LI Simpósio Brasileiro de Pesquisa Operacional
Limeira–SP, 02 a 06 de setembro de 2019

New Formulations for the Variants of the Generalized Capacitated Clustering Problem

Marcos Negreiros
Universidade Estadual do Ceará,
Programa de Pós-Graduação em Administração, PPGA
Av. Silas Munguba, 1700 - Itaperi, Fortaleza/CE, CEP: 60.714-903
marcos.negreiros@uece.br

Nelson Maculan
Universidade Federal do Rio de Janeiro,
Programa de Eng de Sistemas e Computação (PESC-COPPE),
Ilha do Fundão, Sala H-322B, CEP: 21.941-972, Rio de Janeiro, RJ, Brazil
maculan@cos.ufrj.br

RESUMO

O problema generalizado heterogêneo de agrupamento capacitado foi recentemente proposto como uma versão do problema de agrupamento capacitado onde o conjunto de medianas capacitadas é o próprio conjunto de itens. Algumas de suas versões foram estudadas, tendo sido indicado que a melhor forma de resolvê-lo seria através do modelo com centro geométrico. Neste trabalho são colocadas duas novas formulações, uma para o problema generalizado heterogêneo de agrupamento capacitado e outra para o problema de agrupamento capacitado heterogêneo em centro geométrico. Apresenta-se a primeira formulação com muita robustez na resolução de problemas de grande porte aqui investigados usando os *solvers* do mercado, e é refeita a formulação não-linear do problema de agrupamento em centro geométrico, onde se obtém uma versão linearizada competitiva.

PALAVRAS CHAVE. Problema de Agrupamento Capacitado (PAC), PAC em Centro Geométrico, Problema de Atribuição Generalizada (PAG).

Tópicos: Otimização Combinatória, Modelagem Matemática

ABSTRACT

The generalized heterogeneous capacitated clustering problem was recently proposed as a capacitated clustering problem where the set of capacitated medians is the set of items to be covered. Some of its studied versions indicated that the best to be used was the capacitated centre clustering form. In this work we propose two new formulations, one for the heterogeneous capacitated clustering form and the other for the heterogeneous capacitated centred clustering form. The first formulations is more robust than previous to solve large instances here investigated using state-of-the-art *solvers*, and for the second formulation we propose a linearization of its original form that is more competitive.

KEYWORDS. Capacitated Clustering Problems (CCP), CCP in Geometric Center, Generalized Assignment Problem (GAP).

Paper topics: Combinatorial Optimization, Mathematical Modelling

https://proceedings.science/p/106756?lang=pt-br

1. Introduction

Mulvey & Beck (1984) first proposed the Capacitated p median Problem **CpMP** or well known as the Capacitated Clustering Problem **CCP**. It is inherently NP-HARD. The problem considers a capacitated p-median problem, where the capacity of each median can be distinct. The problem can be represented by using the following sets, parameters and variables, Mulvey e Beck [1984]:

Sets I is the set of individuals/items ($|I| = n$); J is the set of medians ($|J| = m$); and $|J|$ is the cardinality of the set J, or a fixed number of *clusters* ($|J| = p$).

Parameters p is the number of *clusters*/medians; d_{ij} is the distance from individual i to its median j; q_i is the demand of an individual i; and Q_j is the maximum capacity of median j.

Variables $x_{ij} = 1$ if the individual i is assigned to the median j or $x_{ij} = 0$, otherwise; $y_j = 1$ if an individual j is assigned to be a median or $y_j = 0$, otherwise.

The CpMP can be formulated as:

$$(\mathbf{CpMP})\ Minimize \sum_{i \in I} \sum_{j \in J} d_{ij} x_{ij} \tag{1}$$

$$\text{such that} : \sum_{j \in J} x_{ij} = 1, \ \ \forall i \in I \tag{2}$$

$$x_{ij} \leq y_j, \ \ \forall i \in I, \forall j \in J \tag{3}$$

$$\sum_{i \in I} q_i x_{ij} \leq Q_j y_j, \ \ \forall j \in J \tag{4}$$

$$\sum_{j \in J} y_j \ \leq \ p \tag{5}$$

$$y_j \in \{0, 1\}, \ \ \forall j \in J \tag{6}$$

$$x_{ij} \in \{0, 1\}, \ \ \forall i \in I, \forall j \in J \tag{7}$$

The objective function (1) minimizes the distance between *medians* and items assigned to each median. The constraint (2) defines one item may be assigned to only one median. The constraint (3) complements constraint (2) to indicate each item may be assigned to some other item that is used as a median. The constraint (4) considers the items demand may not overpass the limited capacity of the median they are assigned. The constraint (5) limits the number of medians used from the set of medians. The constraints (6) and (7) refer to the decision variables of the problem.

There are a number of recently directions on this problem proposing heuristic methods to deal with large scale instances of CCP, Shariff et al. [2010], Deng e Bard [2011], Stefanello et al. [2015], Buhrmann [2016], Lai e Hao [2016], Mai et al. [2018]. Although the discussion of this problem is still valid, the proposal of new variations are not in the same way of the development.

