RAIRO-Oper. Res. 54 (2020) 873–882
https://doi.org/10.1051/ro/2019041

**RAIRO Operations Research**
www.rairo-ro.org

# NEW PROPOSALS FOR MODELLING AND SOLVING THE PROBLEM OF COVERING SOLIDS USING SPHERES OF DIFFERENT RADII

Pedro Henrique González Silva[1,4,*], Ana Flávia U. S. Macambira[2], Renan Vicente Pinto[3], Luidi Simonetti[4], Nelson Maculan[4] and Philippe Michelon[5]

**Abstract.** Given a solid $T$, represented by a compact set in $\mathbb{R}^3$, the aim of this work is to find a covering of $T$ by a finite set of spheres of different radii. Some level of intersection between the spheres is necessary to cover the solid. Moreover, the volume occupied by the spheres on the outside of $T$ is limited. This problem has an application in the planning of a radio-surgery treatment known by Gamma Knife and can be formulated as a non-convex optimization problem with quadratic constraints and linear objective function. In this work, two new linear mathematical formulations with binary variables and a hybrid method are proposed. The hybrid method combines heuristic, data mining and an exact method. Computational results show that the proposed methods outperform the ones presented in the literature.

**Mathematics Subject Classification.** 90C10.

Received September 27, 2017. Accepted April 7, 2019.

## 1. Introduction

Covering problems are often related to the determination of the position of certain objects in order to cover the area or volume of a greater object. To achieve this objective, a level of intersection between the objects is allowed. This class of problems is naturally formulated as a minimization problem. A classical application of covering a two-dimensional region with circles arises in the field of telecommunications. In this context, the equipment covers a circular geographic area and the objective is to attend or to cover a city by installing the smallest amount of equipments.

Another class of problems that is closely related to the covering problems is known as packing problems. In fact, packing and covering are a pair of primal $x$ dual problems. Packing problems consists in positioning the maximum number of objects inside a bigger object, often called container. The objects cannot overlap. A classical application of the packing problem is stacking the maximum number of oranges in a box.

*Keywords.* Problem of covering solids, mathematical programming, hybrid method.

[1] Departamento de Informática, Centro Federal de Educação Tecnológica Celso Suckow da Fonseca, Brazil.

[2] Departamento de Estatística, Universidade Federal da Paraíba, Brazil.

[3] Departamento de Matemática, Universidade Federal Rural do Rio de Janeiro, Brazil.

[4] Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, Brazil.

[5] Laboratoire de Mathématiques d'Avignon, Université d'Avignon et des Pays de Vaucluse, France.

[*]Corresponding author: pehgonzalez@gmail.com; pegonzalez@eic.cefet-rj.br

Article published by EDP Sciences

The problem of covering a tridimensional region with spheres of different radii is presented in this work. This problem combines elements of the two problems described above. It consists in covering the maximum volume of a bigger object, while using the fewer number of the smaller objects as possible. Intersection between them is allowed, because if the spheres only touch each other, there will probably be several uncovered parts of the tridimensional region. Important applications of this problem are the planning treatments of cerebral tumors, named stereotactic radio-surgery. One of the most effective treatment is known as *Gamma Knife* [2]. In this treatment, an equipment is placed in the head of the patient and *shots* of radiation are issued. The region of the cerebral tumor affected by one *shot* can be mathematically approximated by a spherical region. As the equipment has different sizes, its area of effect can be modeled as spheres of 2, 4, 7 or 9 mm radii. The process of planning this treatment is tedious, time-consuming and is highly dependent on the experience of the responsible professional. For these reasons, there is a need for the automation of the planning process of *Gamma Knife* treatment, aiming to find a good covering of the tumor in an acceptable time.

