The effect of strengthened linear formulations on improving the lower bounds for the part families with precedence constraints problem

Lídia Lampreia Lourenço
Faculdade de Ciências e Tecnologia, New University of Lisbon
Centro de Investigação Operacional, Faculdade de Ciências, University of Lisbon
email: lll@fct.unl.pt fax: 351 212 948 391
address: Faculdade de Ciências e Tecnologia - Quinta da Torre 2829-516 Caparica, Portugal

Margarida Vaz Pato
Instituto Superior de Economia e Gestão, Technical University of Lisbon
Centro de Investigação Operacional, Faculdade de Ciências, University of Lisbon
email: mpato@iseg.ulisboa.pt
address: Instituto Superior de Economia e Gestão - Rua do Quelhas, 6, 1200-781 Lisboa, Portugal

Abstract

The part families with precedence constraints problem (PFP) arises in industry, when flexible manufacturing systems are designed within a group technology approach. The aim of this problem is to arrange parts into families by imposing capacity constraints, concerning both the number of parts and processing times, besides precedence constraints in the building of families.

Mixed binary linear programming formulations for the PFP are presented. In endeavoring to strengthen the linear relaxations for the formulations, and hence generate better lower bounds for the optimal value of PFP, some valid inequalities, based on the properties of the problem, were deduced.

The lower bounds obtained significantly improved through the very weak, frequently null, bounds resulting from the original linear relaxation. Moreover, it may be concluded that these models can be a useful methodology to enforce the performance of any branch-and-bound for this very important problem in flexible manufacturing systems.

Keywords: flexible manufacturing systems; part families problem; precedence constraints; mixed binary linear formulations; valid inequalities.
1 Introduction

The part families with precedence constraints problem arises in industrial plants that operate with flexible manufacturing systems (Stecke [21], Ng [15]). A flexible manufacturing system is designed to manufacture a wide range of products ordered, each of which is to be produced in small quantities. The manufacturing of each product demands various types of operations, which are performed by tool machines, following a specific sequence. For this reason, one must know the $N$ products or small lots of products - denoted as parts - ordered. For each part, let $i$ be the part, further information is also required regarding the tools used to produce the part, its execution time $p_i$, considered as independent of the machine, and the parts that have precedence on production over it. In this organization of the production process, one assumes the formation of a maximum of $K$ ($2 \leq K \leq N$) disjoint families of parts, so as to set up groups of parts with similar manufacturing features. The parts assigned to the same family will be processed in one or more machines, whose tools are stored in magazine tools. The machines may be different or otherwise, and their tool magazines always possess clearly defined capacities. It is also assumed that the machines are of the sum-batch type, because the total time taken to produce the parts assigned is the sum of the time for each part. These specific features lead to capacity restrictions in the formation of families, both regarding the number of parts assigned to each family and the total processing time per family.

One considers precedence constraints in the grouping of parts into families as in the production of certain parts other components are incorporated which should have been produced beforehand. Assuming that the production process will be organized by increasing order of the families’ index, the precedence constraints impose that a part can only be assigned to a family if all its predecessors have been assigned to that family or to lower index families.

The preference for grouping parts with similar manufacturing features into the same family has led us to minimize the sum of dissimilarities among parts to be found in the same family which defines the optimization criterion. Here, the dissimilarity between two parts $i$ and $j$ is the Hamming distance, represented by $d_{ij}$, and results from the ratio between the number of different tools and the total number of tools involved in producing these two parts.

In short, the problem of part families with precedence constraints (PFP) requires that the $N$ parts be grouped into $K$ or less disjoint families, subject to capacity constraints as to the number of parts and the processing time per family. This grouping is also subject to the constraints that impose precedence relationships in the assignment of parts to the families. Among all the
solutions that satisfy these constraints, the choice should be determined by the single criterion of minimizing the total dissimilarity among parts within the same family. It should be noted that the optimal solution of a PFP instance may have a number of non-empty families below that of K, which is due to effect of the capacity and precedence constraints. However, should these constraints be redundant, then there exists at least one optimal solution, formed by K non-empty families (Lourenço [9]). The PFP can be viewed as a K-partition problem in a complete graph, whose vertices represent parts, subject to capacities and precedence constraints, with a minimum dissimilarity optimization criterion.

Part of the PFP characterization was inspired by a problem due to Kusiak [8] for which this author presented a p-median formulation. Precedence constraints were mentioned in [4] by Finke and Kusiak and in [14] by Moon, Lee and Seo. Viswanathan [23] refers to the capacity constraints imposed by the number of parts to attribute to a machine or cell of machines, noting that this parameter is indicated by the production design analyst. The PFP was also motivated by the problem proposed by Gunther et al. [6], in which the objective is to assign operations that use common tools to the same work station. However, the PFP is a more general issue insofar it considers all these features simultaneously.

Bellow you will find the data for an instance of the problem with \( N = 10 \) parts, \( K = 4 \) families, processing times \( p_1, p_2, p_3 = 6, p_4 = 3, p_5, p_6, p_7, p_8, p_9, p_{10} = 10 \); capacities of each family relative to the number of parts and the processing time \( M_1 = 4, M_2 = 2, M_3 = 7, M_4 = 2 \) and \( T_1 = 20, T_2 = 5, T_3 = 70, T_4 = 4 \), respectively; direct precedences among the parts, given in figure 1.1, \( 1 \prec 2, 1 \prec 3, 2 \prec 4, 3 \prec 4, 1 \prec 5, 1 \prec 6, 1 \prec 7, 1 \prec 8, 1 \prec 9, 8 \prec 9, 1 \prec 10 \); and dissimilarities represented by matrix \([d_{ij}]\) also shown in figure 1.1.
\[
\begin{bmatrix}
0.5 & 0.6 & 0.1 & 0.1 & 0.6 & 0.7 & 0.9 & 0.3 \\
0.7 & 0.9 & 0.5 & 0.3 & 0.4 & 0.9 & 0.7 & 0.4 \\
0.9 & 0.4 & 0.8 & 0.3 & 0.6 & 0.7 & 0.6 \\
0.3 & 0.7 & 0.7 & 0.6 & 0.2 & 0.5 \\
0.9 & 0.9 & 0.4 & 0.9 & 0.3 \\
0.9 & 0.3 & 0.9 & 0.1 \\
0.7 & 0.6 & 0.3 \\
0.9 & 0.2 \\
0.3 
\end{bmatrix}
\]

Figure 1.1: Precedence network and dissimilarity matrix of a small instance.

The following grouping, represented in figure 1.2, constitutes one feasible solution to this problem: \( F_1 = \{1, 2, 3\}, \ F_2 = \{4\}, \ F_3 = \{5, 6, 7, 8, 9, 10\}, \ F_4 = \{\}. \) As one can verify, the precedence constraints are satisfied, as well as, the capacity constraints in terms of number and processing time of the parts placed there. The total dissimilarity of this feasible solution is equal to 10.4, a value calculated from \((d_{12} + d_{13} + d_{23}) + (d_{56} + d_{57} + d_{58} + d_{59} + d_{5,10} + \ldots + d_{9,10})\). Moreover, this is an optimal solution because 10.4 is precisely the minimum dissimilarity.

