LVI SBPO SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL
DE 04 ATÉ 07 NOVEMBRO 2024
FORTALEZA | CEARÁ

A New Mathematical Model for the Intersection Clustering Problem

David P. Galvão Júnior, Gilberto F. de Sousa Filho, Fabio Protti
Centro de Informática - UFPB - João Pessoa, PB
Instituto de Computação - UFF - Niteroi, RJ
galvao2010@icloud.com, gilberto@ci.ufpb.br, fabio@ic.uff.br

ABSTRACT

The Intersection Clustering Problem (ICP) aims to identify data dependencies by grouping objects according to their use of shared resources, such as words used in writing fake news, collusion companies competing in public tenders or co-authors of scientific articles. In this work, starting from the concept of an intersection graph for ICP, we propose a new mathematical formulation for the problem, aiming to create an optimal partitioning with up to two objects in each part. The experiments carried out on artificial instances show that the new formulation was able to reach the optimum in all cases within the time limit, with a 99.2% reduction in the time taken to solve the instances. At the same time, the solutions obtained maintain the quality of the model in generating groups with better intersections than classical models according to the average silhouette intersection index metric.

KEYWORDS. intersection set, cluster analysis, integer programming model.

Topics: Combinatorial optimization, Mathematical programming, Data mining

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

1. Introduction

Clustering is a technique that groups unlabeled data so that objects within the same class have similar characteristics, and are different from objects within other classes [Aidagulov et al., 2022]. According to [Oyewole and Thopil, 2022], this technique is widely used in industrial activities such as the relationship between road transportation data and macroeconomic indicators, designing of cellular manufacturing, designing of ship loading and diesel fuel consumption, and identification of patterns in healthcare data sets.

To extract useful information from public procurement auction data to detect collusion [García Rodríguez et al., 2022], we stress the importance of grouping competing companies, which in their history jointly participated in public tenders, to, in a second moment, measure how much this concomitant participation did not occur by chance. Therefore, we started searching for an appropriate clustering model for this task; first, we used the most classic clustering model in which the proximity is based on a distance function $d(i, j)$ of objects i and j. The first difficulty was to define what would be a good distance function between two companies since the bids are not ordinal values, but guided by their common characteristics, as in [Lapointe and Legendre, 1994], where the authors use the Jaccard distance function

$$d(i,j) = 1 - \frac{|L_i \cap L_j|}{|L_i \cup L_j|}, \tag{1}$$

which takes value 0 when the intersection of the objects' features is equal to their union (maximum similarity) and value 1 when the intersection is empty. By making L_i the set of bids for company i, we applied the Jaccard distance matrix to the p-median model [Daskin et al., 2015], which seeks to build clusters whose distances to the prototype object are minimized. Observing the formed solution, the companies of each cluster frequently do not have any bidding in common. To understand this phenomenon, observe the companies A, B, and C of Figure 1(a). The company C is the prototype of its cluster, and therefore companies A and B have a good intersection with C; however, this relationship is not transitive, that is, A and B may not have participated in any joint bidding, and this distance is not accounted for in the objective function.

Seeking to avoid the non-transitivity of the Jaccard distance, we applied next the Jaccard matrix to the Cluster Editing model [Böcker and Baumbach, 2013], which seeks to partition the input graph into disjoint cliques by performing a minimum number of modifications of its edges (addition or removal). As illustrated in Figure 1(b), in each cluster, all links between objects are accounted for in the objective function. However, even if companies A, B and C are in the same cluster, minimizing internal links, there is no guarantee that the bids contained in the intersection of A and B are the same as those in the intersection of B with C, so the solution continues with few intersections.

We then used the measure of proximity of objects i and j from a similarity function $\delta(i, j)$, which accounts for the number of dissimilar features between objects, that is, the more similar features more the objects are close [Huang, 1997]. This clustering model is used on non-ordinal categorical data [Bai and Liang, 2015]. We associate each bid with a binary category of the company indicating whether or not it participated in that bid and map the function $\delta(i, j)$ to the p-median model as proposed by [Huang, 1998]. However, the similarity function $\delta(i, j)$ also suffers from non-transitivity in its relations. In Figure 1(c), we can observe in the table of similarities that the company C attracts the companies A and B to its cluster because it has only two dissimilarities; however, when observing $\delta(A, B)$, the dissimilarity is total, thus creating clusters of low intersection.

Abandoning the relationship between companies pair by pair, we tried to adopt the bi-cluster model [Nadif and Govaert, 2013], in which two groups of distinct entities relate to each

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

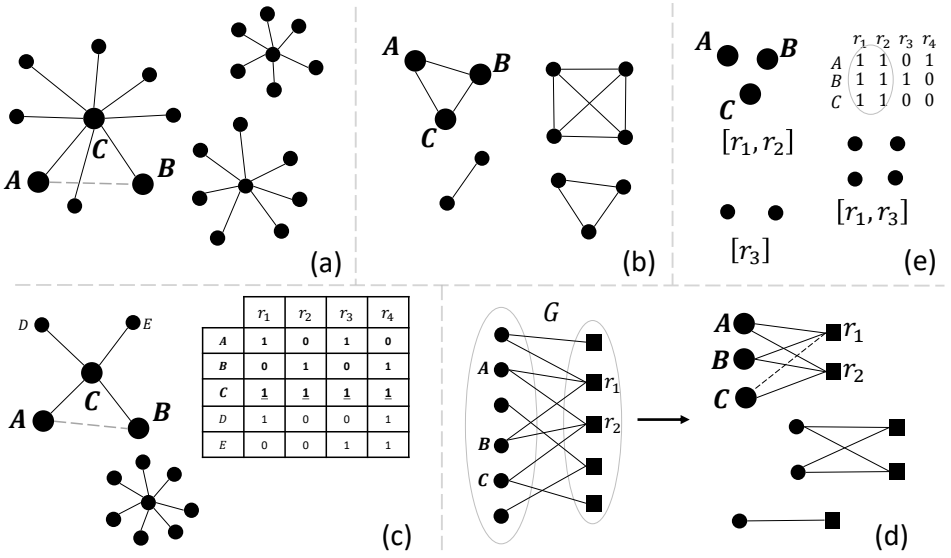


Figure 1: Clustering models.