The generalized capacitated clustering problem was proposed by Negreiros et al. [2017] and considers minimizing the dissimilarities between clusters with the possibility of generating heterogeneous capacitated clusters composed by a given set of items. For the **CpMP** the proposed formulation was only possible solve the homogeneous instances, or when heterogeneous were evaluated, the clusters must have relative slackness to the imposed limits of the groups to return a feasible solution.

https://proceedings.science/p/106756?lang=pt-br

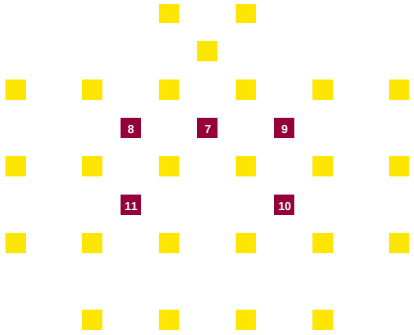


Figure 1: Capacitated p-median instance.

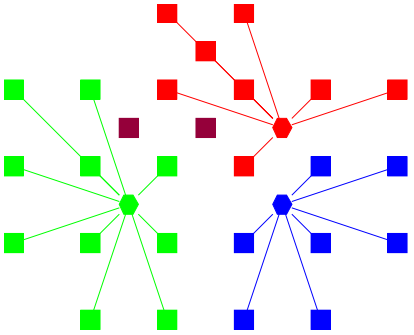


Figure 2: CpMP Solution.

In this paper we consider in section 2 a new variation formulation to deal with any case of the problem, in section 3 we consider the version with a geometric center, composing a new linear approach for the problem, in section 5 we evaluate the results of the formulations over the set of difficult instances from the layout of IT teams.

2. A New Variant to CCP

At first it is considered that the set of medians are disjunct from the set of the items, and as can be observed each median has its own maximum capacity, Mulvey e Beck [1984]. Figure 1 shows an instance of the CpMP where the set of items is in yellow squares ($q_i = 1, \forall i \in I$) and the set of medians appear in brown squares (maximum demand as shown in the label inside the square), $Q = [8, 7, 9, 11, 10]$ from top left to bottom right corner. In Figure 2 we have the optimal solution of the model considering the situation proposed.

If it is desired to extract medians from all the items by using different capacities, the different values of capacities may be repeated for each item in the set of medians and it is necessary to include a new set of constraints to avoid the use of more than one median between copies of the same item. The same model may be repeated, even though the inclusion of this new constraint. Figure 3 shows the situation (instance) where there are no vertices previously defined as medians, instead all of them could be a median with capacity in $Q = [8, 7, 9, 11, 10]$. Figure 4 shows the result of the model obtained for this instance.

Figure 3: New capacitated p-median instance.

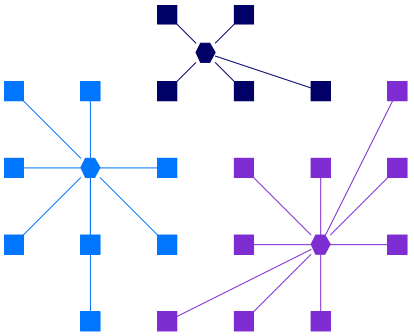


Figure 4: gCpMP Solution.

If one wants to consider a direct model that solves the situation of selecting the best median from the set of items, once all the items can absorb all the capacities, a new model may be

formulated, that we call generic HCMP or simply (gHCMP). In this model, the variable $x_{jj} \in \{0, 1\}$ may represent the item that will be selected as the median from the set I, $(j \in I)$, it will use the appropriate assigned capacity. The order of the median is represented by the variable $g_j^k \in \{0, 1\}$, when it activates the capacity Q_k of median j assigned to group k. Now, it is not known *a priori* what is the assigned capacity for an item that will be chosen as median, and the model decides by itself.

For the gHCMP consider:

Sets:

I - is the set of individuals; J - is the set of possible medians (all individuals); K - is the set of groups;

Parameters:

d_{ij} - is the distance from individual i to its median j; q_i - is the demand of individual i; Q_k - is the maximum capacity of median k;

Variables:

M_j, F_j - Auxiliary variables;

$$g_j^k = \begin{cases} 1, \text{if median } j \text{ an used cluster } k \text{ from the set of possible groups;} \\ 0, \text{otherwise} \end{cases}$$

$$x_{ij} = \begin{cases} 1, \text{if an individual } i \text{ is assigned to individual } j \text{ the median of the group;} \\ 0, \text{otherwise} \end{cases}$$

The gHCMP can be formulated as:

$$(\mathbf{gHCMP}) Minimize \sum_{i \in I} \sum_{j \in I} d_{ij}\, x_{ij} \tag{8}$$

