Annals of Operations Research (2022) 316:1157–1185
https://doi.org/10.1007/s10479-020-03785-4

S.I.: CLAIO 2018

Capacitated clustering problems applied to the layout of IT-teams in software factories

Marcos J. Negreiros[1] · **Nelson Maculan**[2] · **Pablor L. Batista**[1] · **João A. Rodrigues**[1] · **Augusto W. C. Palhano**[1]

Accepted: 1 September 2020 / Published online: 24 September 2020

Abstract

This work studies Heterogeneous Capacitated Clustering Problems (HCCP) which are variations of the Capacitated Clustering Problem (CCP). They are applied to the layout of IT-Teams in software factory. In the real problem, there are workstations placed along the software factory where they are disposed in a pattern the individuals working in the same project form a group that would be geographically close to each other. The groups are heterogeneous in the number of individuals. The general problem discussed here consider a given number of individuals with attributes (weight and coordinates in Euclidean space), and wishes to determine minimum dissimilar *clusters* constrained to a given maximum capacity for each *cluster*. The groups are formed to achieve a specific objective of just forming feasible capacitated compact groups and/or also including the position of the group manager. The models here presented and evaluated consider a new generalized version of the heterogeneous capacitated median problem (gHCMP) which extracts the medians from the set of individuals; a new formulation is introduced for the min-max diameter heterogeneous capacitated clustering problem; a new formulation for the minimum group distance heterogeneous capacitated clustering problem; and a new formulation using the HCCCP. We also discuss ways to obtain the best results from the proposed formulations using adequate solvers for each problem; and it is presented a metaheuristic framework for the HCCCP that solves close to optimality all the real instances here tested. We also present the results obtained from the formulations and metaheuristic related to instances that represent real situations about organizing the layout of two Brazilian software factories. The proposed gHCMP model outperform the solutions obtained for the evaluated instances also using the other models.

Keywords Capacitated clustering problem · Mathematical programming modeling · Solvers

1 Introduction

The focus on resource saving is one of the reasons for the increasing occurrence of large companies in adopting remote research and development (R&D) teams. Apart from other well-known advantages of this trend, some management problems may arise when dealing with dispersed teams: ensuring control in completing critical tasks; supporting a feature that

Extended author information available on the last page of the article needs to be updated regularly; always have some member of the team when a failure is detected or a new task need to be implemented and others.

All of these problems are related to communication barriers. The empirical study of Allen (1977) shows that the probability of communication $P(D)$ between members of a R&D project drastically decreases with the increase of physical distance D. The logarithmic relationship between $P(D)$ and D was demonstrated for small (metric) and large scopes, that is to say, the problems derived from the difficulty of communication between members spread throughout the world can be replicated to a smaller extent to the dispersal in a single company. Although this is an old issue, considering actual availability of very high speed internet/intranet, the concept of agile development used in software factories in Brazil maintain old standards because of labor regulations.

Schedules that alter the contingent of projects, or the opening and closing of these projects, can motivate companies to move members between workstations, or even teams between rooms or floors. When this movement is done without criteria it consequently disorganizes teams, separating members in dispersed places configuring undesirable circumstances such as those already mentioned. The definition of the layout of the teams can be complex in companies that have a high turnover rate and a large number of employees.

This study was elaborated to reach a representation of the real problem, IT-Teams project placement, from Brazilian federal companies, Dataprev and Serpro. These IT companies operate as software factories and rely on standardized development teams skilled in different contexts.

Each project team has a number of people previously defined by managers in accordance with the complexity or work effort estimated to the project. This number is usually different between teams due to the different complexities of projects.

Dataprev's project teams are located in two large halls of a building, on separate floors (3rd and 8th), where workstations are arranged in cells composed up to four stations. In 2016 there were 175 workstations where teams are allotted, see Fig. 1.

At Serpro, teams are dispersed around a large pavilion, sharing the same physical space between customer service and software development demanding. In 2017 there were 350 workstations altogether, where the teams are put on workstations arranged in lots of 8 stations, see Fig. 2.

In both contexts, it is a common problem to structure the layout of the factory workstations in such a way that the distinct groups of the projects may maintain their own members close to each other. There are many ways of building groups with individuals of any kind and characteristic. Although humans are specialized in separating things to perform better decisions when partitions are to be the expected result, this decision process is not possible when the number of individuals is high. When the individual have a demand/offer and a limited capacity is involved while building the groups, the resulting decision problem is more difficult because of its combinatorial counterpart. Also, in this context, there are many ways of doing *clusters*. We will refer to these type of problems as Capacitated Clustering Problem (CCP) from now on.

In the context of CCP, *capacity* is related to the limit of the desks for a specific project. A *diameter* of a *cluster* is the greatest distance between any pair of desks of the same *cluster* or project. A *cluster center* is the barycenter of the *cluster* or its geometric center. Also a *cluster* center can be a project desks' median.

Mulvey and Beck (1984) were the first researchers that proposed a model for the Capacitated Clustering problem. In fact, the studied clustering process was related to a Capacitated p-median Problem (CpMP). In the problem individuals have attributes (or coordinates and weights). The center of the *clusters* are medians from a set of possible places that can absorb

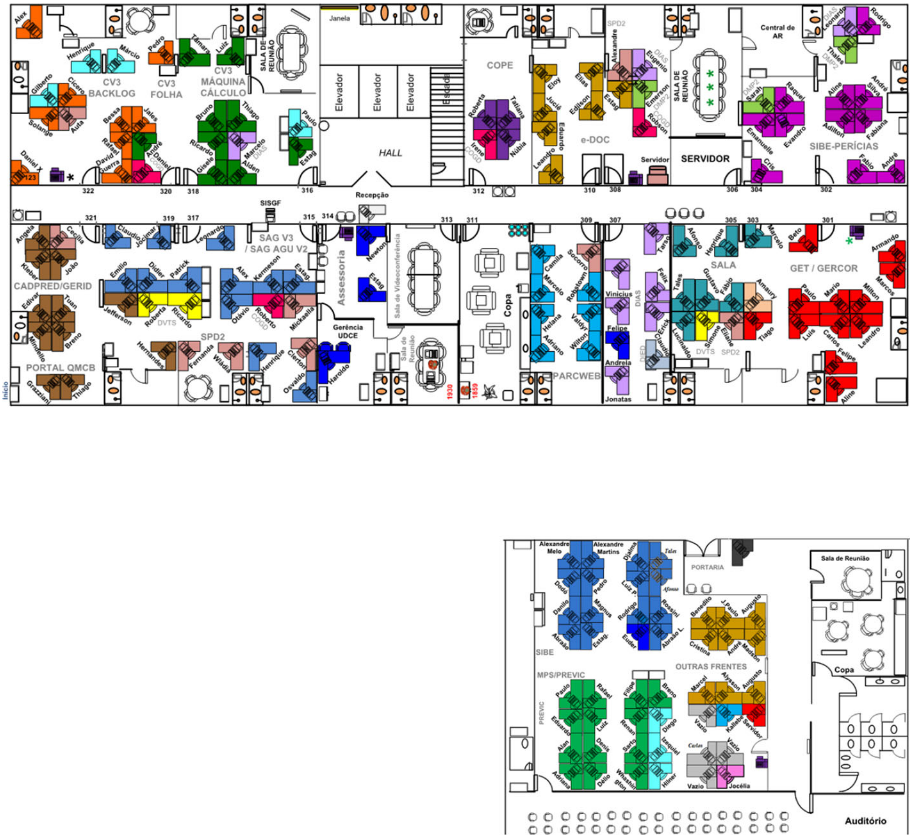


Fig. 1 Dataprev enterprise IT-teams layout in 2 floors

at maximum capacity a sub-set of the individuals. The capacity of the *clusters* may differ for each indicated possible median (Mulvey and Beck 1984).

Negreiros & Palhano (2006) proposed a new problem, considering in the Euclidean plane, a constrained process of doing min-sum clustering from a set of individuals. The center of each *cluster* is defined by the coordinates obtained from the average of the coordinates of individuals in the group. They called the problem as the Capacitated Centred Clustering Problem (CCCP). Different objectives were proposed, indicating variations of the decision process considering a fixed (p) or the best number of *clusters*. In the model, to find the best number of *clusters*, a cost is introduced to open a new *cluster*. Also, the *clusters* have the same capacity. The applications of the problem include garbage collection, dengue disease combat, sales force territory design, newspaper subscribers delivery, vehicle routing, offshore wind farm electrical cable layout, etc (Negreiros and Palhano 2003, 2006; Pillai et al. 2015).