other, but the relationship between elements of the same entity is forbidden. Observing the bipartite graph G in Figure 1(d), we have the companies in one entity (black balls) and the bids in another (black squares), and the edges represent the participation of the companies in the bids. This graph is provided to the Bicluster Editing Problem model [Sousa Filho et al., 2021] which seeks to partition it into disjoint bicliques minimizing the number of modified edges. However, bi-clusterization continued to generate groups with low intersections. Returning to Figure 1(d), the biclique formed by companies A, B, and C has bids r_1 and r_2; however, company C did not participate in r_1 – this relation was added by the model. Therefore, despite the few edge modifications in the original graph, this group only has the bidding r_2 in its intersection.

Therefore, in [Galvão Jr., 2023], the Intersection Clustering Model was proposed. This model does not use the direct relationship between two objects; instead, the evaluation of each cluster is guided by the size of the intersection of its objects' resources. In Figure 1(e), the cluster formed by companies A, B, and C has two bids at the intersection since all companies participated in r_1 and r_2. From this model, a combinatorial optimization problem was proposed, titled Intersection Clustering Problem (ICP), and exact and heuristic solutions to maximize the intersection between groups of objects.

The above discussion was focused on the example of grouping companies based on participation in fraudulent bids, but the ICP can be used for other contexts, such as: grouping words used in the same emails classified as spam, thus allowing to identify the dependence of the words in the context; products grouped by the same characteristics in order to identify their origin, and grouping of professors in a research center by the co-authorship of their scientific articles.

In this work, we propose a new integer programming model, titled Pairwise Model, which was based on a theoretical aspect of the ICP, presenting a good speedup compared to the original model in our computational experiments.

The structure of the paper is as follows. In Sect. 2, we review some clustering models from the literature. In Sect. 3, we present the ICP and discuss some theoretical aspects. In Sect. 4, we propose a new integer programming model for ICP. In Sect. 5, we perform computational tests on artificial instances to benchmark our performance against classical clustering algorithms. In Sect. 6, we present our concluding remarks.

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

2. Literature Clustering Models

[Oyewole and Thopil, 2022] presents a classification criterion of clustering based on the proximity measure being divided into distance matrix and similarity matrix. We use this classification to separate the clustering models that, within our knowledge, could be adapted to identify groups with elements in their intersection, we also present clustering models based on editing their edges.

2.1. Distance matrix

In this model, objects are grouped into different clusters according to certain distance function $d(i, j)$, usually for continuous and ordinal features c. Considering m the number of objects features, the most common function is the Euclidean distance

$$d(i,j) = \sqrt{\sum_{c=1}^{m}(i_c - j_c)^2}.$$

K-means [MacQueen, 1967] is the most popular clustering algorithm, it creates a partitioning P of the set of objects X, where $|P| = k$. It consists in placing each object $x \in X$ in the cluster with the closest *centroid* c, minimizing the sum of distances $d(x, c)$ [Sinaga and Yang, 2020]. A *centroid* is a point, localized exactly at the center of a cluster. As this *centroid* is artificial, k-means was not adopted in this work.

The p-median problem seeks to identify p source locations and assign them to n destinations while minimizing the sum distance between destinations and corresponding sources [Gwalani et al., 2021]. In the data mining area, the most established heuristic for the p-median is called k-medoid [Park and Jun, 2009]. Inspired by the k-means algorithm, the partitions of X are built by choosing a cluster member object as its *centroid*.

[Hennig, 2022] presents 12 different standard indexes in the literature on the internal validation of clustering, being able to evaluate desirable aspects in specific applications. In this work, we adopted the Average Silhouette Width index proposed by [Rousseeuw, 1987] because it attempts to balance two aspects of cluster quality, namely homogeneity, and separation. For $i = 1, ..., n$, and considering l_i as the cluster identifier that contains the object i, we have the "silhouette width"

$$s(i) = \frac{b(i) - a(i)}{max\{b(i), a(i)\}} \in [-1, +1], \tag{2}$$

where $a(i) = \frac{1}{n_{l_i}-1} \sum_{j \in C_{l_i}} d(i,j)$ and $b(i) = \min_{h \neq l_i} \frac{1}{n_h} \sum_{j \in C_h} d(i,j)$.

The Average Silhouette Width is defined as

$$I_{asw}(P) = \frac{1}{n} \sum_{i \in X} s(i).$$

2.2. Similarity matrix

Categorical data is defined from the observation of categorical variables, i.e., variables that have a finite and unordered set of possible values [Huang, 1997]. Formally, let $X = \{1, 2, ..., n\}$ be a set of n objects, $A = \{a_1, a_2, ..., a_m\}$ a set of m attributes used to describe the objects of X. To measure the proximity between the objects $i, j \in X$ we have the similarity function

$$\delta(i,j) = \sum_{p=1}^{m} 1[a_p^{(i)} \neq a_p^{(j)}],$$

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

which computes the number of attributes that are different between i and j, that is, the more equal attributes, the closer the objects are.