The PFP is a very difficult problem. In fact, it was proved to be NP-hard because the clustering problem is one such particular case (Lourenço [9]). Though it has been heuristically solved in Lourenço and Pato [10]. In order to evaluate the quality of such solutions and thus improve the performance of branch and bound methods applied to mixed binary formulations, lower bounds from strengthened formulations were developed by Lourenço [9] and are presented in this paper. Proof of the results may be consulted in Lourenço [9].
In section 2, the authors present a mixed binary linear formulation with aggregated variables for the PFP, denoted by BF1, followed, in section 3, by results concerning the linear relaxation of BF1. In section 4 a preprocessing of the solution is performed by fixing the value of some of the variables. Valid inequalities were developed to strengthen the above referred mixed binary linear formulation and are presented in section 5. Section 6 contains a mixed binary linear formulation with disaggregated variables for the PFP, the so-called BF2. Finally, section 7 describes the computational experiment and section 8 is devoted to the conclusions.

2 Mixed binary linear formulation - BF1

The PFP may be formulated as a mixed binary linear programming problem with aggregated variables (Lourenço and Pato [11]) comprising the following indexes, parameters and variables:

- $i, j$ - part indexes
- $k$ - family index
- $N$ - number of parts ($N \in \mathbb{N}$)
- $K$ - maximum number of families ($K \in \mathbb{N}, 2 \leq K \leq N$)
- $M_k$ - maximum number of parts for family $k$ ($M_k \in \mathbb{N}, M_k \leq N$)
- $p_i$ - production time of part $i$ ($p_i \in \mathbb{R}_0^+$)
- $T_k$ - maximum processing time for family $k$ ($T_k \in \mathbb{R}^+$)
- $g_{ij}$ - direct precedence relationships between parts $i$ and $j$, $i < j$, ($g_{ij} \in \{0, 1\}$)
- $d_{ij}$ - dissimilarity between parts $i$ and $j$, an element of the symmetric matrix $[d_{i,j}]$ whose diagonal elements are zero ($d_{ij} \in \mathbb{R}_0^+$)
- $x_{ik}$ - binary variable indicating whether part $i$ is in family $k$ (=1) or not (=0) ($i = 1, \ldots, N$; $k = 1, \ldots, K$)
- $y_{ij}$ - binary variable indicating whether parts $i$ and $j$ are in the same family (=1) or not (=0) ($i < j$; $i = 1, \ldots, N - 1$; $j = i + 1, \ldots, N$).

The formulation, denoted by BF1, follows:

$$\text{Min } \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} y_{ij} \quad (2.1)$$

s. to $y_{ij} \geq x_{ik} + x_{jk} - 1 \quad i = 1, \ldots, N - 1; j = i + 1, \ldots, N; k = 1, \ldots, K \quad (2.2)$
\[ \sum_{k=1}^{K} x_{ik} = 1 \quad i = 1, \ldots, N \quad (2.3) \]
\[ \sum_{k=1}^{K} kx_{ik} \leq \sum_{j=1}^{K} kx_{jk} \quad (i, j) : g_{ij} = 1 \quad (2.4) \]
\[ \sum_{i=1}^{N} x_{ik} \leq M_k \quad k = 1, \ldots, K \quad (2.5) \]
\[ \sum_{i=1}^{N} p_i x_{ik} \leq T_k \quad k = 1, \ldots, K \quad (2.6) \]
\[ x_{ik} = 0, 1 \quad i = 1, \ldots, N; k = 1, \ldots, K \quad (2.7) \]
\[ y_{ij} = 0, 1 \quad i = 1, \ldots, N - 1; j = i + 1, \ldots, N. \quad (2.8) \]

According to the criterion of minimizing total dissimilarity, the objective function (2.1) represents the value of the total dissimilarity, that is, the sum of the values of dissimilarity between pairs of parts placed in the same family. Constraints (2.2) force each variable \( y_{ij} \) in the optimal solution to take the value of 1 if \( i \) and \( j \) belong to the same family. Should these parts be in different families in the optimal solution, the variable \( y_{ij} \) contributes with the value 0 to the objective function. The set of constraints (2.3) forces each part to belong to one and one family only. The precedence constraints (2.4), defined for the whole pair of parts \((i, j)\) such that \( i \) directly precedes \( j \), ensure that part \( j \) cannot be placed in a family whose index is lower than that of the family to which part \( i \) belongs. Capacity constraints (2.5) and (2.6) do not allow violation of the limits regarding the number of parts and processing time of each family, respectively.

Note that, this linear formulation for the problem of the part families with precedence constraints was used by Aronson and Klein [1] for a classification problem arising in information systems drawn up to support software design.

### 3 Linear Relaxation

The linear relaxation of BF1, denoted by BF1, is the problem resulting from elimination of the integrality of the variables, that is, from substituting \( x_{ik} \in \{0,1\} \) by \( 0 \leq x_{ik} \leq 1 \), with \( i = 1, \ldots, N; k = 1, \ldots, K \) and \( y_{ij} \in \{0,1\} \) by \( 0 \leq y_{ik} \leq 1 \), with \( i = 1, \ldots, N - 1; j = i + 1, \ldots, N \). These bounding values for the variables \( y_{ij} \) were used in the formulation (2.1)-(2.8) for BF1 because in any optimal solution of BF1 the values of variables \( y_{ij} \) are always binary due to the joint effect of minimizing (2.1), while respecting (2.2) and the above bounds.
In some cases, an analysis of the various parameters of an instance of BF1 enables one to verify that the optimal value of its linear relaxation is nil. The result that follows contains three sufficient conditions for the optimal value of linear relaxation to be nil. The first, condition 1.a) refers to the linear relaxation of the particular case of BF1, in which the capacity constraints are redundant. The second, condition 1.b, concerns instances with equal capacities and the last refers to certain cases with different capacities. In any of these cases a solution exists in which the value of all \( x_{ik} \) variables is no greater than 0.5. Thus, from constraints (2.2), \( y_{ij} = 0 \) for all \((i, j)\) and this proves that the BF1 has a null optimal value.

**Result 1.** The linear relaxation of any BF1 instance that is not unfeasible and:

1.a) where the capacity constraints are redundant;

or

1.b) such that the maximum number of parts and the maximum processing time is equal for all the families;

or

1.c) such that \( \sum_{k=1}^{K} \bar{x}_{ik} \geq 1 \) \((i = 1, \ldots, N)\), where the vector \( \bar{x} \) is defined as follows, with

\[
\text{min}_k = \min \left\{ \frac{M_k}{N}, \frac{T_k}{\sum_{i=1}^{N} p_i} \right\}:
\]

\[
\bar{x}_{ik} = \begin{cases} 
\text{min}_k & \text{if } \text{min}_k < 0.5 \\
0.5 & \text{if } \text{min}_k \geq 0.5 
\end{cases} \quad (i = 1, \ldots, N; k = 1, \ldots, K)
\]

has null value.