$$\text{such that}: \sum_{j \in I} x_{ij} = 1, \forall i \in I \tag{9}$$

$$x_{ij} \leq x_{jj}, \forall i \in I, \forall j \in I, \tag{10}$$

$$\sum_{i \in I} x_{ij} \geq M_j, \forall j \in J \tag{11}$$

$$\sum_{k \in K} g_j^k = M_j, \forall j \in J \tag{12}$$

$$\sum_{j \in J} g_j^k = 1, \forall k \in K \tag{13}$$

$$\sum_{i \in I} q_i\, x_{ij} = F_j, \forall j \in J \tag{14}$$

$$F_j \leq \sum_{k \in K} Q_k\, g_j^k, \forall j \in J \tag{15}$$

$$g_j^k \in \{0,1\}, \forall j \in J, \forall k \in K \tag{16}$$

$$x_{ij} \in \{0,1\},\ \forall i \in I, \forall j \in J \tag{17}$$

$$M_j \in \{0,1\}, F_j \geq 0 \tag{18}$$

The objective function (8) minimizes the distance between *medians* and each item assigned to its median. The constraints (9) indicate each item may be assigned to only one median. The constraints (10) define that once a median is used an item can be assigned to it. The constraints (11) join items to medians sets with medians to groups sets. The constraints (12 and 13) warrants that each selected median assigns just one group. The constraints (14) consider the sum of items demand assigned to a median may generate a a forward capacity. The constraints (15) indicate the forward capacity for each selected median may not overpass the limited capacity of its assigned group. The constraints (7-18) refer to the decision variables of the problem.

The gHCMP problem is NP-Hard. The model can be solved fast for instances where the capacities of the *clusters* are homogeneous. For the case of instances with heterogeneous capacities, our experiments return a solution in a reasonable time using Xpress and CPlex, and higher times for Gorubi. All these solvers were tested using AMPL language and run supported from Argonne Labs, Fourer et al. [1993], Gropp e More [1997], Czyzyk et al. [1998], Dolan [2001], Neos [2015]. It is worth to observe that this model is O(n^2), in the number of binary variables and constraints, where n is the cardinality of I. This is a better formulation than CpMP formulation that is O($(n+m)^2$) in the number of variables and constraints.

3. The Heterogeneous Capacitated Centred Clustering Problem

In the original pCCCP model proposed by Negreiros e Palhano [2006], the capacity of the *clusters* are homogeneous, and the objective function considers the minimization of the total variance between items and *cluster* center. Here we consider the *clusters* of distinct capacities and the minimization of the total distance (dissimilarity) between items and assigned *cluster* centers. For this case the problem can be represented by using the following sets, parameters and variables:

Sets K is the dimension of the space ($|K| = 2$, in our case); I is the set of individuals; J is the set of *clusters* centers; and $|J|$ is the cardinality of the set J, or a fixed number of *clusters* ($|J| = p$).

Parameters p is the number of *clusters*; a_i is an r-dimensional vector; q_i is the demand of an individual i and Q_j is the maximum capacity of a *cluster*.

Variables $\bar{x}_j$ is an r-dimensional vector represented by the center coordinates of *cluster* j; ; n_j is the number of individuals in *cluster* j and $y_{ij} = 1$ if an individual i is assigned to the *cluster* j or $y_{ij} = 0$, otherwise.

The pHCCCP can be formulated as following,

$$(\mathbf{pHCCCP})\ Minimize \sum_{i \in I} \sum_{j \in J} ||a_i - \bar{x}_j||\, y_{ij} \tag{19}$$

$$\text{such that}: \sum_{j \in J} y_{ij} = 1, \ \ \forall i \in I \tag{20}$$

$$\sum_{i \in I} y_{ij} \leq n_j, \ \ \forall j \in J \tag{21}$$

$$\sum_{i \in I} a_i\, y_{ij} \leq n_j \bar{x}_j, \ \ \forall j \in J \tag{22}$$

$$\sum_{i \in I} q_i\, y_{ij} \leq Q_j, \ \ \forall j \in J \tag{23}$$

$$\bar{x}_j \in \mathbb{R}^k, n_j \in \mathbb{N} - \{0\}, \forall j \in J, \forall k \in K \tag{24}$$

$$y_{ij} \in \{0, 1\}, \ \ \forall i \in I, \forall j \in J \tag{25}$$

The objective function (19) minimizes the Euclidean distance between *clusters* centers and their assigned item. The constraint (20) assigns one individual to just one *cluster*. The constraint (21) considers the number of individuals per *cluster*. The constraint (22) defines the geometric center of the *clusters*. The constraint (23) limits the assigned individuals to the maximum capacity of the *cluster* j. The constraints (24) and (25) refer to the decision variables of the problem.

pHCCCP is non-linear, non-convex and mixed-binary. It is NP-Hard as the previous problems, once its unconstrained version is also NP-Hard, Hansen e Jaumard [1997]. It seems that the problem is further more difficult to be treated then the other previous mentioned above. The formulation can return quality solutions of any kind, being the set of items disperse or not.

This model is O($n\ m$) in the number of binary variables and O(n) in the number of constraints. It is necessary a non-linear-{0, 1} solver to run this model. Solvers like FilMINT, Knitro and BARON from NEOS Server, and LINGO from LINDO Systems, can solve this model close to optimality for some instances, Abhishek et al. [2010], Sahinidis [1996], Artelys [2019], Neos [2015], LindoSystems [2017].

This model returns two important and desirable features of a solution: geometrically centred groups and a center for each group that is a point close to the best group's manager workstation.

The non-convexity of the HCCCP formulation turn us to follow another direction to reduce this effect.

https://proceedings.science/p/106756?lang=pt-br

Figure 5: Heterogeneous Capacitated Centred Clustering instance.

