minerals

MDPI

Article

A Block Aggregation Method for Short-Term Planning of Open Pit Mining with Multiple Processing Destinations

Saad Salman [1], Khan Muhammad [1,2], Asif Khan [1,3] and Hylke J. Glass [4,*]

[1] National Centre of Artificial Intelligence, Intelligent Information Processing Laboratory, University of Engineering and Technology, Peshawar 25000, Pakistan; saad_salman@uetpeshawar.edu.pk (S.S.); khan.m@uetpeshawar.edu.pk (K.M.); asif.khan@fcm3.paf-iast.edu.pk (A.K.)
[2] Department of Mining Engineering, University of Engineering and Technology, Peshawar 25000, Pakistan
[3] Mineral Resource Engineering, Pak-Austria Fachhochschule: Institute of Applied Sciences and Technology, Mang, Haripur 22621, Pakistan
[4] Minerals Engineering Research Group, Camborne School of Mines, University of Exeter, Penryn, Cornwall TR10 9FE, UK
* Correspondence: h.j.glass@exeter.ac.uk

Abstract: Clustering approaches are widely used to group similar objects and facilitate problem analysis and decision-making in many fields. During short-term planning of open-pit mines, clustering aims to aggregate similar blocks based on their attributes (e.g., geochemical grades, rock types, geometallurgical parameters) while honoring various constraints: i.e., cluster shapes, size, alignment with mining direction, destination, and rock type homogeneity. This approach helps to reduce the computational cost of optimizing short-term mine plans. Previous studies have presented ways to perform clustering without honoring constraints specific to mining. This paper presents a novel block clustering heuristic capable of considering and honoring a set of mining block aggregation requirements and constraints. Constraints can relate to the clustering adjacent blocks, achieving higher destination homogeneities, controlled cluster size, consistency with mining direction, and achieving clusters with mineable shapes and rock types' homogeneity. The proposed algorithm's application on two different datasets demonstrates its efficiency and capability in generating reasonable block clusters while meeting different predefined aggregation requirements and constraints.

Keywords: clustering; short-term mine planning; k-means clustering; blocks aggregation

Citation: Salman, S.; Muhammad, K.; Khan, A.; Glass, H.J. A Block Aggregation Method for Short-Term Planning of Open Pit Mining with Multiple Processing Destinations. *Minerals* **2021**, *11*, 288. https://doi.org/10.3390/min11030288

Received: 3 February 2021
Accepted: 5 March 2021
Published: 10 March 2021

1. Introduction

Deposits are routinely discretized into blocks, which are assigned economic values based on the cost of extraction and each block's expected value [1]. Interpretation of the economic block model defines the Ultimate Pit Limit (UPL) [2], which constrains the open pit dimensions in which maximum undiscounted value generation is predicted while honoring block precedence requirements, as well as physical and operational constraints. Given that a UPL may contain thousands to millions of blocks, defining a multi-period, long-term production schedule is challenging and computationally intensive. The technical and computational complexity can be reduced by merging adjacent blocks with similar properties. Grouping blocks leads to the definition of clusters that can assist both long- and short-term production planning. It should be noted that the long-term production schedule informs the short-term production planning process, albeit subject to its own set of specific constraints [3]. While long-term planning attempts to avoid any precedence constraints between target ore zones, short-term planning seeks to delineate mineable ore shapes.

Establishing block clusters can be addressed through mathematical modeling. However, the computational cost of finding the best solution may be prohibitive: the clustering of large numbers of blocks with multiple properties is a recognized problem [4]. It requires

that homogenous clusters be generated by merging blocks based on grade, rock type, geometallurgical parameters such as Bond work index, recoveries [5–8], and the operational requirements such as proximity of blocks. Heuristic techniques provide a promising way to perform clustering through cognitive learning, experience, and domain knowledge [9]. The two main classes of heuristic techniques are hierarchical (agglomerative and divisive) and partition (K-means, c-means) clustering [10]. Hierarchical agglomerative clustering uses a bottom-up approach that considers each block to be a cluster, merging blocks into a cluster based on best similarity. Hierarchical divisive clustering assumes a top-down approach, in which all blocks are initially part of one cluster, and the most dissimilar blocks are successively separated out into new clusters.

Clustering/Block aggregation has been an integral part of mine planning strategies, both long and short, to reduce their computational complexity [10–20], but this benefit comes at the expense of losing information of various attributes such as grade and rock type at the block level. This may affect the Net Present Value (NPV) of the mining operation at a strategic level and create difficulties in meeting production requirements at an operational level. The clustering process must be able to group blocks while simultaneously considering and satisfying different clustering criteria to minimize information loss. Optimal and practical clusters should have mineable shapes, bounded size, limited destination dilution, and consistent mining direction [11]. Tabesh and Askari-Nasab [10] proposed blocks aggregation for open pit mine planning with hierarchical clustering based on a similarity index that considers block grades, the distance of blocks, rock type penalty, and beneath cluster penalty. Cluster shape refinement was achieved with a Tabu search. However, post-processing using the Tabu search reduced the NPV and homogeneity of clusters by 4 and 50%, respectively. Other researchers have applied the K-means clustering algorithm to cluster blocks by defining a dissimilarity measure based on %Cu, %Mo, speed of extraction, and tonnage in an underground copper mine [14]. Another approach aggregates blocks into fundamental shapes that can be extracted economically while honoring the precedence constraints, which becomes computationally expensive while solving a real-size block model [1].

Aggregation of blocks through clustering techniques offers a useful way to support short-term mine planning. Previously, the application of Fuzzy C-means clustering to short-term planning was investigated, creating groupings of blocks called mining cuts [13]. Mining cuts are used as input to a Mixed-integer linear programming (MILP) formulation for production scheduling with multiple destinations, including stockpiles. However, details about the shape, grade variation, and destination homogeneity of these mining cuts are not reported. Such factors are relevant for short-term planning because the cluster shape must be such that mining equipment has enough space for working efficiently and safely, meeting feed grade requirements at various processing destinations. It is essential for short-term planning to identify mineable shapes, i.e., close to regular shapes such as cubes and cuboids. Ruiseco et al. [17] used a genetic algorithm approach to define the best possible dig-limits/aggregates on a bench while considering grade control data, mining costs, processing, and mining constraints. However, this strategy did not incorporate the multiple rock types and grade targets constraints while deciding the optimal dig-limits. Furthermore, this approach outperformed the traditional hand-drawn dig limit in terms of results (Fitness value) but was computationally expensive. A hierarchical clustering approach with a post-processing stage has been developed, which generated homogeneous clusters in grade and rock types [11]. However, these aggregation approaches do not honor all operational constraints such as multiple destinations, grades, and rock type homogeneity, proximity, and mining direction simultaneously. Furthermore, the addition of multiple parameters in the similarity index adversely affected the quality of the results. The addition of operational flexibility to short-term planning can significantly influence the economic performance of an open-pit mining operation [21]. However, taking advantage of opportunities and addressing challenges during production requires fast revision and adaptations of the short-term mine plan.

An innovative block aggregation heuristic for short-term planning was developed and described in this paper to support decision-making. The proposed approach simultaneously honors a range of block clustering requirements, viz. adjacency of clustered blocks, destination homogeneities, cluster size, consistency with mining direction, mineable shapes, and homogeneity of rock type. K-means clustering is the first step in this approach. Following a brief introduction to K-means clustering, the proposed algorithm, case studies, results, discussion, and conclusions section will be described.

