The Version of Record of this manuscript has been published and is **freely available** in the International Journal of Systems Science: Operations & Logistics, 17/7/2024, https://www.tandfonline.com/doi/full/10.1080/23302674.2024.2378859

A Heuristic for Improving Clustering in Biomass Supply Chains

Ioannis T. Christou[*,#], Fragkoulis Psathas[+], Athanasios Rentizelas[+,1], Thanasis Papadakis[*], Paraskevas N. Georgiou[~], Despina Anastasopoulos[*], Pantelis Lappas[*,^]

[*]Research & Innovation Development, NetCompany-Intrasoft, Luxembourg, Luxembourg

[#]Dept of Information Technology, The American College of Greece, Athens, Greece

[+]School of Mechanical Engineering, National Technical University of Athens, Athens, Greece

~Dept of Mechanical Engineering & Aeronautics, University of Patras, Rio, Greece

[^]Dept of Statistics & Actuarial-Financial Mathematics, University of the Aegean, Samos, Greece

[1]Corresponding author, arent@mail.ntua.gr

Abstract

Clustering is commonly used in various fields such as statistics, geospatial analysis, and machine learning. In supply chain modelling, clustering is applied when the number of potential origins and/or destinations exceeds the solvable problem size. Related methods allow the reduction of the models' dimensionality, hence facilitating their solution in acceptable timeframes for business applications. The weighted minimum sum-of-square distances clustering problem (Weighted MSSC) is a typical problem encountered in many biomass supply chain management applications, where large numbers of fields exist. This task is usually approached with the weighted K-means heuristic algorithm. This study proposes a novel, more efficient algorithm for solving the occurring weighted sum-of-squared distances minimization problem in 2-dimensional Euclidean surface. The problem is formulated as a set-partitioning problem, and a column-generation inspired approach is applied, finding better solutions than the ones obtained from the weighted version of the K-means heuristic. Results from both benchmark datasets and a biomass supply chain case show that even for large values of K, the proposed approach consistently finds better solutions than the best solutions found by other heuristic algorithms. Ultimately, this study can contribute to more efficient clustering, which can lead to more realistic outcomes in supply chain optimization.

Keywords: Clustering, Weighted MSSC, Supply Chain, Biomass, Heuristics.

1.Introduction

Biomass is established as an important alternative to fossil fuels in the quest for renewable energy sources. Biomass plays a versatile role in energy systems, contributing not only to

power and heat generation [1] but also serving as a source for biofuels with widespread applications, particularly in the transportation sector. Currently, biomass accounts for over 70% of the renewable energy production globally, while it contributes to a 4% on the fuel demand in road transport. According to the European Union's (EU) long-term strategy, the objective is to achieve a climate-neutral EU by 2050 [2]. To this end, sustainable biofuels have the potential to play a decisive role in decarbonizing the transport sector (road, aviation, maritime etc.), while ensuring a high level of affordable and accessible transport connectivity [3]. The emphasis is especially on second-generation biofuels (known as advanced biofuels), which originate from dispersed sources, such as agricultural residues, biomass fractions from mixed municipal and industrial wastes, animal manure, wastes and residues from forestry, etc. According to the EU Renewable Energy Directive 2018/2001, advanced biofuels and biogas production are expected to increase at least to 1% of the final energy consumption in the transport sector in 2025 and at least to 3.5% in 2030 [4].

However, the economic viability of using biomass as a renewable energy source is affected by various factors, like uncertainty in feedstock availability, seasonal fluctuations in biomass production, and widespread geographical dispersion [5],[6]. Considering also the volatile and competitive global energy market, the continuous technological developments and the more stringent standards and commitments on promoting Sustainable Development Goals, the design of robust biomass supply chains becomes a highly demanding procedure needing a comprehensive system analysis. The overall challenge is to develop a decision-making framework able to amalgamate a multiplicity of factors, manage uncertainty and provide valuable key performance indicators [7].

A commonly applied approach to support decision-making in such complex supply chain schemes is optimization. Optimization is tightly coupled with studying, modeling and solving a multiplicity of industrial and energy problems at any functional level: short-, medium- and long-term planning [8]. Supply chain optimization models help the decision makers and interested stakeholders make the right strategic and tactical decisions across the upstream, midstream and downstream stages, in order to generate a considerable profit and preserve overall efficiency and sustainability [7]. Mathematical Programming provides the framework to construct such optimization models able to include a variety of feedstock types and sources, intermediate and final products, energy forms and product streams, technologies, and processes, while incorporating economic, environmental, and social aspects related to a typical biomass supply chain. These models can also consider various cost categories (harvesting, processing, transportation etc.) and special constraints while they can express the necessary conditions such as biomass availability and geographical dispersion. Furthermore, these models can have interoperability with Geographical Information Systems (GIS) which provide rich spatial data (e.g., physical locations of fields, warehouses, conversion facilities, roads, etc.) and are equipped with analytical capabilities (e.g. K-means clustering) [9].

The study and analysis of real-life biomass supply chains often includes large numbers of biomass fields and entails the expression of many series and types of variables and mathematical relationships, leading to the construction of detailed and large-scale models. When computational complexity increases, the commonly used exact solution algorithms, can become insufficient to obtain solutions in reasonable timeframes [10]. In such cases, a reduction of the size of the optimization problem, achieved by the reduction of the number of decision variables is required. This can be achieved by employing clustering and aggregation techniques, to formulate manageable and solvable biomass supply chain models [11].

Clustering is a technique commonly used in various fields beyond the biomass supply chain field, such as statistics, geospatial analysis, and machine learning, being tightly linked with the optimization concept. Specifically in supply chain modelling, clustering is naturally required when the number of potential origins and/or destinations is quite large and the native routing assignment significantly enhances the overall combinatorial optimization load and the solvability of the cost/distance minimization problem [12]. Transforming the initial depiction of supply chain nodes to an equivalent partitioned representation can expand the applicability and suitability spectrum in supply chain management gaining benefits in model generation and solution times.

The motivation for dealing with the clustering problem in biomass supply chains originated from the research and modeling objectives of the Horizon 2020 CERESiS project [13], which aims to produce sustainable liquid biofuels through the cultivation of suitable energy crops for the remediation and management of contaminated land. The optimization of the respective biomass supply chain is significantly affected by the inherent geographical dispersion of the biomass sources and the fluctuation of the feedstock supply. Common clustering approaches in the literature either employ the standard K-means algorithm or spatially divide the map into a grid, arbitrarily placing the coordinates of the aggregated field in the center. Both approaches overlook the total hectarage of each field, potentially leading to significantly inaccurate transportation cost estimates, hence affecting the accuracy of the whole supply chain modeling. This happens because these algorithms assume equal-sized clusters, while in real-world applications, biomass fields often have varying hectarage. It should be noted that similar issues exist in many other supply chain optimization contexts where a large number of upstream supply nodes/origins exist, such as recycling, waste collection, agri-food etc.

To address this issue, the present study introduces a novel weighted K-means algorithm accounting for both field coordinates as well as field size, resulting in a more accurate representation of transportation costs. Incorporating this innovative approach, the aim is to render large biomass supply chain models computationally feasible while also mitigating any inaccuracy linked to traditional and arbitrary clustering. The overall modeling improvement facilitates the end goal of drawing robust and economically viable biofuel or bioenergy production plans based on energy crops cultivated in geographically dispersed land parcels.

The methodological novelty lies in the development of a new heuristic approach to solve the weighted sum-of-squared distances minimization problem, that considers field size in the clustering process, thus leading to more accurate clustering results in respect to the real system compared to the weighted version of the well-known K-means heuristic, when the size of each field/origin is important. The application novelty lies in applying a heuristics approach for clustering that can consider the field size specifically in the biomass supply chain context. The proposed approach can be equally applied in other supply chain contexts, where the weight of each origin could denote quantities of materials available or other values, instead of the field size of the biomass supply chain problem.

