A Computational Study of the Pseudo-Boolean Approach to the p-Median Problem Applied to Cell Formation

Boris Goldengorin[1] and Dmitry Krushinsky[2]

[1] Department of Applied Mathematics and Informatics,
Nizhny Novgorod branch of Higher School of Economics, Russia
bgoldengorin@hse.ru
[2] Department of Operations, University of Groningen, The Netherlands
d.krushinsky@rug.nl

Abstract. In this study we show by means of computational experiments that a pseudo-Boolean approach leads to a very compact presentation of p-Median problem instances which might be solved to optimality by a general purpose solver like CPLEX, Xpress, etc. Together with p-Median benchmark instances from OR and some other libraries we are able to solve to optimality many benchmark instances from cell formation in group technology which were tackled in the past only by means of different types of heuristics. Finally, we show that this approach is flexible to take into account many other practically motivated constraints in cell formation.

1 Introduction

The p-Median problem (PMP) is a well-known NP-hard problem which was originally defined by Hakimi [19] and involves the location of p facilities on a network in such a manner that the total weighted distance of serving all demands is minimized. Being a classical problem in combinatorial optimization, the PMP has been widely studied in literature and applied in cluster analysis, quantitative psychology, marketing, telecommunications industry [10], sales force territories design [23], political districting [5], optimal diversity management [9], cell formation in group technology [33], vehicle routing [21], and topological design of computer communication networks [25].

The basic PMP model that has remained almost unchanged during recent 30 years is the so called ReVelle and Swain [27] integer LP formulation (in fact, a Boolean LP formulation). The PMP has since been the subject of considerable research involving the development of adjusted model formats (Rosing *et al.* [29], Cornuejols *et al.* [13], Church [11,12]), and recently by AlBdaiwi *et al.* [2], and Elloumi [15], as well as the development of advanced solution approaches, e.g. Beltran *et al.* [7], Avella *et al.* [4]. For a comprehensive review of the PMP we address the reader to Reese [26], Mladenovich *et al.* [22] and ReVelle *et al.* [28].

In this paper we consider an application of PMP to the industrial engineering problem of cell formation (CF) in group technology. Cell formation suggests decomposition of a manufacturing system into several subsystems such that these subsystems, *manufacturing cells*, are as independent as possible. This ensures that machines processing

J. Pahl, T. Reiners, and S. Voß (Eds.): INOC 2011, LNCS 6701, pp. 503–516, 2011.
© Springer-Verlag Berlin Heidelberg 2011

the same parts are placed closer to each other and time spent by parts on travelling from one machine to another is substantially reduced. Moreover, smaller systems (cells) are easier to manage (e.g. scheduling). PMP was applied to cell formation by a number of authors, e.g. [33]. However, due to NP-hardness of the PMP all of them used heuristics to solve the resulting PMP instances.

The purpose of this paper is two-fold. First, we show by means of numerical experiments that a pseudo-Boolean approach allows a very compact representation of the PMP instance data and can be used to derive an efficient Mixed-Integer Linear Programming (MILP) formulation. Second, we show that the PMP can be used as a flexible framework for cell formation, as in CF applications PMP can be very efficiently solved to optimality and any real-world constraints can be included. Our experiments show that even the largest CF instances used in literature can be solved within a second and the quality of solutions outperforms even the most contemporary heuristics.

In the next section we describe an efficient MILP formulation of the PMP based on a pseudo-Boolean approach and in Section 3 we conclude that all known reductions are contained in our model. Section 4 reports our computational study with benchmark instances. In Section 5 we discuss the possibilities induced by our model in CF applications and Section 6 summarizes this paper and provides future research directions.

2 The Mixed Boolean Pseudo-Boolean Model (MBpBM)

Recall that given sets $I = \{1,2,\ldots,m\}$ of sites in which plants can be located, $J = \{1,2,\ldots,n\}$ of clients, a non-negative matrix $C = [c_{ij}]$ of costs of serving each $j \in J$ from each $i \in I$, the number p of plants to be opened, and assuming a unit demand at each client site, the p-Median Problem (PMP) is one of finding a set $S \subseteq I$ with $|S| = p$, such that the total serving cost is minimized:

$$f_C(S) = \sum_{j\in J} \min\{c_{i,j} | i \in S\} \tag{1}$$

The *Combinatorial Formulation of PMP* is to find a subset $S^\star$ such that

$$S^\star \in \arg\min\{f_C(S) : \emptyset \subset S \subseteq I, \quad |S| = p\} \ . \tag{2}$$

The objective function $f_C(S)$ of the PMP (1) can be reformulated in terms of a pseudo-Boolean polynomial, pBp, (see Hammer [20] and Beresnev [6]) in the following way. Consider a vector $\mathbf{y} = (y_1,\ldots,y_m)$ of Boolean variables such that $y_i = 0$ iff i-th location is opened (i.e. iff $i \in S$). For some client j a corresponding column of C contains the costs of serving this client from any location. Clearly, the demand of client j cannot be satisfied cheaper than $c_{\pi_{1j},j}$, where π_{1j} is the index of the smallest entry in j-th column of the costs matrix. This minimum value is attained only if location π_{1j} is opened and $y_{\pi_{1j}} = 0$. Otherwise, the cheapest way of satisfying client j is to use the second smallest entry in j-th column. In this case the cost is $c_{\pi_{2j},j} = c_{\pi_{1j},j} + y_{\pi_{1j}}(c_{\pi_{2j},j} - c_{\pi_{1j},j})$. If the location corresponding to the second smallest entry is also closed, the minimum costs of serving client j is $c_{\pi_{3j},j} = c_{\pi_{1j},j} + y_{\pi_{1j}}(c_{\pi_{2j},j} - c_{\pi_{1j},j}) + y_{\pi_{1j}}y_{\pi_{2j}}(c_{\pi_{3j},j} - c_{\pi_{2j},j})$. This intuition can be extended further and the costs of serving client j can be expressed as:

$$c_{\pi_{1j},j} + \sum_{k=1}^{m-1} (c_{\pi_{k+1,j},j} - c_{\pi_{kj},j}) \prod_{r=1}^{k} y_{\pi_{rj}} \tag{3}$$