A variation of the Capacitated p-median Problem was proposed by Prata (2015). It indicates that in the new problem each item requires a type of service, and the set of p-medians have an associated capacity. The capacities may be different between types and medians. The author called the problem Multi-Capacitated Clustering Problem (MCCP), and reported possible applications like: assignment of students by their level to schools considering their

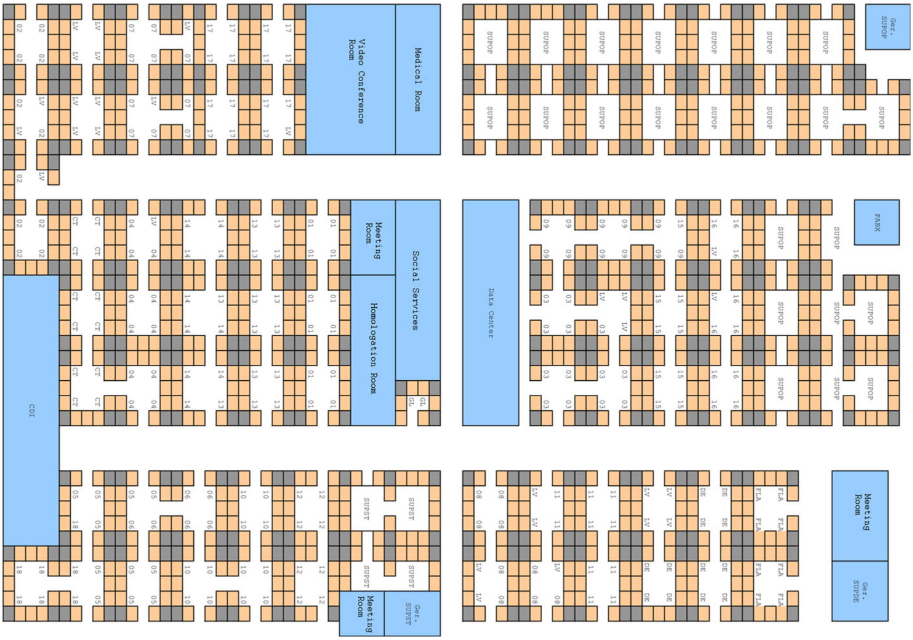


Fig. 2 Serpro enterprise IT-teams layout in one pavilion

capacity to absorb students by each level, clustering customers demands for products from refineries, considering that refineries have different capacities to process each product.

A CCP version for symmetric graphs $G(V, E)$ with positive weights in the vertices (V) and edges (E) was proposed by Deng and Bard (2011). The authors contextualize their CCP to help facility planners at mail processing and distribution centers within the US Postal Service as well as it was further indicated to mobility networks by Moran-Mirabal et al. (2013). The CCP variation called Maximally Diverse Grouping Problem (MDGP), (Brimberg et al. 2018), considers that the set of vertices V are to be partitioned into p *clusters*, such that the sum of the benefits associated with the edges within each *cluster* is maximized, and the sum of the node weights of each *cluster* falls with the interval $[C^{min}, C^{max}]$. In MDGP the weighs of the vertices are all equal to 1. In the mobility network the variation of the CCP problem is called Handover Minimization Problem (HMP). HMP constraints are the same as MDGP, but the weight of the vertices are positive reals, and the objective function minimizes the sum of distances of edges within each *cluster*.

For the MDGP (Deng and Bard 2011) two formulations were proposed to solve the problem: linear-binary and mixed-binary. Since they are NP-Hard, they developed a reactive greedy randomized adaptive search procedures (GRASP) coupled with variable neighborhood descent (VND) variants. In addition, they propose a Path Relinking post-processing procedure which did not result in a significant improvement. Later on, (Lewis et al. 2014) proposed a new quadratic linear-binary formulation, and compared with previous models for the set of CCP benchmark instances running directly by the solvers CPLEX and Gurobi. The quadratic linear-binary formulation took more time to prove optimality than linear-binary, but the results were comparable with the heuristics for the instances evaluated. This same formulation was also tested by Martinez-Gavara et al. (2015) for both MDGP and HMP using Tabu Search and GRASP, and by Brimberg et al. (2018) for generalised variable neighbourhood search (GVNS) and skewed general variable neighbourhood search (SGVNS) metaheuristics. The authors (Brimberg et al. 2018) also proposed a new set of larger benchmark instances, and showed their SGVNS are comparable to the other heuristics to low and average sized instances, and outperformed them for larger instances. Their method also shows to run all instances in reasonable time.

This paper considers different models for the CCP, where there is only one type of service/product per individual to be clustered, and the capacity of the *clusters* may differ. The main goal of our evaluations is to identify the characteristics of each solution obtained, for the same aspect of clustering process. In the context of the difficulty to solve the problem studied, we explain their characteristics considering the aspects related to its mathematical formulation (linear mixed-integer or nonlinear mixed-integer).

In this work we do not combine upper bounds produced by heuristic methods with the solvers resolution, nor include or introduce feasible cuts to achieve better feasible bounds for the different forms of the Capacitated Clustering problem. We have analysed heuristic and metaheuristic approaches for the HCCCP, (Batista et al. 2015), (Muritiba et al. 2012), but not for all the models here evaluated. We are only interested in solving the layout problem arisen from IT-Teams placement, by using appropriate formulations and heuristic methods.

In Sect. 2 we consider the mathematical formulations and aspects for the CpMP, in Sect. 3 we consider the Min-Max Diameter Heterogeneous Capacitated Clustering Problem (MMD-HCCP) and in Sect. 4 we consider the Min Group Distance Heterogeneous Capacitated Clustering Problem (MGD-HCCP), as new versions for the capacitated clustering problem. In Sect. 5 we discuss two mathematical formulations for the heterogeneous capacitated centred clustering problem (HCCCP) and their aspects to be solved using available nonlinear mixed-integer solvers; in Sect. 6 we present a framework of heuristics to the HCCCP that can be applied to this context. In Sect. 7 we show instances, variations of the problem and the results of our models and metaheuristic for the real IT applications in the enterprises previously indicated. In Sect. 8 we conclude indicating the appropriate use of each formulation.

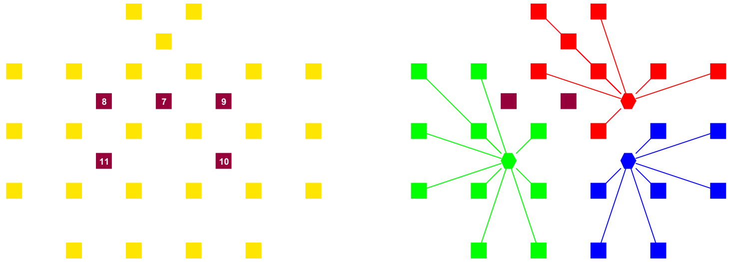
2 The capacitated *p*-median problem and a new variant

Mulvey andBeck (1984) first proposed the **CpMP**. 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 and Beck 1984):

- Sets: let I be the set of individuals/items ($|I| = n$), and let J be the set of medians ($|J| = m$), $|J| = p$ is the fixed number of *clusters*.
- Parameters: p can also be 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:

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



(a) Capacitated p-median instance. **(b)** CpMP optimal solution.

Fig. 3 CpMP instance and solution

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

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

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

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

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

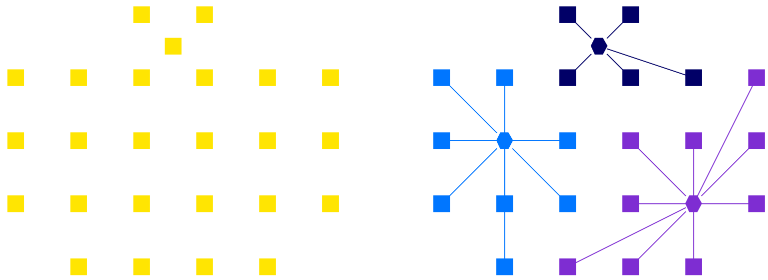
$$x_{ij} \in \{0, 1\}, \quad \forall i \in I, \forall j \in J \tag{7}$$

The objective function (1) minimizes the distance between *medians* and items assigned to each median. The constraint (2) defines that every item is to be assigned to exactly 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 to. 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.

At first it is considered that the set of medians are disjunct from the set of items, and as can be observed each median has its own maximum capacity (Mulvey and Beck 1984). Figure 3a 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 Fig. 3b 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. In this case, 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, with the addition of this new constraint. Figure 4a 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 4b shows the result of the model obtained for this instance.

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 generalized heterogeneous capacitated median problem (gHCMP). In this model, the variable $x_{jj} \in \{0, 1\}$ may represent the item that will be selected as the



(a) A generalized heterogeneous capacitated median instance.

(b) gHCMP optimal solution.

Fig. 4 gHCMP instance and solution

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, the model decides by itself.

For the Generalized Heterogeneous Capacitated Median Problem (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; $g_j^k = \begin{cases} 1, \text{if median j used a cluster k from the set} \\ \text{of possible groups;} \\ 0, \text{otherwise} \end{cases}$

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

The gHCMP can be formulated as:

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

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

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

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

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

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

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

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

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

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

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

The objective function (8) minimizes the distance between *medians* and each item assigned to its median. The constraints (9) defines that every item has to be assigned to exactly 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 and 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 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 (16–18) refer to the decision variables of the problem.