This paper is structured as follows: Section 2 presents the literature review. Section 3 introduces the model formulation. Section 4 describes the column generation approach. Section 5 analyses the results of the computation experiments. Finally, section 6 concludes the study and discusses future research perspectives.

2. Literature Review

Numerous studies deal with the optimization of biomass supply chains considering various logistics and processing stages involved, from the energy crops cultivation or feedstock supply up to biofuel or bioenergy production. For example, Fattahi and Govindan [14] presented a multi-stage stochastic program that minimizes the cost of a biofuel supply chain to satisfy the demand, accounting for seasonal and stochastic biomass availability, and incorporating environmental and social impact. In a study by Momenitabar et al. [15] the biofuel supply chain was optimized with a focus on sustainability criteria to determine the optimal siting of the conversion facilities. Sosa et al. [16] developed a linear programming tool taking into account moisture in biomass, showing its impact on harvest area timeline and storage, as well as transportation elements within the supply chain, including strategies to optimize truckloads within legal weight and volume constraints. In the abovementioned studies as well as in much of the relevant literature, a detailed record of biomass/feedstock field data is often lacking or unavailable due to the inherent difficulty of gathering input from dispersed biomass fields. This is the reason that detailed data from biomass supply hubs' location (e.g. fields) is usually consolidated or assumed to be available in a single geographical point, hence limiting the accuracy of the modeling process outcomes. However, in instances where this abundance of information is available and is used as the starting point for the biomass supply chain modeling, a method of field aggregation may become necessary for grouping the fields, to reduce the overall model complexity and allow its solution in reasonable times using mathematical optimization.

For this purpose, clustering algorithms are important and inextricably linked with biomass supply chain design, management and optimization. A common approach in the biomass supply chain literature lies in treating the substantial volume of data from supply hubs, which renders the supply chain optimization problem complex and in certain cases unsolvable. For example, O'Neill et al. [17] developed a stochastic mixed-integer linear programming (MILP) biofuel supply chain model, aiming simultaneously at the maximization of economic performance and the minimization of the environmental impact. Their dataset contained over 40,000 fields, which rendered the problem unsolvable with exact optimization methods. To reduce computational cost, clustering was employed at resolutions of 25 km^2, 100 km^2, 225 km^2, 400 km^2, 625 km^2 and county level. Uen and Rodríguez [18] presented a MILP model maximizing the net present value of a food waste supply chain for renewable energy generation. The model involved a large dataset with 102 food waste sources and 621 candidate conversion facilities sites (anaerobic digestion and co-digestion), resulting in high computational cost. To address this issue, they used the K-means algorithm to create more representative centroids based on spatial proximity; the final cluster selection consisted of 9 clusters for 291 aerobic digestion candidate sites and 12 clusters for the 330 co-digestion candidates. Psathas et al. [6] developed a biomass-to-biofuel supply chain that integrated large-scale centralized, smaller-scale decentralized as well as mobile processing facilities. The complex MILP model, applied to a use case involving over 3,000 biomass fields, showcased the inability to achieve a solution for such a high-dimension optimization problem, prompting the need for clustering the fields in a more manageable group of 16 clusters using the K-means algorithm.

The application of clustering algorithms in the context of biomass and bioenergy is also apparent in Vehicle Routing Problems (VRPs). Ayoub et al. [19] developed a bioenergy decision support system which combined optimization and simulation for the design of bioenergy

production. In their system, they employed fuzzy C-means clustering to define the location and optimal size of storage and bioenergy conversion facilities based on biomass collection points, with the end goal being the minimization of transportation costs, CO_2 emissions and number of workers. Zamar et al. [20] applied near-neighborhood search and a modified K-means clustering technique for the route optimization in the bale collection problem, aiming to minimize travel time and fuel consumption. The constraint of this new K-means algorithm limited the temporary storage sites to valid locations. Zamar et al. [21] presented a stochastic VRP model for the biomass residue collection from a set of sawmills to identify the optimal routing schedule. Due to variations in biomass quality, the aim was to maximize the ratio of energy returned on energy invested. This combination of uncertainty added into the simulation for the VRP required large computational effort. To make the problem more computationally manageable, the 25 sawmills were spatially clustered based on the travel distance matrix using the K-medoids technique into 4 distinct groups and the itinerary was then rerouted to the center of each cluster.

Based on the above, clustering algorithms play a vital role in addressing challenges associated with upstream, midstream and downstream supply chain stages in biomass supply chain optimization, where typically a large number of distinct locations are involved in real-life systems. They can identify potential processing hubs and deal with large datasets in MILP models with high complexity. Clustering algorithms, such as GIS proximity-based clustering, K-means, K-medoids, C-means prove to be essential in creating manageable data subsets, enabling the reduction of computational burden and ensuring reasonable solution times, allowing the construction and use of decision support systems suitable for business applications.

Very limited applications of the Weighted-MSSC problem can be found; one of them presenting the weighted version of the classical K-means algorithm for a use-case regarding locating distribution centers for COVID-19 vaccines throughout the US [36]. We use the related dataset in our computational results section. Similarly, the query “weighted MSSC clustering” returns a number of papers on clustering ensembles where different clusters have different weights, but individual data points have no weights assigned to them as in our case. Overall, even though there have been a number of papers advancing the state-of-the-art in MSSC, with the exception of the article [36] none have studied the weighted version of the problem. In the next three paragraphs therefore, we revisit a number of papers on the MSSC problem instead.

In [22], the authors present the well-known “K-Means++” heuristic for initializing the centers of the K-means algorithm, by modifying another popular heuristic, the so-called “farthest centers” heuristic, which works by selecting at random one point from the dataset as first center, and then repeatedly chooses the point that is “the furthest distance” away from the current set of centers as the next center. Because this heuristic was prone to choosing outliers in the dataset as the initial centers, the K-Means++ algorithm works by selecting at random but with probability proportional to the shortest distance of any point from the currently selected centers the next point to be selected as the next center. This randomization procedure guarantees theoretical good properties of the initial clustering for the K-means algorithm. Then, in [23], the authors remove the serial bottleneck of the above-described procedure by fully parallelizing it to allow utilizing as many CPU cores as there are available in a cluster, and therefore allow much larger datasets than was possible before to be clustered efficiently; this K-Means variation is denoted as the “K-Means ||”.

Aloise et al. in [24] presented an improved column generation method for the MSSC that was able to solve to optimality problems with up to 2400 points in 2-D Euclidean space and K taking values up to 400. Very recently, the authors of [25] presented a Semi-Definite Programming approach (SDP) to the MSSC, that allows them to solve to provable optimality a number of problems that were not solvable before. In [26], Burgard et al. use advanced mixed integer programming techniques such as cutting planes (including cardinality cuts, outer approximation cuts and bary-center propagation) to enhance the performance of the Open-Source solver SCIP.

In [27], Peng and Xia present a "convexity cut method" to solve to global optimality the MSSC problem. They use CPLEX for solving their corresponding cutting plane problem; they report results for problems up to 4600 data points, but they only work with very small values of K. Finally, in a Master's thesis [28], the author presents a number of clustering algorithms for forest harvesting, mentioning the classical K-means algorithm as one possible method for solving their (unweighted) problem.

According to the literature review, clustering is not fully exploited under the concept of comprehensive biomass supply chain management. In the upstream level, where typically biomass fields of varying size exist, the existing clustering approaches are employed only to treat the location dimensionality while the hectarage (area) factor is ignored. The absence of incorporating hectarage in the cluster calculation may result in forming groups that don't adequately represent the initial set of fields, diminishing accuracy and affecting many aspects of the design and optimization of the whole biomass supply chain such as the transportation costs, the location and capacity of midstream processing facilities etc. The present study aims to fill this methodological gap and develop an effective clustering algorithm that allows more precise representation of clustered sites considering both their size and location. Enhancing and improving the clustering process facilitates the management of more complex problems without increasing solution times, while preserving accuracy and generating robust results which support a more informed decision-making.

