НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
январь–февраль 2026 Том 26 № 1 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
January–February 2026 Vol. 26 No 1 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК
ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ

doi: 10.17586/2226-1494-2026-26-1-85-93
УДК 004.021

# Кластеризация аппроксимированного Парето-фронта

**Александр Григорьевич Юртаев**✉

Саратовский государственный технический университет имени Гагарина Ю.А., Саратов, 410054, Российская Федерация

agyurtaev@mail.ru✉, https://orcid.org/0009-0002-4146-4322

**Аннотация**

**Введение.** В современной инженерной и научно-технической практике многокритериальная оптимизация часто обеспечивает поиск компромиссных решений без задания весовых коэффициентов и границ, формируя Парето-фронт посредством эвристической аппроксимации на основе генетических алгоритмов. Однако даже аппроксимированный Парето-фронт представляет собой множество точек, что затрудняет анализ и отбор решений. Для упорядочения и структурирования полученных решений возможным решением становится кластеризация, позволяющая выделить репрезентативные группы компромиссов. Научная новизна предлагаемого метода кластеризации заключается в комбинации алгоритмов Ordering Points to Identify the Clustering Structure и *k*-means с выделением медоидов, обеспечивающей автоматическое удаление шума и компактное представление репрезентативных стратегий. **Метод.** Предложен метод двухэтапной кластеризации. На первом этапе применен алгоритм Ordering Points to Identify the Clustering Structure, с помощью которого строится упорядоченный профиль плотности и автоматически фильтруются шумовые точки по порогу досягаемости. На втором этапе использован алгоритм *k*-means, выполнено разбиение отфильтрованного ядра Парето-фронта на кластеры и вычислены центроиды, а затем медоиды — реальные представители данных. **Основные результаты.** Проведены два эксперимента на трехмерных множествах точек Парето-фронта (1226 и 2514 ядровых точек после фильтрации). В результате применения предложенной методики получено разбиение на 10 кластеров. Установлено, что после фильтрации доля шумовых точек составила менее 1 % от общего числа решений. Фильтрация позволяет существенно снизить значения метрики, оценивающей качество центров кластеров, при умеренном увеличении суммарного времени выполнения кластеризации. Показано, что малая рассогласованность центроидов и соответствующих медоидов свидетельствует о высокой репрезентативности полученных кластеров. **Обсуждение.** Предложенный комбинированный метод, сочетающий применение алгоритмов Ordering Points to Identify the Clustering Structure и *k*-means, требует настройки двух параметров, автоматически адаптируется к нелинейным плотностям и размерам входных данных. Область применения метода может быть расширена для любых задач многокритериальной оптимизации, решаемых посредством построения и анализа Парето-фронта, включая инженерную оптимизацию, логистику, энергетику и финансовое моделирование. В перспективе возможно внедрение адаптивных методов для автоматического определения оптимальных параметров используемых алгоритмов, а также обеспечения адаптации к динамическим изменениям многокритериальных задач.

**Ключевые слова**

кластеризация, Парето-фронт, Ordering Points to Identify the Clustering Structure, *k*-means, многокритериальная оптимизация

**Ссылка для цитирования:** Юртаев А.Г. Кластеризация аппроксимированного Парето-фронта // Научно-технический вестник информационных технологий, механики и оптики. 2026. Т. 26, № 1. С. 85–93. doi: 10.17586/2226-1494-2026-26-1-85-93

# Clustering of the approximated Pareto front

**Alexander G. Yurtaev**✉

Yuri Gagarin State Technical University of Saratov, Saratov, 410054, Russian Federation

agyurtaev@mail.ru✉, https://orcid.org/0009-0002-4146-4322

**Abstract**

In contemporary engineering and scientific practice, multi-objective optimization often facilitates the search for compromise solutions without prescribing weight coefficients or bounds, forming a Pareto front via heuristic approximation based on genetic algorithms. However, even an approximated Pareto front consists of a large set of points, which complicates analysis and selection of solutions. To organize and structure the obtained results, clustering can be employed to identify representative groups of trade-offs. The scientific novelty of the proposed clustering method lies in the combination of Ordering Points to Identify the Clustering Structure and *k*-means algorithms with the introduction of medoids identification, which ensures automatic noise removal and a compact representation of representative strategies. A two-stage clustering approach is proposed. At the first stage, Ordering Points to Identify the Clustering Structure algorithm is used to construct an ordered density profile and to automatically filter out noise points based on the reachability threshold. At the second stage, the *k*-means algorithm is applied to the filtered Pareto front core to partition it into clusters, compute the centroids, and then determine the medoids — real representative data points. Two experiments were conducted on three-dimensional Pareto front datasets (1226 and 2514 core points after filtering). As a result of applying the proposed approach, a partition into 10 clusters was achieved. It was found that after filtering, the proportion of noise points was less than 1 % of the total number of solutions. The filtering step significantly reduced the metric assessing the quality of cluster centers, with only a moderate increase in the total clustering time. A small discrepancy between centroids and their corresponding medoids indicates the high representativeness of the resulting clusters. The proposed hybrid method, combining Ordering Points to Identify the Clustering Structure and *k*-means algorithms, requires the adjustment of only two parameters and automatically adapts to nonlinear densities and input data scales. The scope of this method can be extended to any multi-objective optimization problems solved through the construction and analysis of the Pareto front, including engineering optimization, logistics, energy systems, and financial modeling. In the future, the approach may be enhanced by integrating adaptive mechanisms for automatic determination of optimal algorithm parameters, as well as dynamically changing multi-objective problem settings.

**Keywords**

clustering, Ordering Points to Identify the Clustering Structure, *k*-means, multi-objective optimization

**For citation:** Yurtaev A.G. Clustering of the approximated Pareto front. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2026, vol. 26, no. 1, pp. 85–93 (in Russian). doi: 10.17586/2226-1494-2026-26-1-85-93

## Введение

В современных инженерных и научно-технических задачах возникает необходимость одновременного учета нескольких, зачастую противоречивых, критериальных функций при выборе проектных или эксплуатационных решений. Такие задачи формулируются в рамках многокритериальной оптимизации (МКО), где целью становится не поиск единственного оптимального решения, а построение множества Парето-оптимальных (недоминируемых) вариантов, образующих Парето-фронт [1, 2]. Точка решения считается Парето-оптимальной, если не существует иной реализации, превосходящей ее по всем критериям одновременно, а само множество таких точек отражает границу компромисса между конфликтующими целями. Это позволяет отказаться от априорного задания весовых коэффициентов и тем самым существенно снизить субъективность выбора.