The gHCMP problem is NP-Hard as CpMP (Mulvey and Beck 1984). 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 Gurobi. All these solvers were tested using AMPL language and run supported from Argonne Labs, NEOS (Czyzyk et al. 1998; Dolan 2001; Gropp and Moré 1997; Neos 2015; Fourer et al. 1993). 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 min–max diameter capacitated clustering problem

In this version of the CCP, the clustering must be done considering the minimization of the maximum internal distance of a *cluster* between their individuals. The unconstrained version of this problem was proposed by Rao (1971), and now we add the capacity constraint. The min-max diameter heterogeneous capacitated clustering problem (MMD-HCCP) can be stated as follows:

Sets	I is the set of individuals ($\|I\| = n$); and M is the set of *clusters* ($\|M\| = m$).
Parameters	p is the maximum number of *clusters*; d_{ij} is the distance from individual i to individual j; q_i is the demand of an individual i; and Q_j is the maximum capacity of a *cluster* j.
Variables	Z is the greatest diameter of a *cluster* between all m *clusters*; and $x_{ik} = 1$ if the individual i is assigned to the *cluster* k or $x_{ik} = 0$, otherwise.

The MMDH-CCP can be formulated as:

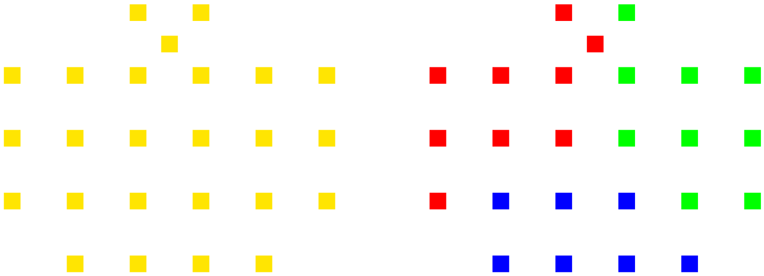
$$(\mathbf{MMD-HCCP}) Minimize \ \ Z \tag{19}$$

$$\text{such that :} d_{ij}(x_{ik} + x_{jk} - 1) \leq Z, \tag{20}$$

$$\forall i \in I, \forall j \in I, \ i < j, \ \forall k \in M \tag{21}$$

$$\sum_{k \in M} x_{ik} = 1, \quad \forall i \in I \tag{22}$$

$$\sum_{i \in I} q_i \ x_{ik} \leq Q_k, \quad \forall k \in M \tag{23}$$



(a) Min-max Diameter Capacitated Clustering instance.

(b) MMDCCP associated optimal solution.

Fig. 5 MMDCCP instance and solution

$$Z \geq 0, \; x_{ik} \in \{0, 1\}, \; \forall i \in I, \forall k \in M \tag{24}$$

The objective function (19) minimizes the diameter of the greatest cluster. The constraint (21) enforces Z to be at least equal to greatest diameter solution. The constraint (22) assigns one item to one cluster. The constraint (23) considers the sum of items demand assigned to a cluster may not overpass its limited capacity. The constraint (24) refers to the decision variables of the problem.

The MMD-HCCP is NP-HARD. This model is O(n m) in the number of constraints as well as for the number of binary variables. It is a good and fast way of doing capacitated clustering since its linear mixed-integer formulation is easier than previous capacitated median problems.

MMD-HCCP can be of adequate use, if the set of items are formed by homogeneous positions in the Euclidean space ($\mathbb{R}^2$) and the groups are of enough capacity to maintain individuals as close as possible, Fig. 5a. The groups, for this situation, may not overlap (compact), and reasonable final geometric solutions may be obtained, as can be seen in Fig. 5b. On the other hand, if the individuals are disperse and/or the capacity of the *clusters* introduces slackness, the solution may result in bad clustering, but in good mixture of groups, see Fig. 6. In this case, the objective may be changed to attend the size of each particular group to be formed.

4 The minimum groups distance capacitated clustering problem

This is a new version of the CCP, here the *clusters* must be created considering the minimization of the total distance of p formed cliques. A mixed-integer linear unconstrained version of this problem was proposed by Brito et al. (2008) and (Brito et al. 2017), here it is reformulated and the heterogeneous capacity constraint is added. The minimum groups distance heterogeneous capacitated clustering problem (MGD-HCCP) can be stated as follows:

Sets	I is the set of individuals ($\|I\| = n$); and M is the set of *clusters* ($\|M\| = m$).
Parameters	d_{ij} is the distance from individual i to individual j; q_i is the demand of individual i; and Q_k is the maximum capacity of a *cluster* k.
Variables	$y_{ik} = 1$ if the individual i is assigned to the *cluster* k or $y_{ik} = 0$, otherwise; and $x_{ij} = 1$ if the individuals i and j are in the same *cluster* or $x_{ij} = 0$, otherwise.

The MGD-HCCP can be proposed as following,

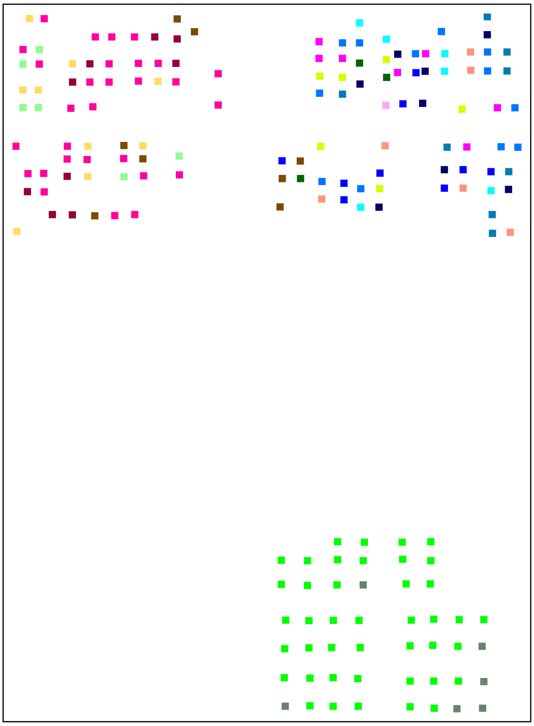


Fig. 6 Min–max diameter capacitated clustering instance and solution degeneration

$$(\mathbf{MGD-HCCP})\ Minimize \sum_{j\in I}\sum_{i\in I|i<j} d_{ij}x_{ij} \tag{25}$$

$$\text{such that}: \sum_{k\in M} y_{ik} = 1, \quad \forall i \in I \tag{26}$$

$$y_{ik} + y_{jk} - 1 \le x_{ij}, \quad \forall i, j \in I \mid i < j, \forall k \in M \tag{27}$$

$$\sum_{i\in I} q_i\ y_{ik} \le Q_k, \quad \forall k \in M \tag{28}$$

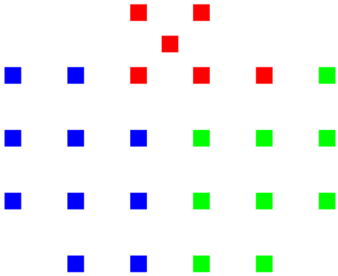
$$x_{i,j} \ge 0, \quad \forall i, j \in I \mid i < j \tag{29}$$

$$y_i^k \in \{0, 1\}, \quad \forall i \in I, \forall k \in M \tag{30}$$

The objective function (25) minimizes the total internal distance of the *clusters*. The constraint (26) indicates an item i may be assigned to a *cluster* k. The constraint (27) states that when items i and j belong to *cluster* k, then they share the same *cluster*. The constraint (28) considers the sum of individuals demand assigned to a *cluster* may not overpass its limited capacity. The constraints (29) and (30) refer to the decision variables of the problem.

The MGD-HCCP is NP-HARD. Its model is of O(n^2 m) in the number of constraints as for the number of binary variables. It is another good way of doing capacitated clustering, since it is also a linear mixed-integer formulation and the final groups are more compact than the dispersed ones resulted in the previous min-max diameter formulation.

(a) MGD-HCCP instance.



(b) MGD-HCCP associated optimal solution.

Fig. 7 MGD-HCCP instance and solution

MGD-HCCP may achieve high quality solutions, if the set of items are formed by compact groups where they have enough capacity to maintain individuals as close as possible, Fig. 7a. The groups, in this situation, may not overlap, and reasonable final geometric solutions may be obtained, as can be seen in Fig. 7b. In other direction, if the individuals are disperse and/or the capacity of the *clusters* introduces slackness, as observed in previous model the solution may result in bad undesirable clustering, Fig. 8.

To reduce the number of variables of this model, we introduce them as

$$\{x_{ij} \geq 0 \mid i < j, \forall i, j \in I\} \tag{31}$$

It implies a reduced value of the total cost of the formed cliques and limits the freedom to generate convex groups.

5 The heterogeneous capacitated centred clustering problem

In the original pCCCP model proposed by Negreiros and 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.