To cluster this kind of data, the k-modes algorithm is one of the most popular, it is very similar to the k-means algorithm, with a divergence in the way the nearest cluster to a point is defined. In this case, the dissimilarity function $\delta(i, j)$ is used [Chaturvedi et al., 2001].

[Bai and Liang, 2015] presents in detail two similarity measures from the literature for clustering categorical solutions:

- The category utility function (CU) attempts to maximize the probability that two data objects in the same cluster obtain the same attribute values;

- The information entropy function (E_{log}) uses the information-theoretic principles and the notion of entropy to measure the clustering results. The basic intuition is that groups of similar objects have lower entropy than those of dissimilar ones.

2.3. Graphs editing model

The models presented in this section work with clique formation in their solutions. A clique is a subset of the vertices such that every two distinct vertices are adjacent.

The cluster editing problem (CEP) consists of giving a graph $G = (V, E)$, transforming G into a vertex-disjoint union of cliques by inserting and deleting a minimum number of edges from E [Böcker and Baumbach, 2013]. A non-automatic version of CEP called p-CEP, builds a partitioning of p cliques minimizing edge edits [Bulhões et al., 2017a].

The bicluster editing problem (BEP) consists of editing a minimum number of edges in order to transform an input bipartite graph $G = (V_1, V_2, E)$ into a vertex-disjoint union of complete bipartite subgraphs; in other words, in a final solution, each connected component is a biclique [Sousa Filho et al., 2017].

The two proximity metrics (objects) and their respective homogeneity index (partition) can be used as input for these models and in the evaluation of their formed solutions.

3. Intersection Clustering Model

The intersection clustering model has two main characteristics: (1) there is no measure of proximity between objects in pairs, the measurement is made over the entire of objects in each cluster and not as a composition of metrics; (2) clustering is guided by the number of resources that all objects have simultaneously.

3.1. Intersection Clustering Problem

Let $X = \{1, 2, .., n\}$ be a set of n objects, L a set of resources shared between objects, $L_i \subseteq L$ the set of resources used by object $i \in X$, $P = \{C_1, C_2, .., C_m\}$ a partition of X, and

$$L_{C_k} = \begin{cases} \bigcap_{i \in C_k} L_i & \text{, if } |C_k| > 1 \\ \emptyset & \text{, otherwise} \end{cases}$$

the intersection set of the resources used by the objects in a cluster C_k. The Intersection Clustering Problem (ICP) aims to find a partition P of X such that the sets L_{C_k} are distinct and the sum of

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

their sizes is maximized. The problem can be formulated as follows [Galvão Jr., 2023] :

$$\text{Maximize} \quad \sum_{k=1}^{m} |L_{C_k}| \tag{3}$$

$$\text{s.a.} \quad \bigcup_{k=1}^{m} C_k = X, \tag{4}$$

$$C_k \neq \emptyset, \qquad \forall k \in \{1, 2, ..., m\}, \tag{5}$$

$$C_k \cap C_l = \emptyset, \qquad \forall k, l \in \{1, 2, ..., m\}, k \neq l, \tag{6}$$

$$L_{C_l} \neq L_{C_k}, \qquad \forall k, l \in \{1, 2, ..., m\}, k \neq l. \tag{7}$$

Restrictions (4), (5) and (6) that each object $i \in X$ belongs to exactly one cluster, and (7) prohibits the appearance of two different clusters with identical intersections in P.

Figure 2(a) illustrates an example instance $\hat{X} = \{1, 2, 3, 4, 5\}$ for the ICP, with the sets of resources $\hat{L}_i$. Figure 2(b) illustrates a partition $P' = \{C'_1, C'_2\}$ of $\hat{X}$, where $C'_1 = \{2, 3, 4, 5\}$ has only the resource b at its intersection, and $C'_2 = \{1\}$, due to being a singleton, does not have resources at its intersection, thus, P' contains only one resource. Figure 2(c) presents the optimal partition of $\hat{X}$, due to the movement of object 3 from C''_1 to C''_2 we get $\hat{L}_{C''_1} = \{b, d\}$ and $\hat{L}_{C''_2} = \{a, b\}$, increasing to four the resources at the intersections of P''.

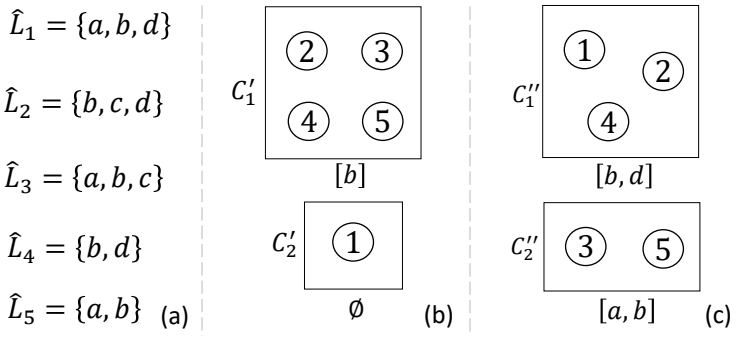


Figure 2: Example solutions for the ICP. (a) List of resources $\hat{L}_i$ of the objects of instance $\hat{X} = \{1, 2, 3, 4, 5\}$, (b) solution P' of $\hat{X}$ and (c) optimal solution P'' of $\hat{X}$.