2. Materials and Methods

2.1. K-Means Clustering

K-means is the fastest centroid-based algorithm [22] among many clustering algorithms [23–25]. The K-means clustering algorithm is an iterative process that assigns each mining block to one of K predefined classes or clusters. As an associated output, K-means clustering returns the cluster centers, or centroids, which correspond to the average value of each cluster's block properties. The steps of the K-means algorithm are as follows:

- Input block properties: $X = \{x_1, x_2 \cdots x_n\}$, where n is the number of blocks on a bench.
- Initialize K geochemical class centers, c_k, with random values.
- (a) Assign each block i to the nearest geochemical class centroid c_k using a distance criterion, m_{ik}:
- $m_{ik} = argmin_{k\in\{1\ldots K\}} |x_i - c_k|^2$
- Recalculate the K geochemical cluster centers c_k as a weighted mean of the blocks that belong to c_k:

$$c_k = \frac{\sum_i^n m_{ik} x_i}{\sum_i^n m_{ik}} \quad (1)$$

- Repeat from step (a) until there is no significant change in cluster centers

This process minimizes the objective function $J = \sum_{i=1}^{n}\sum_{k=1}^{K} m_{ik} |x_i - c_k|^2$ by reducing the intra-cluster variance. In the following, centroids will be referred to as geochemical class centers.

2.2. Validation Criterion for the Number of Classes

Cluster validation is the process of establishing whether the selected number of cluster centers or centroids is optimal. Clustering outcomes are compared using indices, with the most common being those obtained with the silhouette [26], elbow [27], and gap statistic [28] methods. While the elbow method can lead to ambiguous results, the silhouette method can be used with any clustering approach [29] and will be applied here.

2.3. Silhouette Coefficient

For a set of predefined K clusters, the silhouette is calculated as follows:

$$s_i = \frac{b_{ij} - a_{ik}}{\max\{a_{ik}, b_{ij}\}} \quad (2)$$

where

a_{ik} = average dissimilarity of block i from all blocks in a geochemical cluster c_k;
b_{ij} = average dissimilarity of block i from all blocks in the least dissimilar geochemical cluster c_j, where $i \notin c_j \; and \; j \neq k$;
s_i = silhouette.

The silhouette can vary between −1 and +1, with the following interpretation:

$$S_i = \begin{cases} 0 & if\ point\ i\ is\ very\ close\ its \text{ neighbouring clusters} \\ 1 & if\ point\ i\ far\ away\ from\ its \text{ neighbouring clusters} \\ -1 & if\ point\ i\ is\ assigned\ to\ the\ wrong\ clusters \end{cases}$$

The average silhouette value of all blocks is termed the silhouette coefficient. Higher values of the silhouette coefficient indicate better clustering performance and vice versa. The best clustering outcome is associated with the largest attainable silhouette coefficient.

With complex arrangements of mining blocks, it may be advantageous to allow blocks to be part of more than one cluster. In that case, a fuzzy approach can be adopted in which the total block membership, which is equal to unity, is distributed appropriately across different geochemical classes. The mapping of block memberships to cluster centers is defined by Membership Functions (MFs). Several membership functions can be distinguished: Crisp MF, Triangular MF, Trapezoidal MF, Gaussian MF, Generalized bell MF, Sigmoid MF, and Singleton MF [30]. The triangular membership function will be considered in this study as it allows the mixing of blocks from various destinations.

3. The Proposed Block Aggregation Algorithm

A block aggregation algorithm is proposed where mining blocks are combined into a spatial cluster when these

- have similar attributes,
- are located in the same neighborhood, and
- match requirements for different processing destinations shape and size of clusters, adjacency, rock type, and mining direction.

The first step is to cluster blocks with the K-means method to identify geochemical classes. A second step clusters blocks based on the similarity between adjacent blocks considering all relevant properties, including rock types. The second step is repeated until there are no more blocks to merge, i.e., meet a predefined merging threshold, or the required cluster size has been reached. Note that the algorithm starts the aggregation process from the point where mining starts and aligns clustering with the mining direction. The Algorithm 1 is as follows:

Algorithm 1 Proposed Algorithm

Inputs:
Bench Number
Maximum cluster size
Minimum cluster size
Merging threshold-1 between (0–1)
Second, merging threshold-2, i.e., <Merging threshold-1
n = number of blocks on any given bench
i = index of each block
U = number of adjacent blocks, for an ith block on the bench
u = index of U adjacent blocks
MV = Membership Value
For each bench in a dataset:
    Find geochemical cluster centers
    For K = 2 to C_{max}
        Apply K-means for all blocks exceeding waste (as in Section 2)
        Report K geochemical class centroids
        Determine silhouette coefficient for selected K number of classes
        Choose the best K = number of grade classes, among K = 2 to C_{max}
        Include the waste grade class, i.e., $K = K + 1$
    Membership value calculation and similarity calculation
    For each block i on a bench
        Find membership of block i to class k using μ_k (i), for all $k = 1, 2 \ldots K$
        Calculate adjacency function (A_{iu}) for each block i and adjacent block u
        Measure dissimilarity D_{iu} of block i and adjacent block u
        Quantify similarity S_{iu} of block i and u using
        S_{iu} = (Max $D - D_{iu}$)/Max D
        Where Max D = maximum dissimilarity of block i from adjacent blocks
        $u = 1, 2 \ldots U$

Algorithm 1 *Cont.*

Block aggregation
Starting from lowest index min (i): the starting point of mining direction
For min (i) in Bench Number:
    While the size of Cluster < Max Cluster size:
        IF S_{iu} > Merging Threshold-1:
            Merge min (i) with min (u)
            $i = u$
Unclustered block merging
For each min (i) not in any cluster:
    While Merging Threshold-2 > 0
        IF S_{iu} > Merging Threshold-2:
            Merge min (i) with min (u)
            Reduce Threshold-2 i.e., Threshold-2 = Threshold-2 − 0.05
Small cluster adjustment
Inputs:
    GC_i = Grade of the ith cluster
    GA_p = Grade of the pth adjacent cluster
    p = total number of adjacent clusters to ith cluster
    p = each adjacent cluster to ith cluster
    IF size of a cluster i < minimum cluster size:
        For p in P:
            Find $GC_i - GA_p$
            IF $GC_i - GA_p$ = min ($GC_i - GA_p$)
                Merge pth cluster into ith cluster
Shape *refinement*
For a specified number of Iterations:
    Find and remove corner blocks
    Remove any empty clusters

The proposed algorithm is explained in the following sections.

3.1. Identification of Geochemical Classes

K-means clustering identifies grade class centers of clusters consisting of blocks exceeding the waste blocks threshold. In addition to geochemical grades, K-means use spatial coordinates to create compact clusters with better shapes. The number of geochemical classes for a given bench in a block model is determined by evaluating the silhouette coefficient.

3.2. Block Classification Using Membership Functions

Fuzzy membership functions inform decision-making at the transition boundaries between grade classes identified during K-means clustering. Triangular membership functions represent the preceding, current, and successive grade class center values and are used to find the membership of a block for each class in the ore grade region, whereas all waste blocks are assigned membership values of unity to a waste class. Membership values are used to devise quantitative similarity measures between blocks, as discussed in the next section.