Some authors approach the *Gamma Knife* problem as a packing problem [10] and two heuristics were proposed to find a feasible solution and then a *Branch and Bound* technique was applied. In [1], besides a heuristic to find a good starting point to the problem, the authors used an inexact penalization method to remodel the constraints and the least squares method was used to estimate parameters. In [5], at first, a nonlinear non-convex mixed integer model was proposed. The solid was discretized and five reformulations of the first model were proposed. Both the nonlinear non-convex mixed integer and the non-linear convex mixed integer models were solved with a Variable Neighborhood Search method and the linear mixed integer reformulations were solved by the commercial package CPLEX. In [8], the author used an approach based on Graph Theory. Maximum weight cliques were found to generate cuts in a *Branch and Cut* scheme. In [11], the proposed model was solved by penalization techniques and stochastic search heuristics. In [7], a nonlinear model is presented and it is solved by the hyperbolic suavization technique.

The contributions given by this work are the proposal of two linear models with integer variables and a hybrid method to solve the spheres covering problem. The hybrid method combines heuristics, data mining and one of the mathematical models proposed in this work.

This work is organized as follows: in Section 2, the mathematical formulations are presented. In Section 3, the proposed hybrid method is detailed. Section 4 presents the computational results. Besides this, in this section, a comparison is made between the results of the hybrid method, the mathematical formulation and the results of the linear model proposed by [8]. Finally, Section 5 presents the concluding remarks and future works.

## 2. Mathematical programming models

In this Section, the problem is mathematically described. Following this, comes the explanation of the discretization, a procedure used in order to convert the continuous set, given by the solid, in a discrete mesh with a finite set of points. The points of the grid that results from the discretization are sphere centers candidates, which are used by the two models proposed in this work.

Let $S$ be a finite set of spheres of different sizes. The (CSSDRP) – Covering Solids with Spheres of Different Radii Problem, consists of covering a solid $T \subset \mathbb{R}^3$ using spheres from $S$. A total covering occurs when each point $p \in T$ belongs also to some sphere $s \in S$. Intersection between spheres may be necessary, otherwise an amount of points will not be covered by any sphere. It is important to highlight that setting a single parameter to control the intersection between the spheres may forbid a total covering.

Mathematically, the (CSSDRP) can be described as: given a compact set $T \subset \mathbb{R}^3$, a finite set of different radii $R \subset \mathbb{R}_+$, a finite set $N \subset \mathbb{N}$ of indexes of spheres, a function $o : N \to \mathbb{R}^3$ that associates each sphere $i \in N$ to its center $(x_i, y_i, z_i)$, and a function $r : N \to R$ that associates each sphere $i \in N$ to its radius, find a set of spheres $\{B(o(i), r(i)) \mid i \in N\}$, covering each point of $T$, respecting a limit of intersection between the spheres and also respecting a limit of occupation of the external region of $T$ by the spheres.

Let us notice $o_i \in T$, $i \in N$, the decision variables corresponding to $o(i)$, and by $r_i \in \mathbb{R}$ the radius values corresponding to $r(i)$. The nonlinear non-convex model parameterized by $\{c_i, \alpha_i\}_{i \in N}$ and presented below was

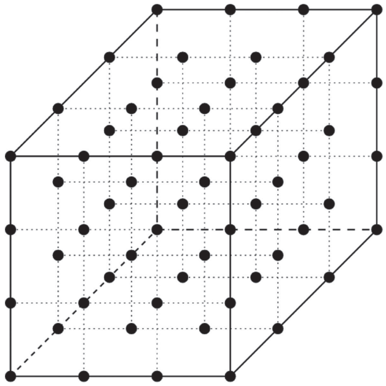

FIGURE 1. Example of general parallelepiped discretization.

proposed by [8].

$$\max \quad \sum_{i=1}^{n} c_i \, y_i \tag{2.1}$$

$$\text{s. t} \quad ||o_i - o_j||^2 \geq (r_i + r_j - \alpha_{r_i r_j})^2 \, (y_i + y_j - 1), \qquad \forall\, 1 \leq i < j \leq n \tag{2.2}$$

$$o \in T^n \tag{2.3}$$

$$y \in \{0,1\}^n. \tag{2.4}$$

The binary variable $y_i$ defines whether or not sphere $i$ is in the solution, $c_i$ are real positive parameters and throughout this paper $\alpha_{r_i r_j}$ is defined as:

$$\alpha_{r_i r_j} = \frac{\min(r_i, r_j)}{2}. \tag{2.5}$$

Constraints (2.2) sets a limit to the intersection between spheres $i$ and $j$, controlled by the parameter $\alpha_{r_i r_j}$. Constraints (2.3) can be modified to limit the occupation of the external volume of $T$ by the spheres.

