An approach to the development of clustering algorithms with a combined use of the Variable Neighborhood Search and Greedy Heuristic Method

V I Orlov, I P Rozhnov, L A Kazakovtsev and E V Lapunova
Reshetnev Siberian State University of Science and Technology, prospekt Krasnoyarsky Rabochiy 31, Krasnoyarsk 660031, Russia

E-mail: levk@bk.ru

Abstract. Authors propose a new approach to the development of clustering algorithms based on parametric optimization models with the combined use of search algorithms with variable randomized neighborhoods and greedy agglomerative heuristic procedures.

1. Introduction
Modern industrial production requires solving many clustering problems that have to be solved relatively quickly, with such a result that is difficult to improve using known methods without a significant increase in time costs.

An analysis of the existing problems of using clustering methods for the problems with high demands on the accuracy and stability of the result, shows a shortage of algorithms that are able produce results in a fixed time that would be difficult to improve using known methods and which would ensure the stability of the results obtained upon repeated runs of the algorithm [1, 2]. Noting a certain deficit in the clustering methods compromised in terms of the quality of the result and the calculation time (by quality we mean accuracy, i.e. the closeness of the objective function value to its global optimum), this study aims to develop improved algorithms for clustering problems with high demands on the accuracy and stability of the result.

2. Modern clustering algorithms
Modern methods of cluster analysis offer a wide selection of means for identifying groups that are heterogeneous in the set of parameters. The most common is the k-means model. A simple and fast algorithm of the same name, applicable to the widest class of problems, allows you to find a local minimum. The algorithm has limitations: the user must specify the number of groups k into which the objects are divided, and the result strongly depends on the initial solution chosen randomly [3, 4].

The algorithm for the k-medoid problem - Partitioning Around Medoids (PAM) is similar to the k-means algorithm: both try to minimize the error in each iteration, however, PAM works with the medoids which are objects that belong to the original data set, while k-means works with centroids which are artificially created objects representing the cluster. The PAM algorithm works with the distance matrix; its goal is to minimize the distance between the medoid of each cluster and its members [5, 6]. Modern popular methods for clustering problems include the Expectation Maximization algorithm. The main idea is to artificially introduce an auxiliary vector of hidden variables, which reduces the difficult
optimization problem to two steps. The E-step performs iterations to recalculate hidden variables according to the current approximation of the parameter vector. The M-step maximizes the likelihood function to find the next approximation of the vector. The Classification EM-algorithm (CEM) is a modification of the EM-algorithm. Unlike the EM, it solves the hard clustering problem [7, 8]. In this case, each object belongs to a single cluster.

Below, the k-means, PAM, and CEM algorithms are denoted as two-step local search algorithms. Currently, in cluster analysis, there is a tendency to the use of collective methods [2, 9]. The Variable Neighborhoods Search (VNS) is a popular method for solving discrete optimization problems proposed by N. Mladenovic and P. Hansen, which allows us to find good suboptimal solutions to comparatively large problems. The idea is to systematically change the appearance of the neighborhood during the local search. Flexibility and high efficiency explain its competitiveness in solving NP-difficult problems [10]. The k-means and k-medoid procedures are, in fact, simple search algorithms with alternating neighborhoods [1, 11].

3. Clustering algorithms of the Greedy Heuristic Method

The greedy agglomerative heuristic procedure for the k-means problem and similar problems consists of two steps. Let there be two known (parent) solutions to the problem (the first of which, for example, is the best known solution), represented by the sets of centers of the clusters \( S \). First, the sets of parental solutions are combined. We get an intermediate invalid solution with an excessive number of clusters. Then the algorithms performs a sequential decrease in the number of centers. In each iteration, it cuts off such a center, that the removal of this center gives us the least significant deterioration in the value of the objective function [12]:

Algorithm 1 Basic Greedy Agglomerative Heuristic Procedure

**Required**: initial number of clusters \( K \), needed number of clusters \( k \), initial solution \( S \), \(|S|=K\).