Однако полное перечисление Парето-фронта в дискретных задачах МКО сопряжено с экспоненциальным ростом числа недоминируемых точек [3] даже при сравнительно небольшом числе критериев и относится к NP-трудным задачам [4]: детерминированные алгоритмы, гарантирующие нахождение всех точек фронта, в худшем случае требуют перебора объемов, превышающих доступные вычислительные ресурсы, и потому практически неосуществимы при прямом переборе альтернатив. В связи с этим для приближенного построения Парето-фронта с заранее контролируемой точностью применяются эвристические методы [5, 6], среди которых генетические алгоритмы [7, 8] занимают ведущее место. За полиномиальное время они формируют репрезентативное приближение множества недоминируемых решений без необходимости скаляризации критериев.

Полученный набор недоминируемых решений, образующий приближенный Парето-фронт, служит исходным материалом для дальнейшего анализа: при большом количестве точек его визуальная и аналитическая оценки затруднительны, а выбор конкретных вариантов — субъективен и трудоемок. Следовательно, одним из ключевых этапов постобработки результатов МКО становится кластеризация Парето-фронта, позволяющая структурировать многочисленные компромиссные решения в несколько наглядных и интерпретируемых групп [9, 10].

Кластеризация Парето-фронта дает следующие преимущества:

— сокращение размерности (вместо десятков или сотен точек для анализа поступают центроиды кластеров, что упрощает визуализацию и последующую обработку данных без существенной потери информации о распределении решений);

— выявление типовых стратегий (каждый кластер объединяет решения со схожим взаимным соотношением критериев, что выявляет основные семейства компромиссов, характерные для рассматриваемой задачи);

— поддержка принятия решений (из каждого кластера достаточно выбрать один или несколько наиболее предпочтительных представителей с учетом внешних факторов, что снижает трудности при отборе оптимальных вариантов).

Применение кластеризации на этапе постобработки Парето-фронта не только упрощает анализ большого множества компромиссных вариантов, но и обеспечивает формализованные критерии для отбора и ранжирования репрезентативных решений, что становится полезным инструментом при практической реализации результатов МКО.

### Обзор алгоритмов кластеризации

Рассмотрим наиболее распространенные и классические алгоритмы кластеризации.

***k*-means [11, 12].** Изначально задаются матрица данных и число кластеров. В ходе работы алгоритма объекты итеративно распределяются по кластерам таким образом, чтобы суммы квадратов расстояний от точек до их центроидов были минимальными. Достоинства применения алгоритма — высокая скорость сходимости, линейная сложность по числу образцов и наглядность интерпретации (центроиды отражают средние решения кластеров). Главный недостаток алгоритма заключается в его чувствительности к выбросам: даже одиночные аномальные точки автоматически относятся к ближайшему центроиду, что может искажать положение центров кластеров и снижать точность разбиения данных.

**Иерархическая кластеризация [13, 14].** Задаются матрица данных, метрика расстояния, метод агломерации и число кластеров для отсечения дендрограммы (если необходимо). Кластеризация начинается с того, что каждая точка рассматривается как отдельный кластер. На каждом шаге объединяются две наиболее близкие группы до получения единого кластера. В результате формируется дендрограмма, допускающая горизонтальный срез на произвольной высоте, после чего каждая пересеченная ветвь трактуется как отдельный кластер. Не требуется заранее заданное число кластеров, но алгоритм обладает высокой вычислительной сложностью, результаты работы чувствительны к выбросам и зависимы от выбора метрики и метода агломерации.

**Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [15, 16].** Изначально задаются матрица данных, радиус и минимальное число точек в окрестности ядра кластера. Выполнение алгоритма начинается с произвольной точки и, если в ее окрестности находится как минимум заданное число образцов, формируется ядро кластера, затем рекурсивно добавляются все точки, попадающие в окрестность любого ядра; оставшиеся образцы маркируются как шум. С помощью данного алгоритма есть возможность выделять кластеры произвольной формы и автоматически игнорировать выбросы. Основные недостатки — необходимость тщательного подбора параметров (радиус и минимальное число точек в окрестности) и снижение качества кластеризации при значительных различиях плотностей между кластерами.

**Gaussian Mixture Models (GMM) [17, 18].** На входе задаются матрица данных, число компонент, тип ковариационной структуры и метод инициализации параметров. Все данные рассматриваются как смесь нескольких нормальных распределений, где каждая точка получает не жесткую, а вероятностную метку принадлежности к кластерам. Заданные параметры настраиваются итеративно методом максимального правдоподобия, что позволяет гибко моделировать эллиптические формы кластеров. Достоинства применения: гибко моделируются эллипсоидальные формы кластеров, предоставляется вероятностная разметка и учитывается перекрытие групп. Недостатки: результат работы алгоритма чувствителен к инициализации и локальным минимумам, а также результаты сложнее интерпретировать по сравнению с жесткими методами кластеризации.

**Spectral Clustering [19–21].** Изначально задаются матрица данных, мера попарной схожести, масштаб ядра и число кластеров. Задача формулируется как поиск в графе: точки связываются ребрами с весами, отражающими их схожесть, после чего выполняется спектральный анализ специальной матрицы, и точки проецируются в низкоразмерное пространство собственных векторов графа. В новом представлении группы отделяются методами плоской кластеризации (например, *k*-means). Применение алгоритма позволяет выявлять сложные, нелинейные структуры данных, но практическое применение ограничивается высокими требованиями к памяти и вычислениям при построении и спектральном разложении матриц большого размера, а также необходимостью настройки сразу нескольких параметров: меры сходства, масштаба ядра и числа кластеров.

**Ordering Points to Identify the Clustering Structure (OPTICS) [22–24].** Задаются матрица данных и минимальное число точек для определения плотности, при этом параметр радиуса не требуется. Алгоритм является развитием плотностного алгоритма DBSCAN: после единой обработки строится упорядоченный список точек с метриками досягаемости и ядрового расстояния. По графику плотности можно интерактивно или автоматически выделять кластеры на различных уровнях без повторных запусков. Главное преимущество применения алгоритма — отсутствие необходимости задавать радиус заранее и гибкое обнаружение кластеров разных масштабов; из минусов — относительная сложность интерпретации результатов без визуального анализа.