## 2.1. Discretization of the volume of the solid

The non convexity presented by constraints (2.2) of model (2.1)–(2.4) can be avoided by applying the procedure of discretization of solid $T$ (see Fig. 1). In this work we choose to use a parallelepiped as base of the discretization. Due to this, given a parameter $\delta$, a discretization of $T$ is a parallelepiped grid where each block of the grid is a cube, with edges of size $\delta$, and the vertices of each cube correspond to a possible center of a sphere. As this work considers spheres of different radii, each vertex $i$ may be associated with one of the considered radii, generating the pair $(i, r_k)$, $r_k \in R$.

In order to guarantee that the spheres do not exceed too much the solid, a safe zone was created by removing candidate centers positioned close to its border. The limits of the safe zone are defined as a maximum allowed distance from the boundaries of the solid and are defined for each test case. The removal is made if some sphere generated by the pair $(i, r_k)$ cross the solid's new boundary.

## 2.2. Linear mathematical formulations

In this subsection, three mathematical formulations are presented considering the discretization presented in Section 2.1. The first formulation (2.6)–(2.8) was presented in [8]. The other two are being proposed in this work and are presented in Section 2.2.2.

*2.2.1. Independent set formulation*

At the same work as model (2.1)–(2.4), an independent set formulation is presented. As this approach will be used later in this paper in order to compare results, its interpretation follows. The vertices of the graph correspond to the discretization points of solid $T$, presented in Section 2.1. There will be an edge among two vertices of the graph if these vertices represents the centers of two spheres that satisfies constraints (2.2). The solution of model (2.1)–(2.4) will be the maximum weight clique of the feasibility graph just described. The model used to find the maximum weighted clique in a graph $G(V, A)$ is presented below

$$\max \quad \sum_{i=1}^{|V|} w_i \, y_i \tag{2.6}$$

$$\text{s. a} \quad y_i + y_j \leq 1\,, \qquad \forall\,(i,j) \notin A \tag{2.7}$$

$$y_i \in \{0,1\}\,, \qquad \forall\, i \in V\,. \tag{2.8}$$

In (2.6), $w_i$ is the weight of vertex $i \in V$ and for model purposes, the correspondence $w_i = c_i$ will hold. In (2.6)–(2.8), variable $y_i = 1$ if vertex $i$ belongs to the maximum weight clique. According to constraints (2.7) if there is no edge among two vertices $i$ and $j$, then, by definition, these two vertices cannot belong to the same clique, preventing both from being in the solution.

Following what was proposed in [8], for the sake of comparability, from this point on $c_i$ will be considered as $r_i^3$. In that case it would mean that an overlap between spheres would lead to parts of the solid counted multiple times in the objective function.

*2.2.2. Newly developed formulations*

Let $P$ be the set of points of the discretization and $I$ a set indexing those points. The distance between each pair of points $p_i, p_j \in P$ can be calculated in advance. Proceeding this way, the left side of equation (2.2) becomes a constant.

$$D_{ij} = ||p_i - p_j||, \quad i, j \in I.$$

For the sake of simplicity, instead of writing $p_i \in P$ to represent a point, from now on it will be written $i \in P$. By doing this, as each point $i$ of the grid is a candidate to be a sphere center, one may define new a variable $y_i^r = 1$ if the point $i$ is chosen as the center of a sphere of radius $r \in R$. So, accordingly to the ideas above, the following reformulation is proposed:

$$\max \quad \sum_{i \in P} \sum_{r \in R} c_i^r y_i^r$$

$$\text{s.a.} \quad D_{ij} \geq (r + s - \alpha_{rs})(y_i^r + y_j^s - 1) \qquad \forall i, j \in P, \quad \forall r, s \in R \tag{2.9}$$

$$\sum_{r \in R} y_i^r \leq 1 \qquad \forall i \in P \tag{2.10}$$

$$y_i^r \in \{0,1\} \qquad \forall i \in P, \forall r \in R \tag{2.11}$$

where $y_i^r$ is a binary variable that indicates whether or not a sphere centered in $i \in P$ and of radius $r \in R$ is in the solution. Constraints (2.9) ensure that the maximum amount of overlap between each pair of spheres is respected. Constraints (2.10) ensure that, for each point $i$ of the discretization, only one radius is selected, or none. Constraints (2.11) describes the domain of variables $y$.