3.3. Calculation of Similarity between Adjacent Blocks

For each block, surrounding blocks with minimum Euclidean distances from the reference block are identified as adjacent blocks. The similarity between adjacent blocks is determined using grade membership values of blocks using Equation (3). For categorical

variables, e.g., rock type, a penalty approach was used as suggested previously [10,31]. The similarity value between any two blocks is defined by Equation (4).

$$GS_{iu} = 1 - \frac{\sqrt{\sum_{k=1}^{K}(M_{ik} - M_{uk})^2}}{\sqrt{DS_{max}}} \quad (3)$$

$$GS'_{iu} = GS_{iu} \times R_{iu} \quad (4)$$

$$R_{iu} = \begin{cases} 1, & if\ block\ i\ and\ u\ has\ the\ same\ rock\ type \\ r, & otherwise\ (taking\ value\ between\ 0\ and\ 1) \end{cases}$$

where i = Block index; $i = 1, 2 \ldots\ldots I$

n = Total number of blocks;
k = Class index; $k = 1, 2 \ldots\ldots K$;
K = Total number of classes;
r = rock type penalty value;
u = adjacent block to block i
M_i = grade membership value of block i to class k;
M_u = grade membership value of block u to class k;
GS_{iu} = similarity between block i *and* u;
R_{iu} = rock type penalty between block i *and* u;
DS_{iu} = dissimilarity between block i *and* u;
DS_{max} = Maximum dissimilarity between block i *and* u.

If adjacent blocks belong to different rock types, a higher value of "r" will give a higher weighting to the grade in determining similarity. Lower "r" values, on the other hand, would reduce the weighting given to grade while calculating the similarity.

3.4. Merging Similar and Adjacent Blocks

Based on similarity value, adjacent blocks are merged to form clusters of a predefined size, expressed in the maximum number of blocks. The algorithm starts the merging process from the lowest indexed block, i.e., the starting point of mining on a bench, and merges subsequent blocks if the similarity between these exceeds a predefined "threshold-1". The algorithm continues the merging process along the intended direction of mining (Figure 1) until a set maximum cluster size is reached. The overall block aggregation procedure on a bench would end when one of the following two conditions is met:

- None of the blocks qualify for merging
- All blocks on a bench are merged

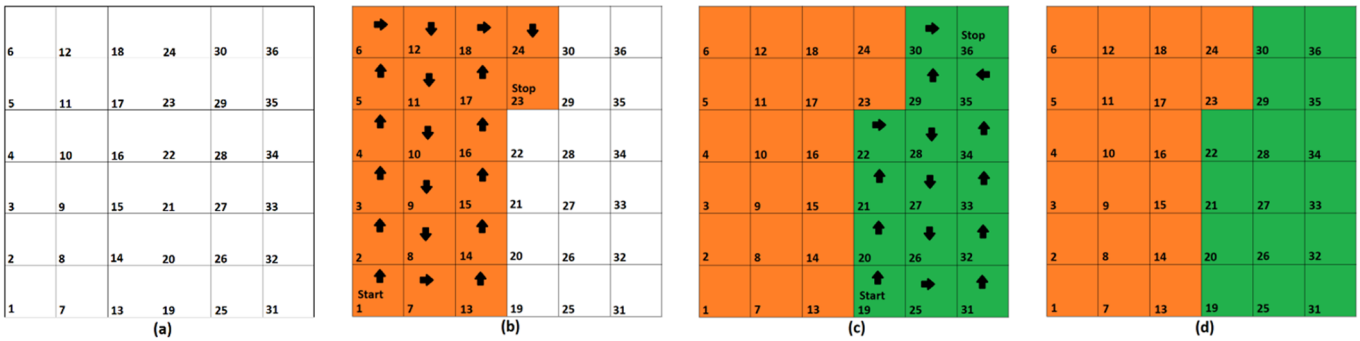

Figure 1. Principle of clustering (**a**) bench under consideration, (**b**,**c**) the process of cluster formation, and (**d**) final result.

3.5. Unclustered Block Merging (UBM)

After the primary merging of blocks, there will likely be unclustered blocks due to not meeting the merging criterion. For these blocks, the merging process is continued while iteratively reducing the similarity value, given by "threshold-2", until all blocks are assigned to a cluster.

3.6. Small Cluster Adjustment (SCA)

Identifying a uniform cluster size supports ease in equipment deployment and movement [32]. Any cluster with a smaller size than the predefined minimum is merged with an adjacent cluster with the smallest difference in average grades among adjacent clusters. However, small ore clusters located between waste clusters or small waste clusters located between ore clusters are lost, i.e., excluded from clustering. This approach helps to prevent excessive dilution.

3.7. Shape Refinement (SR)

Shape refinement is beneficial for efficient blast design and subsequent extraction. Shape refinement begins by identifying sharp-cornered blocks, i.e., blocks with only one adjacent block from the same cluster and more than one adjacent block from another cluster. Corner blocks may be detached from their original cluster and assigned to clusters with which these share more than one side.

A distinction is made between corner ore blocks and corner waste blocks. A detachment of sharp-cornered ore blocks is avoided if this leads to loss of ore. The detachment of sharp-cornered waste blocks and mixing these with the ore cluster is allowed if the resulting ore cluster still qualifies as ore after adding a waste block. The shape refinement process is illustrated in Figure 2. Figure 2a shows the resultant clustering before shape refinement (SR) with sharp corners shown in Figure 2b, where each color represents a separate cluster. The arrangement in Figure 2c would be obtained after the 1st iteration of SR, Figure 2d after the 2nd iteration, and Figure 2e after the 3rd iteration. When the arrangement in Figure 2e would not improve with further shape refinement, it is accepted as the final clustering result.

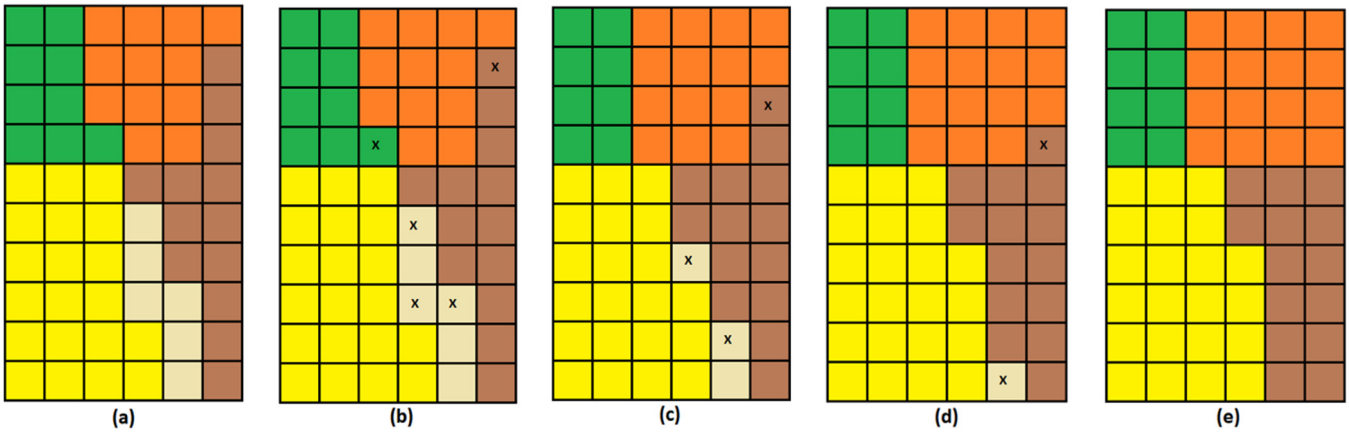

Figure 2. Procedure of Shape Refinement (SR) shown as: (**a**) clusters before the SR step, (**b**) sharp corners indicated as "x", and results after (**c**) 1st iteration, (**d**) 2nd iteration, and (**e**) final iteration of the SR step.