### Разработанный метод двухэтапной кластеризации

При анализе множества точек приближенного Парето-фронта ключевыми становятся два критерия.

Критерий 1. Отделение шума и аномальных решений, не влияющих на общую структуру компромиссов.

Критерий 2. Получение сжатого представления оставшегося ядра решений через небольшой набор репрезентативных стратегий.

Целью настоящей работы является разработка и экспериментальная проверка метода кластеризации при-

ближенного Парето-фронта, обеспечивающего автоматическое отделение шумовых решений и формирование компактного набора репрезентативных компромиссных стратегий при минимальной настройке параметров.

Предлагается последовательное применение плотностного алгоритма OPTICS для выявления значимых областей и последующая кластеризация этих точек с помощью алгоритма *k*-means, что обеспечивает решение обеих задач без необходимости избыточной ручной настройки. При этом алгоритм *k*-means выбран как один из наиболее простых и широко применяемых (при фиксированном числе кластеров время выполнения алгоритма почти пропорционально числу точек; с помощью алгоритма находятся такие центры, чтобы сумма квадратов расстояний точек до своих центров была минимальна), однако результаты применения алгоритма *k*-means чувствительны к выбросам и переменной плотности. Предварительная плотностная фильтрация по порогу досягаемости (с применением алгоритма OPTICS) устраняет эту уязвимость, формируя устойчивое ядро, на котором получены компактные и воспроизводимые кластеры методом *k*-means.

По критерию 1 с помощью алгоритма OPTICS осуществляется обход всего набора точек — для каждой точки определяется минимальный размер окрестности, при котором она становится ядром плотного скопления. Благодаря этому автоматически учитываются локальные особенности распределения: не требуется заранее задавать порог плотности, и вместо жестких границ строится сплошной плотностной профиль всего фронта. В результате формируется упорядоченный список точек с метриками досягаемости, по которому за один запуск можно обнаружить все основные скопления плотных компромиссов и одновременно отсеять разреженные области — шум и выбросы. Тем не менее выбросы не стоит полностью игнорировать: они могут представлять крайние решения Парето-фронта, демонстрирующие границы допустимых компромиссов и содержащие наиболее экстремальные стратегии.

По критерию 2 алгоритм *k*-means применяется только к отфильтрованному ядру данных. Алгоритм детерминировано разбивает ядро на заранее заданное число кластеров, причем задача — получить центроиды, максимально отражающие реальные компромиссные решения. Обеспечивается быстрое и вычислительно экономичное формирование нужного количества кластеров, при этом требуя в качестве единственного параметра число кластеров, определяемое исходя из практических требований — сколько типовых вариантов компромиссов нужно для последующего анализа или презентации.

Альтернативные алгоритмы в настоящей работе не использовались по следующим причинам. Иерархическая кластеризация требует хранения и обновления полной матрицы попарных расстояний (обычно $O(N^2)$ по памяти и не ниже $O(N^2)$ по времени), а также отсутствует процедура выбора медоидов. Применение алгоритма DBSCAN подразумевает согласованную настройку двух параметров одновременно, плохо переносит переменную плотность (типичную для приближенного Парето-фронта) и не предоставляет упорядочения по досягаемости, на основе которого однозначно задается порог шума. Применение алгоритма GMM предполагает гауссову форму кластеров и выбор ковариационной структуры. При наличии выбросов апостериорные вероятности принадлежности становятся размытыми, а из-за отсутствия фильтрации шума оценки центров компонент смещаются и, как правило, не совпадают ни с одной из исходных точек данных. При применении алгоритма Spectral Clustering требуется построения матрицы сходства и спектрального разложения, результаты чувствительны к выбору ядра и масштаба, что повышает долю ручной настройки и снижает воспроизводимость.

Таким образом, двухэтапная кластеризация с применением алгоритмов OPTICS и *k*-means объединяет сильные стороны двух разных методов: с помощью алгоритма OPTICS автоматически очищаются данные от шума и адаптируются к сложной нелинейной плотности, а алгоритмом *k*-means быстро и надежно генерируется компактный набор репрезентативных центроидов. При этом требуется минимум параметров (число соседей в OPTICS и число кластеров в *k*-means), а процедура кластеризации остается масштабируемой для любых объемов входных решений.

### Описание алгоритмов

**Алгоритм OPTICS.** На входе дано множество точек $X = \{x_1, \ldots, x_N\} \subset R^3$, где $N$ — количество точек множества. Задается единственный параметр $m$ — количество соседей для оценки плотности.

Для каждой точки $x_i$ ее расстояние до ядра (*core_dist*) задается как $m$-ое по величине значение среди расстояний $\{d(x_i, x_j)\}_{j \neq i}$, где $d(x_i, x_j) = \|x_i, x_j\|_2$. Если соседей меньше $m$, то $core\_dist(x_i) = +\infty$, $x_i$ не может быть ядром.

Перед началом основного цикла для всех $i$ задаются переменные $P[i] = false$ и $reach[i] = +\infty$. Здесь массив $P$ обозначает, была ли точка уже обработана, а массив *reach* хранит минимальную достижимость точки $x_i$ от уже распознанных ядер плотности.

Выбирается случайно индекс $i$ с $P[i] = false$, затем $P[i] = true$ и $i$ добавляется в конец списка $O$ (при инициализации пустой).

Для всех $j$ с $1 \le j \le N, j \neq i$ и $P[j] = false$ вычисляется $p_{ij} = \max(core\_dist(x_i), d(x_i, x_j))$. Если $p_{ij} < reach[j]$, то $reach[j] = p_{ij}$, значение $(p_{ij}, j)$ добавляется в очередь *seeds* (при инициализации пустая), которая реализуется как приоритетная очередь по возрастанию $p$.

Пока очередь *seeds* не пуста, извлекается запись $(p, j) = \min_p \{p_{kl}\}$ из *seeds*. Если $P[j] = false$, то $P[j] = true$ и в список $O$ добавляется $j$. Если $(core\_dist(x_j) < +\infty$, для всех $k$ с $1 \le k \le N$, $k \neq j$ и $P[k] = false$ вычисляется $p_{jk} = \max(core\_dist(x_j), d(x_j, x_k))$ и так далее.