This representation naturally induces two objects related to j-th column: a permutation $\Pi^j = (\pi_{1j}, \ldots, \pi_{mj})$ that sorts the entries from the corresponding column of the costs matrix in a nondecreasing order, and the vector of differences $\Delta^j = (\delta_{0j}, \ldots, \delta_{m-1,j})$ defined as follows:

$$\begin{aligned} \delta_{0j} &= c_{\pi_{1j},j} \ , \\ \delta_{rj} &= c_{\pi_{r+1,j},j} - c_{\pi_{rj},j} \text{ for } 1 \le r \le m-1 \ , \end{aligned} \tag{4}$$

By extending the above reasoning to all clients and defining a permutation matrix $\Pi = (\Pi^1, \ldots, \Pi^n)$ and a differences matrix $\Delta = (\Delta^1, \ldots, \Delta^n)$ the total cost function (1) can be represented by the following polynomial:

$$\mathscr{B}_{C,\Pi}(\mathbf{y}) = \sum_{j=1}^{n} \left\{ \delta_{0j} + \sum_{k=1}^{m-1} \delta_{kj} \prod_{r=1}^{k} y_{\pi_{rj}} \right\} \ . \tag{5}$$

The expressions $\alpha_S \prod_{i \in S} y_i$ and $\prod_{i \in S} y_i$ are called a *monomial* and a *term*, respectively. In this paper monomials with the same term are called *similar monomials*. We say that a pseudo-Boolean polynomial is in *the reduced form* if for any two of its monomials the corresponding terms differ. In other words, the algebraic summation of similar monomials is called *reduction*.

AlBdaiwi et al. [2] show that the total cost function (5) is identical for all possible permutation matrices Π, hence we can remove it from notations without any confusion.

The Hammer-Beresnev polynomial $\mathscr{B}_C(\mathbf{y})$ contains less than $m \cdot n$ monomials and their number can be further reduced by using that for any feasible solution $\mathbf{y}$ to a PMP instance holds $\sum_{i=1}^{m} y_i = m - p$. This implies that any monomial in the pBp expressed as a constant multiplied by more than $m - p$ variables necessarily evaluates to zero. This is formalized in Theorem 1 (AlBdaiwi *et al.* [2]).

Theorem 1. *For any PMP instance C with $p \le m$ the following assertions hold:*

1. *The degree of a truncated Hammer-Beresnev polynomial $\mathscr{B}_{C,p}(\mathbf{y})$ is at most $m - p$;*
2. *Each column of the costs matrix C can be p-truncated by setting all p largest entries in a column to the value of the smallest entry among these p largest.*

The above theorem allows to substitute $\mathscr{B}_C(\mathbf{y})$ (5) by $\mathscr{B}_{C,p}(\mathbf{y})$ defined as

$$\mathscr{B}_{C,p}(\mathbf{y}) = \sum_{j=1}^{n} \left\{ \delta_{0j} + \sum_{k=1}^{m-p} \delta_{kj} \prod_{r=1}^{k} y_{\pi_{rj}} \right\} \ . \tag{6}$$

We can reformulate (2) in terms of Hammer-Beresnev polynomials as the *pseudo-Boolean formulation of PMP*:

$$\mathbf{y}^\star \in \arg\min\{\mathscr{B}_{C,p}(\mathbf{y}) : \mathbf{y} \in \{0,1\}^m, \sum_{i=1}^{m} y_i = m - p\} \ . \tag{7}$$

Let us denote by $|B|$ the number of monomials in $\mathscr{B}_{C,p}(\mathbf{y})$, by $T_r \in \{1,\dots,m\}$ a set of indices of variables in the r-th monomial of the pBp and by α_r coefficients of the monomials (e.g. $\alpha_0 = \sum_{j=1}^{n} \delta_{0j}$). Now the truncated reduced Hammer-Beresnev polynomial can be expressed as

$$\mathscr{B}_{C,p}(\mathbf{y}) = \alpha_0 + \sum_{r=1}^{m} \alpha_r y_r + \sum_{r=m+1}^{|B|} \alpha_r \prod_{i \in T_r} y_i \tag{8}$$

and by introducing nonnegative variables z_r $(r = m+1,\dots,|B|)$ we have linearised it (see e.g., Wolsey[32]) in order to obtain a linear objective function

$$f(\mathbf{y},\mathbf{z}) = \alpha_0 + \sum_{r=1}^{m} \alpha_r y_r + \sum_{r=m+1}^{|B|} \alpha_r z_r \ . \tag{9}$$