3.Model Formulation

The clustering problem dealt in this study is the Weighted MSSC problem, which generalizes the well-known MSSC problem [29-30] as follows: consider a finite set of n points $S = \{s_1, \dots, s_n\}$ in d-dimensional plane, with associated positive weights $w_1, \dots w_n$.The problem is to find $K > 1$ "centers" in the plane that will minimize the total weighted sum of square distances of each of the n points to their closest center multiplied by the point's corresponding weight. It is worth noting that when all the points have equal weights, the problem reduces to the well-known minimization of sum of square errors problem that appears in many problems in signal processing (code-book design), unsupervised learning (K-Means clustering), supply chain design, and so on.

The problem can be stated then as follows:

We seek to find a partition C of the set S into K disjoint sets $C_i, i = 1, \dots K: C_i \cap C_j = \emptyset \ \forall i \neq j, \ i, j = 1 \dots K, \ \cup_{i=1}^{K} C_i = S$ that minimizes the function $f(C) = \sum_{i=1}^{K} \sum_{s_m \in C_i} w_m \|s_m - \overline{s_i}\|^2$ where the point $\overline{s_i}$ is such that it minimizes the sum of weighted distances of each point in the cluster C_i to it.

Lemma 1: *Given the points s_m in a partition cluster C_i, with accompanying weights w_m, the optimal point in the plane that minimizes the sum of weighted distances of the points in the partition cluster to it, is the (weighted) barycenter* $\overline{s_i} = \frac{\sum_{s_m \in C_i} w_m s_m}{\sum_{m:s_m \in C_i} w_m}$.

Proof: It is enough to show that for the function

$$f(C_i, x) = \sum_{s_m \in C_i} w_m \|s_m - x\|^2 \quad (1)$$

the gradient $\nabla_x f(C_i, \bar{s}_i) = 0$. The gradient of the function at any point x is

$$\nabla_x f(C_i, x) = -2 \sum_{s_m \in C_i} w_m (s_m - x) \quad (2)$$

Setting the above expression to zero, we obtain as the only solution the vector

$$x^* = \overline{s_i} = \frac{\sum_{s_m \in C_i} w_m s_m}{\sum_{m:s_m \in C_i} w_m} \quad (3)$$

QED.

Therefore, mathematically, we seek to optimize the following problem (P):

$$\min_C \sum_{i=1}^K \sum_{s_m \in C_i} w_m \|s_m - \overline{s_i}\|^2 \quad (P)$$

subject to:

$$\begin{cases} \overline{s_i} = \frac{\sum_{s_m \in C_i} w_m s_m}{\sum_{m:s_m \in C_i} w_m}, i = 1 \dots K \\ C_i \cap C_j = \emptyset, \forall i \neq j, \ i, j = 1 \dots K \\ \bigcup_{i=1}^K C_i = S \end{cases}$$

4.A Column Generation Approach to Solving the Weighted MSSC Problem

The minimum sum of square errors clustering problem (MSSC) has been the subject of intense study for many decades [29]. It is a combinatorial problem by nature: given a finite set $S = \{s_1, s_2, \dots s_n\} \in \mathbb{R}^d$ of n data points in some vector space, the problem is to find a partition of the data points in exactly $K > 1$ disjoint partition clusters so that the sum of the square distances of each point to its closest center (cluster mean point) is minimized. The problem is known to be NP-Hard [31]. The MSSC problem is clearly a special case of the *weighted* MSSC problem where the weight of each data point is set to 1.

The most known and frequently used algorithm for solving the standard MSSC problem is the K-Means algorithm [24, 26, 27, 28], an expectation-maximization (EM) type algorithm that works by iterating two major phases: an *expectation* step that assigns points to their closest current centroid (mean), and a *maximization* step that re-computes the centroids as the mean vector of every data point in a cluster, optimizing the objective function under the current assignments. The algorithm iterates until some convergence criteria is met, or -for very large problems- for a fixed number of iterations. For most problems, the initial choice of cluster centers is a major factor that can determine the quality of the final solution as well as the speed of convergence [22, 23].

K-means is a heuristic algorithm that does not guarantee the optimal solution of an MSSC problem, even though it often finds high quality solutions, at least for modest values of K. As described above, the K-means algorithm does not solve the weighted MSSC problem, however it is not difficult to state a variant that attempts to. Essentially, the only modification to the K-means algorithm that is needed, is in the M-step (step 2) where we compute the new cluster centers. Given Lemma 1 in section 2, we see that the modification needed to be done to the original K-means algorithm to result in a weighted K-means algorithm is to compute the center of each cluster in step 2 according to eq. (3).

However, whereas the K-means algorithm, and the modified weighted K-means algorithm usually provide acceptable results for problems with small number of partitions required (say, up to a few tens), the same is not usually the case for larger number of partitions [33]. For such problem instances, we propose a column-generation type approach to a set covering formulation of the problem, in a spirit similar to the approach described in [34] for the standard MSSC problem.

In particular, notice first that the weighted MSSC problem, being also a combinatorial optimization problem, can be written as a set partitioning problem. Consider all possible subsets of the set S and denote by A the $n \times 2^n$ matrix containing ones and zeros, whose columns represent all those possible subsets: every column of the matrix has exactly n components, and the i-th component indicates whether the data point s_i is contained or not in the subset. Given this (rather big) matrix, the weighted MSSC problem can be written down as the set-partitioning problem.

$$\min_x c^T x \qquad \text{(SPP)}$$

$$\text{subject to:} \begin{cases} Ax = e \\ e^T x = K \\ x \in \{0,1\}^n \end{cases}$$

where the column vector $e \in \mathbb{R}^n \equiv [1,1, \dots 1]^T$ and the cost-vector $c = [c_1, c_2, \dots c_{2^n}]^T \in \mathbb{R}^{2^n}$ contains the (weighted) costs of all subsets of the set S. The set-partition problem formulation (SPP) asks to pick exactly K subsets of the set S (constraint $e^T x = K$ and $x_i \in \{0,1\}\ for\ all\ i = 1, \dots 2^n$) such that they completely cover the set S without any overlap (constraints $Ax = e$) and minimize the total weighted cost of each selected subset.

Solving the problem (SPP) is not any easier than solving the problem (P), and both problems are NP-Hard. Even further, just writing down the initial model formulation of the problem (SPP) requires explicitly enumerating and storing all 2^n possible subsets of S which is clearly infeasible for even small values of n. We therefore resort to a column-generation approach (originally described in [34] for the standard MSSC problem) whereby we begin with a base matrix B that contains only a few of the columns of the full matrix A, and afterwards we solve the following variant of the initial problem (SPP), called (SCP_R):

$$\min_x c^T x \qquad (SCP_R)$$

$$\text{subject to:} \begin{cases} Bx \geq e \\ e^T x = K \\ x \in \{0,1\}^n \end{cases}$$

The components of the column-vector c again correspond to the weighted clustering cost of the corresponding subsets described in the columns of the reduced matrix B, only now we

seek to solve the *set covering* version of the problem (constraints $Bx \geq e$) which means we don't care if some data points appear in more than one subsets selected in the final solution, as long as each point appears in at least one selected subset.

In case there exists at least one data point s_{dup} that appears in more than one selected subset in the solution of (SCP_R), we remove it from every selected subset in which it appears except the subset from which if we removed s_{dup} the weighted cluster cost increase would be the smallest among all other selected subsets that contain it; ties are arbitrarily broken. It is easy to see that such a procedure results in strictly lowering the objective function value of the weighted MSSC problem.

Given the new clusters that represent a valid clustering solution to the weighted MSSC, though it may or may not be optimal, *we generate more columns to add to our base matrix* B by *adding the new subsets that we created in the conversion process described above* (if any), and then by running the weighted K-means algorithm on the solution found by the previous step, and *adding all new clusters created during the run of the weighted K-means algorithm* to the new base matrix B. We then solve the new problem (SCP_R) with the new base matrix B again, and repeat this process until in one full iteration we no longer improve on the final objective value of the problem.

