blum, 2008; Sewell and Jacobson, 2012). A very good overview of the methods can be found in the survey of Scholl and Becker (2006).

Published by John Wiley & Sons Ltd, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main St, Malden, MA02148, USA.

The problem is of interest to researchers, as it forms the core of a large class of generalized assembly line balancing problems. These include assembly lines of different layout, for example, U-shaped lines, lines with assignment restrictions, varying task times, or setup times. Becker and Scholl (2006) survey generalized assembly line balancing problems, and the ALB Research Group (2012) provides a property-based search for information on these problems.

Results obtained for the SALBP can often be transferred to generalized problems. The purpose of this paper is to show that this also applies to integer programming models for the SALBP. Clearly, mathematical models solved by standard solvers are not competitive with state-of-the-art methods. They are nevertheless useful since they frequently serve as benchmarks for better methods, and are a tool for studying new general assembly line balancing problems, where such methods are not yet available. Combined with reduction rules and heuristic solutions, integer programming models solved by standard solvers can be a reasonable, prototypical solution method.

To obtain the best possible solution and to guarantee a fair comparison to other methods, it is necessary to select the best model. In this paper, we address this problem comparing theoretically and computationally several models for the SALBP from the literature, and some improved models proposed in this paper. A survey of models for the SALBP can be found in Baybars (1986) and Scholl (1999). To the best of our knowledge, no theoretical comparison of these models has been published before. A computational study of some models has been provided by Pastor et al. (2007).

We argue that the results obtained for the SALBP can be generalized to other assembly line balancing problems. This will be demonstrated by two case studies. We show how the model for the U-shaped assembly line balancing problem (UALBP-1) proposed by Urban (1998) and the model for the bin packing problem with precedence constraints (BPP-P), which has been recently introduced by Dell'Amico et al. (2012) can be improved by the formulations proposed in this paper. Urban's model is widely used in the literature as originally proposed (Aase et al., 2003; Gökçen and Ağpak, 2004; Chiang et al., 2007; Erin, 2007; Kara and Tekin, 2009; Ağpak et al., 2012), and a better model may improve the results in these applications. In the case of the BPP-P, it turns out that the best integer model is competitive with a sophisticated tailored branch-and-bound algorithm, demonstrating its utility as a tool for obtaining a rapid, reasonable solution method for new problems.

The remainder of this paper is organized as follows. In the next section we present a formal definition, basic mixed-integer linear models of the SALBP-1 and the SALBP-2 and further models from the literature. In Section 3, we propose improved formulations of the precedence constraints and the station limits, and theoretically compare the resulting models with the existing ones. The improved formulations are applied to related problems in Section 4. A computational study is presented in Section 5 and we offer some conclusions in Section 6.

2. Integer programming models for the SALBP

In this section, we review mathematical models for SALBP-1 and SALBP-2 that have been proposed in the literature. For a task $i \in N$, let F_i denote the set of its *immediate followers*, and P_i the set of its *immediate predecessors*. Let S be the set of stations. In the following, we suppose that for the SALBP-1 an upper bound $\overline{m}$ on the number of stations is known ($|N|$ is such an upper bound), and that $S = \{1, \dots, \overline{m}\}$. For the SALBP-2, the number of stations m is part of the problem instance. In this case we set $S = \{1, \dots, m\}$.

There have been three kinds of models proposed in the literature. Bowman (1960) (in the revised formulation of White, 1961) proposed two formulations, one using binary *impulse variables* and another based on *time variables*, representing the starting time of the tasks. Scholl (1999) proposed a formulation using binary *step variables* $\overline{x}_{si}$, where $\overline{x}_{si} = 1$ indicates that task $i \in N$ is assigned to station $s \in S$ or some preceding station. Since the formulation using time variables has been found inferior by Pastor et al. (2007) and the formulations using impulse variables are the most common in the literature, we focus in the following on the latter.

2.1. Basic models for SALBP

To represent the assignment of tasks to station, we introduce impulse variables

$$x_{si} = \begin{cases} 1, & \text{if task } i \in N \text{ is assigned to station } s \in S, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Any feasible allocation has to satisfy the *occurrence* constraints

$$\sum_{s \in S} x_{si} = 1, \qquad \forall i \in N, \tag{2}$$

which ensures that every task is allocated to a single station, the *precedence* constraints

$$x_{tj} \le \sum_{s \in S | s \le t} x_{si}, \qquad \forall i \in N, j \in F_i, t \in S, \tag{3}$$

and the *nondivisibility* constraints

$$x_{si} \in \{0, 1\}, \qquad \forall i \in N, s \in S. \tag{4}$$

These constraints have been first proposed by Bowman (1960) and their above form is due to White (1961). For a given cycle time c and an upper bound $\overline{m}$ on the number of stations, the SALBP-1 can be formulated as