1: Improve \( S \) with the two-step local search algorithm if possible.

while \( K\neq k \) do

for each \( i\in [1,K] \)

2: \( S'=S\setminus \{X_i\} \). Calculate \( F' = F(S') \) where \( F(\cdot) \) if the objective function value (for example, (1) for the k-means problem).

end for

3: Select a subset \( S_{\text{elim}} \) of \( n_{\text{elim}} \) centers, \( S_{\text{elim}} \subseteq S \), \(|S_{\text{elim}}|=n_{\text{elim}}\), with the minimal values of the corresponding variables \( F'' \). Here, \( n_{\text{elim}}=\max\{1, 0.2\cdot(|S|-k)\} \).

4: Obtain the new solution \( S=S\setminus S_{\text{elim}} \), \( K=K-1 \), and improve this new solution with the two-step local search algorithm.

end while

In we propose the new heuristic procedures [1, 8] that modify the known solution based on the second known solution (see Algorithm 1).

Algorithm 2 Greedy Procedure 1

**Required**: sets of cluster centers \( S'\subset\{X'_1,\ldots,X'_k\} \) and \( S''\subset\{X''_1,\ldots,X''_k\} \)

for each \( i\in [1,K] \)

1: Merge \( S' \) and one item of the set \( S'' \); \( S=S\cup\{X''_i\} \)

2: Run the Basic Greedy Agglomerative Heuristic Procedure (Algorithm 1) with the initial solution \( S \). Save the obtained result (the set and the corresponding objective function value).

end for

3: Return the best of the solutions saved in Step 2.
In the second version of the procedure, sets are partially combined, while the first set is taken completely, and a random number of elements is randomly selected from the second set.

**Algorithm 3** Greedy Procedure 2

1: Select randomly \( r \in [0;1) \). Assign \( r=\lfloor(k/2-2)\cdot r^2\rfloor+2 \). Here, \( \lfloor . \rfloor \) is the integer part.

2: for \( i \) from 1 to \( k-r \)
   2.1: Form a randomly selected subset \( S''' \) of \( S'' \) of cardinality \( r \). Combine sets \( S = S' \cup S''' \).
   2.2: Run Algorithm 1 with this combined set as an initial solution.
end for

3: Return the best of the solutions obtained in Step 2.2.

A simpler algorithm below combines the full sets.

**Algorithm 4** Greedy Procedure 3

1: Combine sets \( S = S' \cup S'' \), and run Algorithm 1 with the initial solution \( S \).

These algorithms can be used as parts of various global search strategies. Sets of solutions derived ("children") from the the solution \( S' \), formed by combining its items with the items of some solution \( S'' \) and using Algorithm 1, are used as the neighborhoods in which a solution is searched. Thus, the second solution \( S'' \) is a parameter of the neighborhood selected randomly (randomized).

4. **New approach to the development of clustering algorithms**

In [1, 8, 13], authors considered combinations of greedy heuristic algorithms with the Variable Neighborhood Search [10, 14, 15] for the k-means, k-medoid problems as well as the well-known algorithms j-means and CEM. The accuracy of the clustering methods can be improved by a new approach to the development of clustering algorithms based on parametric optimization models, with the combined use of the search algorithms with variable randomized neighborhoods and greedy agglomerative heuristic procedures:

**Algorithm 5** GH-VNS (Greedy Heuristic in the Variable Neighborhood Search)

1: Run the two-step local search algorithm from a random initial solution to obtain solution \( S \).
   Assign \( O=O_{\text{start}} \) (number of the neighborhood type).

2: \( i=0 \), \( j=0 \) (the number of unsuccessful iterations in the current neighborhood and in total).