Average Silhouette Intersection index

To identify the homogeneity of a P solution for the ICP, we propose to define the similarity relation between an object $i \in X$ and a cluster $C_k \in P$ through the distance function

$$d(i, C_k) = \left| L_i \cup L_{C_k} \setminus L_i \cap L_{C_k} \right|,$$

which quantifies the missing resources so that L_i is equal to L_{C_k}. Next, we propose an adaptation of the silhouette index proposed by [Rousseeuw, 1987] for the ICP. To do so, consider $s(i)$ as the value of the silhouette of each object $i \in X$, defined as in Equation (2), where $a(i) = d(i, C_w)$, the distance from object i to the cluster $C_w \subset P$ it is contained in ($i \in C_w$), and $b(i) = min\{d(i, C_t) \mid C_t \in P, C_t \neq C_w\}$, the smallest distance between i and all clusters C_t that i is not contained in. The divisor $max\{b(i), a(i)\}$ of $s(i)$ guarantees its value in the interval $[-1, +1]$, indicating perfect clustering when $s(i) = +1$ and imperfect clustering when $s(i) = -1$. If $a(i) = b(i) = 0$ then $s(i) = 0$.

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

Calculating the average of the silhouette values of the objects $i \in X$ we have

$$I_{asi}(P) = \frac{1}{n} \cdot \sum_{i \in X} s(i), \tag{8}$$

which is the Average Silhouette Intersection index of partition P.

Analyzing the example partition P' from Figure 2(b) we have that $s(1) = -\frac{1}{3}$, $s(2) = \frac{1}{3}$, $s(3) = \frac{1}{3}$, $s(4) = \frac{1}{2}$ and $s(5) = \frac{1}{2}$, so $I_{asi}(P') = \frac{1}{5} \cdot \frac{4}{3} = +0.27$. The partition P'' of Figure 2(c) has $s(1) = 0$, $s(2) = \frac{1}{2}$, $s(3) = \frac{1}{2}$, $s(4) = 1$ and $s(5) = 1$, obtaining $I_{asi}(P'') = \frac{1}{5} \cdot 3 = +0.6$.

3.2. Theoretical aspects

Definition 1. *The (intersection graph) is a directed graph $G = (V, A)$, in which each vertex $i \in V$ is associated to the object $x_i \in X$, the set of edges $A = \{(i, j) \mid i, j \in V, i \leq j\}$ represents the intersection of resources between x_i and x_j. The edges are weighted by*

$$w_{ij} = \begin{cases} |L_i \cap L_j| & \text{, if } i < j \\ 0 & \text{, if } i = j. \end{cases}$$

Figure 3 shows the *intersection graph* $\hat{G}$ associated with the input $(\hat{X}, \hat{L}_i)$ illustrated in Figure 2(a). All edges of $\hat{G}$ are present, therefore, all objects of X have some intersection of resources between them. As an example, objects 2 and 3 have $w_{2,3} = |L_2 \cap L_3| = |\{b, c\}| = 2$.

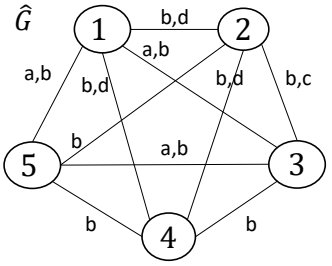


Figure 3: *Intersection graph* $\hat{G}$ of $(\hat{X}, \hat{L}_i)$.

In [Galvão Jr., 2023] the following propositions for the ICP are proved:

Proposition 1. *ICP is NP-Complete.*

Proposition 2. *Let (X, L) be an instance for the ICP, there is at least a optimal partitioning P^* of X with at most two objects in each part.*

4. Pairwise Model for ICP

The partition formed by the Pairwise Model consists of groups containing at most 2 objects, with the optimal solution guaranteed by Proposition 2. Let $G = (V, A)$ be an incidence graph, $A_v = \{(i, j) \in A \mid v = i \text{ or } v = j\}$ be the set of arcs in A that have v as an endpoint vertex, the binary decision variable $y_{ij} = 1$ indicating that vertices i and j form a part $C = \{x_i, x_j\}$ contained in the solution P, and considering Proposition 2, we aim to find a partition P in the ICP that

$$\text{Maximize} \quad \sum_{(i,j) \in A} w_{ij} y_{ij} \tag{9}$$

$$\text{s.t.} \quad \sum_{(i,j) \in A_v} y_{ij} \leq 1, \qquad \forall v \in V, \tag{10}$$

$$y_{ij} \in \{0, 1\}, \qquad \forall (i, j) \in A.$$

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

The objective function (9) seeks to maximize the intersection of shared resources between pairs of objects that compose P. The constraints (10) ensure that each vertex is in at most one group of the partition P. Finally, every vertex v for which $\sum_{(i,j)\in A_v} y_{ij} = 0$ or $y_{vv} = 1$ should be added to a unit group $C = \{x_v\}$ in P without loss of optimality. Compared to the model proposed in [Galvão Jr., 2023], the formulation of the present work has $O(|X|^2)$ variables and $O(|X|)$ constraints, while the previous formulation has $O(|X|^2)$ variables and $O(|X| \times |L|)$ constraints.