$$(\mathrm{BW1}-1) \qquad \text{minimize} \qquad \sum_{s \in S} y_s, \tag{5}$$

$$\text{subject to} \qquad \sum_{i \in N} t_i x_{si} \le c y_s, \qquad \forall s \in S, \tag{6}$$

$$\text{Equations (2) to (4)}, \tag{7}$$

$$y_s \in \{0, 1\}, \qquad \forall s \in S, \tag{8}$$

where the variables y_s indicate the usage of station $s \in S$. For the SALBP-2 the cycle time c is variable and the number of stations m is fixed. It can be formulated as

$$(BW1-2) \qquad \text{minimize} \qquad c, \tag{9}$$

$$\text{subject to} \qquad \sum_{i \in N} t_i x_{si} \le c, \qquad \forall s \in S, \tag{10}$$

$$\text{Equations (2) to (4)}, \tag{11}$$

$$c \in \mathbb{R}. \tag{12}$$

These formulations are due to Baybars (1986). In the following two subsections we present improvements of these models by adding station limits and better formulations of the precedence constraints.

2.2. Station limits

For a given cycle time c and a given number of stations m one often can derive bounds on the stations a task can be assigned to. For task $i \in N$, let $E_i(c, m)$ be the earliest and $L_i(c, m)$ the latest admissible station. (For the SALBP these bounds can be set, for instance, to $E_i(c, m) = \lceil \sum_{j|j \le i} t_j / c \rceil$ and $L_i(c, m) = m + 1 - \lceil \sum_{j|i \le j} t_j / c \rceil$.) Then, we can restrict the domain of the decision variables, substituting (4) by

$$x_{si} \in \{0, 1\}, \qquad \forall i \in N, E_i(c, \overline{m}) \le s \le L_i(c, \overline{m}) \tag{13}$$

in the formulation of the SALBP-1 and by

$$x_{si} \in \{0, 1\}, \qquad \forall i \in N, E_i(\overline{c}, m) \le s \le L_i(\overline{c}, m) \tag{14}$$

in the formulation of the SALBP-2, where $\overline{m}$ is an upper bound on the number of stations, and $\overline{c}$ an upper bound on the cycle time (Patterson and Albracht, 1975).

The station bounds can be strengthened as follows (Pastor and Ferrer, 2009). In the case of the SALBP-1, when taking into account the currently used stations,

$$x_{L_i(c,s),i} \le y_s, \qquad \forall s \in S, i \in N \tag{15}$$

is valid, since if station s is unused, we can assume that this also holds for all later stations and therefore the latest possible station is now $L_i(c, s-1) = L_i(c, s) - 1$.

For the SALBP-2 we can model the cycle time explicitly to obtain better bounds on the stations. Let $\underline{c}$ be a lower bound on the cycle time and let $C = [\underline{c}, \overline{c}]$ be the set of admissible cycle times. Then we can represent the cycle time by

$$c = \sum_{t \in C} t r_t, \tag{16}$$

$$\sum_{t\in C} r_t = 1, \tag{17}$$

$$r_t \in \{0, 1\}, \qquad t \in C. \tag{18}$$

This allows the following inequalities to be added to the formulation of the SALBP-2:

$$x_{ei} \leq 1 - \sum_{t\in C|e<E_i(t,m)} r_t, \qquad \forall i \in N, E_i(\overline{c}, m) \leq e < E_i(\underline{c}, m), \tag{19}$$

$$x_{li} \leq 1 - \sum_{t\in C|L_i(t,m)<l} r_t, \qquad \forall i \in N, L_i(\underline{c}, m) < l \leq L_i(\overline{c}, m). \tag{20}$$

These inequalities are easily seen to be valid. Suppose, for example, that $\sum_{t\in C|e<E_i(t,m)} r_t = 1$ in Equation (19). Then station e comes before the earliest possible station for task i considering the current cycle time, and therefore $x_{ei} = 0$. Similarly, if $\sum_{t\in C|L_i(t,m)<l} r_t = 1$ in Equation (20) station l comes after the latest possible station for task i given the current cycle time, and therefore $x_{li} = 0$.