3.8. Cluster Evaluation Criteria

Following the clustering process, adherence to clustering objectives is evaluated. Clustering aims to generate clusters that are homogeneous in grade, rock type, the destination of material, have mineable shapes of a consistent size, and are aligned with the direction of mining. This study assesses the performance of the proposed algorithm with the following criteria (Tabesh and Askari-Nasab [11]):

Destination Homogeneity (DH): expresses the highest percentage of blocks in a cluster with the same destination. In equation:

$$DH = (c/C) \times 100$$

where DH = destination homogeneity index

c = highest number of blocks in a cluster which share the same destination
C = total number of blocks in a cluster.

Rock Unity (RU): expresses the highest percentage of blocks in a cluster with the same rock type:

$$RU = (r/R) \times 100$$

where RU = rock unity index

r = highest number of blocks in a cluster from the same rock type
R = total number of blocks in a bench.

Total Blocks Clustered (TC): expresses the percentage of blocks which have been clustered on a bench:

$$T = (r/R) \times 100$$

where T = total blocks clustered index

r = Number of blocks that are assigned to a cluster
R = Total blocks on a bench.

The shape of a cluster: is evaluated visually.

4. Case Studies

This section presents the proposed algorithm's application on two datasets available at Minelib (A library of open-pit mining problems). These feature two processing destinations for run-of-mine ore: Leach pad and Flotation Circuit. The Marvin dataset (Case A) relates to a copper deposit with block grades ranging from 0 to 1.46% Cu and comprised of 8516 blocks within the Ultimate Pit Limit. The Alexandra Newman dataset (Case B) relates to a copper mine with block grades ranging from 0 to 3.69% Cu and comprised of 1060 blocks within a predefined Ultimate Pit Limit.

A software code was developed in Python to test the proposed algorithm. For both datasets, block clusters on each bench were generated using the same values for technical and economic parameters (Table 1). The algorithm was applied to the entire 3D dataset in both cases on a bench-by-bench basis (2D). For illustration, only bench 18 of Newman and bench 10 of Marvin dataset are presented in detail, while the entire datasets' results are tabulated at the end.

Table 1. Technical and economic parameters for Case A and B.

Description	Value	Unit
Copper Price	6000	USD per t of Cu
Mining Cost	2	USD per t of Rock
Beneficiation Cost—Crushed Leach	2	USD per t of Ore
Beneficiation Cost—Mill	10	USD per t of Ore
Smelting Cost	1000	USD per t of Cu
Mill Recovery	0.90	%
Leach Pad recovery	0.25	%
Cutoff Grade—Leach Pad	0.16	% Cu
Cutoff Grade—Mill	0.25	% Cu
Marvin Dataset Block Dimensions	30 × 30 × 30	m
Newman Dataset Block Dimensions	12 × 12 × 12	m

The proposed algorithm was tested through simulations in which the merging threshold, rock penalty, and cluster size were varied (Table 2).

Table 2. Algorithmic parameters settings of the three different scenarios for Case A and B.

Scenario	Merging Threshold Range	Rock Penalty Range	Size	
			Max	Min
Case A—Marvin Dataset (Bench 10)				
1 (A)	0.2–0.9	N/A	30	10
1 (B)	0.2–0.9	N/A	30	10
Case B—Alexandra Newman Dataset (Bench 18)				
2	0.2–0.9	0.2–0.8	30	10

Before simulations, the grade/rock type distributions for selected benches of the Marvin and Alexandra Newman datasets are shown in Figures 3a and 4a,b, while blocks in the respective pits can be viewed in Figures 3b and 4c.

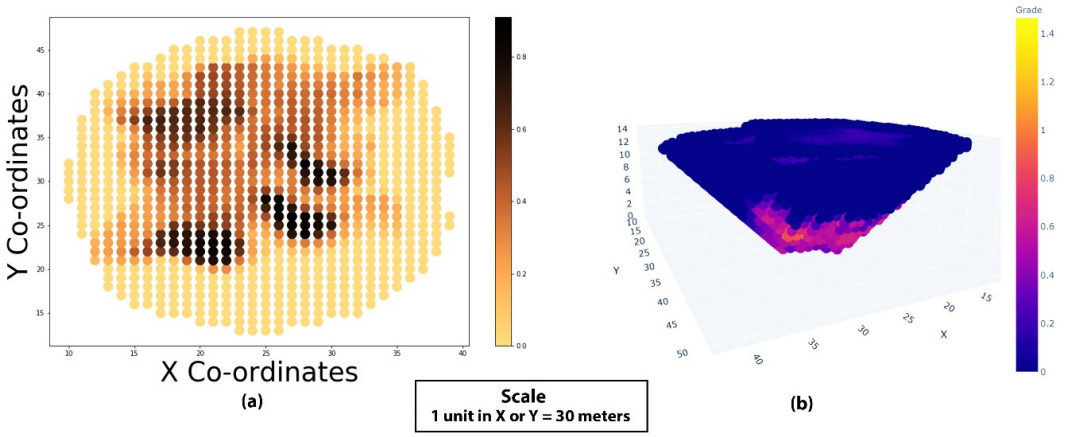

Figure 3. (**a**) Grade distribution on bench 10 of the Marvin dataset (**b**) 3D view of the Marvin dataset.

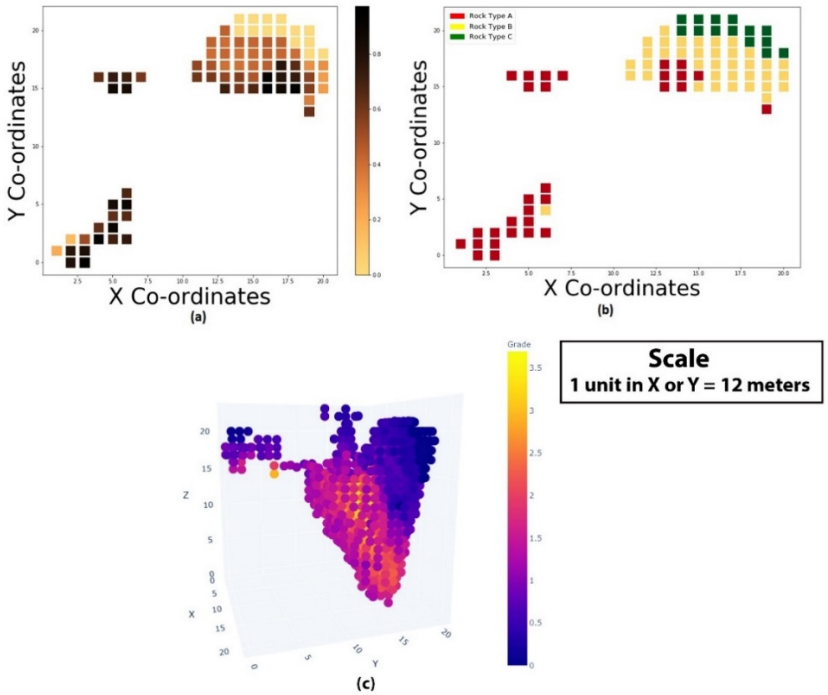

Figure 4. (**a**) Grade, (**b**) rock type distribution on bench 18 of the Newman dataset, and (**c**) 3D view of the Newman dataset. Each block is of 12 × 12 × 12 m dimension.