The following result presents a necessary and sufficient condition for the optimal value of the BF1 to be positive. This condition is satisfied when, in any feasible solution of the linear relaxation, there exists a pair of parts with positive dissimilarity and a family such that the sum of the values of the respective variables (relative to those parts and to that family) is greater than 1. Through constraint (2.2), this same condition enforces the value 1 for the variable \( y_{ij} \). Thus, the objective function, at least, attains the value of the aforesaid positive dissimilarity.

**Result 2.** The optimal value of the linear relaxation of a BF1 instance is positive if and only if in any feasible solution \((\bar{x}, \bar{y})\) of BF1, \( \exists i, j \in \{1, \ldots, N\} \) where \( i < j \) and \( \exists k \in \{1, \ldots, K\} \) such that \( \bar{x}_{ik} + \bar{x}_{jk} > 1 \) and \( d_{ij} > 0 \).
On the strength of the previous results, it is clear that the linear relaxation lower bounds will be weak and, in most practical cases, null.

4 Preprocessing

A priori, one finds that, the value of certain variables in any feasible solution of the BF1 formulation can be determined.

Taking advantage of the capacities of families, one can apply the following rule to fix the values of specific variables of any feasible solution of BF1. Here and throughout the text, $M^* = \max\{M_1, M_2, \ldots, M_K\}$ and $T^* = \max\{T_1, T_2, \ldots, T_K\}$.

**Rule 1.** In any feasible solution $(x, y)$ of BF1

\begin{align*}
x_{ik} &= 0 & i = 1, \ldots, N; k = 1, \ldots, K: p_i > T_k \\
x_{ik} &= 0 \quad \text{and} \quad x_{ik'} = 1 & i = 1, \ldots, N; \exists k' \in \{1, \ldots, K\} \ p_i \leq T_{k'} \quad \forall k \in \{1, \ldots, K\} \setminus \{k'\}: p_i > T_k \\
y_{ij} &= 0 & i = 1, \ldots, N - 1; j = i + 1, \ldots, N: p_i + p_j > T^*.
\end{align*}

Again, the values of some variables can be fixed but now by also taking into consideration the network of PFP precedences. Let $G = (V, E)$ be the oriented network, representing the precedence relationships between the parts where $V$ is the set of vertices (parts) and $E$ the set of arcs: vertices $i$ and $j$ define the initial and final extremities respectively of an arc, provided there exists a direct precedence relationship of $i$ in relation to $j$, that is, $i \prec j$. When, in network $G$, there is a path from vertex $i$ to $j$, one may say that there is a transitive precedence relationship from $i$ to $j$, represented by $i \prec\prec j$.

**Rule 2.** Let $i$ and $j$ be a pair of parts such that $i \prec\prec j$ and $\text{Cam}_{ij} = \{l \in \{1, \ldots, N\} : i \prec\prec l \text{ and } l \prec\prec j\}$. If $|\text{Cam}_{ij}| + 2 > M^*$ or if $(\sum_{l \in \text{Cam}_{ij}} p_l) + p_i + p_j > T^*$ then in any feasible solution $(x, y)$ of BF1 one has

\begin{align*}
y_{ij} &= 0; x_{iK} = 0; x_{j1} = 0 \\
y_{ir} &= 0 & r : j \prec\prec r \\
y_{sj} &= 0 & s : s \prec\prec i \\
x_{ik} + x_{jk} &\leq 1 & k = 1, \ldots, K.
\end{align*}
Aronson and Klein [1] developed a preprocessing result by imposing in constraint (2.3) relative to part \( i \) a lower bound \( L_i \) and an upper bound \( U_i \), both bounds on the index of the family in which this part can be placed. These bounds are calculated in accordance with its transitive predecessors and its transitive successors, respectively. Below, Rule 3 presents this issue for the BF1.

**Rule 3.** In any feasible solution \((x, y)\) of BF1 one has

\[
x_{ik} = 0 \quad i = 1, \ldots, N; \quad k = 1, \ldots, K: \quad k < L_i \text{ or } k > U_i
\]

(4.8)

where \( TPred(i) \) and \( TSuc(i) \) are respectively the set of transitive predecessors and transitive successors of part \( i \) and \( U_i = \min\{u_1^i, u_2^i\} \) and \( L_i = \max\{l_1^i, l_2^i\} \) with

\[
u_1^i = \max\{g = 1, \ldots, K : p_i + \sum_{j \in TSuc(i)} p_j \leq \sum_{k=g}^K T_k\}
\]

\[
u_2^i = \max\{g = 1, \ldots, K : 1 + |TSuc(i)| \leq \sum_{k=g}^K M_k\}
\]

\[
l_1^i = \min\{g = 1, \ldots, K : p_i + \sum_{j \in TPred(i)} p_j \leq \sum_{k=g}^K T_k\}
\]

\[
l_2^i = \min\{g = 1, \ldots, K : 1 + |TPred(i)| \leq \sum_{k=g}^K M_k\}
\]

**5 Valid Inequalities**

The characteristics of the BF1 formulation for the PFP in particular and the different constraint systems associated with it, permit one to establish different types of valid inequalities. Once incorporated in this mixed binary linear formulation, these constraints can strengthen the respective linear relaxation lower bound.

The first valid inequalities are deduced from the grouping characteristics. There follows a study of the inequalities obtained from the precedence constraints and, lastly, from the capacity constraints. Here, the convex hull of the feasible region of BF1 is represented by \( \text{conv}(F(BF1)) \), where the feasible region is denoted by \( F(BF1) \).

**5.1 Inequalities based on the grouping characteristics**

Valid inequalities are developed because the PFP, regarded as an optimization problem on the complete network of the parts can be considered the problem of searching for a partition of a
complete graph into no more than $K$ cliques, minimizing total dissimilarity and subject to additional constraints. Hence, bearing in mind the number of parts and families, one can deduce lower bounds for the total number of $y_{ij}$ variables taking a positive value. This is present in the next result.

**Result 3.** The following constraint is a valid inequality for $\text{conv}(F(BF1))$:

$$l_{\text{min}} \leq \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} y_{ij}$$

where

$$l_{\text{min}} = \begin{cases} \frac{N}{K}(N - \frac{K}{2}(\lceil \frac{N}{K} \rceil + 1)) & \text{if } \lceil \frac{N}{K} \rceil \geq 2 \\ N - K & \text{if } \lceil \frac{N}{K} \rceil < 2. \end{cases}$$

Other types of constraints follow. But they only concern variables $y_{ij}$ and results from the transitive nature of appertain to the family sets.