Let I be the set of individuals and J the set of cluster centers. We consider $|I| > |J|$. Given $a_i \in \mathbb{R}^{|K|}$, $i \in I$, q_i the demand of an individual $i \in I$, and Q_j the maximum capacity of a cluster $j \in J$. Where $K = \{1, 2, ..., k, ..., \alpha\}$, and $\alpha < +\infty$.

Lets consider as variables: $\bar{x}_j \in \mathbb{R}^{|K|}$, $j \in J$ represents the center of *cluster* $j \in J$; n_j an integer, is the number of individuals in *cluster* $j \in J$ and $y_{ij} \in \{0, 1\}$, where $y_{ij} = 1$ if i is an individual in *cluster* j, otherwise $y_{ij} = 0$.

The pHCCCP can be formulated as follows:

$$(\textbf{pHCCCP})\ Minimize \sum_{i \in I} \sum_{j \in J} ||a_i - \bar{x}_j||\ y_{ij} \tag{32}$$

$$\text{such that}: \sum_{j \in J} y_{ij} = 1, \quad \forall i \in I \tag{33}$$

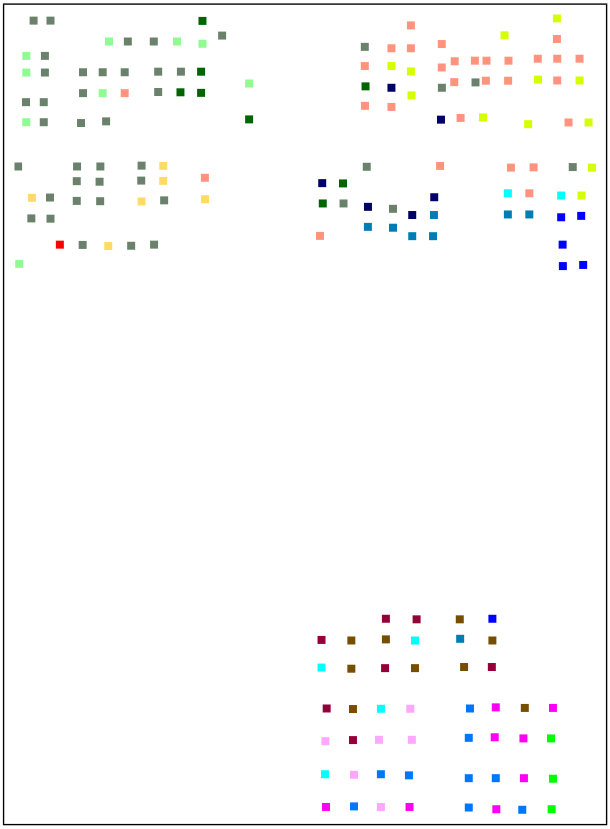


Fig. 8 MGD-HCCP capacitated clustering instance and solution degeneration in a dataprev instance

$$\sum_{i \in I} y_{ij} \leq n_j, \quad \forall j \in J \tag{34}$$

$$\sum_{i \in I} a_i \; y_{ij} \leq n_j \bar{x}_j, \quad \forall j \in J \tag{35}$$

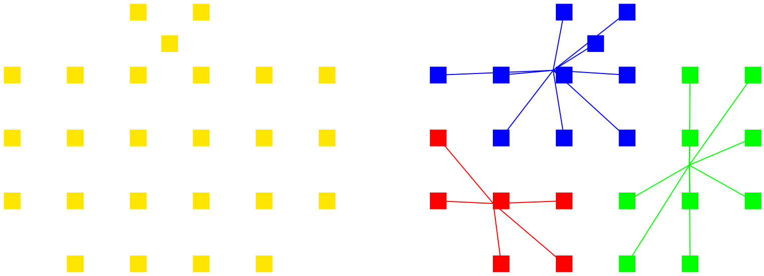
$$\sum_{i \in I} q_i \; y_{ij} \leq Q_j, \quad \forall j \in J \tag{36}$$

$$\bar{x}_j \in \mathbb{R}^{|K|}, n_j \in \mathbb{N} - \{0\}, \forall j \in J \tag{37}$$

$$y_{ij} \in \{0, 1\}, \quad \forall i \in I, \forall j \in J \tag{38}$$

The objective function (32) minimizes the Euclidean distance between *clusters* centers and their assigned item. The constraint (33) assigns one individual to just one *cluster*. The constraint (34) considers the number of individuals per *cluster*. The constraint (35) defines the geometric center of the *clusters*. The constraint (36) limits the assigned individuals to the maximum capacity of the *cluster* j. The constraints (37) and (38) refer to the decision variables of the problem.

pHCCCP is a NP-Hard problem as the previous problems, even its unconstrained version is NP-Hard, (Hansen and Jaumard 1997). The continuous relaxation of pHCCCP is non-convex and non-differentiable.



(a) Heterogeneous Capacitated Centred Clustering instance.

(b) HCCCP associated optimal solution.

Fig. 9 HCCCP instance and optimal solution

This model is $O(n^2|K|)$ in the number of binary variables and also in the number of constraints. It is necessary a non-linear-$\{0, 1\}$ solver to run this model. Solvers like FilMINT (Abhishek et al. 2010), Knitro and BARON from NEOS Server, and LINGO from LINDO Systems, can solve this model close to optimality for some instances, (Neos 2015; LINGO 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, see Fig. 9b.

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

5.1 A new model for pHCCCP

Without lost of generality we consider $\bar{a_i} \in \mathbb{R}_+^{|K|}$. Let $M = \max\{||\bar{a_i}\bar{a_l}|| \mid 1 \leq i < l \leq |K|\}$, and $a_i = \frac{\bar{a_i}}{M}, i \in I, a_i = (a_i^1 a_i^2 \ldots a_i^{|K|})^T$.

Consider the pHCCCP as following:

$$(\textbf{pHCCCP})\ Minimize \sum_{i \in I} \sum_{j \in J} ||a_i - \bar{x}_j||\ y_{ij} \tag{39}$$

$$\text{such that} : \sum_{j \in J} y_{ij} = 1, \quad \forall i \in I \tag{40}$$

$$\sum_{i \in I} y_{ij} \geq 1, \quad \forall j \in J (36) - -(38). \tag{41}$$

The constraints (34) and (35) can be placed as equality constrains:

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

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

$$\text{where,}$$

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

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

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

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

$$\begin{aligned}
(\mathbf{pHCCCP-LC})\ \ \text{Minimize}\ & \textstyle\sum_{i \in I} \sum_{j \in J}\ ||a_i - \bar{x}_j||\ y_{ij} \\
\text{such that :}\quad & \textstyle\sum_{j \in J} y_{ij} = 1,\quad \forall i \in I \\
& \textstyle\sum_{i \in I} y_{ij} \geq 1,\quad \forall j \in J \\
& \textstyle\sum_{i \in I} a_i^k\ y_{ij}\ =\ \sum_{i \in I} t_{ij}^k, \\
& \forall j \in J, \forall k \in K && (42) \\
& 0 \leq\ t_{ij}^k\ \leq y_{ij}^k, \forall j \in J, \forall k \in K && (43) \\
& (y_{ij} - 1) + \bar{x}_j^k \leq t_{ij}^k \leq \bar{x}_j^k + (1 - y_{ij}), \\
& \forall i \in I, \forall j \in J, \forall k \in K \\
& \textstyle\sum_{i \in I} q_i\ y_{ij}\ \leq\ Q_j, \forall j \in J \\
& \bar{x}_j^k \in \mathbb{R}, \forall j \in J, \forall k \in K \\
& y_{ij} \in \{0, 1\}, \forall j \in J && (44)
\end{aligned}$$

The constraints (42), (43) and (44) warranty the optimal solution $\bar{x}_j^\star, \forall j \in I$, 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 (38) 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 and Maculan (2018) we can consider the objective function (32) 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 $s_{ij}^k = (a_i^k - \bar{x}_j^k)$.