2.3. *Alternative formulations of the precedence constraints*

Patterson and Albracht (1975) have proposed to formulate the precedence constraints as

$$\sum_{s\in S} s x_{si} \leq \sum_{s\in S} s x_{sj}, \qquad \forall i \in N, j \in F_i. \tag{21}$$

Thangavelu and Shetty (1971) give the alternative formulation

$$\sum_{s\in S} (m - s + 1)(x_{si} - x_{sj}) \geq 0, \qquad \forall i \in N, j \in F_i. \tag{22}$$

Observe that both formulations are equivalent, since, by the occurrence constraints we have

$$\sum_{s\in S} (m - s) x_{si} + \sum_{s\in S} s x_{si} = m.$$

3. Improved models for the SALBP

3.1. *Station limits*

We can strengthen the station limits proposed by Pastor and Ferrer (2009) as follows. In constraint (15), when a task cannot be assigned to station $L_i(c, s)$, it also cannot be assigned to any later station. This justifies

$$\sum_{u\in S|u\geq L_i(c,s)} x_{ui} \leq y_s, \qquad \forall s \in S, i \in N. \tag{23}$$

Similarly, in constraints (19) and (20) when a task cannot be assigned to a station earlier than e this holds also for stations preceding e, and when a task cannot be assigned later than station l, this holds also for stations following l. Therefore we can strengthen these constraints to

$$\sum_{u\in S|u\leq e} x_{ui} \leq 1 - \sum_{t\in C|e<E_i(t,m)} r_t, \qquad \forall i \in N, E_i(\overline{c}, m) \leq e < E_i(\underline{c}, m), \tag{24}$$

and

$$\sum_{u\in S|u\geq l} x_{ui} \leq 1 - \sum_{t\in C|L_i(t,m)<l} r_t, \qquad \forall i \in N, L_i(\underline{c}, m) < l \leq L_i(\overline{c}, m). \tag{25}$$

3.2. Precedence constraints

We propose the following improved formulation of the precedence constraints:

$$\sum_{s\in S|s\leq k} x_{si} \geq \sum_{s\in S|s\leq k} x_{sj}, \qquad \forall i \in N, j \in F_i, k \in S. \tag{26}$$

The two following propositions state the validity and the theoretical strength of these new constraints.

Proposition 1. *Constraints* (26) *are valid for BW1-1 and BW1-2.*

Proof. If $\sum_{s\in S|s\leq k} x_{si} = 1$ the constraint is trivially satisfied, since $\sum_{s\in S} x_{sj} = 1$. Otherwise, task i is executed on station $k+1$ or later. But since $j \in F_i$ task j cannot be executed on a station preceding station $k+1$, that is, $\sum_{s\in S|s\leq k} x_{sj} = 0$. □

Proposition 2. *Inequalities* (26) *strictly dominate inequalities* (21) *and* (3). *Inequalities* (21) *and* (3) *are incomparable.*

Proof. Patterson and Albracht's inequalities (21) and the equivalent inequalities (22) of Thangavelu and Shetty are aggregated versions of inequalities (26). Indeed, summing up inequalities (26) for $k \in S$, we have, for all $i \in N$, $j \in F_i$,

$$\sum_{k\in S} \sum_{s\in S|s\leq k} x_{si} \geq \sum_{k\in S} \sum_{s\in S|s\leq k} x_{sj}.$$

which can be easily rewritten as Patterson and Albracht's inequalities (21) since

$$\sum_{k\in S} \sum_{s\in S|s\leq k} x_{si} = \sum_{s\in S} (m - s + 1) x_{si}.$$

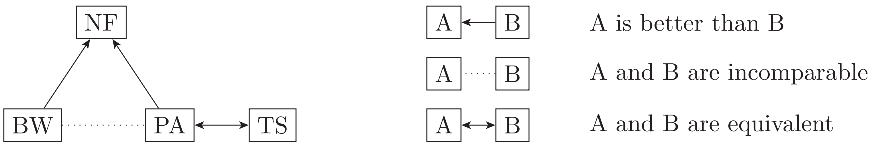

Fig. 1. Relationships between models of the SALBP with different formulations of precedence constraints. They are valid for the SALBP-1 as well as the SALBP-2 with the same station limits.

Table 1
Different formulations of the SALBP-1

Base model					
Suffix	Equations	Var.	Res.		
-1	(2), (4), (5), (6), (8)	$\overline{m}(n+1)$	$\overline{m}+n$		
Precedence constraints			Station limits		
Name	Equations	Res.	Name	Equations	Res.
PA	(21)	on^2	1	–	–
BW	(3)	$on^2\overline{m}$	2	(13)	–
TS	(22)	on^2	3	(13), (15)	$\overline{m}n$
NF	(26)	$on^2\overline{m}$	4	(13), (23)	$\overline{m}n$

Moreover, inequalities (26) also imply Bowman's inequalities (3). Indeed, for all $i \in N$, $j \in F_i$, and $t \in S$, the smaller terms of both inequalities compare as

$$x_{tj} \leq \sum_{s \in S | s \leq t} x_{sj},$$

and since their larger terms are equal, inequalities (26) are lifted versions of (3).