**Result 4.** In considering the three parts $i$, $j$ and $t$, the following constraints, that is, triangular inequalities, are valid inequalities for $\text{conv}(F(BF1))$:

$$y_{ij} \geq y_{it} + y_{tj} - 1 \quad t < i < j \quad (5.1.3)$$

$$y_{ij} \geq y_{it} + y_{tj} - 1 \quad i < t < j \quad (5.1.4)$$

$$y_{ij} \geq y_{it} + y_{jt} - 1 \quad i < j < t. \quad (5.1.5)$$

It should be mentioned that Grötschel and Wakabayashi [5] used the triangular inequalities as constraints to formulate the partition problem of a network with $N$ vertices into cliques. These inequalities are cited in the paper by Park, Lee and Park [17] as facets for the convex hull of the set of feasible solutions of a formulation for the maximum weight clique problem with a capacity constraint in the number of vertices.

### 5.2 Inequalities from precedence constraints

In the next result, conditions are imposed which guarantee that, in the optimal solution, either the index family 1 or the index $K$ family is not empty. Let $Spred$ be the set of parts with no direct predecessors and $Ssuc$ be the set of parts with no direct successors.

**Result 5.**
5.a) Consider a PFP with, at least, one feasible solution where the index 1 family is not empty, then the following constraint is satisfied by at least one optimal solution of BF1:

\[
\sum_{i \in Spred} x_{i1} \geq 1. \tag{5.2.1}
\]

5.b) Consider a PFP with, at least, one feasible solution where the index K family is not empty, then the following constraint is satisfied by at least one optimal solution of BF1:

\[
\sum_{i \in Ssuc} x_{iK} \geq 1. \tag{5.2.2}
\]

Next Result 6 was developed by relating parts pairwise and considering their values, along with the transitive precedence relationships between parts.

**Result 6.** Consider five parts \(i, j, l, t\), and \(u\), such that \(i \prec \prec j \prec \prec t \prec \prec u\) and \(i \prec \prec l \prec \prec t\), the following constraints are valid inequalities for \(\text{conv}(F(BF1))\):

\[
x_{j1} \leq x_{i1}; \quad x_{iK} \leq x_{jK} \tag{5.2.3}
\]

\[
(x_{ik} + x_{tk}) - x_{jk} \leq 1 \quad k = 1, \ldots, K \tag{5.2.4}
\]

\[
y_{ij} \geq y_{it}; \quad y_{jt} \geq y_{it}; \quad y_{it} \leq y_{jt} \tag{5.2.5}
\]

\[
y_{ij} + y_{jt} \geq x_{ik} + x_{jk} + x_{tk} - 1 \quad k = 1, \ldots, K \tag{5.2.6}
\]

\[
y_{ij} + y_{jt} + y_{tu} \geq x_{ik} + x_{jk} + x_{tk} + x_{uk} - 1 \quad k = 1, \ldots, K. \tag{5.2.7}
\]

### 5.3 Inequalities from capacities

The first two groups of inequalities presented in Result 7 relate one part \(j\) with all the others, through variables \(y_{ij}\). Note that these variables do not identify the family to which the parts are assigned, which is why one chooses the greatest capacities: \(M^*\), number capacity for inequality (5.3.1) and \(T^*\), processing time capacity for (5.3.2). Each of these inequalities, for instance, the inequality \(j\), results from consideration of all the possibilities of assigning part \(j\).

In this same result, the next group of inequalities (5.3.3) involves lifting the coefficients of the variables \(x_{ik}\) in the capacity constraints (2.6) of the formulation BF1. One knows that the lifting operation is performed on the coefficients of the variables in a constraint, with a view to obtaining a valid inequality that dominates the inequality from which it resulted (Wolsey [24]). This type of inequalities was developed for the generalized assignment problem by Farias and Nemhauser [3].
**Result 7.** The following constraints are valid inequalities for \( \text{conv}(F(BF1)) \):

\[
\begin{align*}
\sum_{i=1}^{j-1} y_{ij} + \sum_{i=j+1}^{N} y_{ji} & \leq M^* - 1 & j = 1, \ldots, N \\
\sum_{i=1}^{j-1} p_i y_{ij} + \sum_{i=j+1}^{N} p_i y_{ji} & \leq T^* - p_j & j = 1, \ldots, N \\
p_i x_{i1} + (T_k - p_i) x_{i12} + \ldots + (T_k - p_i) x_{i1k} + \ldots + (T_k - p_i) x_{i2k} + \\
\quad + \ldots + (T_k - p_i) x_{i2k} & \leq T_k & i_1, i_2 = 1, \ldots, N; p_{i1} + p_{i2} \geq T_k.
\end{align*}
\]

(5.3.1)  
(5.3.2)  
(5.3.3)  

6  **Mixed binary linear formulation - BF2**

Introduction of yet another index in the variables \( y_{ij} \) caused them to disaggregate. This leads to the appearance of the index of the family in which the two parts \( i \) and \( j \) are grouped. In this way, the variables \( y_{ijk} \), \( \forall i < j, i, j \in \{1, \ldots, N\}, \forall k \in \{1, \ldots, K\} \) characterize an extended formulation for the PFP (Pulleyblank [18]). Though it has more variables and more constraints, it will enable one to develop valid inequalities, which dominate those obtained by replacing \( y_{ij} \) by the sums of \( y_{ijk} \).

The variables used in BF2 and the formulation itself are now defined:

\( y_{ijk} \) - binary variable which indicates whether parts \( i \) and \( j \) are in the same family \( k \) (=1) or not (=0) \( i = 1, \ldots, N - 1; j = i + 1, \ldots, N; k = 1, \ldots, K \)

\[
\begin{align*}
\text{Min} & \quad \sum_{k=1}^{K} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} d_{ij} y_{ijk} \\
\text{s. to} & \quad y_{ijk} \geq x_{ik} + x_{jk} - 1 & i = 1, \ldots, N - 1; j = i + 1, \ldots, N; k = 1, \ldots, K \\
\text{constraints} & \quad (2.3) \text{ to } (2.7) \\
& \quad y_{ijk} = 0,1 & i = 1, \ldots, N - 1; j = i + 1, \ldots, N; k = 1, \ldots, K.
\end{align*}
\]

(6.1)  
(6.2)  
(6.3)

It was proved that the optimal value of the linear relaxation of BF2, represented by BF2, is equal to the optimum of BF1. Hence, in most cases, the bound from BF2 is expected to be nil or
of very poor quality, as the one from BF1. Valid inequalities were then studied to strengthen the optimal value of BF2.

The valid inequalities for the set of feasible solutions of BF1, involving only variables $x_{ik}$, can be directly included in BF2. As for the valid inequalities developed for BF1, which involve variables $y_{ij}$, they require adaptation to BF2, which may consist in replacing $y_{ij}$ by $\sum_{k=1}^{K} y_{ijk}$ or simply substituting variables $y_{ij}$ by $y_{ijk}$ or else, in other deductions performed with the purpose of benefiting the optimum of the linear relaxation of this extended formulation BF2. Such is the case when some constraints are deduced within BF2: use is made of the exact values of the capacity limits regarding the number of parts and total processing time for each family $k$ ((6.4)-(6.5) in Result 8), instead of the maximum of all the capacities, as in the case of Result 7 for BF1.