Figure 4 illustrates an instance $\hat{X} = \{x_1, x_2, x_3, x_4\}$ for the ICP through the resource sets $\hat{L}_i$ and solutions generated by the Pairwise Model. Figure 4(b) illustrates a partition $P' = \{C'_1, C'_2, C'_3\}$ of $\hat{X}$, where $C'_1 = \{x_3, x_4\}$, defined by the variable $y_{3,4} = 1$, has only the resource c in its intersection, so $w_{3,4} = 1$. The parts $C'_2 = \{x_1\}$ and $C'_3 = \{x_2\}$, defined by $y_{1,1} = y_{2,2} = 1$, form unitary groups and have no resources in their intersection, so P' contains only one resource. Figure 4(c) presents the optimal partition of $\hat{X}$, because $C''_2 = \{x_1, x_2\}$, formed by $y_{1,2} = 1$, has $L_{C''_2} = \{c, d\}$, that is, $w_{1,2} = 2$, increasing the resources in the intersections of P'' to three.

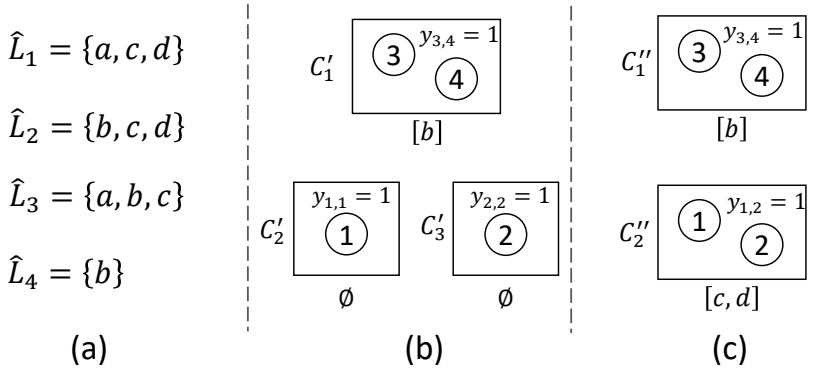


Figure 4: Solutions for the Pairwise Model. (a) Resources of the objects in the instance $\hat{X} = \{x_1, x_2, x_3, x_4\}$, (b) solution P' of $\hat{X}$, and (c) optimal solution P'' of $\hat{X}$.

5. Computational Results

The methods proposed by this work and those mentioned in Section 2 were implemented in the C++ language, using the compiler g++ 5.4.0 (compile option -O2). All mathematical models were solved with the solver IBM CPLEX version 12.9. All experiments were performed on a computer with the following specifications: Intel Xeon E5-2650 v4 2.20 GHz, with 128GB of RAM, and running the operating system Linux Ubuntu 16.04.

Initial studies were conducted on 315 artificial instances. Let $XL(n, m, d)$ denote a random instance with n objects in X, m resources in L, such that each relationship between an object and a resource can independently exist with probability p. First, the subsets X and L are initialized with $n \in [2, 50]$ and $m \in \{16, 21, 23, 30, 33, 35, 39\}$ elements, respectively. Then, for each object $i \in X$, a decision is made with probability p whether to add resource $r \in L$ to L_i. Note that the density of XL is defined as $d = \frac{\sum_{i\in X} |L_i|}{n \cdot m}$.

5.1. Comparison of ICP models

We will use the XL instances to compare the formulation described in Section 4 with the exact strategies of ICP proposed in [Galvão Jr., 2023]. Initially, we compare the execution time between the two formulations. In instances with $|X| > 20$, the original model did not find optimality in any instance after a 1-hour time limit, whereas the Pairwise model solved all instances. Both models were able to solve all instances with $|X| \leq 20$; however, the Pairwise model took an average of $\mu = 5.26$ ms to reach optimality, while the original model took $\mu = 5,914.98$ ms.

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

One possible explanation for this speedup achieved by the Pairwise model lies in the smaller number of variables and constraints in its formulation. We computed the number of variables and constraints generated by each model in the XL instances and presented the boxplot of these values in Figure 5. As we can observe, the Pairwise model showed a lower number, both in the median and in the range of values of the first and third quartiles for the number of variables and constraints. With respect to the number of variables, both models have the same asymptotic complexity, however, the original model has 3 variables with complexity $O(|X|^2)$, compared to 1 variable in the Pairwise model. Regarding the number of constraints, the Pairwise model is asymptotically better, with $O(|X|)$ constraints compared to $O(|X| \times |L|)$ constraints in the original model.

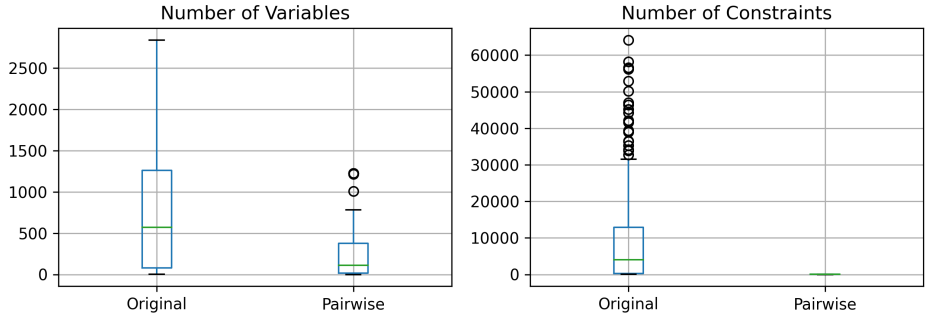


Figure 5: Boxplot of the number of variables and constraints in the original ICP model and the proposed Pairwise model.