By introducing for each variable $z_r = \prod_{i \in T_r} y_i$ the constraints

$$z_r \geq \sum_{i \in T_r} y_i - |T_r| + 1 \quad , \quad z_r \geq 0 \tag{10}$$

we obtained our Mixed Boolean pseudo-Boolean Model (MBpBM):

$$\alpha_0 + \sum_{r=1}^{m} \alpha_r y_r + \sum_{r=m+1}^{|B|} \alpha_r z_r \longrightarrow \min \tag{11}$$

$$\text{s.t.} \sum_{i=1}^{m} y_i = m - p \ , \tag{12}$$

$$\sum_{i \in T_r} y_i - |T_r| + 1 \leq z_r,\ r = m+1,\dots,|B| \ , \tag{13}$$

$$z_i \geq 0, \quad i = m+1,\dots,|B| \ , \tag{14}$$

$$y_i \in \{0,1\},\ i = 1,\dots,m \tag{15}$$

Example 1. Consider a PMP instance from [15] with $m = 4$, $n = 5$, $p = 2$ and

$$C = \begin{bmatrix} 1 & 6 & 5 & 3 & 4 \\ 2 & 1 & 2 & 3 & 5 \\ 1 & 2 & 3 & 3 & 3 \\ 4 & 3 & 1 & 8 & 2 \end{bmatrix} \ . \tag{16}$$

A possible ordering and differences matrices for this problem are given by

$$\Pi = \begin{bmatrix} 1 & 2 & 4 & 1 & 4 \\ 3 & 3 & 2 & 2 & 3 \\ 2 & 4 & 3 & 3 & 1 \\ 4 & 1 & 1 & 4 & 2 \end{bmatrix} \text{ and } \Delta = \begin{bmatrix} 1 & 1 & 1 & 3 & 2 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 2 & 3 & 2 & 5 & 1 \end{bmatrix} \ . \tag{17}$$

The Hammer-Beresnev polynomial representing the total cost function for this instance in the form (5) is

$$\begin{aligned} \mathscr{B}_C(\mathbf{y}) = {} & [1 + 0y_1 + 1y_1y_3 + 2y_1y_2y_3] + \\ & [1 + 1y_2 + 1y_2y_3 + 3y_2y_3y_4] + \\ & [1 + 1y_4 + 1y_2y_4 + 2y_2y_3y_4] + \\ & [3 + 0y_1 + 0y_1y_2 + 5y_1y_2y_3] + \\ & [2 + 1y_4 + 1y_3y_4 + 1y_1y_3y_4] \ . \end{aligned} \tag{18}$$

After reduction of similar monomials and truncation we obtain the following pseudo-Boolean representation of the instance:

$$\begin{aligned} \mathscr{B}_C(\mathbf{y}) = \;& 8 + 1y_2 + 2y_4 + 1y_1y_3 + 1y_2y_3 + 1y_2y_4 + 1y_3y_4 \longrightarrow \min \\ \text{s.t. }\; & y_1 + y_2 + y_3 + y_4 = m - p = 2, \quad \mathbf{y} \in \{0,1\}^m \;. \end{aligned} \tag{19}$$

After introduction of new variables $z_5 = y_1y_3$, $z_6 = y_2y_3$, $z_7 = y_2y_4$, $z_8 = y_3y_4$ the MBpBM is:

$$8 + y_2 + 2y_4 + z_5 + z_6 + z_7 + z_8 \longrightarrow \min \tag{20}$$

$$\text{s.t. } \sum_{i=1}^{4} y_i = m - p = 2 \tag{21}$$

$$z_5 + 1 \geq y_1 + y_3 \tag{22}$$

$$z_6 + 1 \geq y_2 + y_3 \tag{23}$$

$$z_7 + 1 \geq y_2 + y_4 \tag{24}$$

$$z_8 + 1 \geq y_3 + y_4 \tag{25}$$

$$z_i \geq 0,\; i = 5,\ldots,8 \tag{26}$$

$$y_i \in \{0,1\},\; i = 1,\ldots,4 \;. \tag{27}$$

This MBpBM (20)–(27) has the same decision variables y_i as the pseudo-Boolean formulation (7), but its objective function is a linear in y_i and z_i. Note that our model contains 7 coefficients in the objective function, 5 constraints, 4 Boolean and 4 non-negative variables, while for the best Elloumi's model NFexr [15] these numbers are 10, 11, 4 and 7, correspondingly.

In the following Lemma 1 we explain how to reduce the number of Boolean variables y_i involved in the restrictions (13).

Lemma 1. *Let $\emptyset \neq T_r \subset T_q$ be a pair of embedded sets of Boolean variables y_i. Thus, two following systems of inequalities are equivalent:*

$$\begin{array}{ccc} \sum\limits_{i \in T_r} y_i - |T_r| + 1 \leq z_r & & \sum\limits_{i \in T_r} y_i - |T_r| + 1 \leq z_r \\ \sum\limits_{i \in T_q} y_i - |T_q| + 1 \leq z_q & \textit{and} & z_r + \sum\limits_{i \in T_q \setminus T_r} y_i - |T_q \setminus T_r| \leq z_q \\ z_r \geq 0, \quad z_q \geq 0 & & z_r \geq 0, \quad z_q \geq 0 \end{array} \tag{28}$$

Based on a compact representation of a PMP instance within pseudo-Boolean formulation (7) one may conclude that this formulation has extracted only essential information to represent the PMP. It means that we are in a position to check whether our MBpBM is an optimal model within the class of Mixed Boolean LP models. If we were able to show that the matrix of all linear constraints induced by non-linear monomials contains the smallest number of non-zero entries, then taking into account that the objective function has the smallest number of non-zero coefficients and linear constraints induced by non-linear terms one may conclude that our MBpBM is the optimal one [17]. Unfortunately, in general this is not the case. It is not difficult to show that the problem of finding a constraint matrix with the smallest number of non-zero entries is at least as hard as the *classic set covering problem* (see e.g. Garey and Johnson [16]). This means that to find an optimal model within the class of Mixed Boolean LP models is an NP-hard problem, even if the number of linear constraints is a linear function on the number of all decision variables. However, if only numbers of variables, constraints and terms in the objective function are taken into account, MBpBM is an optimal one [17].