Since $0 \leq a_i^k \leq 1$ 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 $\lambda > 0$, as for example $\lambda = 10^{-8}$. The following model is convex, differentiable with linear continuous relaxation in the constraints as shown before. The key to turn this model competitive in relation to the basic model for the pHCCCP is to find $0 \leq a_i^k \leq 1, \forall i \in I, \forall k \in K$.

$$(\mathbf{pHCCCP-CDC})\ \textit{Minimize} \sum_{i \in I} \sum_{j \in J} \sqrt{\sum_{k \in K} (s_{ij}^k)^2 + \lambda^2} \tag{45}$$

$$\text{such that :} \sum_{j \in J} y_{ij} = 1,\ \forall i \in I \tag{46}$$

$$\sum_{i \in I} y_{ij} \geq 1,\ \forall j \in J \tag{47}$$

$$\sum_{i \in I} a_i^k \; y_{ij} \; = \; \sum_{i \in I} t_{ij}^k, \; \forall j \in J, \; \forall k \in K \tag{48}$$

$$0 \leq \; t_{ij}^k \; \leq y_{ij}, \; \forall i \in I, \; \forall j \in J, \; \forall k \in K \tag{49}$$

$$(y_{ij} - 1) + \bar{x}_j^k \leq t_{ij}^k, \; \forall i \in I, \; \forall j \in J, \; \forall k \in K \tag{50}$$

$$t_{ij}^k \leq \bar{x}_j^k + (1 - y_{ij}), \; \forall i \in I, \; \forall j \in J, \; \forall k \in K \tag{51}$$

$$\sum_{i \in I} q_i \; y_{ij} \; \leq \; Q_j, \; \forall j \in J \tag{52}$$

$$(y_{ij} - 1) + (a_i^k - \bar{x}_j^k) \leq s_{ij}^k, \; \forall i \in I, \; j \in J, \; k \in K \tag{53}$$

$$s_{ij}^k \leq (1 - y_{ij}) + (a_i^k - \bar{x}_j^k), \; \forall i \in I, \; j \in J, \; k \in K \tag{54}$$

$$- y_{ij} \leq s_{ij}^k \leq y_{ij}, \; \; \forall i \in I, \; \forall j \in J, \; \forall k \in K \tag{55}$$

$$\bar{x}_j^k \in \mathbb{R}, \; \forall j \in J, \; \forall k \in K \tag{56}$$

$$y_{ij} \in \{0, 1\}, \; \forall i \in I, \; \forall j \in J \tag{57}$$

$$t_{ij}^k \geq 0, \; \forall i \in I, \; \forall j \in J, \; \forall k \in K \tag{58}$$

$$s_{ij}^k \in \mathbb{R}, \; \forall i \in I, \; \forall j \in J, \; \forall k \in K \tag{59}$$

The continuous relaxation of pHCCP-CDC is a convex and differentiable optimization problem.

6 A metaheuristic framework for the HCCCP

A framework of metaheuristic was specially developed to the HCCCP based on another method using Multistart solutions and Tabu Search with Path Relinking, designed by Muritiba et al. (2012).

The Multistart method (pRBFC), 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 apparently do 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, (Muritiba et al. 2012).

Once built a random solution the method 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. Along the process a pool of elite ($\approx$ 10 best) solutions are also incorporated to the framework that is treated by a *forward-based* path relinking method, (Resende and Ribeiro 2016).

For the HCCCP, (Batista et al. 2015) proposed to the previous framework an adjustment to include a new step in the process of building multistart feasible solutions. It is applied a Generalized Assignment Problem (GAP) solver to generate an improved optimal/near optimal solution for the actual assigned centers to the HCCCP instance in evaluation. In this version, it was removed the path relinking process. Considering the framework schema represented in Figure 10, the Multistart method is combined with KBLGAP exact method proposed in Karabakal (1992) or (Martello and Toth 1990), for the Generalized Assignment

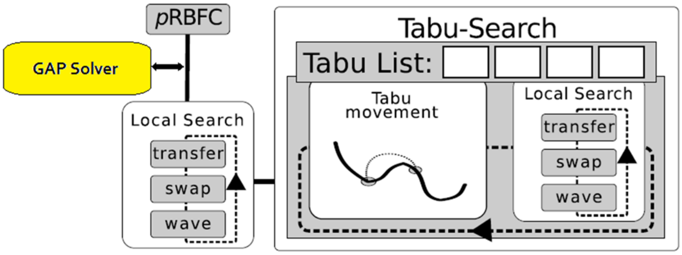


Fig. 10 Metaheuristic framework schema developed to HCCCP by Batista et al. (2015)

Problem (GAP) and to arrange the vertices in *clusters* using a distance matrix 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 Lagrangean method (Karabakal 1992), the best upper-bounds can be obtained for moderate size instances (1,000 individuals). Using an approximate GAP heuristic (Martello and 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 *clusters* centers.

7 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. Another 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.

Each instance was solved for the models here indicated. The HCCCP problem is non-linear and combinatorial, three other models are linear mixed-integer. We evaluated the instances using Argonne NEOS-Server Platform and its solvers to solve them, (Neos 2015). We run the mixed integer linear models (gHCMP, MMD, MGD) using the solvers: Gurobi (7.5.1), CPLEX (12.7.0.0), Xpress-MP (29.01) and LINGO (v17). For the nonlinear mixed integer HCCCP model we used the solvers: FilMINT (AMPL) (v0.0000000001), BARON (16.7.29) and Knitro (10.2.0). In Table 2, it is indicated in the columns related to each problem: the solution value of the objective function (OF), the time spent in seconds to obtain the solution (Time (s)), and the solver used to solve the instance (Solve).

To the linear models (MMD-HCCP) and (MGD-HCCP) their solutions are said to be *optimal solution* for Gurobi and *optimal integer solution within mipgap or absmipgap* for CPLEX, while Xpress indicated *Global search incomplete. Best integer solution found*. In the non-linear pHCCCP model, the solvers indicated the solutions are *optimal* for the tested instances.

7.1 MIN-MAX distance model

As can be seen in Table 2 the solutions of the model with min-max distance were solved to optimality for the first two instances, being CPLEX the solver that returned the best performance. For the other five bigger instances we obtained an upper-bound because each solver reached its running limit time. In any case the instance 6 was solved to optimality (proved by Gurobi). Figure 11 shows the best solution layout for instance DPV-2016, and Fig. 12 shows the best solution obtained for instance SERPRO-2017B.

As can be noticed in both cases, the model MMD-HCCP does not fit to be used to IT-Teams layout in these factories, once the groups degenerate and are distributed along the floors and pavilion of Dataprev and Serpro, although some of the bigger groups in both cases appears well distributed. The results indicated that the final layouts may not reach the desirable groups compaction.

7.2 MGD-HCCP

Table 2 shows the solutions of the model with minimum total distance of groups were solved to optimality for the first two instances as did the previous model by CPLEX, Gurobi and Xpress solvers, with Xpress returning the best performance. For the other five bigger instances we also obtained an upper-bound because of the solvers running time limit. Figure 15 shows the best solution layout for Serpro obtained for instance SERPRO-2017A from CPLEX, and Fig. 16 shows the best solution obtained for instance SERPRO-2017B obtained from Xpress.

As can be noticed in both cases, the model MGD-HCCP fits well to be used to layout of IT-Teams at Serpro and Dataprev software factories but not for one case of Dataprev (see Fig. 13), once the groups degenerate and are distributed along their floors. The third case of Dataprev instance (DPV-2017B) the groups are well distributed, see Fig. 14, but some of them are inside or touch other groups. While in Serpro instances both cases appear well distributed and the teams appear close, many of them shaped like a snip. The results indicated that the final layouts may reach desirable group compaction, but just for Serpro instances.

Table 1 Instances features

Lable	Instance	$\|N\|$	$\|M\|$
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

Label	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 2 Best results obtained from the solvers to each problem and their instances

Instance	MMD-HCCP			MGD-HCCP		
	OF	Time(s)	Solver	OF	Time(s)	Solver
Test 1	316.227	0.004	cplex	14254.936	0.066	xpress
Test 2	360.555	0.019	cplex	16060.498	0.180	cplex
DPV-2016	738.002	11794.600	gurobi	797980.386	11122.300	gurobi
DPV-2017A	186.904	122.213	cplex	70457.500	67.000	cplex
SERPRO-2017A	497.177	10858.900	gurobi	1892651.210	59.581	cplex
DPV-2017B	174.100	795.702	gurobi	119154.862	2761.01	gurobi
SERPRO-2017B	350.157	12138.100	gurobi	684427.252	16.8354	xpress

Instance	pHCCCP		
	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	35.170	knitro
SERPRO-2017A	31295.430	0.088	filmint
DPV-2017B	6465.150	0.148	knitro
SERPRO-2017B	19850.854	0.470	filmint