5.2. ICP versus literature models

In this study, we will use the XL instances to compare the Pairwise model of ICP with other clustering models mapped to maximize the intersection of resources in the formed clusters. For the Pairwise model, the XL instances do not require any treatment. For the CEP, p-CEP, and p-median$_{jac}$ models, we provide a distance matrix computed between all objects $i, j \in X$ using theJaccard distance (Equation 1). For the BEP model, which takes as input the bipartite graph $G = (V_1, V_2, E)$, we map $V_1 = X$, $V_2 = L$, and $E = \{(i, j) \mid i \in X \text{ and } j \in L_i\}$. As for the p-median$_{dis}$ model, we provide a similarity matrix between the objects in X computed by the similarity function $\delta(i, j) = |L_i \setminus L_j|$.

Table 1 presents the metrics for literature models and exact algorithms for ICP on XL instances with $10 \leq |X| \leq 20$. The column group Inst. describes the dimensions of each instance with the columns $|X|$, $|L|$, and d, which represent the number of objects, resources, and density of the instance, respectively. Each proposed algorithm has its specific column group. In each group, the column I_{asi} presents the value of the Average Silhouette Intersection index for the found partitioning (Section 3.1), the column Obj represents the value of the objective function (3), and t(s) is the elapsed time for solving the instance in seconds. In the ICP Pairwise, CEP, and BEP column sets, the column K indicates the number of clusters in their solution. The p-CEP, p-median$_{jac}$, and p-median$_{dis}$ models are non-automatic clustering where the number of clusters to be formed is an input parameter. Therefore, the value of K obtained by ICP Pairwise was provided. A time limit of 1 hour was set for completing the execution in all instances, and the model that exceeded this time limit in any instance is marked with TL in the t(s) column. For the I_{asi}, Obj, and t(s) columns, the best value obtained for the instance among all models is marked in bold.