while \( i<i_{\text{max}} \)
   while \( j<j_{\text{max}} \)
      3: \textbf{if} the STOP conditions are not met (the time limitation is not exceeded), \textbf{then} run the two-step local search algorithm with the randomly chosen initial solution to obtain the solution \( S' \).
      \textbf{repeat}
      4: Depending on \( O \) value (values 1, 2, and 3 are allowed), run the Greedy Procedure 1, 2, or 3, respectively, with the initial solutions \( S \) and \( S' \).
      \textbf{if} the new solution is better than \( S \) \textbf{then}
      store the new result to \( S \); \( i=0; j=0 \).
      \textbf{else} go to 6.
      \textbf{end loop}
      6: \( i=i+1 \).
   \textbf{end while}
   7: \( i=0; j=j+1 \); \( O=O+1 \); \textbf{if} \( O>3 \) \textbf{then} \( O=1 \).
\textbf{end while}
In this algorithm, \( i_{\text{max}} \) is the number of unsuccessful searches in the neighborhood, and \( j_{\text{max}} \) is the number of unsuccessful switching of the neighborhoods. The values of these two parameters are important in the calculations. We used the values: \( i_{\text{max}} = 2k \), \( j_{\text{max}} = 2 \).

The parameter \( O_{\text{ini}} \) which specifies the number of the initial neighborhood is especially important. Depending on the \( O_{\text{ini}} \) value, the algorithms are designated \( \text{GH-VNS1}, \text{GH-VNS2}, \text{GH-VNS3} \) (for the \( k \)-means problem, respectively, \( k\)-\text{GH-VNS1}, \( k\)-\text{GH-VNS2}, \( k\)-\text{GH-VNS3}).

The way of obtaining the second solution \( S' \) on Step 4 is also essential. By default, the second solution contains the same number of items as \( S \). In addition, we used the modifications of Algorithm 5 in which the number of centers in the solution \( S' \) is randomly selected from the set \( \left\{ 2, |S| \right\} \) where \( |S| \) is the number of centers in solution \( S \). In this case, the algorithms are named \( \text{GH-VNS1-RND}, \text{GH-VNS2-RND}, \text{GH-VNS3-RND} \).

5. Computational experiments

We used the datasets from the UCI (Machine Learning Repository) and the Clustering Basic Benchmark repositories [16, 17]. For all data sets, 30 attempts were made to run each of the algorithms. Only the best results achieved in each attempt were recorded, then the values of the objective function were summarized from these results for each algorithm: the minimum and maximum values, average value, and the standard deviation. The \( j \)-means and \( k \)-means algorithms were launched in multi-start mode [1, 12].

The best values of the objective function (minimum, average value, and standard deviation) are highlighted in bold italics in table 1.

For a more complete comparison of the obtained results of computational experiments of new algorithms, we used the data of computational experiments obtained earlier by L. Kazakovtsev [12] for the data sets of a non-destructive testing of the production batches of industrial products by various modifications of the Genetic Algorithm (table 1, the second data set). In the table 1, we use the following abbreviations: \( \text{GA} \) is the Genetic Algorithm, \( \text{GH} \) is the Greedy Heuristic procedure, \( \text{GAGH} \) is the Genetic Algorithm with the Greedy Heuristic procedure and the real alphabet, \( \text{GA F} \) is the Genetic Algorithm with recombination of subsets of a fixed length.

Table 1. Results of the computational experiments, 30 attempts.

| Algorithm          | Min          | Max          | Average       | Std.Dev.       |
|--------------------|--------------|--------------|---------------|----------------|
| Data set BIRCH3 (100000 data vectors of dimensionality 2), 100 clusters, 6 hours |
| \( k \)-means      | 7.92474E+13  | 8.87404E+13  | 8.31599E+13   | 3.088140E+12   |
| \( j \)-means      | 3.76222E+13  | 3.79655E+13  | 3.77715E+13   | 0.116211E+12   |
| \( k\)-GH-VNS1     | 3.72537E+13  | 3.77474E+13  | 3.74703E+13   | 0.171124E+12   |
| \( k\)-GH-VNS2     | 4.21378E+13  | 5.01871E+13  | 4.52349E+13   | 4.333462E+12   |
| \( k\)-GH-VNS3     | 3.72525E+13  | 3.74572E+13  | 3.73745E+13   | 0.074315E+12   |
| \( k\)-GH-VNS1-RND | 3.72541E+13  | 3.77687E+13  | 3.74943E+13   | 0.185483E+12   |
| \( k\)-GH-VNS2-RND | 3.83257E+13  | 4.61847E+13  | 4.0815E+13    | 2.543163E+12   |
| \( k\)-GH-VNS3-RND | 3.73131E+13  | 3.75242E+13  | 3.74164E+13   | 0.061831E+12   |