The entire algorithm, called W-EXAMCE, is a variant of the EXAMCE algorithm specified in [34] adapted to work with weighted clustering, is specified in Fig. 1 and is available from https://www.github.com/ioannischristou/weighted_clustering.

<Insert Figure 1 here>

The procedure Expand(C,τ) expands the set of available solutions to consider in step 5 by producing a (possibly empty) set of clusters that are *perturbations* of the input cluster C. We use a simple heuristic whereby we choose the τ closest neighbors to the center of C that are not in C, as well as the τ members of C that are farthest from the center of C, with τ being a small user-defined parameter (set after experimentation equal to 5). Distances here are weighted, meaning that the distance between a cluster center c and a data point s_i with associated weight w_i is the quantity $\sqrt{w_i}\|c - s_i\|$. We create and return a sequence of 2τ new clusters $N_1^+ \subset N_2^+ \ldots \subset N_\tau^+, N_1^- \supset N_2^- \ldots \supset N_\tau^-$ where each of the N_i^+ clusters contains all points in C plus up to the i-th closest non-member neighbor of C, and each of the N_i^- clusters contains all points in C except up to the i-th farthest member of C.

It is easy to see that in every iteration in steps 3-10 of the W-EXAMCE algorithm, the solution C'' that is found is at least as good as the corresponding solution from the previous iteration:

Lemma 2: *The solutions C'' found in step 7 of the W-EXAMCE algorithm are non-increasing.*

Proof: Notice that the columns of the matrix B in problem (SCP_R) to be solved at each iteration always form a super-set of the columns of the same matrix in the previous iteration, even after that previous matrix is augmented by the solution found in the previous execution of step 7. Therefore, the solution of the problem in step 5 is always at least as good as the solution of step 7 found in the previous iteration. And since both the Rm_Dup(.) procedure in step 6 by design can only further reduce the objective value of the problem, and the same is

true for the application of the W-K-Means algorithm in step 7, the solution C'' found in step 7 at each iteration is always monotonically decreasing. **QED**.

Because of the above monotone convergence of the algorithm (convergence is guaranteed since the solution sequence is monotonically decreasing, and is bounded from below by zero), we have the following:

Corollary 3: *The W-EXAMCE algorithm converges in a finite number of iterations.*

Proof: The convergence of the algorithm is already established. The finiteness of the number of steps follows from the fact that there are only finitely many different partitions to partition the data, and the algorithm stops as soon as no progress is made in two consecutive iterations. **QED.**

A small worked example of the W-EXAMCE algorithm is presented in Appendix A.

5.Computational Results

In all experiments below, we use the GUROBI optimizer (v. 11) to solve the SCP problem in step 5 of the W-EXAMCE algorithm in fig. 1. We use a PC with an Intel Core i9-10920X CPU running at 4.5GHz, equipped with 64GB RAM, running Windows 11. Both the standard K-means and the scalable K-Means|| algorithm implementations are fully parallelized using Java threads and utilize all 24 virtual CPU cores that the CPU offers. All algorithms are written in Java.

In Table 1, the meta-data for the major datasets used for our experiments are presented. All datasets represent points in Euclidean 2-dimensional plane. Due to the lack of publicly available biomass-related land plot datasets, several more datasets from well-known libraries and other sector applications were used to demonstrate the performance of the proposed method. In Table 2 we present the results obtained from 4 datasets in the TSPLIB that are commonly used to benchmark algorithms for the standard MSSC problem (see [30, 33, 34]), with attached random positive weights for each data point. We compare results from applying standard K-means with 100 restarts, with the scalable K-means|| algorithm with 100 restarts, and the W-EXAMCE algorithm that uses the clusters obtained from 10 restarts of the standard K-means algorithm as base clusters. The column labeled "Soln Improvement%" under the W-EXAMCE results columns lists the percentage improvement of W-EXAMCE over the baseline K-means algorithm. As can be seen from the results, W-EXAMCE is always the clear winner of the three algorithms even though it takes a little more time to complete.

In Table 3, we present results from one large-scale use-case for phyto-remediation-based supply chain management, with real land plot data from the Scottish Vacant and Derelict Land Register [30]: the dataset contains the coordinates of 3,398 previously-developed brownfield land plots, in the British National Grid Coordinate system, a *plane* coordinate system that is based upon a Transverse Mercator projection. The dataset also contains the area that each plot covers that serves as the weight of each data point for our clustering purposes. This dataset refers to one of the phytoremediation use cases of the project CERESiS, where the objective was to optimise the respective supply chain of energy crops grown in potentially contaminated land to produce biofuels, while at the same time decontaminating the soil. The dataset is visualized in fig. 2, however the size of each land plot is not drawn to scale.

Finally, to assess the algorithm on another real-life large dataset, in Table 4 we present results using the public (weighted) dataset created for determining the best locations for placing

distribution and vaccination centers for the COVID-19 pandemic in the USA. This dataset contains 4478 data points in 2D. We have converted the original standard lat-lon coordinates of the dataset into a flat grid using Mercator Projection (UTM), and normalized the weights of the original dataset to sum up to 1 (otherwise, the sum of the weighted square distances would lead to arithmetic overflows due to resulting large numbers.)

<Insert Figure 2 here>

<Insert Table 1 here>

<Insert Table 2 here>

In Fig. 3 we show the percentage gap in solution quality between the proposed W-EXAMCE and the weighted K-means with 100 restarts on the Scotland Vacant and Derelict Land Register data; as can be seen clearly both from Fig. 3 and from Tables 2, 3 and 4, the solution found by W-EXAMCE is always superior to that found by 100 restarts of the K-means or the scalable K-Means|| algorithms. It can also be observed that the superiority of the W-EXAMCE algorithm is greatly enhanced when the number of partitions increases, while it requires more time to be solved; however, the time needed is reasonable for practical applications. By employing the proposed algorithm, a more efficient clustering is achieved, in order to support the supply chain optimization process that uses the clusters as input.

<Insert Figure 3 here>

<Insert Table 3 here>

<Insert Table 4 here>

6. Conclusions and Future Directions

This study proposed a novel, more effective clustering method for use within biomass supply chain optimization modeling, as well as for any other case where clustering is required within supply chain analysis and optimization context. In the context of biomass supply chain analysis, clustering is not often utilized; instead, clustering primarily finds applications in VRP problems (such as the “cluster-first-route-second” heuristic by Fisher and Jaikumar [37]), where weights are not usually taken into account.

The proposed algorithm (W-EXAMCE) for the weighted MSSC problem proved to be very effective in identifying more representative clusters of biomass fields than the commonly used technique of the weighted K-means and the scalable K-Means II. The solution quality gap between the proposed algorithm against the weighted K-means increases significantly as the number of partitions increases. Therefore, when more refined supply chain analysis is required, the proposed algorithm can offer more accurate clusters, leading to more accurate transportation cost estimates, hence more accurate supply chain optimization outcomes.

The applicability of the proposed algorithm for the weighted MSSC problem (W-EXAMCE) in biomass supply chain modeling, where exact locations of fields or biomass hubs are known, has been demonstrated for the Scottish Vacant and Derelict Land Register dataset (used as a use case in the CERESiS project [13]), the US COVID-19 cases dataset [36], and several TSPLIB datasets (with random weights). The proposed algorithm offers a viable option for increasing the optimization results’ accuracy. Given the spatial dispersion, typically low availability of

biomass feedstock, and diverse land hectarage of the fields under analysis, a refined approach is necessary, since simple aggregation and clustering techniques may introduce inaccuracies in the results. Here, the weighted K-Means algorithm provides robust solutions for typical biomass to biofuels or bioenergy supply chains, offering a detailed grouped view of the fields based on hectarage that is essential for effective decision-making.

The proposed W-EXAMCE algorithm could have wider applicability in all supply chain modeling contexts where many upstream, midstream or downstream nodes are involved, provided that each node of the physical system carries a different weight. In this case, using the proposed algorithm would lead to more accurate clustering compared to the typical K-means algorithm, ultimately leading to supply chain optimization outputs that are more relevant and based on more accurate inputs.