The strict dominance of inequalities (26) over (21) and (3), and the incomparability of the latter two can be seen by means of an example. Let $N = \{a, b\}$, $a \leq b$ and $m = 3$. It is easy to verify that the fractional solution $x_{1a} = x_{2a} = 1/2$, $x_{1b} = 3/4$, and $x_{3b} = 1/4$ satisfies Equation (21), but neither Equation (26) nor (3), and the fractional solution $x_{1a} = x_{1b} = x_{2b} = x_{3a} = 1/2$ satisfies Equation (3), but neither Equation (26) nor (21). □

The results of Proposition 2 are summarized in Fig. 1, which shows the relationships between the models.

Tables 1 and 2 give a summary of models with different precedence constraints and station limits for the SALBP-1 and SALBP-2 and their number of variables and restrictions. We obtain a different model for each combination of the precedence constraints, the stations limits and the base model. For example, BW2-1 denotes the formulation of the SALBP of type 1, using the precedence constraints of Bowman (3), and station limits (13). Note that Equations (13) and (14) do not increase the number of restrictions, but reduce the number of variables. In both tables o denotes the *order strength* of the instance, that is, the fraction of the at most $\binom{n}{2}$ precedence relations present in the instance. The order strength of an instance is at most 1.

Table 2
Different formulations of the SALBP-2

Base model					
Suffix	Equations	Var.	Res.		
−2	(2), (4), (9), (10), (12)	$nm+1$	$m+n$		
Precedence constraints			Station limits		
Name	Equations	Res.	Name	Equations	Res.
PA	(21)	on^2	1	–	–
BW	(3)	on^2m	2	(14)	–
TS	(22)	on^2	3	(14), (16) to (20)	mn
NF	(26)	on^2m	4	(14), (16) to (18), (24), (25)	$\overline{m}n$

4. Application to related problems

4.1. An improved model for the UALBP-1

U-shaped assembly lines are an alternative to the traditional linear layout, where tasks first pass all stations in the forward direction, and then pass them again in the backward direction, in form of an U. Therefore, the tasks assignable to a station include, besides tasks whose predecessors have been assigned to a preceding station, also the tasks whose successors have been assigned to a preceding station. This added flexibility can improve the line's balance or reduce the number of required stations. The problem of optimally allocating tasks to stations is known as the U-line balancing problem (UALBP). Urban (1998) has proposed an integer linear program for solving the UALBP-1, which is often used in studies of the UALBP (e.g., in Aase et al., 2003; Gökçen and Ağpak, 2004; Chiang et al., 2007; Erin, 2007; Kara and Tekin, 2009; Ağpak et al., 2012).

Introducing decision variables

$$x_{si} = \begin{cases} 1, & \text{if task } i \in N \text{ is executed on station } s \in S \text{ in the forward pass,} \\ 0, & \text{otherwise,} \end{cases} \tag{27}$$

and

$$w_{si} = \begin{cases} 1, & \text{if task } i \in N \text{ is executed on station } s \in S \text{ in the backward pass,} \\ 0, & \text{otherwise,} \end{cases} \tag{28}$$

the UALBP-1 is solved by the integer linear program

$$\text{minimize} \quad \sum_{s \in S} y_s, \tag{29}$$

$$\text{subject to} \quad \sum_{s \in S} x_{si} + w_{si} = 1, \qquad \forall i \in N, \tag{30}$$

$$\sum_{i \in N} t_i (x_{si} + w_{si}) \leq c y_s, \qquad \forall s \in S, \tag{31}$$

$$\sum_{s \in S} (\overline{m} - s + 1)(x_{si} - x_{sj}) \geq 0, \qquad \forall i \in N, j \in F_i, \tag{32}$$

$$\sum_{s \in S} (\overline{m} - s + 1)(w_{si} - w_{sj}) \geq 0, \qquad \forall i \in N, j \in P_i, \tag{33}$$

$$x_{si} \in \{0, 1\}, w_{si} \in \{0, 1\}, y_s\{0, 1\}, \qquad \forall s \in S, i \in N. \tag{34}$$

Due to the U-shaped layout every task may be assigned to the last station, and the station limits for the last station do not apply. Therefore, Urban applies only the bound

$$E_i(c, m) = \min \left\{ \left\lceil \sum_{j|j \leq i} t_j / c \right\rceil, \left\lceil \sum_{j|i \leq j} t_j / c \right\rceil \right\} \tag{35}$$

on the earliest station.