The four last types of inequalities that follow in Result 8 are specifically valid for $F(BF2)$ only. In BF2 the variables $x_{ik}$ and $y_{ijk}$ are defined for a given family. For this reason it is possible to establish a relationship between the two types of variables ((6.6)-(6.7)). As to the valid inequalities (6.8)-(6.9) they result from application of the first level of the linear reformulation technique due to Sherali and Adams [20] to the quadratic formulation for the PFP, given in Lourenço [9].

Result 8. The following constraints are valid inequalities for $\text{conv}(F(BF2))$:

\begin{align}
\sum_{i=1}^{j-1} y_{ijk} + \sum_{i=j+1}^{N} y_{ijk} & \leq (M_k - 1)x_{jk} \quad j = 1, \ldots, N; \ k = 1, \ldots, K \tag{6.4} \\
\sum_{i=1}^{j-1} p_i y_{ijk} + \sum_{i=j+1}^{N} p_i y_{ijk} & \leq (T_k - p_j)x_{jk} \quad j = 1, \ldots, N; \ k = 1, \ldots, K \tag{6.5} \\
y_{ijk} & \leq x_{ik} \quad i = 1, \ldots, N - 1; \ j = i + 1, \ldots, N; \ k = 1, \ldots, K \tag{6.6} \\
y_{ijk} & \leq x_{jk} \quad i = 1, \ldots, N - 1; \ j = i + 1, \ldots, N; \ k = 1, \ldots, K \tag{6.7} \\
\sum_{i=1}^{j-1} y_{ijk} + \sum_{i=j+1}^{N} y_{ijk} & \geq \sum_{i=1}^{N} x_{ik} - (M_k - 1)x_{jk} - M_k \quad j = 1, \ldots, N \quad k = 1, \ldots, K \tag{6.8} \\
\sum_{i=1}^{j-1} p_i y_{ijk} + \sum_{i=j+1}^{N} p_i y_{ijk} - \sum_{i=1}^{N} p_i x_{ik} & \geq (T_k - p_j)x_{jk} - T_k \quad j = 1, \ldots, N \quad k = 1, \ldots, K. \tag{6.9}
\end{align}
Inequalities similar to (6.4)-(6.5) were deduced for the maximal clique problem with a capacity constraint in the number of vertices (Park, Lee and Park [17], Macambira and Souza [13], Hunting, Faile and Kern [7]). These define facets for the polyhedron of the maximal clique problem with a capacity constraint if and only if \( b \leq N - 1 \), where \( b \) is the capacity of the clique and \( N \) the number of vertices of the complete network. It should be noted that the inequalities are an adaptation of (5.3.1)-(5.3.2), developed for BF1. However, they dominate the inequalities obtained by merely introducing index \( k \). In other words, one refers to inequalities that result from the corresponding ones for the previous formulation by replacing \( y_{ij} \) by \( y_{ijk} \).

Finally, one should mention that formulation BF2, strengthened by valid inequalities (6.4)-(6.9), coincides with the mixed binary linear formulation resulting from application of the hierarchical linear reformulation technique due to Sherali and Adams [20] to a quadratic formulation for the PFP, given in Lourenço [9].

7 Computational Experiments

Although the formation of part families within flexible manufacturing systems amounts to a problem whose application is real and often mentioned in the literature, to the authors’ knowledge it has not been considered in the literature. As it is difficult to ascertain what the dimensions of the instances are, besides the appropriate values of the parameters for a real application of the problem, the computational experiment is undertaken with instances that claim to represent different situations.

Two sets, each consisting of 25 instances each, were built, by resorting, in part, from the literature concerning the assembly line balancing problem in industry (Scholl [19] and [16]). This option was made in view of the fact that data is available on both the number of tasks (parts) to be assigned as well as the number of work stations (families), the network of precedences between the tasks and the execution times for completion of the tasks. Data referring to the capacities of families and the dissimilarities between parts were randomly generated by using the random function of the Pascal programming language (Lourenço [9]). In the first set, \( A \) - instances \( I_1 \) to \( I_{25} \) -, each instance has equal capacities for all the families, in terms of the number of parts besides the processing time. In the second set, \( B \) - instances \( I_{26} \) to \( I_{50} \) -, each instance has different capacities for the various families. The capacities were generated and then adapted to give rise to feasible problems.
The data concerned is displayed in Table 7.1. This table also indicates the order strength value, i.e., the ratio between the total number of direct and transitive precedences and $\frac{N(N-1)}{2}$, the maximum number of direct precedences. This value indicates if the network $G$, built from the precedence constraints, is sparse ($0 \leq \text{order strength} \leq 0.5$) or dense ($0.5 < \text{order strength} \leq 1$).