In the future, the W-EXAMCE algorithm could be applied in more diverse datasets of biomass supply chains, to understand its performance gap and expected benefits in a wider set of conditions. Its applicability could also be tested in other supply chain contexts to demonstrate its potential for generalization.

Acknowledgements

This study is part of the CERESiS project which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101006717.

Declaration of interest statement

The authors report there are no competing interests to declare.

Data availability statement

The data in this study can be provided upon request from the authors by emailing them. The entire W-EXAMCE algorithm is available from https://www.github.com/ioannischristou/weighted_clustering

References

[1] E. Tziolas, B. Manos, and T. Bournaris, "Planning of agro-energy districts for optimum farm income and biomass energy from crops residues", *Oper. Res.*, vol. 17, no. 2, pp. 535–546, 2017, doi: 10.1007/s12351-016-0236-y.

[2] UNFCCC, "Long-term low greenhouse gas emission development strategy of the European Union and its Member States", *Eur. Comm.*, vol. 2019, no. March, pp. 1–7, 2020, [Online]. Available: http://www4.unfccc.int/submissions/INDC/Published Documents/Latvia/1/LV-03-06-EU INDC.pdf.

[3] EP, "The European Green Deal - European Parliament resolution of 15 January 2020 on the European Green Deal (2019/2956(RSP))", 2019.

[4] IEA, "Technology roadmap: delivering sustainable bioenergy. Paris, France; 2017", *Phyton (B. Aires).*, vol. 63, pp. 87–91, 2017.

[5] S. S. Hassan, G. A. Williams, and A. K. Jaiswal, "Moving towards the second generation of lignocellulosic biorefineries in the EU: Drivers, challenges, and opportunities", *Renew. Sustain. Energy Rev.*, vol. 101, no. December 2018, pp. 590–

599, 2019, doi: 10.1016/j.rser.2018.11.041.

[6] F. Psathas, P. N. Georgiou, and A. Rentizelas, “Optimizing the Design of a Biomass-to-Biofuel Supply Chain Network Using a Decentralized Processing Approach”, *Energies*, vol. 15, no. 14, 2022, doi: 10.3390/en15145001.

[7] O. Abdussalam, N. Fello, and A. Chaabane, “Exploring options for carbon abatement in the petroleum sector: a supply chain optimization-based approach”, *Int. J. Syst. Sci. Oper. Logist.*, vol. 10, no. 1, 2023, doi: 10.1080/23302674.2021.2005174.

[8] I. E. Grossmann, “Advances in mathematical programming models for enterprise-wide optimization”, *Comput. Chem. Eng.*, vol. 47, pp. 2–18, 2012, doi: 10.1016/j.compchemeng.2012.06.038.

[9] D. F. Lozano-García, J. E. Santibañez-Aguilar, F. J. Lozano, and A. Flores-Tlacuahuac, “GIS-based modeling of residual biomass availability for energy and production in Mexico”, *Renew. Sustain. Energy Rev.*, vol. 120, no. April 2019, 2020, doi: 10.1016/j.rser.2019.109610.

[10] M. Martín, M. Taifouris, and G. Galán, “Lignocellulosic biorefineries: A multiscale approach for resource exploitation”, *Bioresour. Technol.*, vol. 385, no. June, p. 129397, 2023, doi: 10.1016/j.biortech.2023.129397.

[11] S. Potrč, L. Čuček, M. Martin, and Z. Kravanja, “Synthesis of european union biorefinery supply networks considering sustainability objectives”, *Processes*, vol. 8, no. 12, pp. 1–25, 2020, doi: 10.3390/pr8121588.

[12] T. Derya, B. Keçeci, and E. Dinler, “Selective clustered traveling salesman problem”, *Int. J. Syst. Sci. Oper. Logist.*, vol. 10, no. 1, 2023, doi: 10.1080/23302674.2023.2235266.

[13] CERESiS, “ContaminatEd land Remediation through Energy crops for Soil improvement to liquid biofuel Strategies". [Online] Available at: https://ceresis.eu/ (accessed Dec. 17, 2023).

[14] M. Fattahi and K. Govindan, “A multi-stage stochastic program for the sustainable design of biofuel supply chain networks under biomass supply uncertainty and disruption risk: A real-life case study”, *Transp. Res. Part E Logist. Transp. Rev.*, vol. 118, no. September, pp. 534–567, 2018, doi: 10.1016/j.tre.2018.08.008.

[15] M. Momenitabar, Z. Dehdari Ebrahimi, A. Abdollahi, W. Helmi, K. Bengtson, and P. Ghasemi, “An integrated machine learning and quantitative optimization method for designing sustainable bioethanol supply chain networks”, *Decis. Anal. J.*, vol. 7, no. November 2022, 2023, doi: 10.1016/j.dajour.2023.100236.

[16] A. Sosa, M. Acuna, K. McDonnell, and G. Devlin, “Controlling moisture content and truck configurations to model and optimise biomass supply chain logistics in Ireland”, *Appl. Energy*, vol. 137, pp. 338–351, 2015, doi: 10.1016/j.apenergy.2014.10.018.

[17] E. G. O’Neill, R. A. Martinez-Feria, B. Basso, and C. T. Maravelias, “Integrated spatially explicit landscape and cellulosic biofuel supply chain optimization under biomass yield uncertainty”, *Comput. Chem. Eng.*, vol. 160, 2022, doi: 10.1016/j.compchemeng.2022.107724.

[18] T. S. Uen and L. F. Rodríguez, “An integrated approach for sustainable food waste management towards renewable resource production and GHG reduction”, *J. Clean. Prod.*, vol. 412, no. January, 2023, doi: 10.1016/j.jclepro.2023.137251.

[19] N. Ayoub, R. Martins, K. Wang, H. Seki, and Y. Naka, “Two levels decision system for efficient planning and implementation of bioenergy production”, *Energy Convers. Manag.*, vol. 48, no. 3, pp. 709–723, 2007, doi: 10.1016/j.enconman.2006.09.012.

[20] D. S. Zamar, G. Bhushan, and S. Sokhansanj, “A Constrained K-Means and Nearest Neighbor Approach for Route Optimization in the Bale Collection Problem”, vol. 1, pp. 12125–12130, 2017.

[21] D. S. Zamar, B. Gopaluni, and S. Sokhansanj, “Optimization of sawmill residues collection for bioenergy production”, *Appl. Energy*, vol. 202, pp. 487–495, 2017, doi: 10.1016/j.apenergy.2017.05.156.

[22] D. Arthur and S. Vassilvitskii, “K-means++: The advantages of careful seeding”, *Proc. Annu. ACM-SIAM Symp. Discret. Algorithms*, vol. 07-09-January-2007, pp. 1027–1035, 2007.

[23] B. Bahmani, B. Moseley, A. Vattani, R. Kumar, and S. Vassilvitskii, “Scalable κ-means++”, *Proc. VLDB Endow.*, vol. 5, no. 7, pp. 622–633, 2012, doi: 10.14778/2180912.2180915.

[24] D. Aloise, P. Hansen, L. Liberti, "An improved column generation algorithm for minimum sum of squares clustering", Mathematical Programming Ser. A, 131:195--220, 2012.

[25] V. Piccialli, A. M. Sudoso, A. Wiegele, "SOS-SDP: An exact solver for minimum sum of squares clustering", INFORMS Journal on Computing, 34(4):2144--2162, 2021.

[26] J.P. Burgard, C.M. Costa, C. Hojny, T. Kleinert, M. Schmidt, "Mixed-integer programming techniques for the minimum sum of squares clustering problem", Journal of Global Optimization, 87:133--189, 2023.

[27] J. Peng, Y. Xia, "A cutting algorithm for the minimum sum of squared error clustering", Proc. 2005 SIAM Conference on Data Mining, pp. 150--160, 2005.

[28] M. Usman, "An application of geospatial clustering for assets optimisation in forest harvesting", M.Sc. Thesis, Tampere University, 2023.