Results of testing of semiconductor devices 1526TL1 (1234 data vectors of dimensionality 157) 10 clusters, 1 minute

| Algorithm | Min          | Max          | Average       | Std.Dev.       |
|-----------|--------------|--------------|---------------|----------------|
| \( k \)-means | 43 842.10   | 43 844.66    | 43 843.38     | 0.8346         |
| \( j \)-means | 43 841.97   | 43 843.51    | 43 842.59     | 0.4487         |
| \( k\)-GH-VNS1 | 43 841.97   | 43 844.18    | 43 842.34     | 0.9000         |
| \( k\)-GH-VNS2 | 43 841.97   | 43 844.18    | 43 843.46     | 1.0817         |
| \( k\)-GH-VNS3 | 43 841.97   | 43 842.10    | 43 841.99     | 0.0424         |
| \( \text{GAGH} \) | 43 841.98   | 43 844.18    | 43 842.6      | 0.6762         |
| \( \text{GA F} \) | 43 841.98   | 43 842.34    | 43 842.10     | 0.0945         |
| \( \text{GA classical} \) | 43 842.10   | 43 842.88    | 43 842.44     | 0.2349         |
| \( \text{Deterministic GH (IBC)} \) | 45 021.21   | 45 021.21    | 45 021.21     | 0.0000         |
As applied to the problem of k-medoid and maximization of mathematical expectation in Algorithm 5, the two-step local search algorithm is, respectively, the PAM algorithm and the CEM algorithm [8]. The results of computational experiments are presented in tables 2 and 3. The best values of the objective function (minimum, average values and the standard deviation) are shown in bold italics.

Table 2. Results for the data set Ionosphere (351 data vectors of dimensionality 35), 10 clusters, 60 seconds, 30 attempts, L1 distances.

| Algorithm   | Objective function value |       |       |
|-------------|--------------------------|-------|-------|
|             | Min (record)             | Average | Std.Dev. |
| PAM         | 2.688.57                 | 2.704.17  | 12.3308  |
| PAM-GH-VNS1 | 2.607.21                 | 2.607.25  | 0.1497   |
| PAM-GH-VNS2 | 2.607.21                 | 2.607.43  | 0.4303   |
| PAM-GH-VNS3 | 2.607.21                 | 2.607.34  | 0.4159   |
| GA-FULL     | 2.608.22                 | 2.624.97  | 9.5896   |
| GA-ONE      | 2.608.69                 | 2.625.18  | 10.7757  |

In Table 2, GA-FULL is the Genetic Algorithm with Greedy Heuristic and real alphabet, GA-ONE is the Genetic Algorithm which uses Algorithm 3 as the crossing-over procedure.

Table 3. Results for data sets of the non-destructive tests of semiconductor devices (10 clusters, 2 minutes, 30 attempts).

| Algorithm   | Objective function value |       |       |
|-------------|--------------------------|-------|-------|
|             | Min                     | Max    | Average | Std.Dev. |
| 3OT122A     |                         |        |         |         |
| (767 data vectors of dimensionality 13) | | | | |
| CEM         | 120 947.6               | 146 428.5 | 135 777.6 | 7 985.6992 |
| CEM-GH-VNS1 | 121 256.5               | 152 729.1 | 143 956.0 | 8 708.6293 |
| CEM-GH-VNS2 | 123 664.4               | 158 759.2 | 143 028.5 | 10 294.3992 |
| CEM-GH-VNS3 | 128 282.2               | 155 761.9 | 143 506.9 | 10 058.8266 |
| 1526TL1     |                         |        |         |         |
| (1234 data vectors of dimensionality 157) | | | | |
| CEM         | 354 007.3               | 416 538.4 | 384 883.4 | 20 792.8068 |
| CEM-GH-VNS1 | 376 137.1               | 477 124.5 | 438 109.4 | 29 964.0641 |
| CEM-GH-VNS2 | 345 072.6               | 487 498.3 | 444 378.1 | 43 575.3282 |
| CEM-GH-VNS3 | 379 352.3               | 516 777.8 | 456 271.4 | 38 323.0246 |