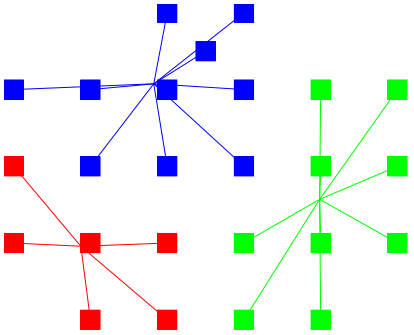


Figure 6: HCCCP associated optimal solution.

3.1. pHCCP - A Convex Differentiable Linear Approach

Consider the pHCCCP as following.

$$(\mathbf{pHCCCP})\ Minimize \sum_{i\in I}\sum_{j\in J} ||a_i - \bar{x}_j||\ y_{ij} \tag{26}$$

$$\text{such that}: \sum_{j\in J} y_{ij} = 1,\ \ \forall i \in I \tag{27}$$

$$\sum_{i\in I} y_{ij} \geq 1,\ \ \forall j \in J \tag{28}$$

$$(23) - (25)$$

The constraints (21) and (22) can be placed as equality constrains:

$$\sum_{i\in I} y_{ij} = n_j, \forall j \in J \text{ and } \sum_{i\in I} a_i\ y_{ij} = n_j \bar{x}_j, \forall j \in J$$

$$\sum_{i\in I} a_i\ y_{ij} = (\sum_{i\in I} y_{ij})\ \bar{x}_j, \forall j \in J$$

where,

$$a_i, \bar{x}_j \in \mathbb{R}^{|K|}, \forall i \in I, \forall j \in J$$

$$0 \leq a_i^k \leq 1,\ 0 \leq \bar{x}_j^k \leq 1, \forall j \in J, k \in K$$

$$\sum_{i\in I} a_i^k\ y_{ij} = \sum_{i\in I} y_{ij}\ \bar{x}_j^k, \forall j \in J, \forall k \in K$$

Therefore a new model with linear constraints can be written as:

$$
\begin{aligned}
(\mathbf{pHCCCP-LC}) \text{Minimize } & \sum_{i\in I}\sum_{j\in J} ||a_i - \bar{x}_j||\, y_{ij} \\
\text{such that}: & \sum_{j\in J} y_{ij} = 1, \ \ \forall i \in I \\
& \sum_{i\in I} y_{ij} \geq 1, \ \ \forall j \in J \\
& \sum_{i\in I} a_i^k\, y_{ij} \ = \ \sum_{i\in I} t_{ij}^k, \forall j \in J, \forall k \in K \qquad (29) \\
& 0 \leq \ t_{ij}^k \ \leq y_{ij}^k, \forall j \in J, \forall k \in K \qquad (30) \\
& (y_{ij}-1) + \bar{x}_j^k \leq t_{ij}^k \leq \bar{x}_j^k + (1-y_{ij}), \forall i \in I, \forall j \in J, \forall k \in K \qquad (31) \\
& \sum_{i\in I} q_i\, y_{ij} \ \leq \ Q_j, \forall j \in J \\
& \bar{x}_j^k \in \mathbb{R}, \forall j \in J, \forall k \in K \\
& y_{ij} \in \{0,1\}, \forall j \in J \\
& t_{ij}^k \in \mathbb{R}, \forall i \in I, \forall j \in J, \forall k \in K \qquad (32)
\end{aligned}
$$

The constraints (29), (30) and (31) warranty the optimal solution, and $\bar{x}_j^\star$ are the barycenter of the points a_i of cluster j.

The continuous relaxation of the constraints, or better, changing the constraints (25) by $0 \leq y_{ij} \leq 1$, $y_{ij} \in \mathbb{R}$ return a set of linear equations and inequalities. Although the objetive function is still non convex.

From Ouzia e Maculan [2018] we can consider the objective function (19) as:

$$
||a_i - \bar{x}_j||\, y_{ij} \ = \ \sqrt{\sum_{k=1}^{2} [y_{ij}(a_i^k - \bar{x}_j^k)]^2}
$$

Let, $s_{ij}^k \ = \ (a_i^k - \bar{x}_j^k) y_{ij}$. When $y_{ij} = 0$ then $s_{ij}^k = 0$ and when $y_{ij} = 1$ then $s_{ij}^k = (a_i^k - \bar{x}_j^k)$.

Once $0 \leq a_i^k \leq 1$ and $0 \leq \bar{x}_j^k \leq 1$ then $-y_{ij} \leq s_{ij}^k \leq y_{ij}, \forall i \in I, j \in J, k \in K$. We can also say that: $(y_{ij}-1)+(a_i^k - \bar{x}_j^k) \ \leq \ s_{ij}^k \ \leq \ (1-y_{ij})+(a_i^k - \bar{x}_j^k)$ and $s_{ij}^k \in \mathbb{R}$, $\forall i \in I, j \in J, k \in K$.