5. Results and Discussion

High merging thresholds produce a clustering pattern with a high homogeneity of leach and mill destinations: when setting the merging "threshold-1" at 0.8, clustering of Marvin data leads to 75% leach, 89% mill destination homogeneities, and an average cluster size of 22. Simulating clustering with a merging "threshold-1" of 0.9, 0.8 0.6, 0.4, to 0.2, as threshold-1 is lowered, more blocks are clustered: 0.68, 0.80, 0.87, 0.92 and 0.98, respectively.

Comparing the results of setting "threshold-1" equal to 0.8 and 0.6 showed that the former produced highly homogeneous blocks and reported more irregular shapes, greater runtime, and smaller average cluster size. This is expected since increasing the similarity threshold will lead to increasingly less prevalent aggregation. The increase in runtime was due to most blocks being merged at the "Unclustered blocks merging" step when "threshold-2" is iteratively decreased until all blocks are clustered. If a particular type of destination blocks is too small in number and scarcely disseminated, therefore reporting small clusters in the first stage of the algorithm, then allowing a higher value of "minimum blocks in a cluster" during the "small cluster adjustment" step leads to the merger of such a cluster with another cluster.

The selection of best values of merging "threshold-1", "threshold-2", cluster size, and rock penalty is investigated in Scenario 1(A) and 1(B) for the Marvin dataset and Scenario 2 for the Alexandra Newman dataset.

5.1. Scenario 1(A)

Figure 5a shows clusters formed at merging "threshold-1" of 0.6 before applying the UBM, SCA, and SR steps. Destination Homogeneities (DH) were reported as 100%, 89%, and 100% for waste, leach, and mill. UBM and SCA's application created clusters with 99%, 78%, and 89% destination homogeneities, while all the blocks clustered are shown in Figure 5b. The shape refinement procedure improved the shapes by removing any sharp corners. The improved shaping clusters are presented in Figure 5c, where DH's of 99%, 75%, and 87% for waste, leach, and mill were achieved (Table 3). The destination homogeneity of leach clusters dropped more compared to other destinations. This could be due to the smaller size and sparse distribution of these clusters. Consequently, the addition of blocks from other destinations into leach clusters during the UBM, SCA, and SR steps resulted in a higher drop in destination homogeneity than mill and waste destinations. Additionally, it was observed that higher "threshold-1" creates small clusters for scarcely disseminated blocks on the bench, and these risks losing their identity to surrounding clusters during the "small cluster adjustment" step. Figure 5a shows the formation of disseminated tiny leach clusters that were later merged with the mill cluster (as seen in Figure 5b) during the "small cluster adjustment" step at the cost of a slight drop in the homogeneity of mill clusters. Figure 5d shows the blocks with their indices and mining direction indicated by the arrowhead. The blocks were indexed along the mining direction to form clusters that were consistent with the direction of mining, consequently reducing the number of drop cuts and increasing their usage.

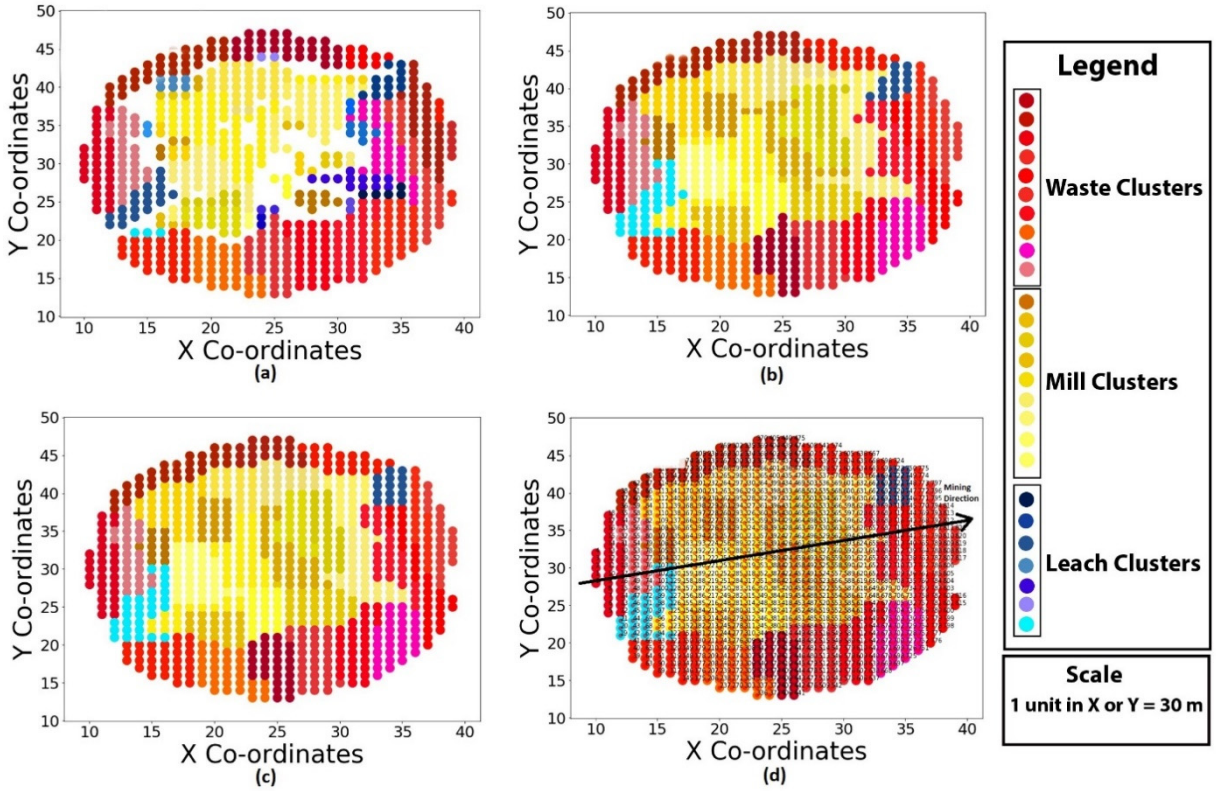

Figure 5. Clustering results from scenario 1(A) applied on bench 10 of the Marvin dataset after: (**a**) block aggregation step, (**b**) unclustered block merging (UBM) and small cluster adjustment (SCA) steps, (**c**) shape refinement (SR) step, and (**d**) indicating the mining direction. Each block is of 30 × 30 × 30 m dimension.

Table 3. Results for bench 10 of Marvin dataset using "threshold-1" of 0.6.

Scenario 1(A) * Destination Homogeneity (DH) per Destination (%)			
Description	**Waste**	**Leach**	**Mill**
Average	99	75	87
Std dev.	1.0	5	18.4
Cluster Size			
Max	**Min**	**Average**	**Std dev.**
32	11	24	6.3

* Total blocks clustered on the bench = 100%; the algorithm run time = 10.97 s.