Since $D_{ij}$ and $(r + s - \alpha_{rs})$ are positive constants, for all $i, j \in P$ and for all $r, s \in R$, constraints (2.9) can be strengthened, tightening the linear relaxation bound. Separating $y_i^r$ and $y_j^s$ from the other terms and using the fact that the variables are binary, constraints (2.9) can be rewritten as

$$\min\left\{ \left\lfloor \frac{D_{ij}}{(r + s - \alpha_{rs})} \right\rfloor, 1 \right\} + 1 \geq y_i^r + y_j^s \qquad \forall i, j \in P, \forall r, s \in R. \tag{2.12}$$

Thanks to that, the mathematical formulation can be rewritten as:

$$\begin{aligned} \max \quad & \sum_{i \in P} \sum_{r \in R} c_i^r y_i^r \\ \text{s.a.} \quad & (2.10), (2.11), (2.12). \end{aligned}$$

The first formulation presented in this subsection is referred as Proposed Formulation, while the last one is referred as Enhanced Formulation. For the sake of simplicity, from here on lets define $c_i^r = r^3$.

## 3. Hybrid method

In this section, the methodology developed in this work is presented. Besides testing the model proposed in Section 2, a hybrid algorithm that combines heuristics, data mining and the modified enhanced formulation, presented in Section 3.3, was developed aiming at finding good solutions in a reasonable computational time. At first, the hybrid method is briefly explained and, after the presentation of Algorithm 1, each one of the components of the method is presented as well as their interaction with one another.

First, the heuristic provides high quality solutions in a short computational time. The data mining technique is then applied on these high quality solutions, in order to extract patterns from them. In this work, the patterns correspond to certain centers of spheres. The patterns found are sent to the modified enhanced formulation as input, forcing the formulation to use these sphere centers.

Algorithm 1 presents the hybrid method proposed in this work.

**Algorithm 1:** Hybrid method.

```
Input: P, R
begin
    {s_best, ES} ← Heuristic(P, R);
    PS ← Miner(ES);
    s_best ← Modified Enhanced Mathematical Formulation(P, R, PS, s_best);
    return s_best
```

In Algorithm 1, $ES$ represents the set of elite solutions, populated by the heuristic, and PS represents the maximal pattern found by the mining process applied to the elite solution pool.

### 3.1. Heuristic – LocalSolver

The heuristic component of the hybrid method uses the LocalSolver framework [6]. Unlike other mathematical programming solvers, LocalSolver is not based in only one optimization technique. This framework has a hybrid approach of neighborhood search and this feature allows it to combine different optimization techniques in a dynamic way during the solution of the problem. LocalSolver combines local search techniques, constraint propagation and inference techniques, linear mixed integer programming techniques as well as non linear programming techniques, in order to solve problems as better as possible.

LocalSolver framework is the first mathematical programming solver integrating local search techniques in order to solve combinatorial and continuous optimization problems. Thus, LocalSolver is able to solve problems with millions of variables, which is not in the scope of the classic solvers, particularly of the linear mixed integer solvers.

The modified enhanced formulation was implemented using the LocalSolver API in C++.

### 3.2. Data mining

The raw data stored in large datasets can hide useful information for companies, government, researchers, etc. For example, it is very convenient to the owner of a supermarket to identify a buying pattern of a group of clients. In order to find association rules among data items of a specific knowledge domain, data mining makes use of methods which may involve machine learning and statistics.

Given a set of items $I = \{i_1, i_2, ..., i_n\}$ and the subsets $X \subseteq I$ and $Y \subseteq I$, where $X$ and $Y$ are nonempty, data mining algorithms produce an association rule of the form if $X$ then $Y$, which can describe the buying pattern of a group of clients of the example above.