[29] S. Theodoridis and K. Koutroumbas, *Pattern recognition*, 2nd ed. San Diego, CA: Academic Press, 2006.

[30] P. Hansen and N. Mladenović“, J-Means: a new local search heuristic for minimum sum of squares clustering”, *Pattern Recognit.*, vol. 34, no. 2, pp. 405–413, 2001, doi: 10.1016/S0031-3203(99)00216-2.

[31] I. T. Christou, *Quantitative Methods in Supply Chain Management: Models and Algorithms*. London, UK: Springer, 2011.

[32] S. P. Lloyd, “Least Squares Quantization in PCM”, *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, 1982, doi: 10.1109/TIT.1982.1056489.

[33] M. Laszlo and S. Mukherjee, “A genetic algorithm using hyper-quadtrees for low-dimensional k-means clustering”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 533–543, 2006, doi: 10.1109/TPAMI.2006.66.

[34] I. T. Christou, “Coordination of cluster ensembles via exact methods”, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 279–293, 2011, doi: 10.1109/TPAMI.2010.85.

[35] SVDLS, “Scottish Vacant and Derelict Land Survey”, 2022. https://www.gov.scot/publications/scottish-vacant-and-derelict-land-survey---site-register/ (accessed Nov. 25, 2023).

[36] https://towardsdatascience.com/determining-optimal-distribution-centers-locations-using-weighted-k-means-1dd099726307 [Online] (accessed April 2, 2024)

[37] M. Fisher and R. Jaikuman, “A generalized assignment heuristic for vehicle routing”, *Networks*, vol. 11, pp. 109-124, 1981.

Appendix A

For a better understanding of the workings of the W-EXAMCE Algorithm described in Fig. 1, we demonstrate its application with a small example.

In Fig. 4 below, we show a dataset of 10 points in 2 dimensions. Each data-point has an associated weight according to Table 5 below, and the “X” that is used to draw each point in Fig. 4 is drawn *approximately* according to the scale shown in Table 5.

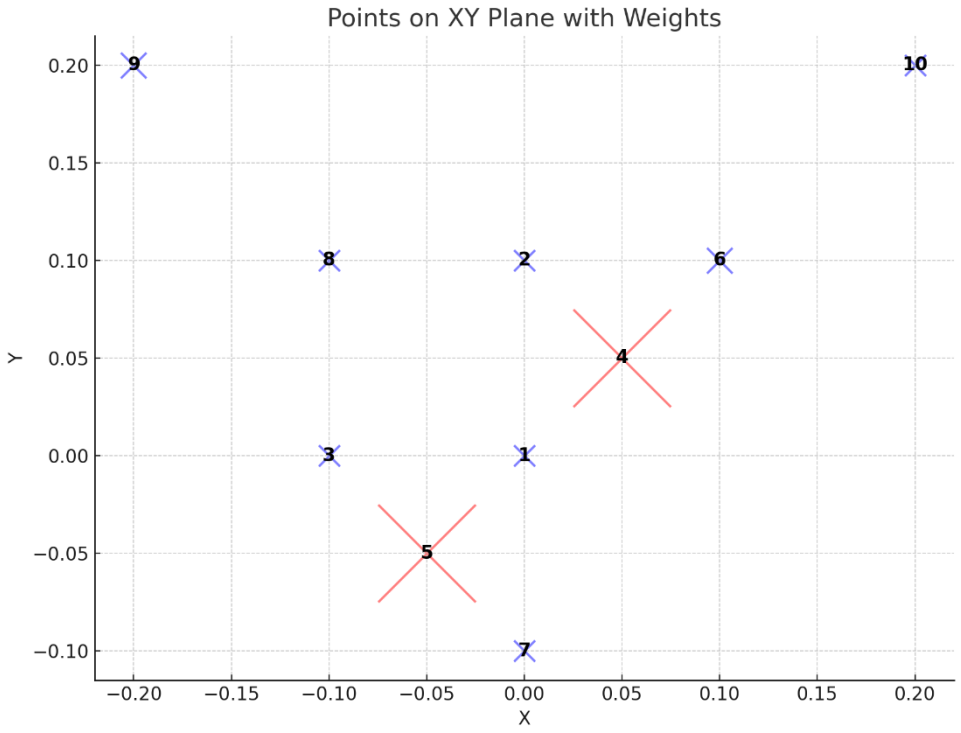

Figure 1: Toy dataset with 10 Points in 2D to trace the steps of algorithm W-EXAMCE

Table 5: Toy dataset weights

Point Index	Weight
1	0.01
2	0.01
3	0.01
4	0.45
5	0.45
6	0.02
7	0.01
8	0.01
9	0.02
10	0.01

We are going to trace the execution of the algorithm W-EXAMCE with k=4. For simplicity, we also set τ=0 for our procedure $Expand(C, \tau)$ so that in essence the procedure does not do anything. We produce the initial matrix A_B by executing the clusterers in the set Ba = (W-KMeans(4), W-KMeans(4), W-KMeans(4), W-KMeans(3), W-KMeans(3), W-KMeans(5), W-KMeans(5)); this means that we apply 3 times the W-KMeans algorithm with k=4 (starting from a different random initial solution each time), and then we apply 2 times the W-KMeans algorithm with k=3, and with k=5 (step 1). This results in a total of 28 clusters whose members and total weighted costs are shown in Table 6 below. The clustering solutions are color-coded in Table 6 below, so that the first clustering solution comprises of the clusters 1-4 (inclusive), the second solution comprises of clusters 5-8 and so on.

Table 6: Results of applying clusterers in the set Ba to the dataset.

Cluster Index	Cluster Members	Weighted Cluster Cost
1	2, 4, 6, 10	5.73E-04
2	1, 5, 7	9.79E-05
3	9	0
4	3, 8	5.00E-05
5	3, 5, 7	1.00E-04
6	8, 9	1.33E-04
7	2, 4, 6, 10	5.73E-04
8	1	0
9	1, 2, 4	9.79E-05
10	6, 10	1.33E-04
11	3, 5, 7	1.00E-04
12	8, 9	1.33E-04
13	1, 3, 5, 7	1.49E-04
14	8, 9	1.33E-04
15	2, 4, 6, 10	5.73E-04
16	1, 2, 3, 4, 5, 6, 7, 8	5.30E-03
17	10	0
18	9	0
19	2, 4, 6	1.45E-04
20	10	0

21	5, 7	4.89E-05
22	8, 9	1.33E-04
23	1, 3	5.00E-05
24	10	0
25	3	0
26	1, 5, 7	9.79E-05
27	2, 4, 6	1.45E-04
28	8, 9	1.33E-04

The best W-KMeans(4) solution found in this step has value 4.65E-04, calculated as the sum of the values in the column "Weighted Cluster Cost" for rows 9-12.

Given the data in the 2nd column of Table 6, the constraints of the problem in step 5 are then as follows:

```
0   1   0   0   0   0   0   1   1   0   0   0   1   0   0   1   0   0   0   0   0   0   1   0   0   1   0   0   >= 1
1   0   0   0   0   0   1   0   1   0   0   0   0   0   1   1   0   0   1   0   0   0   0   0   0   0   1   0   >= 1
0   0   0   1   1   0   0   0   0   0   1   0   1   0   0   1   0   0   0   0   0   0   1   0   1   0   0   0   >= 1
1   0   0   0   0   0   1   0   1   0   0   0   0   0   1   1   0   0   1   0   0   0   0   0   0   0   1   0   >= 1
0   1   0   0   1   0   0   0   0   0   1   0   1   0   0   1   0   0   0   0   1   0   0   0   0   1   0   0   >= 1
1   0   0   0   0   0   1   0   0   1   0   0   0   0   1   1   0   0   1   0   0   0   0   0   0   0   1   0   >= 1
0   1   0   0   1   0   0   0   0   0   1   0   1   0   0   1   0   0   0   0   1   0   0   0   0   1   0   0   >= 1
0   0   0   1   0   1   0   0   0   0   0   1   0   1   0   1   0   0   0   0   0   1   0   0   0   0   0   1   >= 1
0   0   1   0   0   1   0   0   0   0   0   0   1   0   1   0   0   0   1   0   0   0   1   0   0   0   0   1   >= 1
1   0   0   0   0   0   1   0   0   1   0   0   0   0   1   0   1   0   0   1   0   0   0   1   0   0   0   0   >= 1
1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   1   = 4
```

Figure 2: Matrix representation of the set covering problem formulation constraints

Where we have only written the matrix A_B as the LHS and the values of the RHS vector of the constraints.