Considering $\lambda \approx 0$, and $\lambda \neq 0$, as for example $\lambda = 1.10^{-8}$. The following model is convex, differentiable with linear continuous relaxation in the constraints as shown before.

$$(\mathbf{pHCCCP-CDC})\textit{Minimize} \sum_{i\in I}\sum_{j\in J}\sqrt{\sum_{k\in K}({s_{ij}^k}^2+\lambda^2)} \tag{33}$$

$$\text{such that}: \sum_{j\in J} y_{ij} = 1,\ \forall i \in I \tag{34}$$

$$\sum_{i\in I} y_{ij} \geq 1,\ \forall j \in J \tag{35}$$

$$\sum_{i\in I} a_{ik}\, y_{ij} \ = \ \sum_{i\in I} t_{ij}^k, \forall j \in J, \forall k \in K \tag{36}$$

$$0 \leq \ t_{ij}^k \ \leq y_{ij}^k, \forall i \in I, \forall j \in J, \forall k \in K \tag{37}$$

$$(y_{ij}-1) + \bar{x}_j^k \leq t_{ij}^k, \forall i \in I, \forall j \in J, \forall k \in K \tag{38}$$

$$t_{ij}^k \leq \bar{x}_j^k + (1-y_{ij}), \forall i \in I, \forall j \in J, \forall k \in K \tag{39}$$

$$\sum_{i\in I} q_i\, y_{ij} \ \leq \ Q_j, \forall j \in J \tag{40}$$

$$(y_{ij}-1) + (a_i^k - \bar{x}_j^k) \leq s_{ij}^k, \forall i \in I, j \in J, k \in K \tag{41}$$

$$s_{ij}^k \leq (1-y_{ij}) + (a_i^k - \bar{x}_j^k), \forall i \in I, j \in J, k \in K \tag{42}$$

$$-y_{ij} \leq s_{ij}^k \leq y_{ij},\ \forall i \in I, \forall j \in J, \forall k \in K \tag{43}$$

$$\bar{x}_j^k \in \mathbb{R}, \forall j \in J, \forall k \in K \tag{44}$$

$$y_{ij} \in \{0,1\}, \forall i \in I, \forall j \in J \tag{45}$$

$$t_{ij}^k \in \mathbb{R}, \forall i \in I, \forall j \in J, \forall k \in K \tag{46}$$

$$s_{ij}^k \in \mathbb{R}, \forall i \in I, \forall j \in J, \forall k \in K \tag{47}$$

4. A Metaheuristic Framework for the HCCCP

A framework of metaheuristic was specially developed to the HCCCP based on a previous framework designed by Muritiba et al. [2012]. That previous framework includes a new step in the process of building multistart feasible solutions by applying a Generalized Assignment Problem solver Karabakal [1992] or a GAP-Approximate heuristic Martello e Toth [1990] to generate an improved optimal/near optimal solution for the actual assigned centers to the HCCCP instance in evaluation. This solution passes through a local search set of methods (transfer, swap and wave) which feeds the Tabu-List, and tabu continues its evaluation until a new solution needs to be built, Figure 7.

The Multistart method (pRBFC) Muritiba et al. [2012], distributes items randomly through k clusters and controls the capacity of each Q_k group. First of all, each *cluster* receives a random vertex and then the others are inserted into the *cluster* nearest to them, as long as the capacity of each *cluster* is not exceeded. The solutions generated by this method do apparently not have good spatial configuration. However, by our experiments with the diverse sets of instances, the centers of the *clusters* are positioned in an interesting way. Based on this observation, it was decided to combine this method with another, taking advantage of the centers easily computed by this method.

Considering the framework schema represented in Figure 7, the Multistart method is combined with KBLGAP method proposed in Karabakal [1992] or Martello e Toth [1990], for the Generalized Assignment Problem (GAP) and to arrange the vertices in *clusters* using a distance matrix

https://proceedings.science/p/106756?lang=pt-br

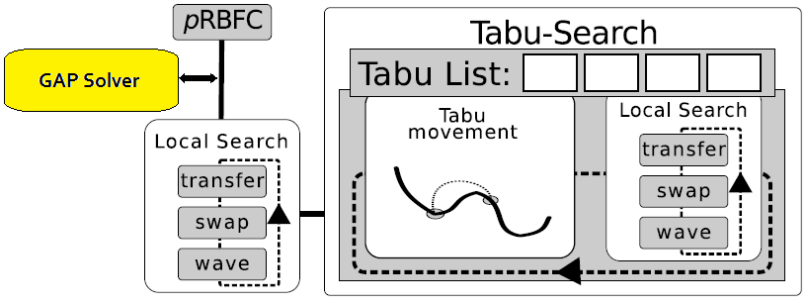


Figure 7: Metaheuristic framework schema developed to HCCCP.

calculated by the distances between the vertices and the centers of the *clusters* generated in the multistart phase. The two combined methods can find optimal solutions or near-optimal as observed in our computational experiments.

The GAP method makes the Multistart solutions more compact, and turn them computationally less expensive, by reducing the number of required Tabu movements to achieve the optimal solution. If we use a B&B Lagrangian method Karabakal [1992], the best upper-bounds can be obtained for moderate size instances (1,000 individuals), using an approximate GAP heuristic Martello e Toth [1990] it is possible to deal with larger size instances and new upper-bounds can be obtained.

The major gain with the use of GAP methodology in the framework of heuristics is that we redesigned the multistart method to find solutions for distinct *cluster* capacities. The procedure is basically the same, but the distribution can achieve different limited capacity values for the card players (known capacities for the *clusters* centers).

5. Instances and Results

There are some features related to the real problem as indicated by Dataprev and Serpro enterprises. Common to both are the fact that some teams must remain together once they perform software tests or maintenance. Other important situation is that people may be fixed in some workstation because of mobility problems or fixed manager rooms or even closeness to the main server. The dynamic of the process and layout is defined by the number of contracts running and the new ones to be run. The dynamic changes are not that much for both cases actually they can be observed like a semestral change or at most an annual change. When moves occur they great affect workers positioning in the factory.