Despite the efficiency of MBpBM it can be further reduced based on a decomposition of the whole search space into subspaces induced by terms in a pBp. By using upper and lower bounds on the cost of optimal solutions one may prove that some subspaces do not contain optimal solutions (see Goldengorin *et al.* [18]). Suppose, we know from some heuristic a (global) upper bound f^{UB} on the cost of optimal solutions. Let us now consider some term $\mathscr{T}$ and the set of indices of its variables T. One can also compute a lower bound $f^{LB}_{\mathscr{T}}$ over a subspace of solutions for which $\mathscr{T}$ is nonzero, i.e. all locations from T are closed. In case $f^{LB}_{\mathscr{T}} > f^{UB}$ one can conclude that for every optimal solution $\mathscr{T}$ evaluates to 0 and a constraint for the corresponding z-variable can be modified respectively. Moreover, all terms containing $\mathscr{T}$ also evaluate to 0 and the corresponding z-variables and constraints can be dropped. We call the model with this reduction MBpBMb and the lower bound that we used is given by:

$$f^{LB}_{\mathscr{T}} = f_C(\overline{T}) + \min_{k_i \in \overline{T}} \sum_{i=1}^{|\overline{T}|-p} [f_C(\overline{T} \setminus \{k_i\}) - f_C(\overline{T})], \tag{29}$$

where $f_C(.)$ – cost function of the PMP and $\overline{T}$ denotes the complement of T, i.e. $\overline{T} = \{1,\ldots,m\} \setminus T$.

3 Reduction Tests for the p-Median Problem

Looking at PMP models described in the literature it can be noticed that all improvements over the classical formulation by ReVelle and Swain [27] are done by application of various reduction tests to the instances of the problem. These tests can be classified into the following two broad groups: pure and optimizational.

The first group includes all kinds of tests exploiting structural redundancies in the input data. For example, presence of equal entries in a column of the costs matrix within a MILP model was first used by Cornuejols [13] and is present in most of the recent models, including Elloumi's ([15]), Church's [11,12] and our MBpBM (11)-(15). Another pure reduction excludes from the formulation largest $p-1$ entries of each column of the costs matrix. It was used by Avella *et al.* ([3], reduction test $FIX1$) and Church [11], and in our model is done by truncation of the pBp. This reduction has a universal nature in a sense that it allows truncation of exactly $p-1$ entries from each column of the cost matrix, irrespectively of the particular instance data. The next structural peculiarity that can be exploited for strengthening the formulation stems from the order in which potential locations are listed if being sorted by increasing distance from a client. If for two clients these orders are equal, they may be considered as one aggregated client. In our model this rule is applied by reducing similar monomials, while in Elloumi's NF (reduction rules R2 and R3 in [15]) and Church's COBRA [11] this is done by "substitution of equivalent variables" (as it is called in [11]). Finally, we would like to

Table 1. Effect of reduction tests for selected benchmark instances

instance	m	p	$\|C\|$	$\|B\|$	**red**.(%)	instance	m	p	$\|C\|$	$\|B\|$	**red**.(%)
pmed26	600	5	360000	25440	**92.93**	rw100	100	10	10000	5683	**43.17**
pmed27	600	10	360000	24117	**93.30**		100	20	10000	5045	**49.55**
pmed28	600	60	360000	19142	**94.68**		100	30	10000	4404	**55.96**
pmed29	600	120	360000	17724	**95.08**		100	40	10000	3771	**62.29**
pmed30	600	200	360000	16920	**95.30**		100	50	10000	3138	**68.62**
pmed31	700	5	490000	25940	**94.71**	rw500	500	10	250000	154622	**38.15**
pmed32	700	10	490000	26384	**94.62**		500	50	250000	141991	**43.20**
pmed33	700	70	490000	21030	**95.71**		500	100	250000	126281	**49.49**
pmed34	700	140	490000	18684	**96.19**		500	150	250000	110487	**55.81**
pmed35	800	5	640000	27788	**95.66**		500	250	250000	78946	**68.42**
pmed36	800	10	640000	28579	**95.53**	rw1000	1000	10	1000000	625052	**37.49**
pmed37	800	80	640000	23324	**96.36**		1000	50	1000000	599698	**40.03**
pmed38	900	5	810000	29230	**96.39**		1000	100	1000000	568197	**43.18**
pmed39	900	10	810000	27638	**96.59**		1000	300	1000000	441946	**55.81**
pmed40	900	90	810000	24165	**97.02**		1000	500	1000000	315682	**68.43**

mention Elloumi's reduction rule R1. The essence of this rule is that some z-variables can be expressed in terms of y-variables in a linear way. Our model implies R1 as we do not introduce any new variables for linear monomials. Thus, our model incorporates all known pure reductions by excluding monomials with zero coefficients, truncation and reduction of similar monomials. The effect of pure reductions can be illustrated by Table 1 where reduction tests were applied to several selected benchmark instances that are widely used for testing PMP-related algorithms. The first three columns contain the title of an instance, numbers of potential locations m and medians p, correspondingly. The next two columns indicate the number of entries in a costs matrix $|C|$ and the number of non-zero coefficients in the pBp $|B|$. The last column displays the achieved reduction (based on truncation and reducing similar monomials) that we computed as $red. = (|C| - |B|)/|C| \times 100\%$.