When the MIP solver solves the SCP problem with the above data, the best solution it finds picks the columns 6, 13, 17 and 19 which represent the following clusters:

{8, 9}

{1, 3, 5, 7}

{10}

{2, 4, 6}

This solution has total weighted cost equal to 4.27E-04 (=1.33E-04 + 1.49E-04 + 0 + 1.45E-04) which happens to be less than any cost found by the application of the base clusterers for $k = 4$. Because the solution found is also a solution to the set partitioning problem (SPP), the step 6 is a no-op (does not do anything). Similarly, step 7 (applying W-KMeans(4) from the solution just found) does not create any new clusters because the new solution found is already a local optimum for the W-MSSC problem. Therefore (given that τ=0) steps 8-10 are also no-ops, and no new columns are added to the matrix A_B. Therefore, in the next iteration of steps 3-10, no improvement will occur, and the algorithm stops after two iterations of its main loop. The best solution found by applying the W-EXAMCE algorithm is shown in Fig. 6 below.

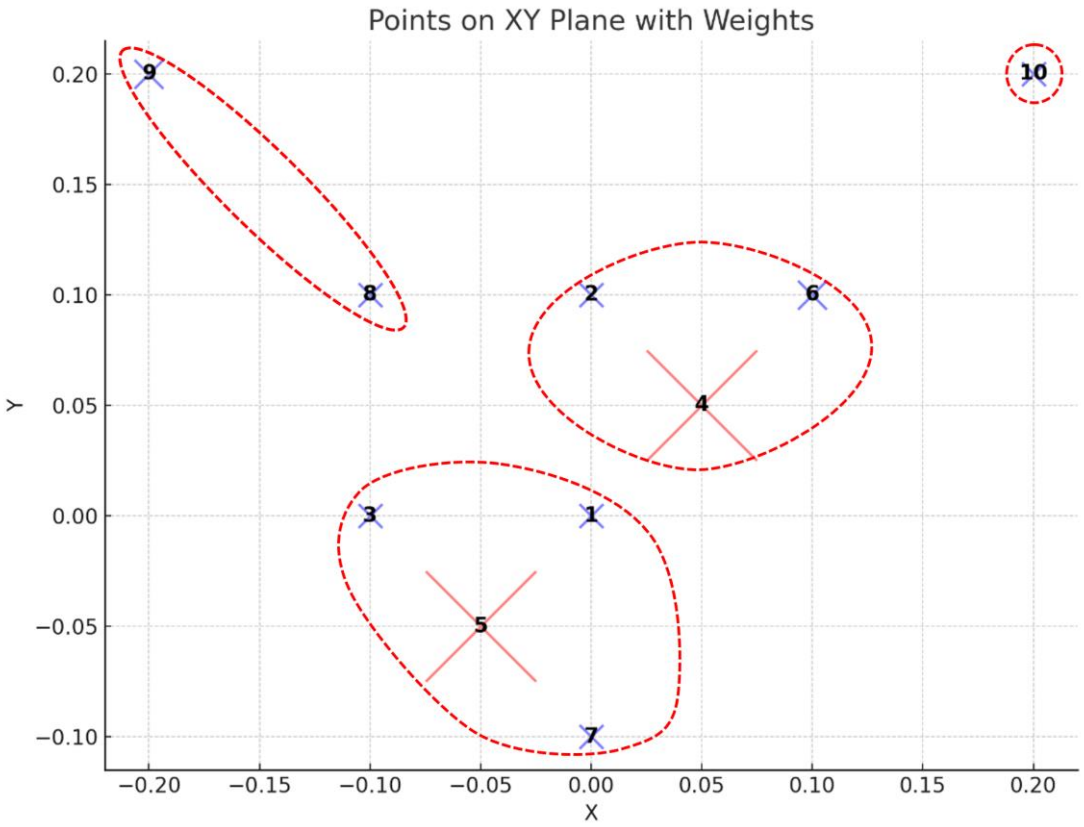

Figure 6: Best clustering solution found by W-EXAMCE on toy dataset

FIGURES

Algorithm 1 W-EXAMCE

Require:

0: S: a finite collection of points in R^d
0: w: the vector of positive weights associated with each point in S
0: k: the number of clusters to partition the data set
0: $c(.,.)$: the cost function $f(C_i, \bar{s}_i)$ described in eq. (1)
0: Ba: a set of base weighted-clustering algorithms that produce disjoint clusters of the set S that cover exactly the set S
0: $Rm\_Dup(.)$: the procedure that removes duplicates in the clusters of a clustering solution described in section 3
0: $Expand(C, \tau)$: the function $Expand : (2^S, N) \rightarrow 2^{2^S}$ is a heuristic procedure that expands the set of solutions described in section 3
0: $W\_KMeans(C)$: the weighted K-Means algorithm starting with the solution C that iterates the two weighted K-Means steps until no improvement in the objective function can be made, and returns all clusters created during the process

Ensure:

0: A partitioning P of the data set S among k disjoint clusters that is locally optimal with respect to the cost function $c(.)$
1: $S_B \leftarrow$ apply(B_a, S) {Apply base clustering algorithms in B_a to produce an initial set S_B of clusters}
2: **repeat**
3: $N \leftarrow |S_B|$
4: $(A_B)_{|S| \times N} \leftarrow$ indicator_matrix(S_B) {Set to the matrix whose columns are the membership indicator vectors of the clusters of SB}
5: $x \leftarrow$ SCP_solve$(A_B, \bar{s}_{A_B}, k)$ {Solve the following Set-Covering problem with side-constraint:}

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{N} c([A_B]_i, \bar{s}_{[A_B]_i}) x_i \\ \text{subject to} & A_B x \geq e \\ & e^T x = k \\ & x_i \in \{0,1\} \quad \forall i = 1, \ldots, N \end{array}$$

6: $C' \leftarrow$ Rm_Dup(x, S_B, c) {Removes any duplicates from the clusters selected by x}
7: $C'' \leftarrow W\_KMeans(C')$
8: $C''' \leftarrow C' \cup C''$
9: $C^4 \leftarrow \bigcup_{c \in C'''}$ Expand(c, τ)
10: $SB \leftarrow SB \cup C^4 \cup C'''$
11: **until** no further improvement
12: **return** C''

Figure 3: W-EXAMCE algorithm

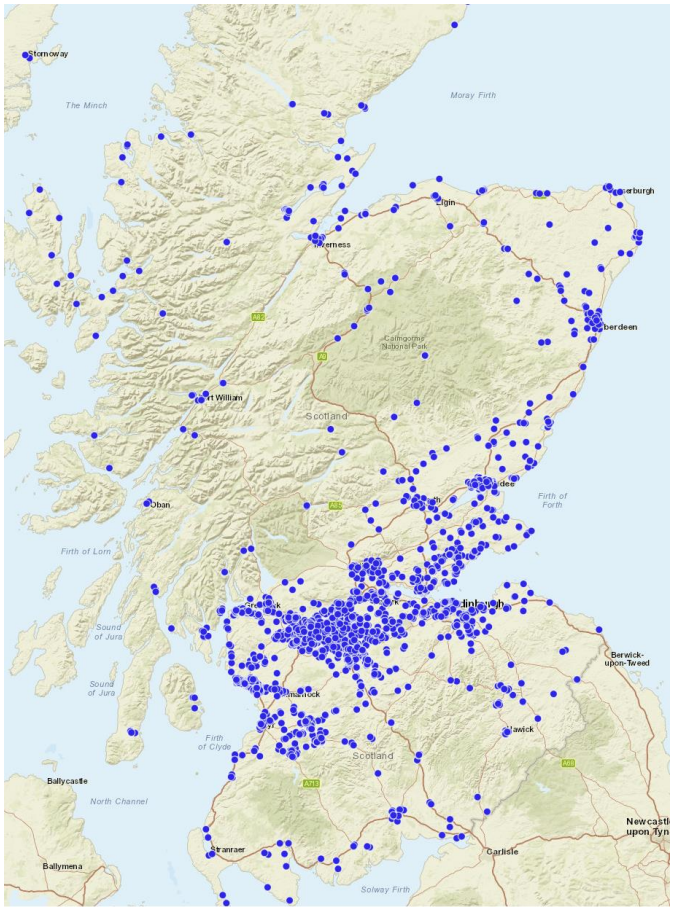

Figure 4: Locations of Vacant and Derelict Land Register Sites in Scotland in the British Coordinate Grid. The size of each plot is not drawn to scale.