Когда очередь *seeds* опустеет, берется любой индекс $i$ с $P[i] = false$ и с этой точкой проводят те же самые действия.

После этого в $O$ окажутся все индексы точек, а в массиве *reach* для каждой точки хранится ее минимальная досягаемость.

По построенному списку $O$ и массиву *reach* можно получить конкретное разбиение. Выбирается порог $\tau$ на досягаемость. $O$ последовательно просматривается и

выделяются из него максимальные последовательности индексов ($i_k$, $i_{k+1}$, …, $i_{k+\ell}$), для которых $reach[i_{k+r}] \le \tau$ для всех $1 \le r \le \ell$ . Каждая такая подпоследовательность становится отдельным кластером. Все точки $j$ с $reach[j] > \tau$ считаются шумом и не входят ни в один из кластеров.

Таким образом, в результате применения алгоритма OPTICS строится упорядоченный профиль плотности. Затем с помощью порога $\tau$ множество решений разбивается на два подмножества: основное и шум.

**Алгоритм *k*-means.** После фильтрации с помощью алгоритма OPTICS остается набор ядровых точек $Y = \{y_1, \ldots, y_n'\} \subset R^3$. Задается параметр числа кластеров $K$.

Ищется разбиение $C = \{C_1, \ldots, C_K\}$ точек и центроиды $\{\mu_1, \ldots, \mu_K\}$ в результате минимизируя сумму внутрикластерных квадратов $\min\limits_{C,\mu} \sum\limits_{k=1}^{K} \sum\limits_{y_i \in C_k} \|y_i - \mu_k\|_2^2$.

Инициализируется $K$ различных центроидов $\mu_k^{(0)}$ путем случайного выбора без повторений из множества $Y$.

Итерации $t = 0,1,2\ldots$ повторяются до выполнения критерия сходимости $\max\limits_k \|\mu_k^{(t+1)} - \mu_k^{(t)}\|_2 < tol$, где $tol$ — заданное значение порога сходимости, или до достижения заданного числа итераций.

Для каждой точки $y_i$ находится ближайший центроид $z_i = \operatorname{argmin}\limits_{1 \le k \le K} \|y_i - \mu_k^{(t)}\|_2^2$, и $y_i$ добавляется в кластер $C_{zi}$.

Для каждого кластера $C_k$ пересчитывается центроид $\mu_k^{(t+1)} = \dfrac{1}{|C_k|} \sum\limits_{y_i \in C_k} y_i$.

Таким образом, на выходе получается массив меток $\{z_i\}$, где $z_i \in \{1, \ldots, K\}$ — индекс кластера для $y_i$, центроиды $\{\mu_k\}$, оптимальные в смысле минимизации суммы квадратов.

В каждой группе $C_k$ также выделяется медоид по сумме линейных расстояний $m_k = \operatorname{argmin}\limits_{y \in C_k} \sum\limits_{y' \in C_k} \|y - y'\|_2$, чтобы получить представителя, принадлежащего самому набору данных.

В результате кластеризации получается компактный набор медоидов, репрезентирующих различные стратегии компромисса без утраты информации о ключевых областях Парето-фронта.

**Метрики качества и вычислительных затрат.** Качество кластеризации оценивается метрикой репрезентативности центров $\Delta_{cm}$, измеряющей согласованность среднего центра с реальной точкой данных. Для каждого кластера $C_k$ с центроидом $\mu_k$ определяется медоид $m_k \in C_k$ как точка, минимизирующая суммарную внутрикластерную дистанцию. Кластерный разрыв «центроид–медоид» равен $\Delta_k^{(cm)} = \|\mu_k - m_k\|_2$. Рассчитывается сводный показатель по разбиению $\Delta_{cm} = \dfrac{1}{|\mathcal{K}|} \sum\limits_{k \in \mathcal{K}} \Delta_k^{(cm)}$, где усреднение ведется по непустым кластерам $\mathcal{K} \subseteq \{1, \ldots, K\}$. Чем меньше $\Delta_{cm}$, тем надежнее медоиды представляют кластеры.

Вычислительные затраты фиксируются и сопоставляются для двух конфигураций. В случае применения только алгоритма *k*-means измеряется время кластеризации полного множества. Для последовательного применения алгоритмов OPTICS и *k*-means время делится на фазы: построение упорядочения и пороговая фильтрация, и кластеризация алгоритмом *k*-means на ядре решений. Оба варианта сравниваются при одинаковых параметрах кластеризации.

## Экспериментальная часть и обсуждение результатов

Для проверки эффективности предложенного метода двухэтапной кластеризации были проведены два эксперимента на наборе трехмерных точек Парето-фронта, сформированных по критериям надежности $\lambda$, стоимости $Pr$ и занимаемой площади $S$ компонентов [25]. Во всех испытаниях параметр числа соседей для алгоритма OPTICS выбран $m = 3$, параметр числа кластеров для алгоритма *k*-means — $K = 10$. Визуализации исходных множеств точек представлены на рис. 1.