The above model uses precedence constraints as proposed by Thangavelu and Shetty (1971) for the SALBP. It can therefore be improved by substituting Equations (32) and (33) by the precedence constraints

$$\sum_{s \in S | s \leq k} x_{si} \geq \sum_{s \in S | s \leq k} x_{sj}, \qquad \forall i \in N, j \in F_i, k \in S, \tag{36}$$

$$\sum_{s \in S | s \leq k} w_{si} \geq \sum_{s \in S | s \leq k} w_{sj}, \qquad \forall i \in N, j \in P_i, k \in S. \tag{37}$$

Furthermore, the station limits may be applied separately to the forward and backward pass

$$x_{si} \in \{0, 1\}, \qquad \forall i \in N, E_i(c, \overline{m}) \leq s, \tag{38}$$

$$w_{si} \in \{0, 1\}, \qquad \forall i \in N, L_i(c, \overline{m}) \leq s, \tag{39}$$

where $E_i(c, m) = \lceil \sum_{j|j \leq i} t_j / c \rceil$ and $L_i(c, m) = \lceil \sum_{j|i \leq j} t_j / c \rceil$.

4.2. An improved model for the BPP-P

The bin packing problem with precedence constraints asks to pack a set of items into the smallest number of bins of a fixed size, with the additional restriction that an item cannot share a bin with one of its predecessors or successors. It has been studied recently by Dell'Amico et al. (2012), who propose a mathematical model, lower bounds, as well as heuristic and exact algorithms.

The BPP-P can be seen as a variant of the SALBP-1 with strict precedences. Therefore, the improvements for SALBP-1 can be applied to the model for the BPP-P, substituting the precedence constraints (26) by their strict variant

$$\sum_{s\in S|s<k} x_{si} \geq \sum_{s\in S|s\leq k} x_{sj}, \qquad \forall i \in N, j \in F_i, k \in S. \tag{40}$$

The improved constraints for the station bounds (23) also apply to the BPP-P. They are stronger in the BPP-P, because the bounds on the earliest station E_i and the latest station L_i a task can be assigned to improve when taking the strict precedences into account. In our experiments, we use the better among the limits for the SALBP-1 given in Section 2.2 and the limits imposed by the longest chain of predecessors or successors for the earliest and latest station, respectively, for each task.

5. Computational experiments

We empirically evaluated the performance of all formulations of the SALBP-1 and the SALBP-2 presented in Tables 1 and 2 and the improved formulations of the UALBP-1 and the BPP-P.

For the SALBP, we limited the comparison to the best-known formulations PA and BW from the literature and the new formulation NF for the precedence constraints combined with all four sets of equations for the station limits, for a total of 12 SALBP-1 and SALBP-2 formulations. For the UALBP-1 and the BPP-P, we compare the model as originally proposed with the theoretically best model. In this latter case, we also compare the results of the new model with the results obtained with the tailored branch-and-bound algorithm of Dell'Amico et al. (2012). The detailed results reported in the tables below are available online at http://www.inf.ufrgs.br/algopt/albp.

5.1. Results for SALBP-1 and SALBP-2

The formulations for the SALBP have been tested on the standard benchmark that contains 269 instances of the SALBP-1 and 302 instances of the SALBP-2. All instances are available online (ALB Research Group, 2012). Currently the optimal value is known for the 269 SALBP-1 instances and all except 14 of the SALBP-2 instances. In the evaluations below, solutions for instances without a known optimum were considered optimal only if the solver could prove so. When comparing solution values obtained by different formulations, a result is considered better when the null hypothesis of no improvement in solution values can be rejected at a significance level $p = 0.05$. In all tests, the test statistic used is a conservative, nonparametric paired sign test, where half of the ties were assigned to each sample (Dixon and Mood, 1946).

The experiments were performed on a PC with an Intel Core i7 CPU running at 2.8 GHz and 12 GB of main memory. We used the solver CPLEX 12.4 with standard options, except for an MIP optimality gap of 10^{-5}, running in a deterministic mode with two threads for a maximum time of 600 seconds. The computation times reported are in seconds of real time. Following Pastor and Ferrer (2004) we use in our experiments for the SALBP-1 the lower bound $\underline{m} = \lceil \sum_{i\in N} t_i/c \rceil$ and the upper bound $\overline{m} = \min\{2\underline{m}, |N|\}$ on the number of stations, and for the SALBP-2 the lower bound $\underline{c} = \max\{\max_{i\in N} t_i, \lceil \sum_{i\in N} t_i/m \rceil\}$ and the upper bound $\overline{c} = 2\underline{c}$ on the cycle time.