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

Table 1: Metrics of different clustering models in selected artificial instances

Inst.			ICP Pairwise				CEP				BEP				p-CEP				p-median$_{dis}$				p-median$_{jac}$			
$\|X\|$	$\|L\|$	d	I_{asi}	Obj	K	t(s)	I_{asi}	Obj	K	t(s)	I_{asi}	Obj	K	t(s)	I_{asi}	Obj	t(s)		I_{asi}	Obj	t(s)		I_{asi}	Obj	t(s)	
10	16	0.3	0.11	**12**	5	**0.0**	0.0	0	2	**0.0**	0.0	0	5	0.06	**0.18**	**12**	0.14		0.0	2	0.01		0.0	1	0.01	
10	16	0.5	0.13	**27**	5	**0.0**	0.0	0	1	**0.0**	0.0	0	2	0.47	**0.18**	**27**	0.09		0.0	5	0.01		0.0	6	0.01	
10	16	0.6	0.15	**34**	5	**0.0**	0.0	0	1	**0.0**	0.0	6	4	1.92	**0.16**	**34**	0.09		0.0	16	0.01		0.0	10	0.01	
10	16	0.7	**0.16**	**45**	5	**0.0**	0.0	1	1	**0.0**	0.0	1	2	4.32	**0.16**	**45**	0.08		0.0	13	0.01		0.0	13	0.01	
10	16	0.8	**0.12**	**54**	5	**0.0**	0.0	1	1	**0.0**	0.09	11	2	0.05	0.03	43	0.1		0.0	7	0.01		0.0	7	0.01	
10	21	0.27	**0.07**	**12**	5	**0.0**	0.0	0	3	0.01	0.0	0	6	0.04	0.06	**12**	0.15		0.0	11	0.01		0.0	6	0.01	
10	21	0.33	**0.31**	**15**	3	**0.0**	0.0	13	6	**0.0**	0.0	1	3	0.01	0.0	12	0.07		0.2	13	0.01		0.2	13	0.01	
16	30	0.3	0.14	**34**	8	**0.0**	0.0	0	1	0.01	0.0	2	9	205.6	0.12	**34**	TL		0.0	12	0.02		0.0	10	0.01	
16	30	0.4	**0.19**	**55**	8	**0.0**	0.0	0	1	0.02	0.0	6	7	TL	**0.19**	**55**	TL		0.0	26	0.01		0.0	17	0.02	
16	30	0.5	0.15	**76**	8	**0.0**	0.0	0	1	**0.0**	0.0	7	6	TL	**0.17**	**76**	TL		0.0	43	0.01		0.0	28	0.36	
16	30	0.6	**0.15**	**104**	8	**0.0**	0.0	0	1	**0.0**	0.05	4	2	TL	**0.15**	**104**	TL		0.0	55	0.02		0.0	40	0.22	
16	30	0.7	**0.12**	**131**	8	**0.0**	0.0	0	1	**0.0**	0.1	9	2	TL	**0.12**	**131**	TL		0.0	56	0.02		0.0	38	0.02	
16	30	0.8	0.11	**164**	8	**0.0**	0.0	1	1	**0.0**	0.0	1	2	TL	**0.12**	**164**	TL		0.0	51	0.01		0.0	27	0.01	
16	30	0.9	**0.06**	**200**	8	**0.0**	0.0	5	1	**0.0**	0.0	5	1	1709.83	0.01	174	TL		0.0	14	0.01		0.0	14	0.01	
19	23	0.36	**0.04**	**62**	10	**0.0**	0.0	1	3	0.03	0.0	0	7	2.14	**0.04**	**62**	TL		0.0	30	0.01		0.0	18	0.01	
19	23	0.42	**0.01**	**53**	10	**0.0**	0.0	0	1	0.02	0.0	5	9	TL	**0.01**	**53**	TL		0.0	16	0.01		0.0	14	0.01	
19	33	0.18	**0.14**	**32**	8	**0.0**	0.0	2	7	0.03	0.0	0	5	0.35	0.0	19	TL		0.0	18	0.01		0.0	2	0.01	
19	33	0.37	0.19	**68**	8	**0.0**	0.0	0	5	0.02	0.0	4	4	TL	0.0	45	TL		0.0	30	0.01		0.0	8	0.01	
19	35	0.37	**0.03**	**96**	9	**0.0**	0.0	3	4	0.02	0.0	0	2	892.55	0.0	74	TL		0.0	51	0.01		0.0	43	0.01	
19	35	0.42	**0.01**	**78**	10	**0.0**	0.0	0	1	**0.0**	0.0	19	6	TL	**0.01**	**78**	TL		0.0	27	0.01		0.0	23	0.01	
19	35	0.95	**0.01**	**263**	10	0.1	0.0	0	1	**0.0**	0.0	0	1	TL	0.0	207	TL		0.0	49	0.01		0.0	49	0.01	
19	39	0.53	**0.17**	**153**	8	**0.0**	0.0	12	5	0.02	0.0	14	3	36.67	0.0	93	TL		0.0	122	0.01		0.0	41	0.01	
19	39	0.68	**0.11**	**197**	9	**0.0**	0.0	0	3	0.02	0.0	0	4	TL	0.0	157	TL		0.0	79	0.01		0.0	2	0.01	
20	23	0.2	0.13	**24**	10	**0.0**	0.0	2	4	0.13	0.0	1	9	0.45	**0.16**	**24**	TL		0.0	7	0.01		0.0	7	0.01	
20	23	0.3	0.15	**37**	10	**0.0**	0.0	0	1	0.03	0.0	7	10	906.97	**0.17**	**37**	TL		0.0	9	0.02		0.0	11	0.01	
20	23	0.5	**0.12**	**75**	10	**0.0**	0.0	0	1	**0.0**	0.0	5	6	TL	0.11	**75**	TL		0.0	28	0.01		0.0	21	0.01	
20	23	0.6	**0.11**	**101**	10	**0.0**	0.0	0	1	**0.0**	0.0	6	4	TL	**0.11**	**101**	TL		0.0	54	0.01		0.0	37	0.01	
20	23	0.7	0.11	**133**	10	**0.0**	0.0	0	1	**0.0**	0.01	1	2	TL	**0.12**	**133**	TL		0.0	71	0.01		0.0	18	0.01	
20	23	0.8	**0.09**	**161**	10	**0.0**	0.0	0	1	**0.0**	0.0	0	1	TL	0.01	144	TL		0.0	21	0.01		0.0	20	0.02	
20	35	0.2	0.09	**27**	10	**0.0**	0.0	0	2	0.07	0.0	2	7	621.58	**0.15**	**27**	TL		0.0	5	0.01		0.0	9	0.01	
20	35	0.3	0.08	**52**	10	**0.0**	0.0	0	1	0.03	0.0	8	7	TL	**0.13**	**52**	TL		0.0	13	0.01		0.0	17	0.01	
20	35	0.5	0.14	**112**	10	**0.0**	0.0	0	1	**0.0**	0.0	9	6	TL	**0.15**	**112**	TL		0.0	74	0.01		0.0	31	0.01	
20	35	0.6	0.15	**146**	10	**0.0**	0.0	0	1	**0.0**	0.0	0	3	TL	**0.16**	**146**	TL		0.0	72	0.01		0.0	38	0.01	
20	35	0.7	0.09	**193**	10	**0.0**	0.0	0	1	**0.0**	0.0	2	3	TL	0.03	182	TL		0.0	91	0.02		0.0	94	0.01	
20	35	0.8	**0.11**	**239**	10	**0.0**	0.0	1	1	**0.0**	0.0	1	1	TL	0.01	214	TL		0.0	8	0.01		0.0	9	0.01	
20	39	0.28	**0.06**	**72**	9	**0.0**	0.0	4	6	0.02	0.0	0	7	27.29	0.0	52	TL		0.0	26	0.02		0.0	1	0.49	
20	39	0.34	**0.11**	**89**	9	**0.0**	0.0	17	6	0.02	0.0	0	5	0.23	0.0	61	1.58		0.0	44	0.02		0.0	16	0.01	

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

Observing the values obtained by the models in the Obj column, we can see that the CEP and BEP models, which are edge-editing automatic clustering models, achieved the lowest values of resource intersection in their solution clusters, reaching on average 1.84% and 4% of the best-known values, respectively. The p-median$_{jac}$ and p-median$_{dis}$ models, which cluster their objects based on proximity to the centroid of each cluster, improved the intersection of their solutions compared to the edge-editing models, but still achieved an average 32.15% and 37.05% of the best-known values, respectively. In the majority of instances, these four models obtained $I_{asi} = 0.0$ in their solutions.

The Pairwise model proposed for ICP was able to solve every instance. The p-CEP model was able to solve only 8 instances within 1 hour of execution. However, considering its intermediate solutions, it found 22 optimal solutions for ICP. In the 14 instances where p-CEP did not reach the optimal value, there was a gap of 88.89%.