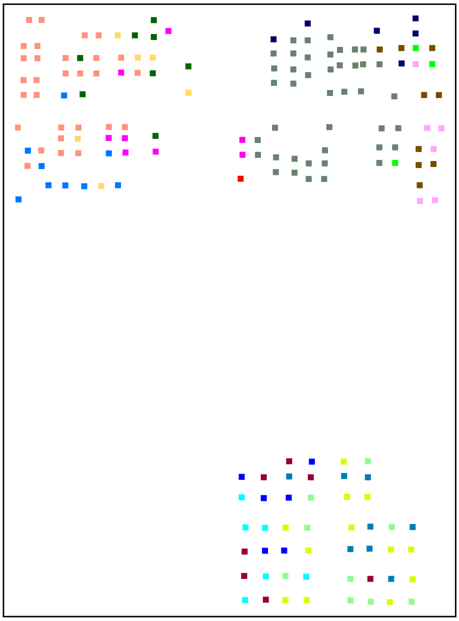


Fig. 11 Best solution for DPV-2016

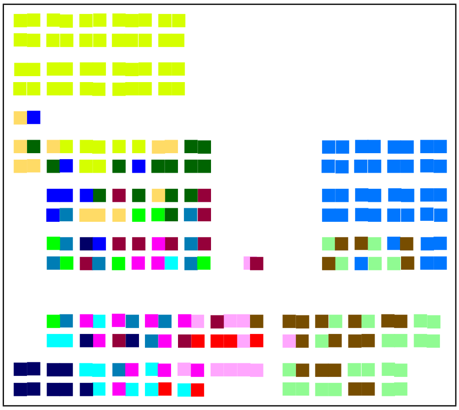


Fig. 12 Best solution for SERPRO-2017B

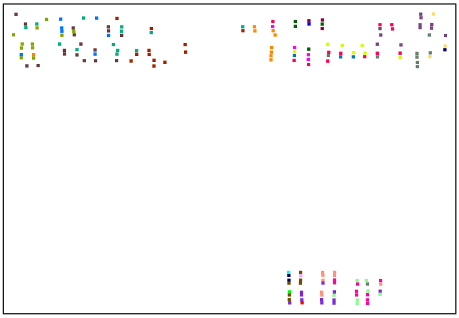


Fig. 13 Best solution for DPV-2016 using **MGD-HCCP** model with CPLEX

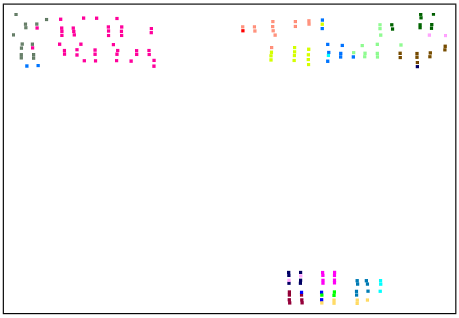


Fig. 14 Best solution for DPV-2017B using **MGD-HCCP** model

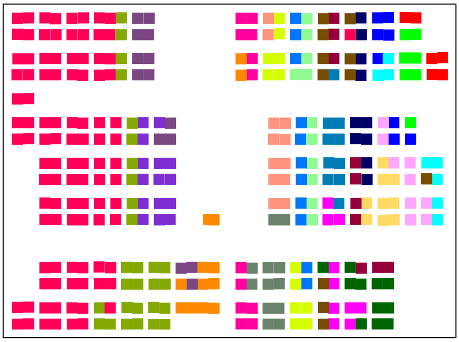


Fig. 15 Best solution for SERPRO-2017A

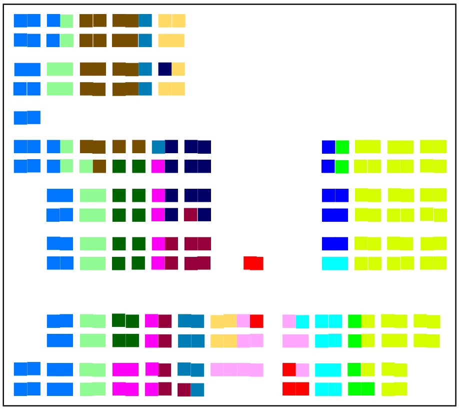


Fig. 16 Best solution for SERPRO-2017B

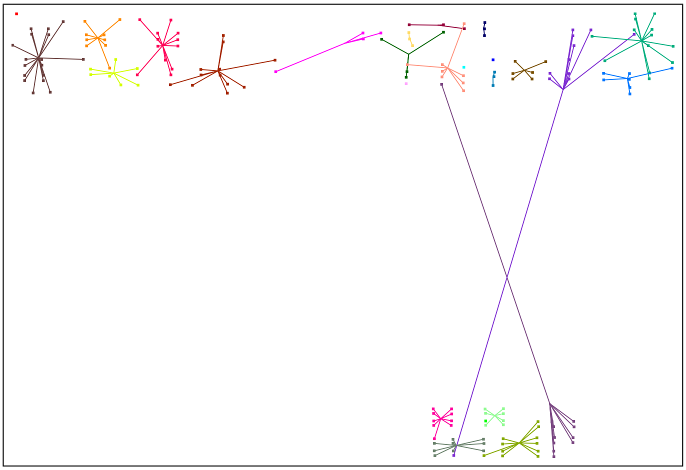


Fig. 17 Solution for DPV-2017A—Knitro

7.3 pHCCCP

The pHCCCP solutions, presented in Table 2, were solved with great difficulty by the solvers used. In general FilMINT solved the instances quite well but not at optimal solution for 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 17 and 18 show the best solution layout for Dataprev obtained for instance DPV-2017A from Knitro, and for instance DPV-2017B both obtained from Knitro. It can be noted that Knitro solver obtained good layouts, although in the DPV-2017A two missing workstations are dividing two groups in the two floors.

Figures 19 and 20 show the best solution layout for Serpro obtained for both instances SERPRO-2017A and SERPRO-2017B from FilMINT. It is noticeable that the solver obtained compact layouts. For both cases the pHCCCP model is adequate and fit well the evaluation of the process. The solutions are quite practical and can be directly implemented by the IT-managers.

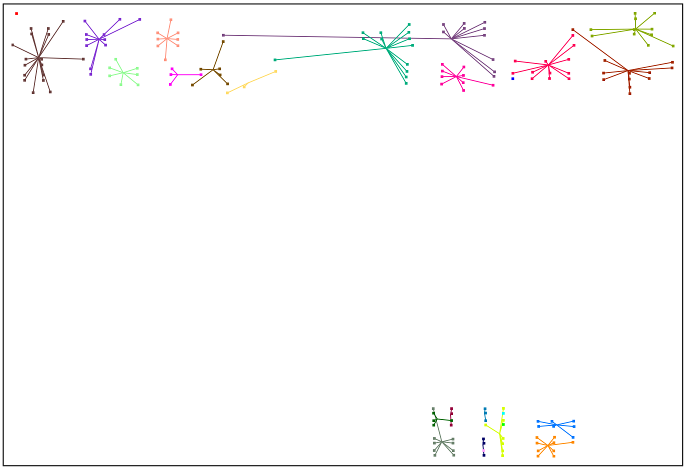


Fig. 18 Best solution for DPV-2017B —Knitro

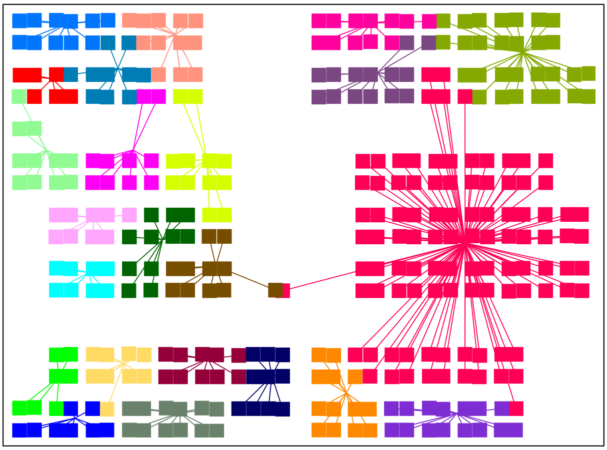


Fig. 19 Best solution for SERPRO-2017A—pHCCCP

Although its inherent complexity, the CCCP problem is solved close to optimality by heuristic methods with very good upper bounds, Chaves and Lorena 2010, 2011; Muritiba et al. 2012. Batista et al. (2014) and most recently Chaves et al. (2018) showed the results of the best known heuristics in comparison to the NEOS solvers resolution for the homogeneous version, where the heuristic results are better than the solvers. The solvers achieved at most the best upper bound already obtained by CCCP heuristics from the benchmark instances.

In Table 3 we consider the comparison of the solutions obtained between the best solver upper-bound and the metaheuristic results for the pHCCCP. Column Gap% indicates the percentage difference between the Objective (OF) results. When the value is negative means gap is favorable to the Metaheuristic and positive to the Solver.