Table 3
Comparison of formulations of the SALBP-1 on 269 classical benchmark problems (Scholl, 1993)

	Optimal			Suboptimal			
Model	Proven	Time	Unprov.	Feas.	Infeas.	Best	Time
PA1	167	25.5	24	43	35	208	22.2
PA2	182	22.0	17	35	35	216	5.9
PA3	187	26.8	16	24	42	208	8.9
PA4	189	25.4	14	22	44	209	6.6
BW1	187	39.3	17	40	25	210	15.5
BW2	194	30.4	16	44	15	220	4.3
BW3	196	33.5	14	19	40	220	5.3
BW4	196	28.3	14	26	33	228	3.4
NF1	194	22.7	17	57	1	248	3.5
NF2	197	20.5	17	54	1	254	3.1
NF3	201	21.5	12	55	1	252	2.3
NF4	200	20.7	16	49	4	259	3.8

Table 4
Comparison of formulations of the SALBP-2 on 302 classical benchmark problems (Scholl, 1993)

	Optimal			Suboptimal			
Model	Proven	Time	Unprov.	Feas.	Infeas.	Best	Time
PA1	162	49.4	18	122	0	192	39.8
PA2	173	51.4	15	114	0	202	31.7
PA3	187	40.5	9	91	15	204	8.2
PA4	187	33.7	11	88	16	206	11.3
BW1	188	38.6	11	103	0	216	16.1
BW2	185	36.5	16	101	0	220	11.2
BW3	192	26.0	10	94	6	218	5.4
BW4	187	29.7	8	106	1	211	8.8
NF1	187	33.7	16	99	0	237	8.5
NF2	191	39.2	17	94	0	247	6.2
NF3	200	36.9	8	94	0	224	6.2
NF4	196	33.9	11	95	0	225	6.4

Table 3 contains the results for the SALBP-1 and Table 4 for the SALBP-2. For each tested model, we report the number of instances for which the branch-and-cut solver of CPLEX found a provably optimal solution within the time limit (Proven) and the average solution time for these instances (Time). For the remaining instances the solver did not terminate within the time limit. For these runs, we report the number of instances for which an optimal solution was found, but could not be proven to be optimal (Unprov), the number of instances for which a feasible, but not optimal solution was found (Feas), and the number of instances for which the solver was unable to find a feasible solution (Infeas). Column "Best" presents the number of instances where the formulation obtained the best value found over all 12 formulations of the same problem type. The last column

Table 5
Comparison of the standard formulation and an improved formulation for the UALBP-1 on 269 classical benchmark problems (Scholl, 1993)

	Optimal			Sub-optimal			
Model	Proven	Time	Unprov.	Feas.	Infeas.	Best	Time
Standard	175	32.3	21	64	9	226	26.2
NF4	190	44.7	23	55	1	256	32.5

gives the average solution time for the instances that could be solved optimally with all models. For the SALBP-1 this was the case for 160 instances, and for the SALBP-2 for 147 instances.

For both SALBP types, the results show a clear tendency to find and prove more optimal solutions with better station limits and precedence constraints. However, station limits of type 3 and 4 tend to make it more difficult to find feasible solutions, and consequently reduce the number of best solutions found. Statistically, the solution values obtained by formulations NFn are significantly less than those obtained by the corresponding formulations PAn and BWn for both problem types (with the exception of NF3, which is only marginally better than BW3 for the SALBP-2). There is neither a significant difference of solution values between precedence constraints BW and PA, nor between different station limits.

The solution times also tend to decrease with better constraints, but the reduction again is less pronounced or nonexistent for station limits of type 3 and 4. Statistically, formulations NFn solve the instances in significantly shorter time than formulations PAn and BWn, and station limits of type 2 are better than those of type 1, which is expected since they only reduce the number of variables.

In summary, we find that station limits of type 2 help to reduce the solution time, but in general better precedence constraints are more important for improving solutions and reducing the solution time than the improved station limits. In a previous study, Pastor et al. (2004) found no significant difference between formulations PA and BW, but observe that formulation BW leads to shorter solution times. Our results confirm this, but we find the reduction in solution time only significant for the SALBP-2. This may come from the difference between the used solvers (CPLEX 8.0 and CPLEX 12.4). In another study Pastor and Ferrer (2009) find that the dynamic station limits (PA3) increase the number of provably optimal solutions over formulation PA2, which is corroborated by our findings. The solution values, on the other hand, do not decrease significantly, and the dynamic station limits make it more difficult to find feasible solutions.

From a practical point of view one may prefer the formulations NF4 for the SALBP-1 and NF2 for the SALBP-2 which achieve the largest number of best solutions in a short time.

5.2. Results for the UALBP-1

We tested Urban's formulation of the UALBP-1 and the improved formulation proposed in Section 4.1 on the 269 instances of the SALBP-1. The experimental settings were the same as in the tests of the SALBP related above. Table 5 shows the comparison of the two models.