Following the definitions presented in [9], $D$ is defined as a set of transactions (a transactional database) defined over $I$, where each transaction $t$ is a subset of I ($t \subseteq I$). The association rule found holds in database $D$ with support $s$ and confidence $c$ if, respectively, $s$% of the transactions in $D$ contain $X \cup Y$, and $c$% of the transactions in $D$ which contain $X$ also contain $Y$. These values of support and confidence are specified by the user and so computational experiments are needed to determine suitable values. Thus, in this work the terms *minsup* and *minconf* represent, respectively, the user specified minimum support and confidence. The term itemset defines sets of items that occur in at least *minsup*% of the database transactions.

The problem of mining association rules commonly has two phases. The first phase identifies all frequent itemsets and the second one shows, for each identified itemset $Z$, all association rules $A \Rightarrow B$ with confidence greater or equal to *minconf*, such that $A \subset Z$, $B \subset Z$, and $A \cup B = Z$. The first phase is named Frequent Itemset Mining (FIM) problem and demands more computational effort than the second one and has been intensively addressed [3].

In this work, the useful patterns to be mined are sets of elements that commonly appear in sub-optimal and yet high quality solutions of the CSSDRP. In this frequent itemset mining application, the set of items $I = \{i_1, i_2, ..., i_n\}$ is a set of centers. Each transaction $t$ of database $D$ represents an elite solution of the CSSDRP. A frequent itemset mined from $D$ with support $s$ represents a set of centers used in s elite solutions.

The Data Mining part of the hybrid method sought to find frequent sets. A frequent itemset is named maximal if there is no superset that is also frequent. The algorithm FPmax* was used to find these maximal frequent sets [4].

After extracting the patterns, the maximal frequent itemset is used to build the modified enhanced formulation, which is presented in the next subsection. Thus, the centers in this itemset are necessarily present at the solutions obtained from the modified enhanced formulation.

### 3.3. Modified enhanced formulation

The necessary modifications that were made on the Enhanced Formulation are presented in this subsection. These modifications were made in order to make it consider the maximal pattern mined in the data mining step. The new model presented is the result of these modifications.

$$\begin{aligned} \max \quad & \sum_{i \in P} \sum_{r \in R} y_i^r r^3 \\ \text{s.a.} \quad & (2.10), (2.11), (2.12) \\ & \sum_{r \in R} y_i^r = 1 \qquad \forall i \in \text{PS} \qquad (3.1) \end{aligned}$$

where the set of constraints (3.1) ensure that all mined centers must be used by the Modified Enhanced Formulation.

## 4. Computational results

In this section, computational results obtained with the presented methods are shown. The solution quality will be analyzed considering objective function value and computational time (Tab. 1) as well as percentage of covering of the solid $T$ (Tab. 2).

The algorithms were implemented in C++ along using *CPLEX 12.6* and *LocalSolver 6.0*. All experiments have been conducted on a PC machine with *Intel* Processor ® *XeonCore* ®*CPU X5675 @ 3,07GHz*, and 48GB of *RAM* memory. In order to run the tests, 15 instances were randomly generated. These instances, related

TABLE 1. Computational results.