Analyzing the value of I_{asi}, we can see that the Pairwise model found the best value in only 22 instances, even though it found the optimal solution in all 36 instances. Looking at the other 14 instances, we can observe that the best I_{asi} value was found either by the Pairwise model or by p-CEP, but in all cases, the Obj value is the same. This means that there are multiple optimal solutions equally evaluated by the objective function of ICP but evaluated differently by I_{asi}, which takes into account the distance of each object to its nearest external cluster.

6. Final Considerations and Future Work

In this work we address the Intersection Clustering Problem, which aims to group objects according to their intersection of resources. In this sense, classical clustering models that use concepts of such as distance matrix, similarity or graph editing fail to generate a partitioning that maximizes the intersection of resources across their objects.

Starting from the proposition that there is an optimal partitioning for the ICP with up to two objects in each part, the present work introduces the concept of intersection graph to represent the connection between two objects and uses this concept to propose a new formulation for the ICP, called Pairwise Model.

The experiments carried out on artificial instances revealed that the proposed model, compared to a previous formulation, can obtain optimal solutions in considerably less time, including being able to solve all tested instances within the time limit. At the same time, compared to classical clustering models, the proposed model showed, on average, better metrics, measured by I_{asi}, in grouping objects according to their intersection of resources.

Future work includes proposing meta-heuristics for ICP and exploring applications of this clustering model in real problems, such as data classification.

References

Aidagulov, R. R., Glavatsky, S. T., and Mikhalev, A. V. (2022). Clustering models. *Journal of Mathematical Sciences*, 262:603–616.

Bai, L. and Liang, J. (2015). Cluster validity functions for categorical data: a solution-space perspective. *Data Mining and Knowledge Discovery*, 29.

Böcker, S. and Baumbach, J. (2013). Cluster editing. In Bonizzoni, P., Brattka, V., and Löwe, B., editors, *The Nature of Computation. Logic, Algorithms, Applications*, p. 33–44, Berlin, Heidelberg. Springer Berlin Heidelberg. ISBN 978-3-642-39053-1.

Bulhões, T., de Sousa Filho, G. F., Subramanian, A., and dos Anjos F. Cabral, L. (2017a). Branch-and-cut approaches for p-cluster editing. *Discrete Applied Mathematics*, 219:51–64. ISSN 0166-218X.

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br

Chaturvedi, A., Green, P. E., and Caroll, J. D. (2001). K-modes clustering. *Journal of classification*, 18:35–55.

Daskin, M. S., Maass, K. L., et al. (2015). Chapter 2-the p-median problem. *Locationscience. Springer.*

Galvão Jr., D. P. (2023). Identificação de fraudes em licitações: uma abordagem utilizando agrupamento por interseção. Master's thesis, Universidade Federal da Paraíba.

García Rodríguez, M. J., Rodríguez-Montequín, V., Ballesteros-Pérez, P., Love, P. E., and Signor, R. (2022). Collusion detection in public procurement auctions with machine learning algorithms. *Automation in Construction*, 133:104047. ISSN 0926-5805.

Gwalani, H., Tiwari, C., and Mikler, A. R. (2021). Evaluation of heuristics for the p-median problem: Scale and spatial demand distribution. *Computers, Environment and Urban Systems*, 88: 101656. ISSN 0198-9715.

Hennig, C. (2022). An empirical comparison and characterisation of nine popular clustering methods. *Advances in Data Analysis and Classification*, 16:201–229. ISSN 1862-5355.

Huang, J. Z. (1997). A fast clustering algorithm to cluster very large categorical data sets in data mining. In *Workshop on Research Issues on Data Mining and Knowledge Discovery*.

Huang, Z. (1998). Extensions to the k-means algorithm for clustering large data sets with categorical values. *Data Mining and Knowledge Discovery*, 2(3):283–304.

Lapointe, F.-J. and Legendre, P. (1994). A classification of pure malt scotch whiskies. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 43(1):237–257.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. In Cam, L. M. L. and Neyman, J., editors, *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, p. 281–297. University of California Press.

Nadif, M. and Govaert, G. (2013). *Co-clustering: models, algorithms and applications.* John Wiley & Sons. ISBN 978-1-848-21473-6.

Oyewole, G. J. and Thopil, G. A. (2022). Data clustering: application and trends. *Artificial Intelligence Review*.

Park, H.-S. and Jun, C.-H. (2009). A simple and fast algorithm for k-medoids clustering. *Expert Systems with Applications*, 36(2, Part 2):3336–3341. ISSN 0957-4174.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65. ISSN 0377-0427.

Sinaga, K. P. and Yang, M.-S. (2020). Unsupervised k-means clustering algorithm. *IEEE access*, 8: 80716–80727. ISSN 2169-3536.

Sousa Filho, G. F., Bulhões, T., Cabral, L. A. F., Ochi, L. S., and Protti, F. (2017). New heuristics for the bicluster editing problem. *Annals of Operations Research*, 258:781–814.

Sousa Filho, G. F., Bulhões, T., dos Anjos F. Cabral, L., Ochi, L. S., Protti, F., and Pinheiro, R. G. (2021). The biclique partitioning polytope. *Discrete Applied Mathematics*, 301:118–130. ISSN 0166-218X.

DOI: 10.59254/sbpo-2024-193740
https://proceedings.science/p/193740?lang=pt-br