The instances here evaluated were divided in seven types. The first two (Test 1 and 2) were designed to evaluate the models and their results. The DPV-2016 is the demand configuration of Dataprev IT-teams for the year of 2016. DPV-2017A is the situation regarding the 2017 Dataprev running projects, considering the overall positioning where there are no fixed or no possible closest teams to be formed. DPV-2017B considers the situation of six fixed individuals. Their places were removed from the layout. Other 41 individuals from four teams must form one team to be close. The same situation is evaluated in SERPRO-2017A where all the workstations can be used by any team in SERPRO-2017B. The fixed workstations were excluded, and the teams that must remain together form two new groups that is the sum of the teams they belong to. Table 1 indicates these distribution and features of each instance we solved, where $|N|$ is the number of individuals to be placed, $|M|$ is the number of groups and Q is a multiset of groups to be placed. For the multiset Q, it is defined by a collection of (Q_k, f)-tuples where Q_k is the maximum capacity of group k and f is the number of groups to be formed with Q_k maximum capacity.

Instance	Lable	$\|N\|$	$\|M\|$
1	Test 1	20	3
2	Test 2	25	6
3	DPV-2016	175	17
4	DPV-2017A	174	26
5	SERPRO-2017A	350	6
6	DPV-2017B	168	18
7	SERPRO-2017B	252	15

Instance	Q
1	$Q = \{(Q_k, f) : (3,1), (5,1), (12,1)\}$
2	$Q = \{(Q_k, f) : (6,1), (9,1), (10,1)\}$
3	$Q = \{(Q_k, f) : (1,1), (3,1), (6,5), (7,1), (8,3), (9,3), (12,1), (29,1), (42,1)\}$
4	$Q = \{(Q_k, f) : (1,5), (3,4), (4,2), (6,1), (7,3), (8,1), (9,3), (11,3), (12,2), (13,1), (17,1)\}$
5	$Q = \{(Q_k, f) : (6,1), (7,1), (8,2), (9,2), (10,4), (11,4), (12,4), (13,1), (14,1), (15,1), (29,1), (100,1)\}$
6	$Q = \{(Q_k, f) : (1,1), (3,2), (4,2), (6,1), (7,3), (8,1), (9,2), (11,2), (12,2), (13,1), (41,1)\}$
7	$Q = \{(Q_k, f) : (6,1), (7,1), (8,1), (11,3), (12,3), (13,1), (14,1), (21,1), (25,1), (38,1), (51,1)\}$

Table 1: Instances features.

Each instance were solved for the models here indicated. Once the HCCCP problem is combinatorial, and the other model is linear mixed-integer. We evaluated the instances using Argonne NEOS-Server Platform and its solvers to solve them. We run the the mixed integer linear model using the solvers: GUROBI, CPLEX, XPRESS and LINGO. For the nonlinear mixed integer HCCCP model we used the solvers: FilMINT, BARON and Knitro, **?**, Sahinidis [1996], Artelys [2019].

5.1. gHCMP

For the **gHCMP** all instances were solved using the proposed model but for some selected solvers: CPLEX (IBM ILOG CPLEX), XPRESS (FICO XPRESS), GUROBI and LINGO. LINGO does not obtain any feasible solution for most of them unless two test instances. The instances presented total capacity equal to the sum of the capacities of each *cluster*. This may interfere in obtaining feasible upper bounds. Table 2 shows the dominance in time of XPRESS in obtained the solutions over all other solvers used. The solutions indicated as (*) are the non optimal solutions, the others are the optimal solutions obtained from each solver.

Instance	XPRESS		CPLEX		GUROBI		LINGO	
	OF	Time (s)	OF	Time (s)	OF	Time (s)	OF	Time (s)
Test 1	2207.11	0.008998	2207.11	0.010998	2207.11	0.12998	2207.1	23.
Test 2	2736.7	0.005999	2736.7	0.009998	2736.7	0.012998	2736.7	34:39.
DPV-2016	27435.8	6.41702	27435.8	39.773	27435.8	431.365	NFSF	>1h
DPV-2017A	3706.19	3.56246	NFSF	> 1h	3706.19	114.609	NFSF	>1h
SERPRO-2017A							NFSF	>1h
DPV-2017B	5249.65*	5.6214	NFSF	> 1h	5249.65*	393.65	NFSF	>1h
SERPRO-2017B	30596.3*	30.7853	31316.8*	44.5862	30430.8*	11812.8	NFSF	>1h

Table 2: Best results obtained from the solvers to each problem and their instances.

Figures 8 and 9 from XPress and Gurobi. The solutions presents compact layouts. For both cases the **gHCMP** model is adequate for IT-Teams layout and indicate the position of the manager, the solutions are quite practical. It can be noted little advantage of the objective value obtained from Gurobi over Xpress for the instance SERPRO-2017B.

Figures 10 and 11 show the best solution layout for Serpro obtained for both instances SERPRO-2017A and SERPRO-2017B from Xpress and Gurobi. The solutions presents compact layouts. For both cases the **gHCMP** model is adequate and fit well the evaluation of the process. Also notable here Gurobi advantage in best objective value but in much time to obtain. The solutions are quite practical and can be directly implemented by the IT-managers.