5.2. Scenario 1(B)

This scenario demonstrates how setting a smaller merging "threshold-1" value during the block aggregation step affects the algorithm's subsequent steps, as shown in Figure 6a–d and Table 4. Since the mill blocks were sufficient in number and surrounded by waste, setting merging "threshold-1" to 0.35 had little effect on the mill blocks homogeneity. However, for this threshold, scarcely disseminated leach clusters led to three clusters (Figure 6c) compared to the two clusters reported in scenario 1(A). A lower "threshold-1" prevented the loss of leach cluster identity since this cluster had grown into sufficient cluster size by merging less similar blocks. Additionally, a lower "threshold-1" value merged fewer mill blocks with the leach clusters, reducing the average homogeneity of leach clusters.

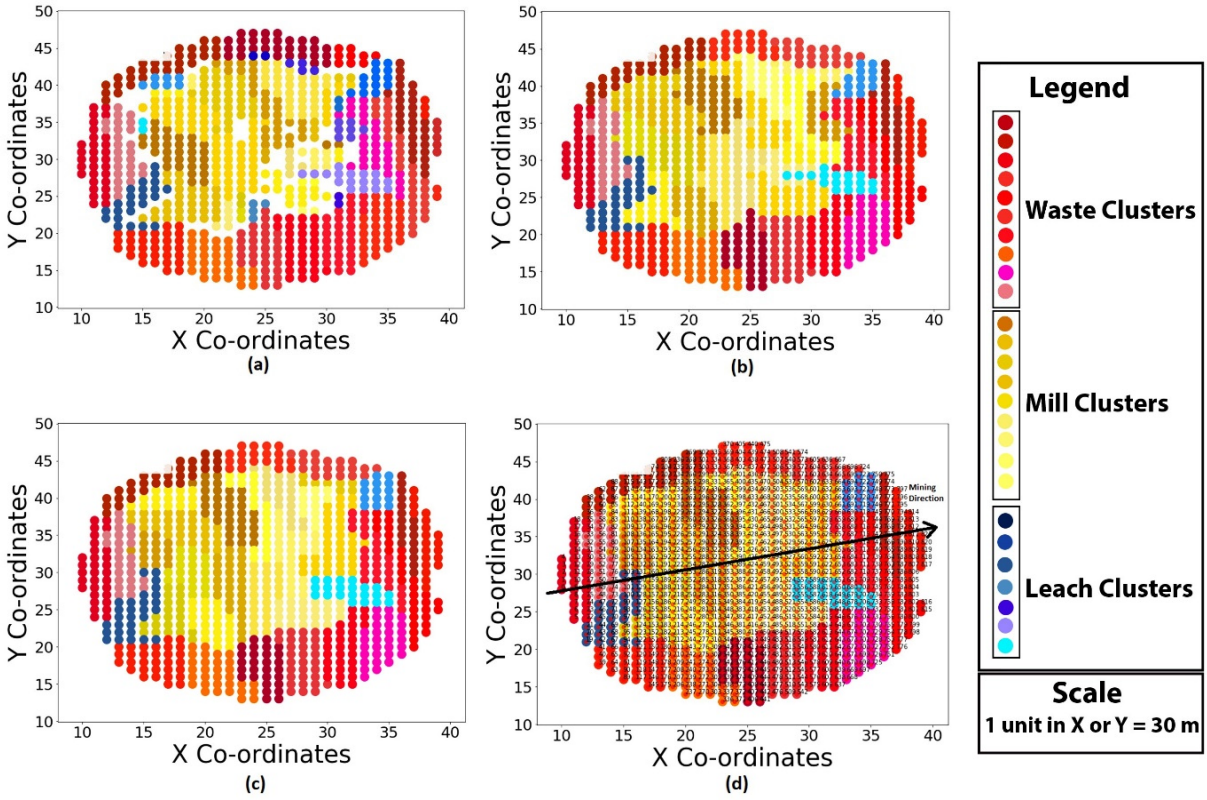

Figure 6. Clustering results from scenario 1(B) applied on bench 10 of the Marvin dataset after (**a**) block aggregation step, (**b**) UBM and SCA steps, (**c**) SR step, and (**d**) indicating the mining direction. Each block is of 30 × 30 × 30 m dimension.

Table 4. Results for bench 10 of the Marvin dataset using "threshold-1" of 0.35.

Scenario 1(B) * Destination Homogeneity (DH) per Destination (%)			
Description	**Waste**	**Leach**	**Mill**
Average	99.7	67.5	88
Std dev.	1.0	13.5	16.6
Cluster Size			
Max	**Min**	**Average**	**Std dev.**
35	6	23	7.2

* Total blocks clustered on the bench = 100%; algorithm run time = 10.57 s.

5.3. Scenario 2

The Alexandra Newman dataset comprises blocks with three different rock types. The best value of rock type penalty "r" was chosen from a range between 0.2 and 0.8, which reported maximum rock unity values, destination homogeneities, and produced mineable shapes (as given in Table 5). Figure 7a indicate the clusters formed after the "block aggregation" step, Figure 7b reports result after the UBM and SCA steps, that remain the same after SR step, as shown in Figure 7c where none of the clusters possessed sharp corners and Figure 7d indicates the direction of mining. For rock penalty, "r" = 0.4, the reported clusters were highly homogenous in grades with DH's 100% for waste, 97% for the mill, and average rock unity of 92%. No leach pad clusters were reported since none of the clusters qualified for this destination due to very few leach blocks on this bench. These results highlight the algorithm's performance in achieving its goals using different scenarios. It is evident that clusters obtained from applying the algorithm achieved high destination homogeneities, rock unity, mineable shapes, consistency with the mining direction, and required size.

Table 5. Results of bench 18 of Newman dataset for "threshold-1" = 0.5 and "r" = 0.4.

Scenario 2 * Destination Homogeneity (%) per Destination			
Description	**Waste**	**Leach**	**Mill**
Average	100	-	97
St. deviation	0	-	5.8
Rock Unity (%)			
Max	**Min**	**Average**	**St. deviation**
100	66.6	92	14.4
Cluster Size			
Max	**Min**	**Average**	**St. deviation**
24	6	16	6.79

* Total blocks clustered on the bench = 100%; algorithm run time = 1.24 s.

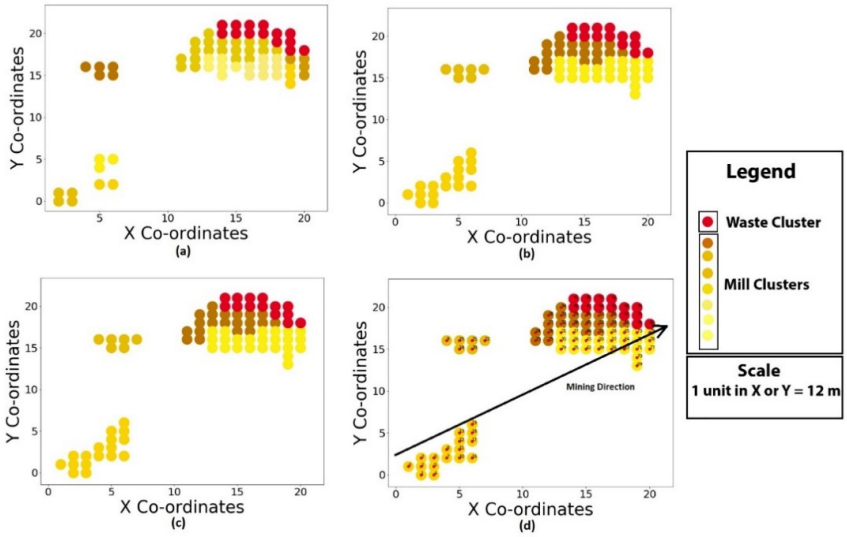

Figure 7. Clustering results from scenario 2 applied on bench 18 of the Newman dataset after (**a**) block aggregation step, (**b**) UBM and SCA steps, (**c**) SR step, and (**d**) indicating the mining direction. Each block is of 12 × 12 × 12 m dimension.