In the experiments here evaluated, the pHCCCP model can also be used with FilMINT and Knitro with reasonable upper bounds at the end for the majority of the instances. Their results are close to the metaheuristic results in cost, however in time to obtain the solution they are very distant. FilMINT obtained the only best upper bound for solvers for instance SERPRO-2017A, with 1.48% gap from metaheuristic result. It is noticeable the regular superiority of

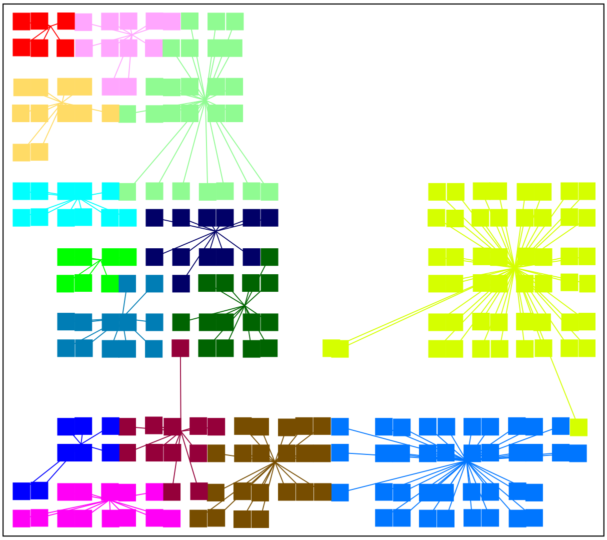


Fig. 20 Best solution for SERPRO-2017B—pHCCCP

Table 3 Best results obtained from solvers and Metaheuristic to pHCCCP instances

Instance	pHCCCP		Solver	OF	Time(s)	Gap%
	OF	Time(s)				
Test 1	2199.185	0.046	filmint	2187.276	4.053	–0.54
Test 2	2774.545	0.063	filmint	2704.303	0.975	–2.53
DPV-2016	31053.900	48.992	knitro	30495.406	7.071	–1,79
DPV-2017A	6916.457	35.000	knitro	4612.834	2.777	–33.31
SERPRO-2017A	31295.430	0.088	filmint	31761.471	13.947	1.48
DPV-2017B	6465.150	0.148	knitro	5673.703	2.933	–12.24
SERPRO-2017B	19850.854	0.471	filmint	19740.297	27.465	–0.55

the results obtained from heuristics in comparison to solvers. The solution obtained by the heuristic method for instance DPV-2017A is much better (33.31%) than the one obtained by the solver Knitro (the best between solvers), Fig. 21. In Fig. 22 we have the final configuration of the best known result for instance SERPRO-2017B, obtained from pHCCCP metaheuristic.

It can be noted that the solutions obtained for the gHCMP and pHCCCP can be compared in cost. gHCMP can be used as an upper bound for the pHCCCP as indicated by Negreiros and Palhano (2006), but here we can find the dominance of gHCMP for the solutions of the five last higher size instances evaluated, but taking much longer time consuming.

7.4 gHCMP

For the gHCMP all instances were solved using the proposed model but for some selected solvers: CPLEX, 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 4 shows the dominance in time of Xpress in obtained the solutions over all other solvers

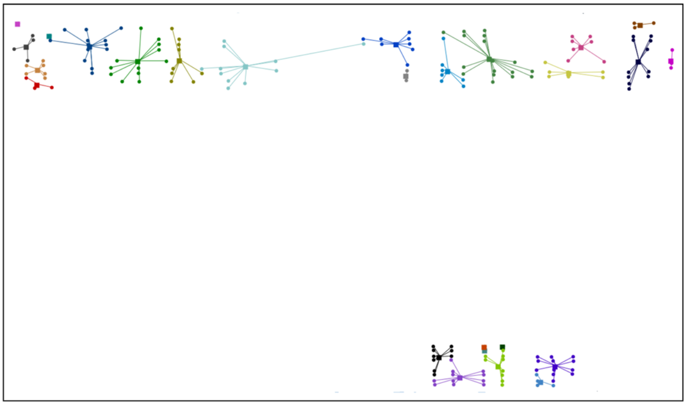


Fig. 21 Best known solution obtained by HCCCP metaheuristic to DPV-2017A

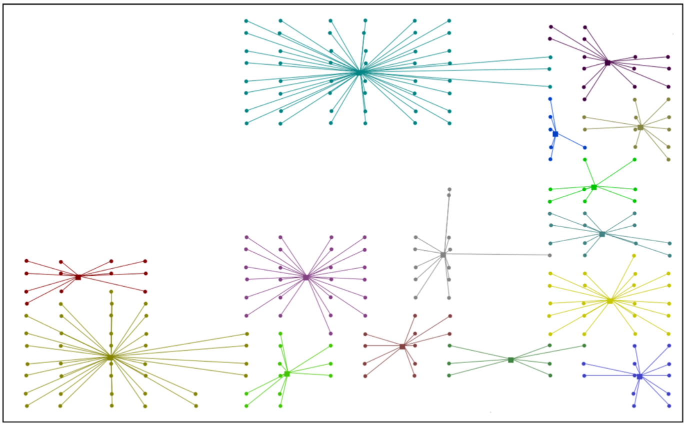


Fig. 22 Best known solution obtained by HCCCP metaheuristic to SERPRO-2017B

used. The solutions indicated as (*) are the non optimal solutions, the others are the optimal solutions obtained from each solver.

Figures 23 and 24 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-2017A.

Figures 25 and 26 show the best solution layout for Serpro obtained for both instances SERPRO-2017A from Gurobi and SERPRO-2017B from Xpress. 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.

Table 4 Best results obtained from the solvers to each problem and their instances

Instance	XPRESS		CPLEX		GUROBI		LINGO	
	OF	Time(s)	OF	Time(s)	OF	Time(s)	OF	Time(s)
Test 1	2207.11	0.0089	2207.11	0.0109	2207.11	0.1299	2207.1	23.
Test 2	2736.7	0.0059	2736.7	0.0099	2736.7	0.0129	2736.7	34:39.
DPV-2016	27435.8	6.4170	27435.8	39.773	27435.8	431.365	NFSF	>1h
DPV-2017A	3706.19	3.5624	NFSF	> 1h	3706.19	114.609	NFSF	>1h
SERPRO-2017A	30596.3*	30.7853	31316.8*	44.5862	30430.8*	11812.8	NFSF	>1h
DPV-2017B	5249.65*	5.6214	NFSF	> 1h	5249.65*	393.65	NFSF	>1h
SERPRO-2017B	19185	36.6914	19418.8	49.4195	NFSF	>1h	NFSF	>1h

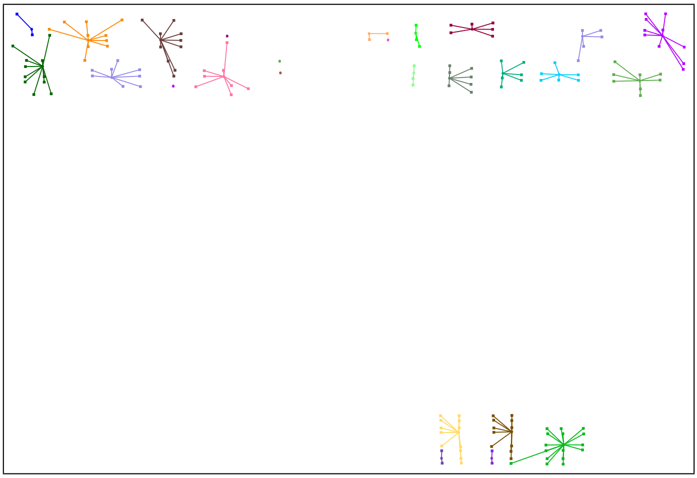


Fig. 23 Best solution for DPV-2017A using gHCMP model

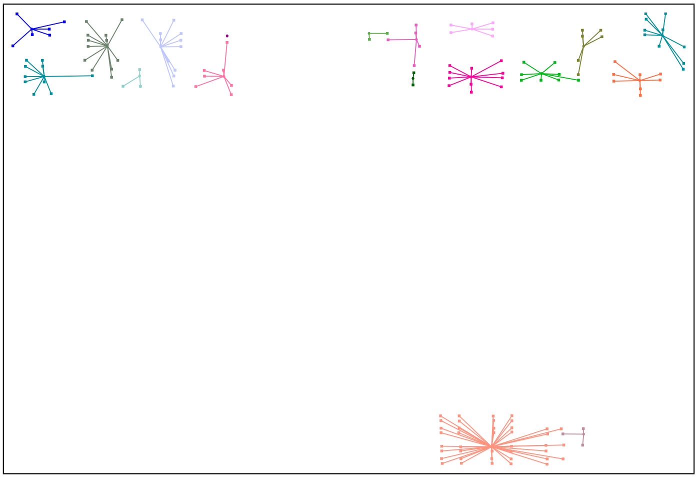


Fig. 24 Best solution for DPV-2017B using gHCMP model

8 Conclusions

The capacitated clustering models can be adjusted to achieve different objectives. Adaptations on the basic models are needed to achieve the best way to reproduce the desired solution. Here we showed four ways of doing capacitated clustering, where the *clusters* capacity may differ. We introduced and evaluated solutions obtained for models related to medians, in this case one wishes to extract from the set of individuals the *clusters* and their centers; most compact groups, where the formulation consider only the groups and the best partition; and a new alternative formulation for the heterogeneous capacitated centred clustering problem.