Table 7.1: Data regarding the test instances

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mertens</th>
<th>Bowman</th>
<th>Jaeschke</th>
<th>Jackson</th>
<th>Mansoor</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N/K$</td>
<td>$I_1$:$I_{26}$</td>
<td>$I_2$:$I_{27}$</td>
<td>$I_3$:$I_{28}$</td>
<td>$I_4$:$I_{29}$</td>
<td>$I_5$:$I_{30}$</td>
</tr>
<tr>
<td>no. of direct precedence relationships</td>
<td>7/2</td>
<td>8/5</td>
<td>9/3</td>
<td>11/4</td>
<td>11/3</td>
</tr>
<tr>
<td>parts' processing time</td>
<td>6</td>
<td>8</td>
<td>11</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>maximum no. of parts per family</td>
<td>1-6</td>
<td>3-17</td>
<td>1-6</td>
<td>1-7</td>
<td>2-45</td>
</tr>
<tr>
<td>maximum time per family</td>
<td>5:4</td>
<td>5-7:2</td>
<td>3-8:3</td>
<td>6-9:3</td>
<td>6-8:4</td>
</tr>
<tr>
<td>dissimilarity between two parts</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
</tr>
<tr>
<td>order strength</td>
<td>0.52</td>
<td>0.75</td>
<td>0.83</td>
<td>0.58</td>
<td>0.60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mitchell</th>
<th>Roszieg</th>
<th>Heskia</th>
<th>Buxey</th>
<th>Sawyer</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N/K$</td>
<td>$I_6$:$I_{31}$</td>
<td>$I_7$:$I_{32}$</td>
<td>$I_8$:$I_{33}$</td>
<td>$I_9$:$I_{34}$</td>
<td>$I_{10}$:$I_{35}$</td>
</tr>
<tr>
<td>no. of direct precedence relationships</td>
<td>21/5</td>
<td>25/6</td>
<td>28/5</td>
<td>29/7</td>
<td>30/7</td>
</tr>
<tr>
<td>parts' processing time</td>
<td>27</td>
<td>32</td>
<td>39</td>
<td>36</td>
<td>32</td>
</tr>
<tr>
<td>maximum no. of parts per family</td>
<td>1-9</td>
<td>1-13</td>
<td>1-108</td>
<td>19-21</td>
<td>1-25</td>
</tr>
<tr>
<td>maximum time per family</td>
<td>4-10:5</td>
<td>5-7:5</td>
<td>6-10:6</td>
<td>5-10:5</td>
<td>6-8:5</td>
</tr>
<tr>
<td>dissimilarity between two parts</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
</tr>
<tr>
<td>order strength</td>
<td>0.70</td>
<td>0.70</td>
<td>0.22</td>
<td>0.51</td>
<td>0.44</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Lutz1</th>
<th>Gunther</th>
<th>Kilbridge</th>
<th>Hahn</th>
<th>Warnecke</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N/K$</td>
<td>$I_{11}$:$I_{36}$</td>
<td>$I_{12}$:$I_{37}$</td>
<td>$I_{13}$:$I_{38}$</td>
<td>$I_{14}$:$I_{39}$</td>
<td>$I_{15}$:$I_{40}$</td>
</tr>
<tr>
<td>no. of direct precedence relationships</td>
<td>32/7</td>
<td>35/8</td>
<td>45/8</td>
<td>53/8</td>
<td>58/10</td>
</tr>
<tr>
<td>parts' processing time</td>
<td>38</td>
<td>45</td>
<td>62</td>
<td>82</td>
<td>70</td>
</tr>
<tr>
<td>maximum no. of parts per family</td>
<td>100-1400</td>
<td>1-30</td>
<td>3-29</td>
<td>40-1775</td>
<td>7-52</td>
</tr>
<tr>
<td>maximum time per family</td>
<td>6-9:5</td>
<td>7-11:5</td>
<td>4-12:7</td>
<td>9-12:7</td>
<td>10-12:8</td>
</tr>
<tr>
<td>dissimilarity between two parts</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
</tr>
<tr>
<td>order strength</td>
<td>0.83</td>
<td>0.59</td>
<td>0.45</td>
<td>0.84</td>
<td>0.59</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Tonge</th>
<th>Wee-Mag</th>
<th>Arc83</th>
<th>Lutz2</th>
<th>Lutz3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N/K$</td>
<td>$I_{16}$:$I_{41}$</td>
<td>$I_{17}$:$I_{42}$</td>
<td>$I_{18}$:$I_{43}$</td>
<td>$I_{19}$:$I_{44}$</td>
<td>$I_{20}$:$I_{45}$</td>
</tr>
<tr>
<td>no. of direct precedence relationships</td>
<td>70/23</td>
<td>75/30</td>
<td>83/21</td>
<td>89/15</td>
<td>89/24</td>
</tr>
<tr>
<td>parts' processing time</td>
<td>86</td>
<td>87</td>
<td>113</td>
<td>118</td>
<td>118</td>
</tr>
<tr>
<td>maximum no. of parts per family</td>
<td>1-152</td>
<td>5-27</td>
<td>233-2881</td>
<td>1-10</td>
<td>1-74</td>
</tr>
<tr>
<td>maximum time per family</td>
<td>5-10:4</td>
<td>5-10:4</td>
<td>6-10:5</td>
<td>4-12:8</td>
<td>5-10:5</td>
</tr>
<tr>
<td>dissimilarity between two parts</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
<td>0.1-1.0</td>
</tr>
<tr>
<td>order strength</td>
<td>0.59</td>
<td>0.23</td>
<td>0.59</td>
<td>0.77</td>
<td>0.77</td>
</tr>
</tbody>
</table>
Besides these instances, four other instances’ sets were tested (Lourenço [12]), three of which were randomly generated for a grouping problem in the information systems. The fourth concerns the assembly line balancing problem (Tonge [22]). The values for the parameters of these instances were taken from Aronson and Klein [1]. However, the authors opted against presenting the results of these tests here as the instances possess parameters that are very similar to some of the instances belonging to the sets A and B outlined above, besides which the computational results were similar.

The computational experiment was performed on a PC with a Pentium 4 processor, with 2.53 Ghz and 512 Mb of RAM memory. To solve the continuous or mixed binary linear models, one used CPLEX software version 8.1 [2], in which all the parameters assumed pre-defined values, with the exception of the time limit parameter used in the search for the optimal solution.

The lower bounds obtained from the linear relaxations $\text{BF}_1$ and $\text{BF}_2$ for all the instances, except $I_{26}$, are null. The optimal value of both linear relaxations for this instance is 0.31, while the minimum total dissimilarity equals 3.9. One may conclude from this experiment that this linear lower bound for the PFP is very weak indeed. Hence, the strengthened formulations were tested with a view to improving the lower bounds.

Table 7.2 displays data, results and execution times referring to each instance given in column (1). Column (2) lists the number of parts and the number of families, whereas column (3) shows the optimal value or, where labeled with *, the best known upper bound (see [9]). Columns (4) to (9) show the results of application of CPLEX software to the ILP or LP problems, including the computational times, in seconds or hours, bounded by 10 hours. Column (4) contains information on the optimal value obtained by applying the CPLEX branch-and-bound to the formulation $\text{BF}_1$. As only small instances lead to an optimum result, for the remaining instances the lower and the upper bounds are given. Column (5) records the results obtained from the strengthened formulation $\text{BF}_1$.
with preprocessing and all valid inequalities developed - Rules 1 to 3 and Results 3 to 7 - denoted by BF1cut, and column (6) indicates the results of application of the CPLEX linear programming optimizer to the respective linear relaxation, BF1cut. Column (7) was built by using BF2 and columns (8) and (9) respectively contain figures obtained from BF2 strengthened by all the valid inequalities developed (Results 3 to 7 adapted to BF2 and Result 8), denoted BF2cut, and the results from the respective linear relaxation, BF2cut. The last rows of the table show two simple statistics: the number of instances exactly solved, i.e., the number of optimal values found and the average gap for each of the columns (4), (5), (7) and (8). The gap of each instance was calculated by \( \frac{\text{upper} - \text{lower}}{\text{upper}} \).