Table 6
Comparison of the results of a branch-and-bound algorithm and a formulation proposed by Dell'Amico et al. (2012, BW1) to the improved formulation NF4 of the BPP-P

		Branch-and-bound			BW1			NF4		
Name	Ins.	Dev.	Time	Un.	Dev.	Time	Un.	Dev.	Time	Un.
Arcus1	16	0.00	5.69	–	0.00	11.09	–	0.00	0.07	–
Arcus2	17	0.00	211.49	–	1.70	2192.00	5	0.00	1,76	–
Barthold	8	0.00	4.41	–	0.00	80.66	–	0.00	0.18	–
Barthol2	27	0.18	575.35	2	8.68	7200.00	27	1.89	5333.87	20
Bowman	1	0.00	0.05	–	0.00	0.04	–	0.00	0.00	–
Buxey	7	0.00	171.94	–	0.00	0.40	–	0.00	0.02	–
Gunther	7	0.00	114.83	–	0.00	0.54	–	0.00	0.02	–
Hahn	5	0.00	0.22	–	0.00	0.20	–	0.00	0.01	–
Heskiaoff	6	0.00	0.22	–	0.00	0.20	–	0.00	0.01	–
Jackson	6	0.00	0.06	–	0.00	0.05	–	0.00	0.00	–
Jaeschke	5	0.00	0.03	–	0.00	0.03	–	0.00	0.00	–
Kilbridge	10	0.00	0.58	–	0.00	1.15	–	0.00	0.02	–
Lutz1	6	0.00	0.13	–	0.00	0.11	–	0.00	0.00	–
Lutz2	11	0.00	106.99	–	6.63	3736.89	5	0.00	1.35	–
Lutz3	12	0.00	1.66	–	0.00	0.80	–	0.00	0.06	–
Mansoor	3	0.00	0.08	–	0.00	0.04	–	0.00	0.00	–
Mertens	6	0.00	0.03	–	0.00	0.02	–	0.00	0.00	–
Mitchell	6	0.00	0.13	–	0.00	0.10	–	0.00	0.00	–
Mukherje	13	0.00	76.92	–	0.61	1154.17	2	0.00	2.30	–
Roszieg	6	0.00	66.86	–	0.00	0.13	–	0.00	0.00	–
Sawyer	9	0.00	134.48	–	0.00	5.41	–	0.00	0.04	–
Scholl	26	0.00	114.39	–	31.52	6160.11	21	0.00	1.66	–
Tonge	16	0.00	15.74	–	1.51	1426.45	3	0.00	0.20	–
Warnecke	16	0.00	870.38	1	7.56	5513.81	12	0.00	109.73	–
Wee-Mag	24	0.00	2.24	–	1.29	2337.76	6	1.06	1089.12	2
Tot./Avg.	269	0.01	157.20	3	4.98	2289.93	81	0.12	40.06	22

As expected, the conclusions for the SALBP-1 also hold for the UALBP-1. The improved model finds and proves more optimal solutions, and finds more and significantly better solution values (for $p = 0.05$) in about the same time used by the original model.

5.3. *Results for the BPP-P*

We finally tested the formulation of the BPP-P proposed in Section 4.2 and compared it to the results obtained by Dell'Amico et al. (2012). These results are available online (Dell'Amico et al., 2010) and have been obtained in an environment similar to ours (a PC with a Pentium processor running at 3 GHz, and CPLEX 12). To be able to make a direct comparison we used the same settings running the solver with only one thread and a time limit of two hours.

The results can be seen in Table 6. Results are reported for each group of instances the number of instances (Ins), the results for the Branch-and-bound algorithm as well as the model proposed by Dell'Amico et al. (2012), and for the model proposed here. For each approach the table gives the average relative deviation from the best known lower bound (Dev), the average solution time (Time), and the number of instances which could not be solved within the time limit (Un). Such instances contribute with the time limit of 7200 seconds to the average execution time. Note that the relative deviations slightly differ from those reported in Dell'Amico et al. (2010), since we updated the lower bounds according to our results.

The new formulation drastically improves the results obtained by the basic model. The number of instances that could not be solved in two hours reduces from 81 to 22, and the average execution time is a factor of 50 faster. The best model is competitive with the branch-and-bound algorithm specifically designed for this problem: although it solves 19 problems less, it obtains a comparable relative deviation on the remaining instances, and is able to solve them a factor of almost four faster. Again, the conclusion for the SALBP holds also for the BPP-P, and the improved constraints seem to be even more effective for strict precedences.

6. Conclusions

We have proposed an improved formulation of the precedence constraints and the station limits for the SALBP, and shown that they theoretically dominate other constraints proposed in the literature. They are applicable to related assembly line balancing problems with similar constraints. Comparing the new and existing models, we have provided a classification of the relationships between models using impulse variable used in the literature.

Computational experiments confirm the theoretical comparison. The proposed precedence constraints can improve upon the constraints of Patterson and Albracht (1975) and Bowman (1960), finding and proving the optimality of more solutions, and finding more best values. A conservative statistical test shows that the improvement of the solution value is significant.