The second group of reduction tests includes optimizational approaches that suggest (pre-)solving the problem. These are reductions based on comparison of upper and (restricted) lower bounds on the optimal solution. Such a reduction was used by Avella *et al.* ([3], reduction test $FIX2$) and is implemented in our formulation MBpBMb leading to even more compact model and, as will be shown in the next section, reduced computing times.

The presented analysis can be summarized as follows: MBpBM in a natural way incorporates all available in literature pure reductions and can be subjected to optimizational problem size reduction techniques.

4 Computational Results for Benchmark Instances

In order to show the applicability of our compact MBpBM formulation, a number of computational experiments were held. We used benchmark instances from two of the

Table 2. Comparison of computing times for our and Elloumi's NF formulations, and Avella et al.'s B&C&P algorithm (15 largest OR-library instances)

instance	m	p	MBpBM	MBpBMb	Elloumi	Avella et al.
pmed26	600	5	163.84	**111.81**	180.31	187
pmed27	600	10	27.59	**21.31**	43.73	47
pmed28	600	60	2.48	**2.13**	3.61	
pmed29	600	120	1.78	**1.31**	2.91	
pmed30	600	200	1.50	**0.78**	4.81	
pmed31	700	5	153.22	**57.05**	90.95	106
pmed32	700	10	**33.13**	43.39	37.64	65
pmed33	700	70	3.09	**2.69**	4.73	
pmed34	700	140	3.72	**1.97**	7.11	
pmed35	800	5	**70.30**	154.41	514.72	189
pmed36	800	10	2256.83	4252.13	6737.25	**453**
pmed37	800	80	3.91	**3.08**	7.00	
pmed38	900	5	1328.34	2041.28	**307.00**	320
pmed39	900	10	572.81	444.08	473.95	**271**
pmed40	900	90	5.39	**4.02**	8.42	

Table 3. Comparison of computing times for our and Elloumi's NF formulations (Resende and Werneck random instances)

instance	m	p	MBpBM	MBpBMb	Elloumi
rw100	100	10	678.91	**452.52**	845.30
	100	20	4.00	**2.22**	5.25
	100	30	0.09	**0.03**	0.13
	100	40	0.08	**0.02**	0.14
	100	50	0.06	**0.02**	0.13
rw500	500	10	–	–	–
	500	100	–	–	–
	500	150	2.97	**1.22**	12.27
	500	200	2.25	**0.28**	4.11
	500	250	1.77	**0.13**	4.36
rw1000	1000	10	–	–	–
	1000	200	–	–	–
	1000	300	118.91	**13.40**	234.99
	1000	400	11.49	**1.16**	21.81
	1000	500	9.08	**0.77**	28.47

– Not solved within 1 hour.

most widely used libraries: J. Beasley's OR-library [24] and randomly generated RW instances by Resende and Werneck (see e.g. Elloumi [15]).

The experiments were conducted on a PC with Intel 2.33 GHz 1.95 GB and Xpress-MP as an MILP solver. Tables 2 and 3 summarize the computational results obtained for the 15 largest OR instances and RW instances, correspondingly. The first three columns contain the name of an instance, the number $m = |I| = |J|$ of nodes and the number p of medians. The next three columns are related to the running times (in seconds) of our models: MBpBM and its modification with reduction based on bounds. The next column reflects computing times for Elloumi's NF that we implemented and tested within the mentioned environment to ensure consistent comparison. The last column displays times reported by Avella et al. [4] for their Branch-and-Cut-and-Price (B&C&P) algorithm (Intel 1.8 GHz 1 GB).

Computational experiments with OR and RW instances can be summarized as follows. Our basic MBpBM formulation outperforms Elloumi's NF and Avella et al.'s B&C&P in most of the tested cases. The reduction based on bounds MBpBMb outperformed other considered models for all but five ORlib instances (in two of these cases our MBpBM was faster).

We also mention an instance from TSP library [30] fl1400 with $p = 400$ unsolved in Avella *et al.* [4] and solved to optimality by MBpBM in 598.5 sec. Note that Beltran's *et al.* [7] advanced semi-Lagrangean approach based on Proximal-Analytic Center Cutting Plane Method has not solved fl1400 with $p = 400$ to optimality and returns an approximation within 0.11% in 678 sec.

5 Application to Cell Formation

The p-Median Problem (PMP) was applied to cell formation in group technology by a number of researchers (see [33], [14] and references within). However, to the best of our knowledge, in all CF related papers PMP (as well as any other model based on graph partitioning or MILP) is solved by some heuristic method. At the same time, for the p-Median problem there exist efficient formulations (like MBpBM or the one in [15]) that allow solving large instances to optimality.