The gHCMP, MMD-HCCP and MGD-HCCP are linear mixed-binary models, their complexity are highly dependent on the sets of *clusters* and items. The first model was tested in the set of instances of IT-Teams and achieved solutions for the heterogeneous cases and it is

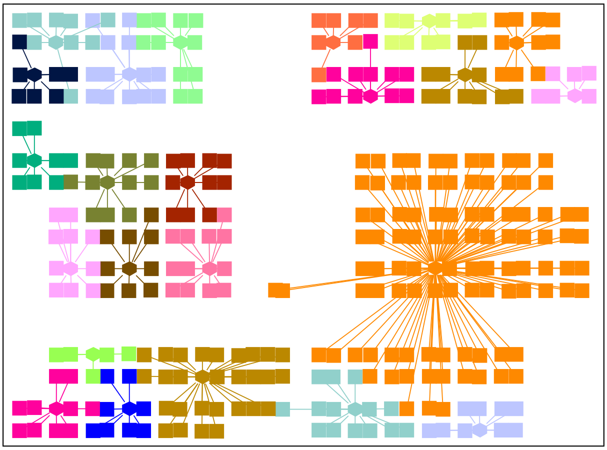


Fig. 25 Best solution for SERPRO-2017A—gHCMP

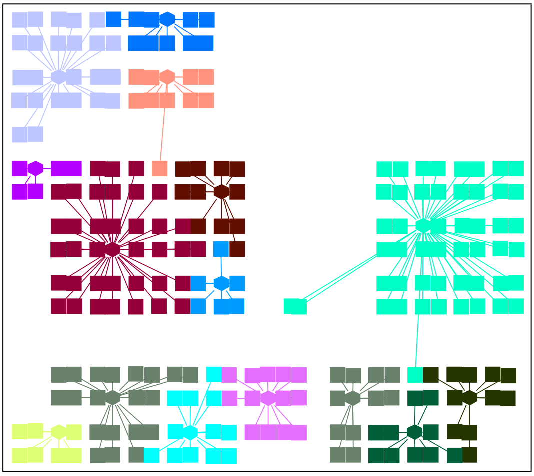


Fig. 26 Best solution for SERPRO-2017B—gHCMP

fully appropriate to be used in the IT-Teams context. The second and third model are shorter and may be convenient for many reasons. For the instances arisen from layout IT-teams in a software factory the solvers worked well in the majority of the situations, but dispersion of groups were annotated. It is noticeable the degeneration of the solutions for the real cases, showing that they are not appropriate for the application in study.

MGD-HCCP showed to be appropriate for layout purposes in situations like at Serpro. The cases with two floors as proposed in the same plane but far apart each other may not be of adequate use.

For the case of the *p*HCCCP the model is more complex and must be solved using nonlinear mixed integer solvers. Although the number of constraints and binary variables are less than the previous models, the proposed model for the pHCCCP may be better solved by a metaheuristic method because the solvers are for differentiable and convex problems. The optimality proof obtained by the evaluated solvers is not always correct, due to the pHCCCP basic model being non differentiability of the objective function and the non convexity of some of its constraints. The use of the proposed metaheuristic obtains superior results for most instances evaluated in reasonable time.

The gHCMP model is more adequate to solve the IT-Teams layout problem as proposed, and the solver Xpress obtained the best results for the major set of instances here evaluated for this model.

Acknowledgements We would like to thank CNPq for Grant 481919/2013-9, which made this research possible, and GRAPHVS Ltda. for providing the support and resources necessary to perform this research. We also thank Frederico Roldan from Serpro and IT-Teams from Dataprev for their kind support in this research.

References

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

Allen, T. J. (1977). Managing the flow of technology: Technology transfer and the dissemination of technological information within the R&D organisation. Cambridge: MIT Press.

Batista, P., Negreiros, M., Muritiba, A., Palhano, A. (2015). New framework of metaheuristics for the capacitated centred clustering problem. In *Preceedings of XI MIC, Metaheuristic International Congress, 2015*.

Batista, P., Negreiros, M., Palhano, A. (2014). Solvers resolution perspective to the capacitated centred clustering problem. In *Annals of XLVI SBPO, SOBRAPO* (pp. 2458–2468).

Brimberg, J., Mladenovic, N., Todosijevic, R., & Urosevic, D. (2018). Solving the capacitated clustering problem with variable neighborhood search. *Annals of Operations Resesearch*, *272*, 289–321.

Brito, J., Brito, L. (2008). Algoritmos vns e genéticos aplicados ao problema de agrupamento com soma mínima. In *Annals of XL SBPO, SOBRAPO* (pp. 1150–1161).

Brito, J., Fadel, A., Semann, G., Brito, L. (2017). Algoritmo gentico de chaves aleatrias viciadas aplicado ao problema de agrupamento com soma mínima. In *Annals of XLIX SBPO, SOBRAPO* (pp. 2325–2336).

Chaves, A., Goncalves, J., & Lorena, L. (2018). Adaptive biased random-key genetic algorithm with local search for the capacitated centered clustering problem. *Computers & Industrial Engineering*, *124*, 331–346.

Chaves, A. A., & Lorena, A. (2010). Clustering search algorithm for the capacitated centred clustering problem. *Computers and Operations Research*, *37*, 552–558.

Chaves, A. A., & Lorena, A. (2011). Hybrid evolutionary algorithm for the capacitated centered clustering problem. *Expert Systems with Applications*, *38*, 5013–5018.

Czyzyk, J., Mesnier, M. P., & Moré, J. J. (1998). The neos server. *IEEE Journal on Computational Science and Engineering*, *5*(3), 68–75.

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

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

Fourer, R., Gay, D., & Kernigham, B. (1993). *AMPL: A modeling language for mathematical programming*. Boyd & Fraser Publishing.

Gropp, W., & Moré, J. J. (1997). Optimization environments and the neos server. In M. D. Buhman & A. Iserles (Eds.), *Approximation Theory and Optimization* (pp. 167–182). Cambridge: Cambridge University Press.

Hansen, P., & Jaumard, B. (1997). Cluster analysis and mathematical programming. *Mathematical Programming*, *1*, 191–215.

Karabakal, N. (1992). *Kblgap: A C code for solving the generalized assigment problem, Technical Report 92-32*. University of Michigan.

Lewis, M., Wang, H., & Kochenberger, G. (2014). Exact solutions to the capacitated clustering problem: A comparison of two models. *Annals of Data Science*, *1*, 15–23.

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

Martello, S., & Toth, P. (1990). *Knapsack Problems. Algorithms and Computer Implementations*. Amsterdam: Wiley.

Martinez Gavara, V., Campos, M., Gallego, M., & Laguna, R. Mart. (2015). Tabu search and grasp for the capacitated clustering problem. *Computer Optimization and Appliccations*, *62*, 589–607.

Moran-Mirabal, J., Gonzlez-Velarde, M. G., & Resende, R. M. (2013). Silva, Randomized heuristics for handover minimization in mobility networks. *Journal of Heuristics*, *19*, 845–880.

Mulvey, J., & Beck, M. (1984). Solving capacitated clustering problems. *European Journal of Operational Research*, *18*, 339–348.
Muritiba, A., Negreiros, M., Souza, M., Oria, H. (2012). A tabu search for the capacitated centred clustering problem. In *Annals of SBPO/CLAIO 2012, ALIO/SOBRAPO, 2012* (pp. 2344–2359).
Negreiros, M., Palhano, A. (2003). Uma aplicao para o problema generalizado de percurso de veculos. In *Annals of the XXV SBPO-Natal/RN, SOBRAPO, 2003*.
Negreiros, M., & 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., Maculan, N. (2018). Mixed integer non linear programming (minlp) models for the euclidean steiner tree problem in rn, Technical Report - LIP6,UPMC - Paris (p. 8).
Pillai, A., Chick, J., Johanning, L., Khorasanchi, M., & de Laleu, V. (2015). Offshore wind farm electrical cable layout optimization. *Engineering Optimization*, *1*, 1689–1708.
Prata, B. (2015). The multi capacitated clustering problem, Technical report, Universidade Federal do Ceará.
Rao, M. R. (1971). Cluster analysis and mathematical programming. *Journal of the American Statistical Association*, *66*(335), 622–626.
Resende, M., & Ribeiro, C. (2016). *Optimization by GRASP: Greedy randomized adaptative search procedures*. Berlin: Springer.

Affiliations

Marcos J. Negreiros[1] · Nelson Maculan[2] · Pablor L. Batista[1] · João A. Rodrigues[1] · Augusto W. C. Palhano[1]

✉ Marcos J. Negreiros
marcos.negreiros@uece.br

Nelson Maculan
maculan@cos.ufrj.br

Pablor L. Batista
opablofernandes@hotmail.com

João A. Rodrigues
joaoamilcar@gmail.com

Augusto W. C. Palhano
augustopalhano@graphvs.com.br

1 Universidade Estadual do Ceará, Mestrado Profissional em Computação Aplicada (MPCUMP-UECE-IFCE) Laboratório de Computação Científica (LCC), Av Silas Munguba, 1700, Campus do Itaperi, Fortaleza, CE CEP 60714.903, Brazil

2 Universidade Federal do Rio de Janeiro Programa de Eng de Sistemas e Computação (PESC-COPPE), Ilha do Fundão, Sala H-322B, Rio de Janeiro, RJ 21.941-972, Brazil