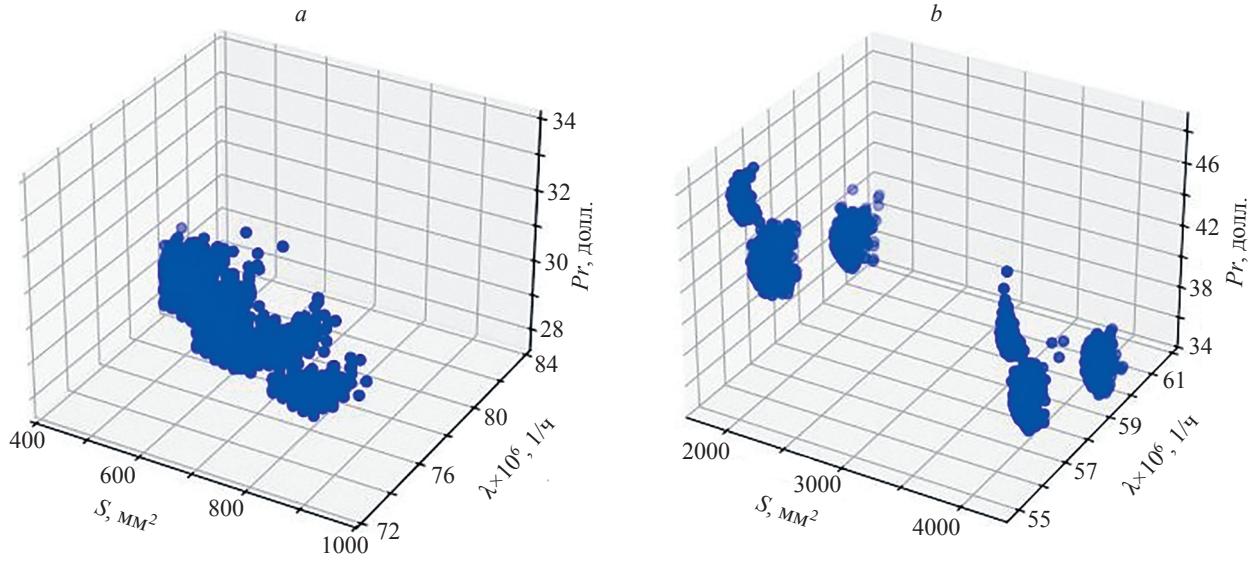


*Рис. 1.* Исходные множества: 1 (*a*) и 2 (*b*)

*Fig. 1.* Initial sets: 1 (*a*) and 2 (*b*)

Под множеством 1 в работе понимается первое исходное множество точек приближенного Парето-фронта, характеризующееся компактным распределением в пространстве критериев. Под множеством 2 понимается второе исходное множество точек Парето-фронта, отличающееся большей протяженностью и неоднородностью плотности в пространстве критериев. Оба множества получены в рамках одной постановки задачи МКО, но различаются характером распределения и числом найденных недоминируемых решений.

Эксперименты проводились на одном персональном компьютере с использованием языка программирования Python. Для обеспечения статистической достоверности каждое экспериментальное испытание запускалось несколько десятков раз, а полученные результаты по представленным метрикам усреднялись.

На рис. 2 представлена диаграмма досягаемости, сформированная в ходе работы алгоритма OPTICS на множестве 1. На оси абсцисс каждая точка соответствует одному решению, упорядоченному алгоритмом OPTICS, на оси ординат отложено расстояние досягаемости, необходимое для присоединения текущей точки к кластеру и вычисляемое как евклидово расстояние в пространстве критериев, представленное в условных единицах.

Результаты фильтрации: примерный порог для шума — 6,176; количество шумовых точек — 12; количество ядровых точек — 1226.

После фильтрации алгоритмом OPTICS оставшееся ядро из 1226 точек было разбито алгоритмом *k*-means на 10 кластеров. Их характеристики приведены в табл. 1.

Итоговый график распределения кластеров и шумовых точек множества 1 показан на рис. 3.

В рамках эксперимента показатели репрезентативности центров $\Delta_{cm}$ составили: для двухэтапной кластеризации — 2,133, для кластеризации алгоритмом *k*-means — 5,094. Время выполнения для двухэтапной кластеризации составило 2,379 с (фильтрация алгоритмом OPTICS — 1,306 с и кластеризация алгоритмом *k*-means — 1,073 с) и для кластеризации алгоритмом *k*-means — 1,233 с.

На рис. 4 представлена диаграмма досягаемости, сформированная в ходе работы алгоритма OPTICS на множестве 2.

Резкий вертикальный выброс на диаграмме досягаемости рис. 4, наблюдаемый в средней части упорядоченного профиля, соответствует одиночной точке с ано-

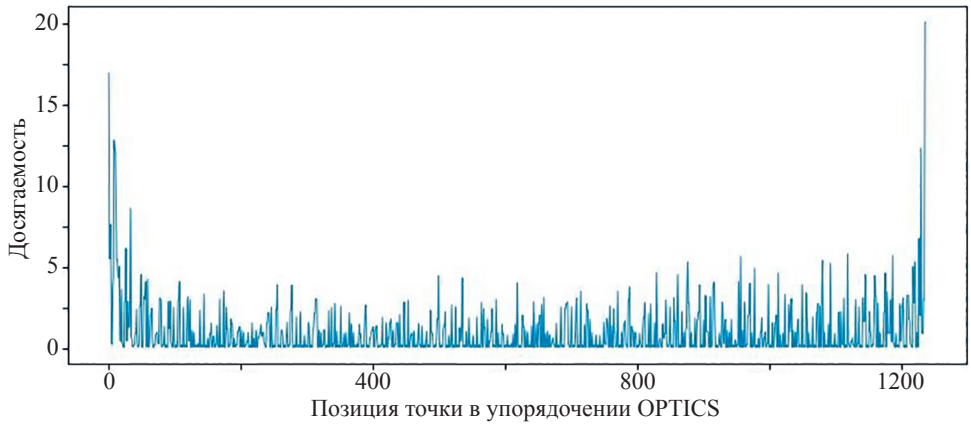


*Рис. 2*. Диаграмма досягаемости множества 1

*Fig. 2*. Reachability plot for dataset 1

*Таблица 1*. Кластеризация множества 1

*Table 1*. Clustering of dataset 1

| Кластер | Центроиды | | | Размер | Медоиды | | |
|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | | 1 | 2 | 3 |
| 0 | 727,490 | 74,063 | 30,762 | 167 | 727,230 | 74,303 | 30,441 |
| 1 | 916,351 | 72,651 | 30,895 | 115 | 914,548 | 72,924 | 30,140 |
| 2 | 560,797 | 76,405 | 29,781 | 117 | 561,392 | 76,361 | 29,609 |
| 3 | 789,880 | 73,883 | 30,417 | 131 | 789,240 | 73,587 | 30,620 |
| 4 | 527,423 | 76,941 | 29,568 | 84 | 528,923 | 77,811 | 28,256 |
| 5 | 596,331 | 75,709 | 30,095 | 132 | 599,380 | 75,573 | 29,772 |
| 6 | 674,077 | 74,589 | 30,562 | 165 | 670,082 | 74,926 | 30,237 |
| 7 | 631,200 | 75,246 | 30,324 | 148 | 631,860 | 75,150 | 30,380 |
| 8 | 475,529 | 78,292 | 29,134 | 47 | 480,660 | 77,477 | 29,142 |
| 9 | 848,513 | 73,368 | 30,441 | 120 | 850,800 | 72,775 | 30,850 |