6. Conclusions
The computational experiments results show that the new algorithms of the Greedy Heuristic Method for clustering problems with increased requirements for the accuracy of the result (by the value of the objective function), with the combined use of the search algorithms with variable randomized neighborhoods (GH-VNS), have more stable (with less standard deviation of the objective function values) and more accurate results (lower average objective function value), in comparison with classical algorithms (k-means, j-means, PAM and CEM). With a significant increase in the calculation time, the known Genetic Algorithms of the Greedy Heuristic Method show slightly better results in comparison with the new algorithms. Nevertheless, we can talk about the competitiveness of new algorithms both in comparison with the classical k-means, PAM and j-means algorithms, as well as with genetic algorithms, including Greedy Heuristic Method algorithms including deterministic ones.

Acknowledgements
Results were obtained in the framework of the state task No. 2.5527.2017/8.9 of the Ministry of Education and Science of the Russian Federation.
References

[1] Orlov V, Kazakovtsev L, Rozhnov I, Popov N and Fedosov V 2018 *IOP Conf. Series: Materials Science and Engineering* **450** 022035

[2] Rozhnov I P, Orlov V I and Kazakovtsev L A 2018 Increase in Accuracy of the Solution of the Problem of Identification of Production Batches of Semiconductor Devices *14th International Scientific-Technical Conference on Actual Problems of Electronic Instrument Engineering, APEIE 2018* **363** 67

[3] MacQueen J B 1967 Some Methods of Classification and Analysis of Multivariate Observations *Proceedings of the 5th Berkley Symposium on Mathematical Statistics and Probability* **1** 281-97

[4] Farahani R and Hekmatfar M 2009 *Facility location: Concepts, models, algorithms and case studies* (Heidelberg:Springer-Verlag Berlin)

[5] Hansen P, Brimberg J, Urosevic D and Mladenovic N 2009 Solving large p-median clustering problems by primal dual variable neighborhood search *Data Mining and Knowledge Discovery* **19** 351–75

[6] Kaufman L and Roussieeuw P 1987 *Clustering by means of Medoids. Statistical Data Analysis Based on the L1-Norm and Related Methods* (Springer US) 405-16

[7] Celeux G and Govaert A 1991 *Classification EM Algorithm for Clustering and Two Stochastic Versions Rapport de Recherche de l’INRIA RR-1364* (Centrede Rocquencourt)

[8] Rozhnov I, Kazakovtsev L, Bezhitskaya E and Bezhitskiy S 2019 *IOP Conf. Series: Mater. Sci. Eng.* **537** 052032

[9] Rozhnov I, Orlov V and Kazakovtsev L 2018 Ensembles of clustering algorithms for problem of detection of homogeneous production batches of semiconductor devices *Proceedings “OPTA-SCL 2018. Proceedings of the School-Seminar on Optimization Problems and their Applications”. CEUR-WS* **2098** pp 338-48

[10] Hansen P and Mladenovic N 2005 *Variable Neighborhood Search Search Methodology* (Springer US) pp 211-38

[11] Kochetov Y, Alekseeva E, Levanova T and Loresh M 2005 Large Neighborhood Local Search for the p-Median Problem *Yugoslav Journal of Operations Research* **15** 53–63

[12] Kazakovtsev L and Antamoshkin A 2015 Greedy heuristics method for location problems *Vestnik SibGAU* **16** 317-25

[13] Kazakovtsev L, Rozhnov I, Popov E, Karaseva M and Stupina *IOP Conf. Series: Mater. Sci. Eng.* **537** 022052

[14] Alp O, Erkut E and Drezner Z 2003 An Efficient Genetic Algorithm for the p-Median Problem *Annals of Operations Research* **122** 21–42

[15] Kochetov Y, Mladenovic N and Hansen P 2003 Local search with alternating surroundings. *Discrete analysis and research opera* **s 2(10:1)** 11–43

[16] UCI Machine Learning Repository http://archive.ics.uci.edu/ml

[17] Clustering basic benchmark http://cs.joensuu.fi/sipu/datasets