Recall that for a directed weighted graph $G = (V,A,C)$ with $|V|$ vertices, set of arcs $(i, j) \in A \subseteq V \times V$ and weights (distances, dissimilarities, etc.) $C = \{c_{ij} : (i, j) \in A\}$, the PMP consists of determining p nodes (the *median nodes*, $1 \leq p \leq |V|$) such that the sum of weights of arcs joining any other node and one of these p nodes is minimized. In terms of cell formation, vertices represent machines and weights c_{ij} represent dissimilarities between machines i and j. These dissimilarities can be derived from the sets of parts that are being processed by either of the machines (e.g. if two machines process almost the same set of parts they have small dissimilarity and are likely to be in the same cell) or from any other desired characteristics (e.g. workers skill matrix, operational sequences, etc.). Moreover, usually there is no need to invent a dissimilarity measure as it can be derived from one of the available similarity measures using an expression $d(i, j) = c - s(i, j)$, where $d(.,.)/s(.,.)$ is a dis/similarity measure and c – some constant large enough to keep all dissimilarities non-negative. As can be seen from the literature, several similarity measures were proposed and the particular choice can influence results of cell formation. For our experiments we have chosen one of the most widely used – Wei and Kern's "commonality score" [31], and derived our dissimilarity measure as

$$d(i, j) = r \cdot (r-1) - \sum_{k=1}^{r} \Gamma(a_{ik}, a_{jk}) \tag{30}$$

where

$$\Gamma(a_{ik}, a_{jk}) = \begin{cases} (r-1), & if\ a_{ik} = a_{jk} = 1 \\ 1, & if\ a_{ik} = a_{jk} = 0 \\ 0, & if\ a_{ik} \neq a_{jk} \end{cases} \tag{31}$$

where r - number of parts, a_{ij} - entries of the $m \times r$ *machine-part incidence matrix* (i.e. $a_{ij} = 1$ if part j needs machine i and $a_{ij} = 0$, otherwise).

Thus, if applied to cell formation, the p-Median problem means finding p machines that are best representatives (centres) of p manufacturing cells, i.e. the sum over all cells of dissimilarities between such a centre and all other machines within the cell is minimized. Once p central machines are found, the cells can be generated by assigning each other machine to the central one such that their dissimilarity is minimum. Note that the desired number of cells p is part of the input for the model and must be known beforehand. Otherwise, it is possible to solve the problem for several numbers of cells and pick the best solution.

Example 2. Let the instance of the cell formation problem be defined by the machine-part incidence matrix:

$$\begin{array}{c c} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix} \end{array} \tag{32}$$

with 4 machines and 5 parts. One can construct the machine-machine dissimilarity matrix C by applying the defined above dissimilarity measure (30):

$$C = \begin{bmatrix} 6 & 20 & 10 & 20 \\ 20 & 9 & 19 & 9 \\ 10 & 19 & 9 & 19 \\ 20 & 9 & 19 & 9 \end{bmatrix} \tag{33}$$

For example, the top left entry a_{11} is obtained in the following way:

$$\begin{aligned} a_{11} &= r(r-1) - \sum_{k=1}^{r} \Gamma(a_{1k}, a_{1k}) = \\ &= 5(5-1) - \Gamma(0,0) - \Gamma(1,1) - \Gamma(0,0) - \Gamma(1,1) - \Gamma(1,1) = \\ &= 20 - 1 - 4 - 1 - 4 - 4 = 6 \end{aligned} \tag{34}$$

If one is interested in having two manufacturing cells then the number of medians p in should be set to 2 and the MBpBM formulation of the given instance of cell formation is as follows:

$$f(\mathbf{y}, \mathbf{z}) = 33 + 4y_1 + 1y_3 + 20z_5 + 20z_6 \longrightarrow \min \tag{35}$$

$$y_1 + y_2 + y_3 + y_4 = 2 \tag{36}$$

$$z_5 \geq y_1 + y_3 - 1 \tag{37}$$

$$z_6 \geq y_2 + y_4 - 1 \tag{38}$$

$$z_i \geq 0,\ i = 5, 6 \tag{39}$$

$$y_i \in \{0, 1\},\ i = 1, \ldots, 4 \tag{40}$$

Its solution $y = (0,0,1,1)^T$, $z = (0,0)^T$ leads to the following cells:

$$\begin{array}{c} \\ 1 \\ 3 \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccccc} 2 & 4 & 5 & 1 & 3 \end{array} \\ \left[\begin{array}{ccc|cc} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{array}\right] \end{array} \tag{41}$$

We conducted numerical experiments in order to compare this approach with other contemporary ones. The aim of our numerical experiments was twofold. First, we would like to show that the model based on PMP produces high-quality cells and in many cases outperforms other contemporary heuristics, thus making their use questionable. Second, by showing that computation times are negligibly small, we argue the use of heuristics for solving PMP itself. In order to compare solutions quality two following measures were used:

$$GCI = 1 - \frac{e}{o} \times 100\% \quad \text{and} \quad \eta = \frac{1}{2}\left[\frac{\alpha}{\alpha + v} + \frac{\beta}{\beta + e}\right] \times 100\%, \tag{42}$$

where *exceptional elements* are those nonzero entries of the block-diagonalized machine-parts incidence matrix that lie outside the blocks, m – the number of machines, r – the number of parts, o – the total number of ones in the machine-part incidence matrix, e – the number of exceptional elements, v – the number of zeroes in diagonal blocks, $\alpha = o - e$ and $\beta = mr - o - v$.