The results from the proposed algorithm applied to the complete set of the Marvin and Newman data are summarized in Tables 6 and 7. All blocks were clustered, with high average destination homogeneities of 99.4%, 73.5%, and 91.3% for waste, leach, and mill for the entire Marvin dataset and 99.5% and 99.4% for waste and mill region of the entire Alexandra Newman deposit with average rock unity more than 90%.

Table 6. Bench wise average destination homogeneities after applying the proposed algorithm for 14 benches of the Marvin dataset.

B #	Waste (%)	Leach (%)	Mill (%)	Time (s)	B #	Waste (%)	Leach (%)	Mill (%)	Time (s)
0			100	1.34	8	99.40		91.37	7.18
1			100	1.42	9	100	61.35	92.15	8.41
2			100	1.82	10	99.73	75.00	86.75	10.97
3			99.00	2.81	11	100	75.00	85.05	16.48
4			97.00	2.67	12	100	71.00	80	17.77
5	100		91.00	3.75	13	95.60	76.90	72.1	20.50
6	100		93.42	4.48	14	99.39	81.69		12.36
7	100		90.31	5.36					

Table 7. Bench wise average destination homogeneities and rock unity (RU) after applying the proposed algorithm to 22 benches of the Newman dataset.

B #	Waste (%)	Leach (%)	Mill (%)	Time (s)	RU (%)	B #	Waste (%)	Leach (%)	Mill (%)	Time (s)	RU (%)
1			100	0.5	100	12			98.07	1.13	92.30
2			100	0.67	100	13			100	1.24	90.20
3			100	0.75	100	14			97.6	1.26	90.80
4			100	0.74	100	15			96.9	1.26	93.70
5			100	0.76	100	16	100		98	1.26	82.20
6			100	0.76	100	17	100		99	1.22	83.80
7			100	0.80	100	18	100		97	1.24	92.15
8			100	0.97	100	19	100		100	1.11	100
9			100	0.90	97.20	20	100		100	1.01	100
10			100	1.01	96.80	21	96.87		100	0.78	98.43
11			100	1.12	93.30	22			100	0.68	100

The case studies suggest that very high or very low merging "threshold-1" and rock penalty "r" values are not recommended. Higher values of merging "threshold-1" adversely affect results in terms of shape and cluster size, while a low "threshold-1" causes much dilution and low homogeneities. Best results were found for intermediate values of merging "threshold-1" of 0.6 and rock penalty "r" of 0.4. Given that the algorithm is computationally efficient, simulations with various rock penalty values "r", "threshold-1", and "threshold-2" and size can be carried out to identify the best outcome.

In addition to the previously mentioned literature, block aggregation methods for long- [33–36] and short-term [16,18,20,37] open-pit mine optimization have been reported. Mathematical [16,20] to Deep Learning [37] block aggregation methods have been proposed very recently for underground [20] and open-pit short-term scheduling [16], emphasizing the need to account for operational mining constraints; however, none take account of them simultaneously. The proposed algorithm accounts for multiple operating constraints related to short-term planning. These comprise achieving mineable cluster shapes, high destination homogeneity, rock unity, cluster alignment with mining direction, and computationally efficiency. Additionally, the allowance for user-defined cluster size, merging threshold, and rock penalty give user the command to utilize his knowledge of the deposit and experiment to find the best possible results within reasonable time. Furthermore, the shape refinement procedure is highly valuable for addressing the equipment access constraints during mining operations. Computational efficiency, user-defined inputs, and simultaneously addressing many block aggregation constraints are advancements over previous work. Additionally, computational speed was a significant feature of the proposed algorithm. Clustering was conducted on Intel(R) Core (TM) i7-CPU 1.80 GHz with 20 GB RAM, reporting 10.57 s for bench 10 of the Marvin dataset containing 821 blocks and 1.24 s for bench 18 of the Newman dataset with 80 blocks. However, this algorithm still needs to be developed to deal with more complex multi-element multi-destination deposits and incorporate grade uncertainty and slope requirements into the aggregation strategy.

6. Conclusions

A novel, computationally-efficient block aggregation algorithm is presented that is capable of producing high destination and rock type homogeneities, mineable shapes, bounded size, block adjacency, and able to align clusters towards mining direction. These capabilities can play an essential role in facilitating the decision-making process of short-term open pit mine planning. The proposed algorithm is applied on two publicly available mine datasets, the Newman and Marvin datasets, which demonstrated its performance and efficiency in generating block clusters while simultaneously honoring these constraints. This algorithm can be applied to any 3D mineral dataset, particularly on metallic deposits with higher geologic complexity, on a bench-by-bench basis. It could be extended to cluster

multi-element datasets with multi-processing options while incorporating grade, market uncertainties, and slope requirements into the block aggregation process.

Author Contributions: Conceptualization, S.S., K.M., and A.K.; Methodology, S.S., K.M., and A.K.; Software, S.S.; Validation, S.S., K.M., and A.K.; Formal analysis, S.S., K.M., A.K., and H.J.G.; Investigation, S.S., K.M., A.K., and H.J.G.; Resources, K.M., H.J.G., and A.K.; Writing—original draft preparation, S.S., K.M., A.K., and H.J.G.; Writing—review and editing, S.S., K.M., A.K., and H.J.G.; Visualization, S.S.; Supervision, K.M. and A.K.; Project administration, K.M.; Funding acquisition, K.M. All authors have read and agreed to the published version of the manuscript.

Funding: "This research was funded by Higher Education Commission of Pakistan (HEC), National Centre of Artificial Intelligence" and "The APC was funded by the Authors".

Data Availability Statement: Data supporting reported results can be found at: http://mansci-web.uai.cl/minelib/Datasets.xhtml accessed on 9 March 2021.