| **Instances** | **V** | **Formulation [8]** | | **Proposed formulation** | | **Enhanced formulation** | | **Hybrid method** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | **Sol** | **T** | **Sol** | **T** | **Sol** | **T** | **Sol** | **TLS** | **TM** | **TMF** | **T** |
| **5-20-8** | 800 | 304 | 3600 | 240 | 3600 | 336 | 18.67 | 336 | 103 | 0.007 | 16.15 | 119.157 |
| **6-12-10** | 720 | 200 | 3600 | 224 | 3600 | 256 | 2307 | 256 | 103 | 0.005 | 2199.47 | 2302.475 |
| **7-10-19** | 1330 | 312 | 3600 | 296 | 3600 | 416 | 3600 | **464** | 115 | 0.016 | 3484.984 | 3600 |
| **8-7-17** | 952 | 240 | 3600 | 240 | 3600 | 288 | 3600 | 288 | 200 | 0.014 | 3399.986 | 3600 |
| **8-19-16** | 2432 | 472 | 3600 | 472 | 3600 | 672 | 3600 | **688** | 155 | 0.036 | 3444.964 | 3600 |
| **9-16-23** | 3312 | 600 | 3600 | 600 | 3600 | 856 | 3600 | **968** | 310 | 0.064 | 3289.936 | 3600 |
| **10-16-18** | 2880 | 504 | 3600 | 504 | 3600 | 768 | 3600 | **856** | 283 | 0.063 | 3316.937 | 3600 |
| **11-19-6** | 1254 | 256 | 3600 | 328 | 3600 | 360 | 3600 | **368** | 129 | 0.01 | 3470.99 | 3600 |
| **13-7-10** | 910 | 208 | 3600 | 232 | 3600 | 304 | 3600 | 304 | 119 | 0.011 | 3480.989 | 3600 |
| **14-12-10** | 1680 | 376 | 3600 | 272 | 3600 | 496 | 3600 | **544** | 299 | 0.033 | 3300.967 | 3600 |
| **15-20-14** | 4200 | 592 | 3600 | 592 | 3600 | 1120 | 3600 | **1175** | 506 | 0.09 | 3093.91 | 3600 |
| **18-15-18** | 4860 | – | 3600 | 983 | 3600 | 1392 | 3600 | 1392 | 429 | 0.093 | 3170.907 | 3600 |
| **19-10-13** | 2470 | 448 | 3600 | 448 | 3600 | 704 | 3600 | 704 | 169 | 0.036 | 3430.964 | 3600 |
| **19-12-5** | 1140 | 400 | 3600 | 352 | 3600 | 424 | 3600 | 424 | 113 | 0.006 | 3486.994 | 3600 |
| **21-9-14** | 2646 | 464 | 3600 | 628 | 3600 | 840 | 3600 | 840 | 255 | 0.039 | 3344.961 | 3600 |

to the volume to be covered, are all parallelepipeds with dimensions in a range between 5mm and 30 mm. A maximum execution time of one hour has been set up.

The first column of Table 1 presents the dimensions of the parallelepipeds. The second column, **V**, displays the volume of the parallelepipeds in $mm^3$. The third and fourth columns present, respectively, the value of the solution found with the formulation (2.6)–(2.8) proposed by [8] and the computational time. The fifth and sixth columns present, respectively, the value of the solution found with the enhanced formulation proposed in Section 2 and the computational time. The last five columns present information related to the hybrid method. Column **Sol** presents the value of the solution, columns **TLS**, **TM** and **TMF** present the computational time used by LocalSolver, Miner and the modified enhanced formulation, respectively. The last column, **T**, shows the total computational time used by the hybrid method.

The symbol "–" in the first column **Sol** indicates that the formulation presented in [8] found no feasible solution in the maximum allowed execution time. The underline represents that the mathematical formulation was able to prove optimality. The results in bold indicate that the solution provided by hybrid method is better than the one obtained by enhanced formulation.

Table 1 shows that the value of the solution found by the enhanced formulation outperformed all the ones obtained by the other mathematical formulations, considering the limit of time. A more interesting comparison lies between the results given by the enhanced formulation and the hybrid method. The values of the objective function of the enhanced formulation and the hybrid method are reported in the seventh and ninth columns of Table 1, respectively. In this way, it is shown that the hybrid method found better solutions for 7 of the 15 instances and both methods found the same value for 8 of the 15 instances. In terms of computational time, the hybrid method spent more computational time than the enhanced formulation on the first instance, but on the second instance, it was more efficient. All the other tests were interrupted when the maximum computational time was reached for both enhanced formulation and hybrid method.