5.2. HCCCP

The HCCCP solutions, presented in table **??**, were solved with great difficulty by the solvers used. In general FilMINT solved the instances quite well but not at optimal solution for

https://proceedings.science/p/106756?lang=pt-br

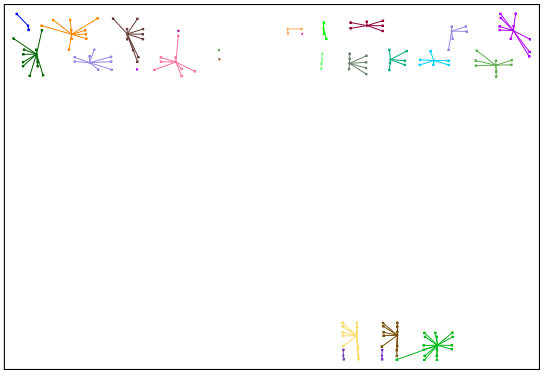


Figure 8: Best solution for DPV2017A using **gCpMP** model.

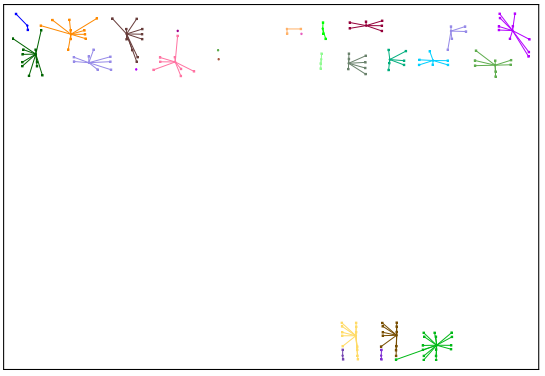


Figure 9: Best solution for DPV2017B using **gCpMP** model.

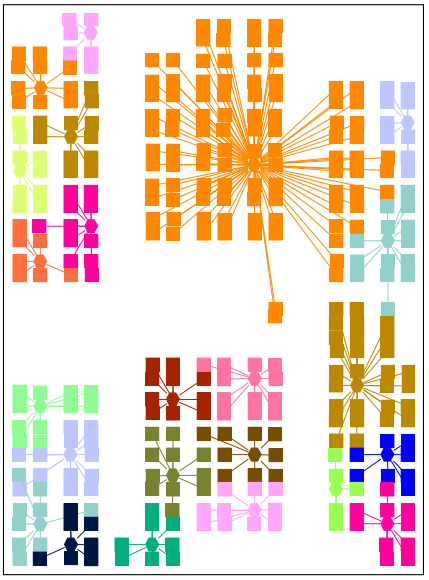


Figure 10: Best solution for SERPRO-2017A.

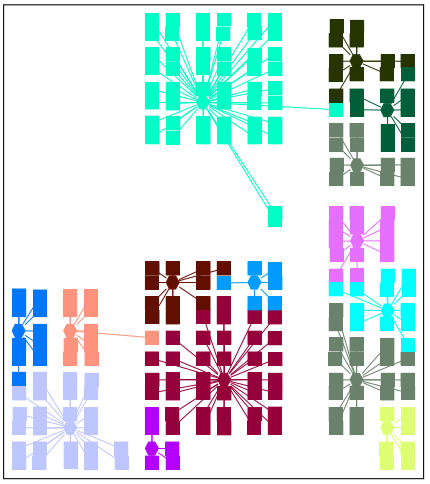


Figure 11: Best solution for SERPRO-2017B.

Dataprev and Serpro instances. Knitro solved the five bigger instances where two were the best upper-bound found. FilMINT dominates three of them and Knitro dominated the other two.

Figures 12 and 13 show the best solution layout for Dataprev obtained for instance DPV2017A from Knitro, and for instance DPV2017B both obtained from Knitro. It can be noted that Knitro solver obtained good layouts, although in the DPV2017A two missing workstations are dividing two groups in the two floors.

In Table 3 we consider the comparison of the solutions obtained between the best solver upper-bound and the metaheuristic results for the HCCCP.

6. Conclusion

The new formulations here proposed were evaluated for a set of instances selected from IT Teams project layout, whose features indicates that the groups to be formed must be fulfilled to the end. Also the capacities of each group is very different although the items have unit demand.

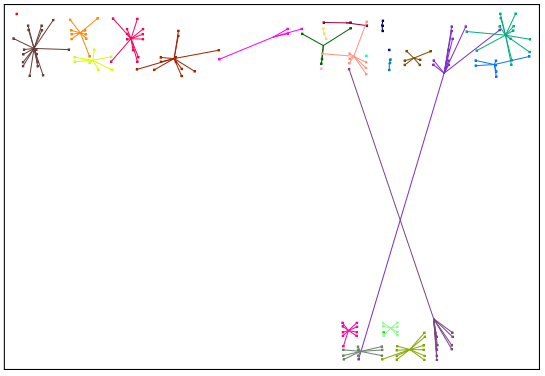


Figure 12: Best solution for DPV2017A.