Acknowledgments: This work was supported by the Higher Education Commission (HEC) of Pakistan under Grant: National Centre of Artificial Intelligence.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ramazan, S. The new Fundamental Tree Algorithm for production scheduling of open pit mines. *Eur. J. Oper. Res.* **2006**, *177*, 1153–1166. [CrossRef]
2. Hochbaum, D.S.; Chen, A. Performance analysis and best implementations of old and new algorithms for the open-pit mining problem. *Oper. Res.* **2000**, *48*, 894–914. [CrossRef]
3. Blom, M.; Pearce, A.R.; Stuckey, P.J. Short-term planning for open pit mines: A review. *Int. J. Min. Reclam. Environ.* **2019**, *33*, 318–339. [CrossRef]
4. Gonzalez, T.F. On the computational complexity of clustering and related problems. In *System Modeling and Optimization*; Springer: Berlin/Heidelberg, Germany, 1982; pp. 174–182.
5. Dominy, S.C.; O'connor, L.; Parbhakar-Fox, A.; Glass, H.J.; Purevgerel, S. Geometallurgy—A route to more resilient mine operations. *Minerals* **2018**, *8*, 560. [CrossRef]
6. Lund, C.; Lishchuk, V.; Koch, P.; Lamberg, P. The geometallurgical framework. Malmberget and Mikheevskoye case studies. *Min. Sci.* **2015**, *22*, 57–66.
7. Beaumont, C.; Musingwini, C. Application of geometallurgical modelling to mine planning in a copper-gold mining operation for improving ore quality and mineral processing efficiency. *J. S. Afr. Inst. Min. Metall.* **2019**, *119*, 243–252. [CrossRef]
8. Mohammadi, S.; Rezai, B.; Abdol, A.; Mortazavi, S.M. Evaluation of the Geometallurgical Indexes for Comminution Properties at Sarcheshmeh Porphyry Copper Mine. *Preprints* **2019**, 2019120103. [CrossRef]
9. Romanycia, M.H.J.; Pelletier, F.J. What is a heuristic? *Comput. Intell.* **1985**, *1*, 47–58. [CrossRef]
10. Tabesh, M.; Askari-Nasab, H. Two-stage clustering algorithm for block aggregation in open pit mines. *Trans. Inst. Min. Metall. Sect. A Min. Technol.* **2011**, *120*, 158–169. [CrossRef]
11. Tabesh, M.; Askari-Nasab, H. Automatic creation of mining polygons using hierarchical clustering techniques. *J. Min. Sci.* **2013**, *49*, 426–440. [CrossRef]
12. Tabesh, M.; Mieth, C.; Askari-Nasab, H. A multi-step approach to long-term open-pit production planning. *Int. J. Min. Miner. Eng.* **2014**, *5*, 273–298. [CrossRef]
13. Eivazy, H.; Askari-Nasab, H. A mixed integer linear programming model for short-term open pit mine production scheduling. *Trans. Inst. Min. Metall. Sect. A Min. Technol.* **2012**, *121*, 97–108. [CrossRef]
14. Weintraub, A.; Pereira, M.; Schultz, X. A Priori and A Posteriori Aggregation Procedures to Reduce Model Size in MIP Mine Planning Models. *Electron. Notes Discret. Math.* **2008**, *30*, 297–302. [CrossRef]
15. Deutsch, M. A branch and bound algorithm for open pit grade control polygon optimization. In Proceedings of the 19th APCOM, State College, PA, USA, 14–18 April 2017; pp. 1–8.
16. Nelis, G.; Morales, N. A mathematical model for the scheduling and definition of mining cuts in short-term mine planning. *Optim. Eng.* **2021**. [CrossRef]
17. Ruiseco, J.R.; Williams, J.; Kumral, M. Optimizing Ore–Waste Dig-Limits as Part of Operational Mine Planning Through Genetic Algorithms. *Nat. Resour. Res.* **2016**, *25*, 473–485. [CrossRef]
18. Tabesh, M.; Askari-Nasab, H. Clustering mining blocks in presence of geological uncertainty. *Min. Technol. Trans. Inst. Min. Metall.* **2019**, *128*, 162–176. [CrossRef]
19. Nezhadshahmohammad, F.; Pourrahimian, Y. A Clustering Algorithm for Block-Cave Production Scheduling. *Glob. J. Earth Sci. Eng.* **2018**, *5*, 45–53. [CrossRef]
20. Nezhadshahmohammad, F.; Pourrahimian, Y.; Aghababaei, H. Presentation of a multi-index clustering technique for the mathematical programming of block-cave scheduling. *Int. J. Min. Sci. Technol.* **2018**, *28*, 941–950. [CrossRef]

21. Jamshidi, M.; Osanloo, M. Multiple destination influence on production scheduling in multi-element mines. *Int. J. Eng. Trans. A Basics* **2018**, *31*, 173–180.
22. Uppada, S.K. Centroid Based Clustering Algorithms- A Clarion Study. *Int. J. Comput. Sci. Inf. Technol.* **2014**, *5*, 7309–7313.
23. Li, Y.; Wu, H. A Clustering Method Based on K-Means Algorithm. *Phys. Procedia* **2012**, *25*, 1104–1109. [CrossRef]
24. Harikumar, S.; Surya, P.V. K-Medoid Clustering for Heterogeneous DataSets. *Procedia Comput. Sci.* **2015**, *70*, 226–237. [CrossRef]
25. Bezdek, J.C.; Ehrlich, R.; Full, W. FCM: The fuzzy c-means clustering algorithm. *Comput. Geosci.* **1984**, *10*, 191–203. [CrossRef]
26. Rousseeuw, P. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.* **1986**, *20*, 53–65. [CrossRef]
27. Kodinariya, T.M.; Makwana, D.P.R. Review on Determining of Cluster in K-means Clustering Review on determining number of Cluster in K-Means Clustering. *Int. J.* **2016**, *1*, 90–95.
28. Yuan, C.; Yang, H. Research on K-Value Selection Method of K-Means Clustering Algorithm. *J Multidiscip. Sci. J.* **2019**, *2*, 226–235. [CrossRef]
29. Kaufman, L.; Rousseuw, P.J. *Finding Groups in Data: An Introduction to Cluster Analysis*; John Wiley & Sons, Inc.: Hoboken, NH, USA, 2009; Volume 344.
30. ROSS, J.T. *Fuzzy Logic with Engineering Applications*, 3rd ed.; John Wiley & Sons Ltd.: West Sussex, UK, 2010.
31. Dósea, M.; Silva, L.; Silva, M.A.; Cavalcanti, S.C.H. Adaptive Mean-Linkage with Penalty: A new algorithm for cluster analysis. *Chemom. Intell. Lab. Syst.* **2008**, *94*, 1–8. [CrossRef]
32. Topal, E.; Ramazan, S. A new MIP model for mine equipment scheduling by minimizing maintenance cost. *Eur. J. Oper. Res.* **2010**, *207*, 1065–1071. [CrossRef]
33. Lotfian, R.; Gholamnejad, J.; Mirzaeian Lardkeyvan, Y. Effective solution of the long-term open pit production planning problem using block clustering. *Eng. Optim.* **2020**, 1–16. [CrossRef]
34. Jélvez, E.; Morales, N.; Nancel-Penard, P.; Peypouquet, J.; Reyes, P. Aggregation heuristic for the open-pit block scheduling problem. *Eur. J. Oper. Res.* **2016**, *249*, 1169–1177. [CrossRef]
35. Mai, N.L.; Topal, E.; Erten, O. A new open-pit mine planning optimization method using block aggregation and integer programming. *J. S. Afr. Inst. Min. Metall.* **2018**, *118*, 705–714. [CrossRef]
36. Elsayed, S.; Sarker, R.; Essam, D.; Coello Coello, C.A. Evolutionary approach for large-Scale mine scheduling. *Inf. Sci.* **2020**, *523*, 77–90. [CrossRef]
37. Williams, J. Assessing Clustering of Selective Mining Units through Convolutional Neural Networks in a Genetic Algorithm Dig Limit Optimization. Master's Thesis, McGill University, Montréal, QC, Canada, 2019.