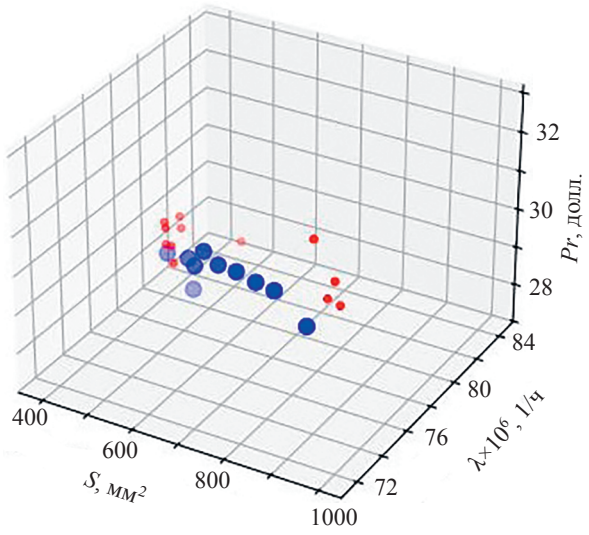


*Рис. 3*. График распределения медоидов кластеров (синие точки) и шумовых точек (красные точки) исходного множества 1

*Fig. 3*. Clusters (blue points) and noise points (red points) distribution plot for initial dataset 1

мально большим значением расстояния досягаемости. Данный пик возникает в момент перехода алгоритма OPTICS между различными плотностными областями и указывает на изолированное решение, удаленное от соседних кластерных структур, которое классифицируется как шумовое.

Результаты фильтрации: примерный порог для шума — 5,587; количество шумовых точек — 25; количество ядровых точек — 2514.

После фильтрации алгоритмом OPTICS оставшиеся 2514 точек были разбиты алгоритмом *k*-means на 10 кластеров. Их характеристики приведены в табл. 2.

Итоговый график распределения кластеров и шумовых точек множества 2 показан на рис. 5.

Показатели репрезентативности центров $\Delta_{cm}$ составили: для двухэтапной кластеризации — 2,845, для кластеризации алгоритмом *k*-means — 6,137. Время выполнения для двухэтапной кластеризации равно 5,348 с

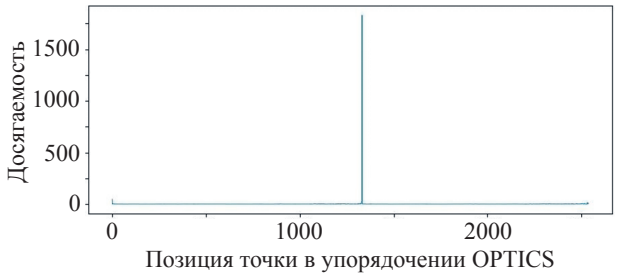


*Рис. 4*. Диаграмма досягаемости множества 2

*Fig. 4*. Reachability plot for dataset 2

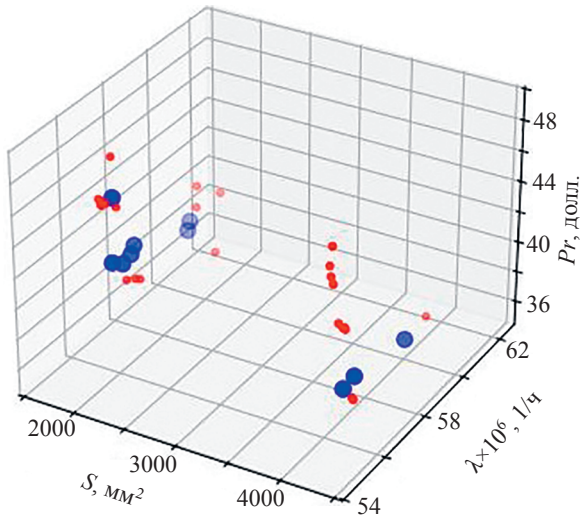


*Рис. 5*. График распределения медоидов кластеров (синие точки) и шумовых точек (красные точки) исходного множества 2

*Fig. 5*. Clusters (blue points) and noise points (red points) distribution plot for initial dataset 2

(фильтрация алгоритмом OPTICS 3,470 с и кластеризация алгоритмом *k*-means 1,778 с) и для кластеризации алгоритмом *k*-means — 2,736 с.

В экспериментах с помощью алгоритма OPTICS была выявлена небольшая доля шумовых точек (менее

*Таблица 2*. Кластеризация множества 2

*Table 2*. Clustering of dataset 2

| Кластер | Центроиды | | | Размер | Медоиды | | |
|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | | 1 | 2 | 3 |
| 0 | 1969,513 | 56,279 | 43,005 | 146 | 1968,812 | 55,931 | 41,746 |
| 1 | 1925,393 | 57,083 | 41,695 | 186 | 1924,404 | 56,576 | 40,822 |
| 2 | 2026,010 | 55,465 | 45,677 | 86 | 2022,312 | 55,767 | 46,228 |
| 3 | 1837,390 | 58,257 | 40,074 | 250 | 1836,749 | 57,474 | 40,810 |
| 4 | 4236,148 | 55,802 | 40,815 | 269 | 4228,508 | 55,720 | 38,470 |
| 5 | 4037,979 | 59,060 | 36,535 | 454 | 4042,924 | 59,796 | 36,082 |
| 6 | 1740,556 | 60,346 | 39,054 | 150 | 1743,888 | 60,494 | 38,692 |
| 7 | 4132,440 | 57,469 | 37,698 | 473 | 4131,237 | 56,769 | 37,714 |
| 8 | 1881,333 | 57,740 | 40,724 | 243 | 1882,129 | 57,130 | 40,705 |
| 9 | 1791,665 | 59,090 | 39,710 | 120 | 1792,496 | 60,161 | 38,545 |

1 %) без ручной настройки радиуса: по профилю досягаемости были выбраны порог шума около 6,176 для множества 1 и около 5,587 для множества 2. При этом из общего числа точек сохранилось более 99 % ядровых точек, что свидетельствует о малой чувствительности метода к локальным неоднородностям плотности.

Применение алгоритма *k*-means обеспечило репрезентативное разбиение ядра в обоих случаях: размеры кластеров варьируются от 47 до 167 в эксперименте 1 и от 86 до 473 в эксперименте 2. Большие кластеры (кластеры 5 и 7 в эксперименте 2) отражают наиболее плотные области компромиссов, малые (кластер 8) — редко встречающиеся варианты.