<table>
<thead>
<tr>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
<th>(5)</th>
<th>(6)</th>
<th>(7)</th>
<th>(8)</th>
<th>(9)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>BF1 optimum or best known</td>
<td>BF1cut optimum or [lower,upper]</td>
<td>BF1cut optimum or [lower,upper]</td>
<td>BF2 optimum or [lower,upper]</td>
<td>BF2cut optimum or [lower,upper]</td>
<td>BF2cut optimum or [lower,upper]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(time)</td>
<td>(time)</td>
<td>(time)</td>
<td>(time)</td>
<td>(time)</td>
<td>(time)</td>
</tr>
<tr>
<td>I1</td>
<td>7/2</td>
<td>3.9</td>
<td><strong>3.9</strong></td>
<td>3.9</td>
<td><strong>3.9</strong></td>
<td><strong>3.9</strong></td>
<td><strong>3.9</strong></td>
<td><strong>3.9</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.0 s)</td>
<td>(0.1 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
</tr>
<tr>
<td>I2</td>
<td>8/5</td>
<td>1.7</td>
<td><strong>1.7</strong></td>
<td><strong>1.7</strong></td>
<td><strong>1.7</strong></td>
<td>1.7</td>
<td><strong>1.7</strong></td>
<td>1.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.1 s)</td>
<td>(0.0 s)</td>
<td>(0.1 s)</td>
</tr>
<tr>
<td>I3</td>
<td>9/3</td>
<td>5.5</td>
<td><strong>5.5</strong></td>
<td><strong>5.5</strong></td>
<td><strong>5.5</strong></td>
<td><strong>5.5</strong></td>
<td><strong>5.5</strong></td>
<td>5.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
</tr>
<tr>
<td>I4</td>
<td>11/4</td>
<td>3.3</td>
<td>3.3</td>
<td><strong>3.3</strong></td>
<td><strong>3.3</strong></td>
<td>3.3</td>
<td>3.3</td>
<td>3.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(0.3 s)</td>
<td>(0.0 s)</td>
<td>(0.0 s)</td>
<td>(0.3 s)</td>
<td>(0.1 s)</td>
<td>(0.0 s)</td>
</tr>
<tr>
<td>I5</td>
<td>11/3</td>
<td>5.0</td>
<td>5.0</td>
<td>5.0</td>
<td>4.0</td>
<td><strong>5.0</strong></td>
<td>5.0</td>
<td>4.2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(34.1 s)</td>
<td>(0.5 s)</td>
<td>(0.0 s)</td>
<td>(0.1 s)</td>
<td>(0.3 s)</td>
<td>(0.1 s)</td>
</tr>
<tr>
<td>I6</td>
<td>21/5</td>
<td>17.0</td>
<td>17.0</td>
<td><strong>17.0</strong></td>
<td><strong>17.0</strong></td>
<td>17.0</td>
<td>17.0</td>
<td><strong>16.2</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(34.1 s)</td>
<td>(4.9 s)</td>
<td>(0.3 s)</td>
<td>(17.4 s)</td>
<td>(2.7 s)</td>
<td>(0.9 s)</td>
</tr>
<tr>
<td>I7</td>
<td>25/6</td>
<td>18.4</td>
<td>18.4</td>
<td><strong>18.4</strong></td>
<td><strong>18.4</strong></td>
<td>18.4</td>
<td>18.4</td>
<td><strong>15.1</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(1.6 h)</td>
<td>(51.0 s)</td>
<td>(1.2 s)</td>
<td>(7.4 h)</td>
<td>(71.5 s)</td>
<td>(4.5 s)</td>
</tr>
<tr>
<td>I8</td>
<td>28/5</td>
<td>22.9</td>
<td><strong>22.9</strong></td>
<td>[19.6,24.5]</td>
<td>16.1</td>
<td>[14.9,24.2]</td>
<td>[18.2,25.3]</td>
<td><strong>16.2</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(7.5 h)</td>
<td>(10.0 h)</td>
<td>(3.1 s)</td>
<td>(10.0 h)</td>
<td>(10.0 h)</td>
<td>(52.2 s)</td>
</tr>
<tr>
<td>I9</td>
<td>29/7</td>
<td>18.0</td>
<td>[16.0,18.0]</td>
<td>[14.8,18.6]</td>
<td>11.7</td>
<td>18.0</td>
<td>[15.0,18.2]</td>
<td><strong>11.9</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(10.0 h)</td>
<td>(10.0 h)</td>
<td>(3.0 s)</td>
<td>(7.7 h)</td>
<td>(10.0 h)</td>
<td>(60.0 s)</td>
</tr>
<tr>
<td>I10</td>
<td>30/7</td>
<td>19.0*</td>
<td>[11.4,20.5]</td>
<td>[16.1,22.0]</td>
<td>11.5</td>
<td>[11.7,19.3]</td>
<td>[18.1,19.0]</td>
<td><strong>12.4</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(10.0 h)</td>
<td>(10.0 h)</td>
<td>(1.7 s)</td>
<td>(1.0 h)</td>
<td>(10.0 h)</td>
<td>(56.6 s)</td>
</tr>
<tr>
<td>I11</td>
<td>32/7</td>
<td>27.5</td>
<td>[21.3,28.5]</td>
<td>[27.5,-]</td>
<td>23.2</td>
<td>[13.4,28.3]</td>
<td><strong>27.5</strong></td>
<td><strong>24.3</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(10.0 h)</td>
<td>(10.0 h)</td>
<td>(3.7 s)</td>
<td>(1.0 h)</td>
<td>(63.4 s)</td>
<td>(436.0 s)</td>
</tr>
<tr>
<td>I12</td>
<td>35/8</td>
<td>24.3*</td>
<td>[8.7,24.3]</td>
<td>[16.6,27.0]</td>
<td><strong>14.2</strong></td>
<td>[9.8,25.2]</td>
<td>[14.5,30.0]</td>
<td><strong>14.2</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(10.0 h)</td>
<td>(10.0 h)</td>
<td>(18.5 s)</td>
<td>(1.0 h)</td>
<td>(10.0 h)</td>
<td>(453.0 s)</td>
</tr>
<tr>
<td>I13</td>
<td>45/8</td>
<td>48.3*</td>
<td>[3.8,48.3]</td>
<td>[26.3,-]</td>
<td>23.9</td>
<td>[7.2,49.7]</td>
<td>[27.0,49.5]</td>
<td><strong>24.7</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(10.0 h)</td>
<td>(10.0 h)</td>
<td>(10.7 s)</td>
<td>(1.0 h)</td>
<td>(1.0 h)</td>
<td>(0.8 h)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( I_{14} )</td>
<td>53/8</td>
<td>77.1(^*)</td>
<td>[4.8, -]</td>
<td>[65.8, 78.6]</td>
<td>60.0</td>
<td>[2.6, -]</td>
<td>[65.3, 78.2]</td>
<td>62.7</td>
</tr>
<tr>
<td>( I_{15} )</td>
<td>53/10</td>
<td>61.3(^*)</td>
<td>[-, -]</td>
<td>[37.8, -]</td>
<td>35.9</td>
<td>[0.8, -]</td>
<td>[38.3, 64.6]</td>
<td>36.7</td>
</tr>
<tr>
<td>( I_{26} )</td>
<td>7/2</td>
<td>3.9</td>
<td>3.9</td>
<td>3.9</td>
<td>3.9</td>
<td>3.9</td>
<td>3.9</td>
<td></td>
</tr>
<tr>
<td>( I_{27} )</td>
<td>8/5</td>
<td>1.3</td>
<td>1.3</td>
<td>0.7</td>
<td>1.3</td>
<td>1.3</td>
<td>0.7</td>
<td></td>
</tr>
<tr>
<td>( I_{28} )</td>
<td>9/3</td>
<td>5.5</td>
<td>5.5</td>
<td>5.5</td>
<td>5.5</td>
<td>5.5</td>
<td>5.5</td>
<td></td>
</tr>
<tr>
<td>( I_{29} )</td>
<td>11/4</td>
<td>3.3</td>
<td>3.3</td>
<td>3.3</td>
<td>3.3</td>
<td>3.3</td>
<td>3.3</td>
<td></td>
</tr>
<tr>
<td>( I_{30} )</td>
<td>11/3</td>
<td>4.4</td>
<td>4.4</td>
<td>4.4</td>
<td>4.4</td>
<td>4.4</td>
<td>4.4</td>
<td></td>
</tr>
<tr>
<td>( I_{31} )</td>
<td>21/5</td>
<td>15.4</td>
<td>15.4</td>
<td>15.4</td>
<td>15.4</td>
<td>15.4</td>
<td>15.4</td>
<td></td>
</tr>
<tr>
<td>( I_{32} )</td>
<td>25/6</td>
<td>17.4</td>
<td>17.4</td>
<td>17.4</td>
<td>17.4</td>
<td>17.4</td>
<td>17.4</td>
<td></td>
</tr>
<tr>
<td>( I_{33} )</td>
<td>28/5</td>
<td>22.2*</td>
<td>[13.6, 23.3]</td>
<td>[20.5, 22.2]</td>
<td>15.9</td>
<td>[16.9, 22.9]</td>
<td>[19.5, 22.2]</td>
<td>15.9</td>
</tr>
<tr>
<td>( I_{34} )</td>
<td>29/7</td>
<td>17.5*</td>
<td>[15.9, 17.8]</td>
<td>[13.8, 18.5]</td>
<td>11.3</td>
<td>[12.8, 17.8]</td>
<td>[14.0, 18.6]</td>
<td>11.4</td>
</tr>
<tr>
<td>( I_{35} )</td>
<td>30/7</td>
<td>18.4*</td>
<td>[14.8, 18.4]</td>
<td>[13.2, 19.4]</td>
<td>10.3</td>
<td>[14.5, 17.0]</td>
<td>[13.5, 18.5]</td>
<td>10.4</td>
</tr>
<tr>
<td>( I_{36} )</td>
<td>32/7</td>
<td>23.7</td>
<td>[17.1, 24.1]</td>
<td>[20.5, 22.2]</td>
<td>23.7</td>
<td>[22.1, 24.5]</td>
<td>[19.5, 22.2]</td>
<td>20.0</td>
</tr>
<tr>
<td>( I_{37} )</td>
<td>35/8</td>
<td>25.0*</td>
<td>[11.1, 25.0]</td>
<td>[14.9, 30.1]</td>
<td>13.7</td>
<td>[11.8, 23.8]</td>
<td>[13.8, 27.8]</td>
<td>13.7</td>
</tr>
<tr>
<td>( I_{38} )</td>
<td>45/8</td>
<td>40.8*</td>
<td>[6.1, 46.1]</td>
<td>[24.2, 66.0]</td>
<td>22.6</td>
<td>[7.2, 43.9]</td>
<td>[22.7, 58.2]</td>
<td>22.7</td>
</tr>
<tr>
<td>( I_{39} )</td>
<td>53/8</td>
<td>72.9*</td>
<td>[4.9, -]</td>
<td>[57.0, 118.1]</td>
<td>56.5</td>
<td>[8.1, 84.4]</td>
<td>[57.8, 87.9]</td>
<td>56.8</td>
</tr>
<tr>
<td>( I_{40} )</td>
<td>53/10</td>
<td>62.2</td>
<td>- -</td>
<td>[33.3, 80.5]</td>
<td>31.9</td>
<td>[18.4, 71.9]</td>
<td>[35.2, 85.2]</td>
<td>33.6</td>
</tr>
<tr>
<td><strong>number of optimums</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>average gap</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note that, the instances \( I_{16} \) to \( I_{25} \) and \( I_{41} \) to \( I_{50} \) do not appear in the Table 7.2 because the CPLEX software gave the message 'out of memory', resulting in no value. The best result between column (6) and (9) per instance is shown in bold symbol. Bold symbols are also used to identify the optimal values among the results of columns (4), (5), (7) and (8).
Firstly, one began by testing the formulations without and with strengthening, using the branch-and-bound algorithm of the CPLEX package and verified that the strengthened formulations (BF1cut and BF2cut in columns (5) and (8)), for the medium size instances, generated bounds for the optimum of PFP that are significantly better than those given by the corresponding basic formulations BF1 and BF2 (columns (4) and (7)). However, in the case of the larger instances ($I_{34}$ to $I_{40}$), the strengthened formulation BF2, due to the excess of variables and constraints included for strengthening, required an additional computational effort and was unable to obtain better upper bounds than the respective non-strengthened formulation. Even so, the average gap from BF2cut is the best among the gaps given by all the formulations.