We compared our experimental results with those reported in four recent papers. The main focus was made on the largest 24 instances we could find in literature on CF with $m \times r$ ranging from 8×20 to 50×150. The first paper [33] uses a p-Median approach but solves PMP by a heuristic procedure. It uses Wei and Kern's [31] similarity measure and *GCI* as a quality measure. We were not able to derive the value of η because solutions are not provided in that paper. The second paper [34] applies the ART1 neural network to cell formation, thus using a completely different approach. The authors used η-measure to estimate solution quality and included solutions (block-diagonalized matrices) in their paper, thus making it possible for us to compute *GCI*. The third paper [1] demonstrates an application of a decision-making technique to the cell formation problem. Authors report values of η and we derived values of *GCI* from their solutions.

The most recent paper considered in our computational experiments is [8]. It uses a model that is very similar to the p-Median problem but differs in the following detail: a restriction specifying the number of cells is replaced by a constraint ensuring that each cell has at least two machines. We implemented the models for machine cell formation and part assignment from [8] in Xpress to perform the experiments. Like in the previous cases we used only machine-part incidence matrices as an input and Wei and Kern's (dis)similarity measure. Taking into account that the model from [8] automatically defines the best number of cells, we had to solve our PMP based model for all possible values of p.

The results of the experiments can be summarized as follows. With regard to the solutions quality our model outperforms all its considered counterparts both in *GCI* and η measures. Even though there were scarce instances for which our approach was slightly dominated (this can be explained by the fact that the PMP does not explicitly optimize the considered quality measures), both average and best improvement of quality is noticeable. The difference between quality of our solutions and the ones reported in [33], [34], [1] and obtained by the model from [8] both in terms of *GCI* and η can be summarised as follows:

$$
\begin{array}{lccc}
 & \textit{worst} & \textit{average} & \textit{best} \\
\Delta GCI & -1.90 & 2.86 & 15.20 \\
\Delta \eta & -0.55 & 3.82 & 17.41
\end{array}
\tag{43}
$$

Concerning the computing times, we would like to mention that each of the considered instances was solved within 1 second on a standard PC using Xpress. Even if some other approach can do faster, the difference is negligible.

5.1 Additional Constraints

Above we considered a PMP approach to the CF problem in its simplest form with only the machine-part relations taken into account. Yet, in real manufacturing systems there exist additional factors that must be considered to generate "reasonable" cells. Clearly, there are three places in our model where additional factors can be incorporated:

- dissimilarity coefficients;
- objective function (structure);
- constraints;

The easiest way of introducing additional factors into the model is via the dissimilarity coefficients. For example, dissimilarity between a pair of machines can be made dependent not only on the number of parts that need these machines but also on the weight, volume, processing time of these parts or their operational sequences. Moreover, workers able to operate these machines can be taken into account. Thus, suitable choice of dissimilarity coefficients allows to account for capacity, workload and skills issues without changing the structure of the model. At the same time, a variety of restrictions can be inserted either by changing the structure of the objective function (e.g. by adding terms penalising assignment of some machines to the same cell) or by adding new constraints. It is easy to understand that the only requirement on new constraints is their linearity. The fact that almost any real-world constraints can be either expressed or approximated in a linear form makes the PMP-based approach quite flexible. Taking into account that our model for PMP is very compact, any additional constraints are welcome.

6 Summary and Future Research Directions

The paper presented a new approach to formulation of models for the p-Median problem. We started with a pseudo-Boolean formulation with just one constraint on the number of medians. We then reduced the size of the objective function by truncation and reducing similar monomials, and linearised all non-linear terms in it with additional variables and constraints. The resulting model, a Mixed Boolean pseudo-Boolean Model, incorporates all known reductions and has the smallest number of constraints related to non-negative variables. As we have shown, the matrix of constraints would be as sparse as possible if we were able to solve a generalization of the classic set covering problem defined on the set of all terms involved in the pseudo-Boolean formulation of PMP. Unfortunately, this set covering problem is NP-hard [16]. Anyway, if we evaluate the size of a model by the number of non-zero coefficients in the objective function and corresponding constraints, our MBpBM is the smallest one (within mixed Boolean LP models) and instance specific! Note, however, that a smaller formulation does not guarantee smaller solution time (due to NP-hardness of the problem).

The MBpBM can also be applied to cell formation problems, leading to an improved solutions quality compared to the most contemporary approaches. As well, computing times for the largest CF instances are within 1 second and thus are competitive with any heuristic. In the numerical experiments we considered the simplest possible approach to cell formation aimed at block-diagonalising the machine-part incidence matrix without taking into account additional real-world factors. There are two reasons for this. First, we wanted to demonstrate that even a computationally intractable model of cell formation (at least in its simplest form) can be solved to optimality, and this possibility, to the best of our knowledge, was overlooked in literature. Second, this choice was partially governed by available recent papers in the field with which we wanted to compare our results. At the same time, we showed that several types of constraints can be incorporated into the PMP-based CF model thus making it more realistic and allowing to use all the available information about the manufacturing system.

To summarize, in this paper we have shown that our model extends the possibilities of solving p-Median problem instances to optimality by means of general-purpose MILP software, e.g. Xpress.