## 4.1. Disparity between objective function and covering

The models presented at Section 2.2, despite producing a good approximation of the real problem, reach different configurations of covering. It can be seen in Table 1 that the hybrid method found equal or better

TABLE 2. Covering percentual.

| Instances | Enhanced formulation | | Hybrid Method | | Grid |
|---|---|---|---|---|---|
| | Spheres | Covering | Spheres | Covering | |
| **5-20-8** | 42 | 0.9562116 | 42 | 0.9562116 | 826200 |
| **6-12-10** | 32 | **0.9050921** | 32 | 0.8998767 | 745481 |
| **7-10-19** | 52 | 0.8592247 | 58 | 0.9056118 | 1362490 |
| **8-7-17** | 36 | 0.8373494 | 29 | **0.8554597** | 983421 |
| **8-19-16** | 70 | 0.8308372 | 58 | 0.8515358 | 2477790 |
| **9-16-23** | 100 | 0.7834927 | 79 | 0.8883688 | 3369730 |
| **10-16-18** | 89 | 0.7994327 | 79 | 0.8864075 | 2943241 |
| **11-19-6** | 38 | 0.8261945 | 32 | 0.8686519 | 1286490 |
| **13-7-10** | 38 | 0.8721249 | 27 | 0.8721249 | 939401 |
| **14-12-10** | 62 | 0.8417983 | 61 | 0.9000709 | 1723161 |
| **15-20-14** | 133 | 0.8045428 | 91 | 0.8857743 | 4258200 |
| **18-15-18** | 174 | 0.8502051 | 174 | 0.8502051 | 4946911 |
| **19-10-13** | 88 | 0.8392121 | 88 | 0.8392121 | 2513890 |
| **19-12-5** | 53 | 0.9254484 | 53 | **0.929277** | 1172490 |
| **21-9-14** | 105 | 0.8872923 | 105 | 0.8872923 | 2694510 |

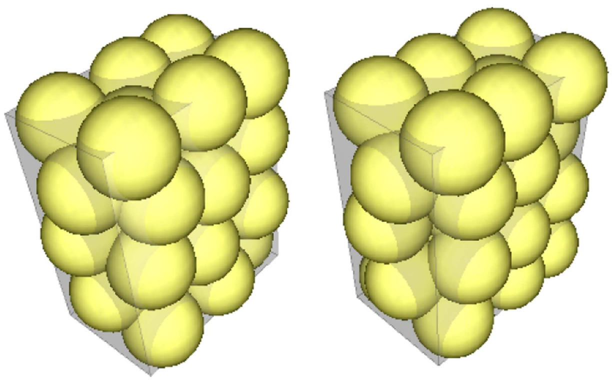

FIGURE 2. Graphic comparison between enhanced formulation and hybrid method – 6-12-10-1.

solutions compared to the enhanced formulation in all instances. However, for one instance, the covering of the enhanced formulation is better than the hybrid one as it can be seen in Table 2.

Table 2 is organized as follows: the first column shows the dimensions of the parallelepipeds. The second column displays the number of spheres used in the solution. The third column displays the covering of the solid given by the methods presented in this work. The last column shows the number of points used in the discretization in order to calculate the covering. An important remark is that a covering equal to one means a total covering of the solid.

As shown in Tables 1 and 2, in 3 instances, namely, (6-12-10, 8-7-17 and 19-12-5), both methods found the same value of the objective function, but with different solutions. These solutions have generated different covering of the solids. The enhanced formulation reached a better percentage of covering in just one of these three instances. The six pictures associated with the solution given by the enhanced formulation and the hybrid method for the three instances are shown below and reveal the existent differences between the solutions. For each presented image, (Figs. 2–4), the first solution is the one obtained by the enhanced formulation, while the second one is the one obtained by the hybrid method.

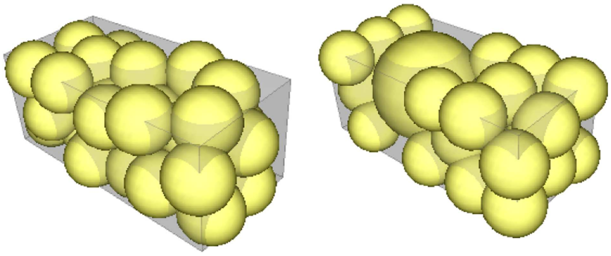

Figure 3. Graphic comparison between enhanced formulation and hybrid method – 8-7-17-1.