Moreover, in Table 7.2 one may see that the optima given by the linear relaxations of the strengthened formulations (columns (6) and (9)) significantly improve the initial null values which do not appear in the table. Additionally, one finds that BF2cut produced lower bounds that are appreciably better than those obtained from BF1cut, and this difference between the two lower bounds is slightly more marked and frequent in the case of instances with equal capacity for the families (for example $I_{6}$). Note that, in BF2cut, the strengthening includes inequalities which have no correspondent in BF1cut, hence forcing a best lower bound from BF2cut.

As to the effect of the order strength of the instances on the computational results, not shown in this table, one found that the greater the order strength the greater the proportion of variables with a fixed value through the preprocessing phase, i.e., application of the Rules 1, 2, and 3.

Finally, through the computational experiments it was found that, of the various valid inequalities studied, the one that had greatest importance in the improvement of the lower bounds is the valid inequality (5.1.1). This inequality is also beneficial, in relation to the remaining ones, in that it can be adapted to any problem of grouping elements, with or without constraints.

8 Conclusions

Following an analysis of the PFP and its properties one performed solution preprocessing, thus resulting the fixing of value for a large number of variables. In the experimentation undertaken, this preprocessing benefits application of the CPLEX software to both mixed binary linear formulations, BF1 and BF2, especially, in the case of the higher dimension instances.
Several valid inequalities were also developed to strengthen linear relaxation of the formulations. For the smaller instances one may conclude that the strengthened formulations enable the ILP algorithm of CPLEX to perform better in determining the optimum, as they require less time. For the medium-sized instances these formulations found upper and lower bounds for the optimum in a relatively short time and here the strengthened BF2 formulation obtains better lower bounds than the BF1. Finally, for the larger instances, the strengthened formulations with all valid inequalities seem by now to be inappropriate in view of its great consumption of computational resources.

The experiments undertaken indicate that, by using more computational resources and resorting to a more discerning study of the additional constraints to include, the disaggregated formulation BF2 less duly strengthened with valid inequalities will constitute the basis of a solution methodology for this important, though difficult problem, which may be used within the flexible manufacturing systems.

Acknowledgments

The authors are grateful to the two anonymous referees for their valuable suggestions and comments.

Research partially supported by POCTI/ISFL 152-1.

References