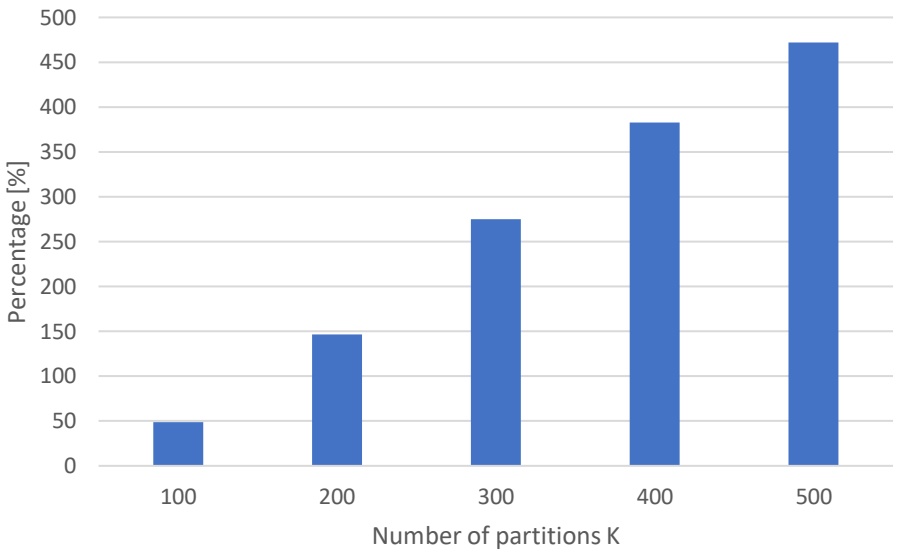

Figure 5: Comparison chart of the solution gap between W-EXAMCE and weighted K-Means with 100 Restarts on the Scotland Land Register dataset

TABLES

Table 1: Used dataset characteristics

Dataset	Library	Studies using the Dataset	Number of points	Number of dimensions	Weights
u1060	TSPLIB	[33, 34]	1,060	2	Random
pcb3038	TSPLIB	[33, 34]	3,038	2	Random
pr2392	TSPLIB	[24]	2,392	2	Random
rl11849	TSPLIB	[24,34]	11,849	2	Random
Scotland Vacant and Derelict Land Register	Land Register	[6]	3,398	2	Actual Plot Areas
US COVID-19 Cases	N/A	[36]	4,478	2	Actual number of cases per district

Table 2: Comparison of results of W-EXAMCE, K-Means and scalable K-Means|| w/ 100 Restarts on TSPLIB datasets

Dataset	k	Best K-Means / 100 Restarts		Best K-Means|| / 100 Restarts		W-EXAMCE		
		Solution	Time (secs)	Solution	Time (secs)	Solution	Solution Improvement (%)	Time (secs)
rl11849	100	1.74E+09	7.78	1.73E+09	10.2	1.72E+09	1.5	155.0
	200	8.36E+08	8.9	8.24E+08	13.6	8.13E+08	2.8	242.0
	300	5.40E+08	9.9	5.30E+08	16.7	5.14E+08	5.2	133.0
	400	3.95E+08	10.24	3.82E+08	19.5	3.68E+08	7.4	177.0
	500	3.11E+08	10.3	2.94E+08	24	2.81E+08	10.4	707.0
pr2392	50	4.74E+08	3.8	4.81E+08	3.98	4.70E+08	0.8	21.0
	100	2.23E+08	3.28	2.18E+08	4.06	2.02E+08	10.2	26.5
	200	9.67E+07	1.77	9.57E+07	5.45	8.43E+07	14.7	36.6
	300	6.02E+07	1.94	5.64E+07	5.41	4.73E+07	27.4	73.1
	400	4.09E+07	1.98	3.75E+07	6.19	3.01E+07	36.2	40.7
	500	2.99E+07	1.99	2.64E+07	7.94	2.08E+07	44.1	47.9
u1060	50	2.97E+08	2.1	2.90E+08	2.62	2.72E+08	9.1	30
	100	1.32E+08	3	1.31E+08	3.5	1.05E+08	25.6	11.5
	150	8.48E+07	3	7.96E+07	4.5	6.45E+07	31.4	20.4
	200	6.14E+07	4.5	6.15E+07	4.6	4.21E+07	46.0	16.1
	250	4.42E+07	4.1	4.13E+07	4.8	3.02E+07	46.3	20.4
	300	3.41E+07	4.9	3.26E+07	5.5	2.30E+07	48.2	21.4
pcb3038	50	1.05E+08	5.1	1.04E+08	5.5	4.94E+07	113.4	15.3
	100	5.29E+07	6	5.09E+07	6.1	2.39E+07	121.4	36
	200	2.59E+07	12	2.08E+07	12	1.06E+07	144.9	16
	300	1.65E+07	22.1	1.02E+07	23.1	6.21E+06	166.3	25.4
	400	1.25E+07	20.1	1.00E+07	22.9	4.12E+06	203.6	30.8
	500	9.84E+06	25	7.53E+06	26.2	2.87E+06	242.9	34

Table 3: Comparison of results of W-EXAMCE, K-Means w/ 100 Restarts and Scalable K-Means|| w/ 100 Restarts on Scotland Vacant and Derelict Land Register dataset

Dataset	k	Best K-Means / 100 Restarts		Best K-Means|| / 100 Restarts		W-EXAMCE		
		Solution	Time (secs)	Solution	Time (secs)	Solution	Solution Improvement (%)	Time (secs)
Scotland Vacant and Derelict Land Register	100	2.50E+11	4.5	2.11E+11	7.8	1.68E+11	48.8	37.8
	200	1.49E+11	19.8	1.01E+11	25.3	6.05E+10	146.3	80.9
	300	8.89E+10	16.9	9.91E+10	30.1	2.37E+10	275.1	77.9
	400	6.95E+10	21.8	4.99E+10	25.9	1.44E+10	382.6	91.8
	500	4.94E+10	24	2.51E+10	28.5	8.64E+09	471.8	126.2

Table 4: Comparison of results of W-EXAMCE, K-Means w/ 100 Restarts and Scalable K-Means|| w/ 100 Restarts on the US COVID-19 Cases dataset.

Dataset	k	Best K-Means / 100 Restarts		Best K-Means|| / 100 Restarts		W-EXAMCE		
		Solution	Time (secs)	Solution	Time (secs)	Solution	Solution Improvement (%)	Time (secs)
US COVID-19 Cases	50	1.16E+10	8	1.06E+10	37	9.63E+09	20.1	30.6
	100	4.92E+09	7.5	4.35E+09	34.9	3.70E+09	33.0	35.7
	200	1.94E+09	9.8	2.16E+09	30.9	1.31E+09	47.7	51.3
	300	1.16E+09	9.1	1.31E+09	37	6.66E+08	74.5	71.5
	400	7.86E+08	12.7	9.00E+08	34.7	4.03E+08	94.8	99.7
	500	5.55E+08	11.6	6.31E+08	37.5	2.73E+08	103.1	103