References

1. Ahi, A., Aryanezhad, M.B., Ashtiani, B., Makui, A.: A novel approach to determine cell formation, intracellular machine layout and cell layout in the CMS problem based on TOPSIS method. Comput. Oper. Res. 36(5), 1478–1496 (2009)
2. AlBdaiwi, B.F., Ghosh, D., Goldengorin, B.: Data aggregation for p-median problems. J. Comb. Optim. 21(3), 348–363 (2011)
3. Avella, P., Sforza, A.: Logical reduction tests for the p-median problem. Ann. Oper. Res. 86, 105–115 (1999)
4. Avella, P., Sassano, A., Vasil'ev, I.: Computational study of large-scale p-median problems. Math. Prog., Ser. A 109, 89–114 (2007)
5. Belenky, A.S. (ed.): Mathematical modeling of voting systems and elections: Theory and Applications. Math. Comput. Model. 48(9-10), 1295–1676 (2008)
6. Beresnev, V.L.: On a problem of mathematical standardization theory (in Russian). Upravliajemyje Sistemy 11, 43–54 (1973)
7. Beltran, C., Tadonki, C., Vial, J.-P.H.: Solving the p-median problem with a semi-Lagrangian relaxation. Comput. Optim. Appl. 35, 239–260 (2006)
8. Bhatnagar, R., Saddikuti, V.: Models for cellular manufacturing systems design: matching processing requirements and operator capabilities. J. Opl. Res. Soc. 61, 827–839 (2010)
9. Briant, O., Naddef, D.: The optimal diversity management problem. Oper. Res. 52, 515–526 (2004)

10. Brusco, M.J., Köhn, H.-F.: Optimal partitioning of a data set based on the *p*-median problem. Psychometrika 73(1), 89–105 (2008)
11. Church, R.L.: COBRA: a new formulation of the classic p-median location problem. Ann. Oper. Res. 122, 103–120 (2003)
12. Church, R.L.: BEAMR: An exact and approximate model for the p-median problem. Comput. Oper. Res. 35, 417–426 (2008)
13. Cornuejols, G., Nemhauser, G., Wolsey, L.A.: A canonical representation of simple plant location problems and its applications. SIAM J. Matrix Anal. A (SIMAX) 1(3), 261–272 (1980)
14. Deutsch, S.J., Freeman, S.F., Helander, M.: Manufacturing cell formation using an improved p-median model. Comput. Ind. Eng. 34(1), 135–146 (1998)
15. Elloumi, S.: A tighter formulation of the p-median problem. J. Comb. Optim. 19, 69–83 (2010)
16. Garey, M.R., Johnson, D.S.: Computers and Intractability. Freeman, New York (1979)
17. Goldengorin, B., Krushinsky, D.: Complexity evaluation of benchmark instances for the *p*-median problem, Math. Comput. Model. 53, 1719–1736 (2011)
18. Goldengorin, B., Tijssen, G.A., Ghosh, D., Sierksma, G.: Solving the simple plant location problems using a data correcting approach. J. Global Optim. 25, 377–406 (2003)
19. Hakimi, S.L.: Optimum locations of switching centers and the absolute centers and medians of a graph. Oper. Res. 12, 450–459 (1964)
20. Hammer, P.L.: Plant location – a pseudo-Boolean approach. Israel J. Technol. 6, 330–332 (1968)
21. Koskosidis, Y.A., Powell, W.B.: Clustering algorithms for consolidation of customer orders into vehicle shipments. Transp. Res. 26B, 365–379 (1992)
22. Mladenovic, N., Brimberg, J., Hansen, P., Moreno-Peréz, J.A.: The p-median problem: A survey of metaheuristic approaches. Eur. J. Oper. Res. 179, 927–939 (2007)
23. Mulvey, J.M., Beck, M.P.: Solving capacitated clustering problems. Eur. J. Oper. Res. 18, 339–348 (1984)
24. OR-Library, `http://people.brunel.ac.uk/~mastjjb/jeb/orlib/pmedinfo.html`
25. Pirkul, H.: Efficient algorithms for the capacitated concentrator location problem. Comput. Oper. Res. 14(3), 197–208 (1987)
26. Reese, J.: Solution Methods for the p-Median Problem: An Annotated Bibliography. Networks 48, 125–142 (2006)
27. ReVelle, C.S., Swain, R.: Central facilities location. Geogr. Anal. 2, 30–42 (1970)
28. ReVelle, C.S., Eiselt, H.A., Daskin, M.S.: A bibliography for some fundamental problem categories in discrete location science. Eur. J. Oper. Res. 184, 817–848 (2008)
29. Rosing, K.E., ReVelle, C.S., Rosing-Vogelaar, H.: The p-Median and its Linear Programming Relaxation: An Approach to Large Problems. J. Oper. Res. Soc. 30, 815–822 (1979)
30. TSP–library, `http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/`
31. Wei, J.C., Kern, G.M.: Commonality analysis, a linear cell clustering algorithm for group technology. Int. J. Prod. Res. 27(12), 2053–2062 (1989)
32. Wolsey, L.: Mixed integer programming. In: Wah, B. (ed.) Wiley Encyclopedia of Computer Science and Engineering, John Wiley & Sons, Inc., Chichester (2008)
33. Won, Y., Lee, K.C.: Modified p-median approach for efficient GT cell formation. Comput. Ind. Eng. 46, 495–510 (2004)
34. Yang, M.-S., Yang, J.-H.: Machine-part cell formation in group technology using a modified ART1 method. Eur. J. Oper. Res. 188(1), 140–152 (2008)