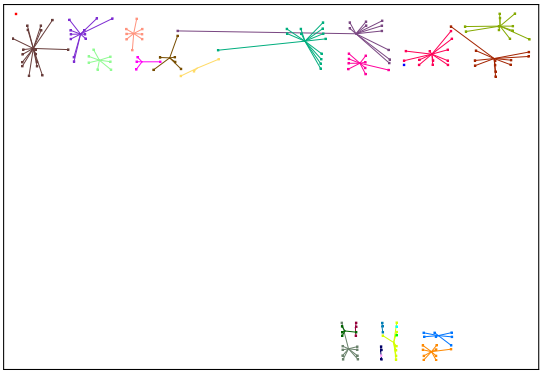


Figure 13: Best solution for DPV2017B.

Instance	HCCCP		
	OF	Time (s)	Solver
Test 1	2199.185	0.046	filmint
Test 2	2774.545	0.063	filmint
DPV-2016	31053.900	48.992	knitro
DPV-2017A	6916.457	35.000	knitro
SERPRO-2017A	31295.430	0.088	filmint
DPV-2017B	6465.150	0.148	knitro
SERPRO-2017B	19850.854	0.471	filmint

Table 3: Best results obtained from solvers to HCCCP instances.

This favour to evaluate the process of obtaining solutions using our models from the well known solvers, and extracts from them the best on cost to obtain optimal of even good feasible solutions.

It is clear that the use of cutting methods and /or math-metaheuristics could be here welcome, even combining to achieve much better solutions for any other context.

References

Abhishek, K., Leyffer, S., e Linderoth, J. (2010). Filmint: An outer-approximation-based solver for nonlinear mixed integer programs. *INFORMS Journal on Computing*, 22:555–567.

Artelys (2019). Artelys knitro 10.3. `https://www.artelys.com/fr/actualites/actualites-solveurs/knitro-10-3/`.

Buhrmann, J. (2016). *The Effects of Clustering on The Medium and Large-Scale Capacitated Location-Routing Problem*. PhD Thesis, University of the Witwatersrand.

Czyzyk, J., Mesnier, M., e Moré, J. (1998). The neos server. *IEEE Journal on Computational Science and Engineering*, 5:68–75.

Deng, Y. e Bard, J. (2011). A reactive grasp with path relinking for capacitated clustering. *Journal of Heuristics*, 17:119–152.

Dolan, E. (2001). The neos server 4.0 administrative guide. Technical Memorandum ANL/MCS-TM-250, Mathematics and Computer Science Division, Argonne National Laboratory.

Fourer, R., Gay, D., e Kerningham, B. (1993). *AMPL: A Modeling Language for Mathematical Programming*. Boyd & Fraser Publishing.

https://proceedings.science/p/106756?lang=pt-br

Gropp, W. e More, J. (1997). Optimization environments and the neos server. In Buhman, M. D. e Iserles, A., editors, *Approximation Theory and Optimization*, p. 167 – 182. Cambridge University Press.

Hansen, P. e Jaumard, B. (1997). Cluster analysis and mathematical programming. *Mathematical Programming*, 79:191–215.

Karabakal, N. (1992). Kblgap: A c code for solving the generalized assignment problem. Technical Report, University of Michigan.

Lai, X. e Hao, J.-K. (2016). Iterated variable neighborhood search for the capacitated clustering problem. *Engineering Applications of Artificial Intelligence*.

LindoSystems (2017). *LINGO 17.0 - Optimization Modeling Software for Linear, Nonlinear, and Integer Programming*. LINDO SYSTEMS.

Mai, F., Fry, M., e Ohlmann, J. (2018). Model-based capacitated clustering with posterior regularization. *European Journal of Operational Research*.

Martello, S. e Toth, P. (1990). *Knapsack Problems, Algorithms and Computer Implementations*. John Wiley and Sons.

Mulvey, J. e Beck, M. (1984). Solving capacitated clustering problems. *European Journal of Operational Research*, 18:339–348.

Muritiba, A., Negreiros, M., Souza, M., e Oria, H. (2012). A tabu search for the capacitated centred clustering problem. In *Annals of SBPO/CLAIO 2012*, p. 2344–2359. ALIO/SOBRAPO.

Negreiros, M., Batista, P., e Rodrigues, J. (2017). Optimization models for capacitated clustering problems. In *Annals of the XLIX SBPO*. SOBRAPO.

Negreiros, M. e Palhano, A. (2006). The capacitated centred clustering problem. *Computers and Operations Research*, 33:1639–1663.

Neos (2015). Neos (network-enabled optimization system). `http://www.neos-server.org/neos/solvers/index.html`.

Ouzia, H. e Maculan, N. (2018). Mixed integer non linear programming (minlp) models for the euclidean steiner tree problem in rn. *Technical Report - LIP6,UPMC - Paris*, p. 8p.

Sahinidis, N. (1996). Baron: A general purpose global optimization software package. *Journal of Global Optimization*, 8:201–205.

Shariff, S., Moin, N., e Omar, M. (2010). An alternative approach using genetic algorithm based heuristics for capacitated maximal covering location allocation problem. *Operations and Supply Chain Management*, 3:36–49.

Stefanello, F., DeAraujo, O., e Muller, F. (2015). Matheuristics for the capacitated p-median problem. *International Transactions in Operational Research*, 22:149–167.

https://proceedings.science/p/106756?lang=pt-br