Two case studies on the UALBP-1 and the BPP-P indicate that the conclusions for the SALBP also apply to related problems, which further highlights the importance of the proposed models.

References

Aase, G.R., Schniederjans, M.J., Olson, J.R., 2003. U-OPT: an analysis of exact U-shaped line balancing procedures. *International Journal of Production Research* 41, 17, 4185–4210.

Ağpak, K., Fatih Yegl, M., Gökçen, H., 2012. Two-sided U-type assembly line balancing problem. *International Journal of Production Research* 50, 18, 5035–5047.

ALB Research Group, 2011. Homepage for assembly line optimization research. Available at http://www.assembly-line-balancing.de.

Baybars, I., 1986. A survey of exact algorithms for the simple assembly line balancing problem. *Management Science* 32, 909–932.

Becker, C., Scholl, A., 2006. A survey on problems and methods in generalized assembly line balancing. *European Journal of Operational Research* 168, 3, 694–715.

Blum, C., 2008. Beam-ACO for simple assembly line balancing. *INFORMS Journal on Computing* 20, 4, 618–627.

Bowman, E.H., 1960. Assembly-line balancing by linear programming. *Operations Research* 8, 3, 385–389.

Chiang, W.-C., Kouvelis, P., Urban, T.L., 2007. Line balancing in a just-in-time production environment: balancing multiple U-lines. *International Journal of Production Research* 39, 347–359.

Dell'Amico, M., Díaz Díaz, J.C., Iori, M., 2010. Bin packing problem with precedence constraints. Technical report, University of Modena and Reggio Emilia, Italy.

Dell'Amico, M., Díaz Díaz, J.C., Iori, M., 2012. The bin packing problem with precedence constraints. *Operations Research* 60, 6, 1491–1504.

Dixon, W.J., Mood, A.M., 1946. The statistical sign test. *Journal of the American Statistical Association* 41, 557–566.

Erin, R., 2007. Mixed-integer linear programming approach to U-line balancing with objective of achieving proportional throughput per worker in a dynamic environment. Master's thesis, Kocaeli University, Turkey.

Fleszar, K., Hindi, K.S., 2003. An enumerative heuristic and reduction methods for the assembly line balancing problem. *European Journal of Operational Research* 145, 606–620.

Gökçen, H., Ağpak, K., 2004. A goal programming approach to simple U-line balancing problem. *European Journal of Operational Research* 171, 577–585.

Kara, Y., Tekin, M., 2009. A mixed integer linear programming formulation for optimal balancing of mixed-model U-lines. *International Journal of Production Research* 47, 15, 4201–4233.

Pastor, R., Corominas, A., Lusa, A., 2004. Different ways of modelling and solving precedence and incompatibility constraints in the assembly line balancing problem. In *Frontiers in Artificial Intelligence and Applications*, Vol. 113. IOS Press, Amsterdam, pp. 359–366.

Pastor, R., Ferrer, L., 2009. An improved mathematical program to solve the simple assembly line balancing problem. *International Journal of Production Research* 47, 11, 2943–2959.

Pastor, R., Ferrer, L., García, A., 2007. Evaluating optimization models to solve SALBP. In Gervasi, O., Gavrilova, M. (eds) *Proceedings of ICCSA 2007*, Vol. 4705, Part I of *LNCS*. Springer, Berlin, pp. 791–803.

Patterson, J.H., Albracht, J.J., 1975. Assembly line balancing: 0-1 programming with Fibonacci search. *Operations Research* 23, 166–174.

Scholl, A., 1993. Data of assembly line balancing problems. Technical report 16/1993, TH Darmstadt, Germany. Schriften zur Quantitativen Betriebswirtschaftslehre.

Scholl, A., 1999. *Balancing and Sequencing of Assembly Lines* (2nd edn). Physica Verlag, Heidelberg.

Scholl, A., Becker, C., 2006. State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research* 168, 3, 666–693.

Scholl, A., Klein, R., 1999. Balancing assembly lines effectively – a computational comparison. *European Journal of Operational Research* 114, 50–58.

Scholl, A., Voß, S., 1996. Simple assembly line balancing – heuristic approaches. *Journal of Heuristics* 2, 217–244.

Sewell, E.C., Jacobson, S.H., 2012. A branch, bound, and remember algorithm for the simple assembly line balancing problem. *INFORMS Journal of Computing* 24, 3, 433–442.

Thangavelu, S.R., Shetty, C.M., 1971. Assembly line balancing by zero-one integer programming. *AIIE Transactions* 3, 1, 61–68.

Urban, T.L., 1998. Note, optimal balancing of U-shaped assembly lines. *Management Science* 44, 5, 738–741.

White, W.W., 1961. Comments on a paper of Bowman. *Operations Research* 9, 2, 274–276.