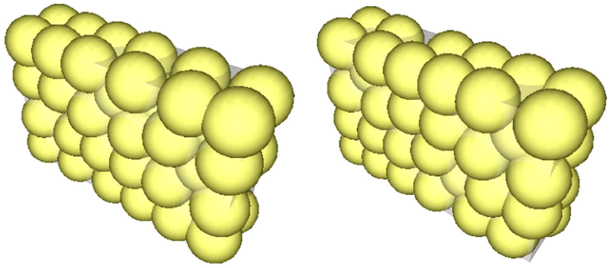

Figure 4. Graphic comparison between enhanced formulation and hybrid method – 19-12-5-1.

An important remark is that the hybrid method has produced better covering for two of the three instances. The hybrid one better covers the solids, and the average gain of covering is more than 5% regarding the enhanced formulation. In the single instance in which the enhanced formulation has a better covering, the difference from the covering produced by the hybrid method is only of 0.5% approximately. The hybrid method used the same number or less spheres than the enhanced formulation in 14 of the 15 instances.

## 5. Conclusion and future works

The results show that the enhanced formulation presented in this work finds better quality results regarding the value of the objective function, in comparison to the mathematical formulation presented by [8]. The results also show that mixing it with a data mining procedure produced an efficient hybrid method. When comparing the objective function of the hybrid method and the enhanced formulation, the hybrid method reaches better solutions to 7 of the 15 instances and finds equal solutions in 8 of the 15 instances. Regarding the covering, even when the hybrid method finds a worse solution, it happens in only one of the fifteen instances. In this case, the difference of the percentage of covering is of, approximately, 0.5%, while the average percentage of the difference of gain, in absolute value, is above 5%.

According to the presented results, few possible ways of developing this work are:

- Find valid cuts so a Branch-and-Cut algorithm could be developed;
- Study different ways to calculate $\alpha_{rs}$;
- Develop a concurrent version of the hybrid method in such a way that several patterns could be used simultaneously, instead of the proposed approach, where just one pattern is used in the Modified Enhanced Formulation.

*Acknowledgements.* This work has been partially supported by CNPq, CAPES-MEC, FAPERJ, COPPETEC Foundation.

## References


[1] M.C. Ferris, L. Jinho and D.M. Shepard, An optimization approach for radiosurgery treatment planning. *SIAM J. Optim.* **13** (2002) 921. Available from: http://search-ebscohost-com.ez15.periodicos.capes.gov.br/login.aspx?direct=true&db=iih&AN=10281146&lang=pt-br&site=ehost-live.

[2] J.C. Ganz, Gamma Knife Surgery. Springer-Verlag Wien, Austria (1997).

[3] B. Goethals and M.J. Zaki, Advances in frequent itemset mining implementations: report on FIMI'03. *Acm Sigkdd Explor. Newslett.* **6** (2004) 109–117.

[4] G. Grahne and J. Zhu, Reducing the main memory consumptions of FPmax* and fpclose. In: *Proc. Workshop Frequent Item Set Mining Implementations (FIMI 2004, Brighton, UK), Aachen, Germany*. Citeseer (2004).

[5] L. Liberti, N. Maculan and Y. Zhang, Optimal configuration of gamma ray machine radiosurgery units: the sphere covering subproblem. *Optim. Lett.* **3** (2009) 109–121.

[6] Localsolver framework. Available from: http://www.localsolver.com. (2020)

[7] D.C. Lubke, *Cobertura de corpos por esferas utilizando suavização hiperbólica*. Master's thesis, COPPE/UFRJ, Rio de Janeiro, 2 (2014).

[8] R.V. Pinto, *O Problema de Recobrimento de Sólidos por Esferas de Diferentes Raios*. Ph.D. thesis, COPPE/UFRJ, Rio de Janeiro, 3 (2015).

[9] M.H. Ribeiro, A. Plastino and S.L. Martins, Hybridization of grasp metaheuristic with data mining techniques. *J. Math. Model. Algorithms* **5** (2006) 23–41.

[10] A. Sutou and Y. Dai, Global optimization approach to unequal global optimization approach to unequal sphere packing problems in 3D. *J. Optim. Theory App.* **114** (2002) 671–694.

[11] M.B. Karam Venceslau, *O Problema de Recobrimento Mínimo de um Corpo em Três Dimensões por Esferas de Diferentes Raios*. Ph.D. thesis, COPPE/UFRJ, Rio de Janeiro, 5 (2015).