В обоих наборах точек фильтрация алгоритмом OPTICS существенно улучшила репрезентативность кластеров: $\Delta_{cm}$ снизилась с 5,094 до 2,133 (примерно 42 %) и с 6,137 до 2,845 (примерно 46 %) по сравнению с кластеризацией *k*-means без фильтрации. Улучшение качества достигается увеличением дополнительного времени на фильтрацию: суммарное время двухэтапной кластеризации составило 2,379 с против 1,233 с без фильтрации для множества 1, и 5,248 с против 2,736 с — множества 2 (увеличение порядка 90 % в обоих экспериментах). С учетом того, что с помощью алгоритма OPTICS автоматически было отсечено порядка 1 % шума и сохранено более 99 % ядровых точек, полученное снижение $\Delta_{cm}$ можно считать обоснованной ценой: медоиды удобны для интерпретации как типовые решения, а центры кластеров — как устойчивые ориентиры для выбора.

В большинстве кластеров расстояние между центроидом и соответствующим медоидом невелико, что указывает на хорошую репрезентативность медоидов.

Различие абсолютных масштабов координат (первые кластеры в эксперименте 2 лежат в диапазоне около 2000 по критерию 1, в то время как в эксперименте 1 — в диапазоне около 400) указывает на то, что схема эффективно адаптируется к разным масштабам данных без изменения параметров алгоритмов.

Таким образом, проведенные эксперименты подтвердили работоспособность и универсальность метода двухэтапной кластеризации для анализа приближенного Парето-фронта: алгоритмы стабильно отфильтровывают шум и формируют компактный набор репрезентативных решений без ручной настройки.

## Заключение

В работе предложена и исследована двухэтапная кластеризация приближенного Парето-фронта, основанная на последовательном применении плотностного алгоритма Ordering Points to Identify the Clustering Structure и алгоритма *k*-means с последующим выделением медоидов. В обоих проведенных экспериментах с помощью алгоритма Ordering Points to Identify the Clustering Structure надежно отделены шумовые точки (менее 1 % от общего числа) без ручной настройки параметров, сохранив более 99 % ядровых решений. Последующая кластеризация ядра алгоритмом *k*-means дала репрезентативное разбиение на 10 групп, отличающихся плотностью и масштабом областей компромиссов. При этом по метрике репрезентативности центров зафиксирован существенный выигрыш относительно варианта без фильтрации, что достигнуто при умеренном росте суммарного времени выполнения. Расхождение между центроидами и медоидами оказалось незначительным, что подтверждает практическую применимость выделенных медоидов в качестве реальных вариантов решений.

Сфера применения этой методики широка — она используется в инженерной оптимизации, логистике, энергетике, финансовом моделировании и управлении рисками, а также в медицине и других областях, где требуется компактный анализ компромиссных решений многокритериальных задач.

Дальнейшие исследования будут направлены на внедрение адаптивных методов для автоматического определения оптимальных параметров используемых алгоритмов, а также обеспечение адаптации в процессе работы к динамическим изменениям многокритериальных задач.

## Литература


1. Зак Ю.А. Множество Парето для критериев эффективности, представленных стохастическими и нечеткими данными // Искусственный интеллект и принятие решений. 2014. № 2. С. 89–101.
2. Ногин В.Д. Множество и принцип Парето: Учебное пособие. СПб: Издательско-полиграфическая ассоциация высших учебных заведений. 2022. 110 с.
3. Брестер К.Ю., Становов В.В., Семенкина О.Э., Семенкин Е.С. О применении эволюционных алгоритмов при анализе больших данных // Искусственный интеллект и принятие решений. 2017. № 3. С. 82–93.
4. Garey M.R., Johnson D.S. Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman and Company, 1979. 340 p.
5. Zitzler E., Deb K., Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results // Evolutionary Computation. 2000. V. 8. N 2. P. 173–195. https://doi.org/10.1162/106365600568202
6. Ватутин Э.И. Решение дискретных комбинаторных оптимизационных задач с использованием эвристических методов: методические указания по выполнению лабораторных и практических работ по дисциплине «Основы комбинаторной оптимизации». Курск: Юго-Западный государственный университет, 2016. 30 с.


## References


1. Zack Yu.A. The set of Pareto efficiency criteria presented stochastic and fuzzy data. *Artificial Intelligence and Decision Making*, 2014, no. 2, pp. 89–101. (in Russian)
2. Nogin V.D. *The Pareto Set and Principle*. St. Petersburg, Izdatelsko-Poligraficheskaya Assotsiatsiya Vysshikh Uchebnykh Zavedeniy, 2022, 110 p. (in Russian)
3. Brester Ch.Yu., Stanovov V. V., Semenkina O. E., Semenkin E. S. About the use of evolutionary algorithms in big data analysis. *Artificial Intelligence and Decision Making*, 2017, no. 3, pp. 82–93. (in Russian)
4. Garey M.R., Johnson D.S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, 1979, 340 p.
5. Zitzler E., Deb K., Thiele L. Comparison of multiobjective evolutionary algorithms: empirical results. *Evolutionary Computation*, 2000, vol. 8, no. 2, pp. 173–195. https://doi.org/10.1162/106365600568202
6. Vatutin E.I. *Solving Discrete Combinatorial Optimization Problems using Heuristic Methods: Methodological Guidelines for Laboratory and Practical Works in the Course «Fundamentals of Combinatorial Optimization»*. Kursk, South-West State University Publ., 2016, 30 p. (in Russian)
7. Gladkov L.A., Kureichik V.V., Kureichik V.M. *Genetic Algorithms*. Moscow, Fizmatlit Publ., 2010, 368 p. (in Russian)

7. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. М.: ФИЗМАТЛИТ, 2010. 368 с.
8. Вирсански Э. Генетические алгоритмы на Python. М.: ДМК Пресс, 2020. 286 с.
9. Everitt B.S., Landau S., Leese M., Stahl D. Cluster Analysis. John Wiley & Sons, 2011. 346 p.
10. Клинов Д.А., Григорян К.А. Разработка методики сегментации пользователей с помощью алгоритмов кластеризации и расширенной аналитики // Электронные библиотеки. 2022. Т. 25. № 2. С. 137–147. https://doi.org/10.26907/1562-5419-2022-25-2-137-147
11. Бондаренко И.Б., Гатчин Ю.А., Гераничев В.Н. Синтез оптимальных искусственных нейронных сетей с помощью модифицированного генетического алгоритма // Научно-технический вестник информационных технологий, механики и оптики. 2012. № 2 (78). С. 51–55.
12. Барсегян А.А. Анализ данных и процессов. СПб: БХВ-Петербург, 2009. 512 с.
13. Семериков А.В., Глазырин М.А. Кластеризация студентов университета иерархическим и KMeans методами // ИТ Арктика. 2021. № 3. С. 41–58.
14. Кисляков А.Н., Поляков С.В. Иерархические методы кластеризации в задаче поиска аномальных наблюдений на основе групп с нарушенной симметрией // Управленческое консультирование. 2020. № 5 (137). С. 116–127. https://doi.org/10.22394/1726-1139-2020-5-116-127
15. Иванов А.А. Кластеризация данных на основе марковской цепи с помощью алгоритма DBSCAN // Новые информационные технологии в автоматизированных системах. 2018. № 21. С. 315–319.
16. Трушкова К.Н., Калайда В.Т. Кластеризация спутниковых изображений облачных полей на основе алгоритма DBSCAN // Известия вузов. Физика. 2013. Т. 56. № 8-3. С. 356–358.
17. Murphy K.P. Machine Learning: A Probabilistic Perspective. The MIT Press, 2012. 1104 p.
18. Гарафиев И.З., Гарафиева Г.И. Кластеризация вакансий инженеров: методы K-средних и модель гауссовой смеси // Управленческий учет. 2024. № 9. С. 234–240.
19. Chung F.R.K. Spectral Graph Theory. American Mathematical Society, 1997. 207 p.
20. Баринов А.Е., Захаров А.А., Жизняков А.Л. Алгоритм спектральной кластеризации с ограничениями для выделения лица человека на изображениях // Динамика систем, механизмов и машин. 2016. № 4. С. 222–228.
21. Пылов П.А., Протодьяконов А.В. Спектральная кластеризация для сегментации изображения // Инновации. Наука. Образование. 2020. № 23. С. 274–277.
22. Han J., Kamber M., Pei J. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2011. 744 p.
23. Ankerst M., Breunig M.M., Kriegel H.P., Sander J. OPTICS: ordering points to identify the clustering structure // Proc. of the 1999 ACM SIGMOD International Conference on Management of Data. 1999. P. 49–60. https://doi.org/10.1145/304182.304187
24. Попова О.А. Анализ и выбор метода кластеризации для текстовых документов короткой длины с целью реализации в модуле рекомендательной системы вуза // XXI век: итоги прошлого и проблемы настоящего плюс. 2023. Т. 12. № 4 (64). С. 89–102.
25. Юртаев А.Г., Степанов М.Ф. Решение задачи выбора компонентной базы при проектировании электронных блоков авионики // Математические методы в технологиях и технике. 2025. № 2. С. 170–174.

8. Wirsansky E. *Hands-On Genetic Algorithms with Python: Applying Genetic Algorithms to Solve Real-World Deep Learning and Artificial Intelligence Problems*. Packt Publishing, 2020, 346 p.
9. Everitt B.S., Landau S., Leese M., Stahl D. *Cluster Analysis*. John Wiley & Sons, 2011, 346 p.
10. Klinov D.A., Grigorian K.A. Development of a method for user segmentation using clustering algorithms and advanced analytics. *Russian Digital Libraries Journal*, 2022, vol. 25, no. 2, pp. 137–147. (in Russian). https://doi.org/10.26907/1562-5419-2022-25-2-137-147
11. Bondarenko I.B., Gatchin Y.A., Geranichev V.N. Synthesis of optimal artificial neural networks by modified genetic algorithm. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2012, no. 2 (78), pp. 51–55. (in Russian)
12. Barsegyan A.A. *Data and Process Analysis*. St. Petersburg, BHV-Peterburg, 2009, 512 p. (in Russian)
13. Semerikov A.V., Glazyrin M.A. Clustering university students by hierarchical and KMeans methods. *IT Arctica*, 2021, no. 3, pp. 41–58. (in Russian)
14. Kislyakov A.N., Polyakov S.V. Hierarchical clustering methods in a task to find abnormal observations based on groups with broken symmetry. *Administrative Consulting*, 2020, no. 5 (137), pp. 116–127. (in Russian). https://doi.org/10.22394/1726-1139-2020-5-116-127
15. Ivanov A.A. Data clustering based on a Markov chain using the DBSCAN algorithm. *Novye Informatsionnye Tekhnologii v Avtomatizirovannykh Sistemakh*, 2018, no. 21, pp. 315–319. (in Russian)
16. Trushkova K.N., Kalaida V.T. The clusterization of satellite images of cloudy fields on basis of algorithm DBSCAN. *Izvestiya Vuzov. Fizika*, 2013, vol. 56, no. 8-3, pp. 356–358. (in Russian)
17. Murphy K.P. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012, 1104 p.
18. Garafiev I.Z., Garafieva G.I. Clustering engineering vacancies: K-means and gaussian mixture model. *Management Accounting*, 2024, no. 9, pp. 234–240. (in Russian)
19. Chung F.R.K. Spectral Graph Theory. American *Mathematical Society*, 1997, 207 p.
20. Barinov A.E., Zakharov A.A., Zhiznyakov A.L. Spectral clustering algorithm with constraints for human face extraction in images. *Dinamika Sistem, Mekhanizmov i Mashin*, 2016, no. 4, pp. 222–228. (in Russian)
21. Pylov P.A., Protodyakonov A.V. Spectral clustering for image segmentation. *Innovatsii. Nauka. Obrazovanie*, 2020, no.23, pp. 274–277. (in Russian)
22. Han J., Kamber M., Pei J. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2011, 744 p.
23. Ankerst M., Breunig M.M., Kriegel H.P., Sander J. OPTICS: ordering points to identify the clustering structure. *Proc. of the 1999 ACM SIGMOD International Conference on Management of Data*, 1999, pp. 49–60. https://doi.org/10.1145/304182.304187
24. Popova O.A. Analysis and selection of a method of clusterization of short length text documents for implementation in the university recommender system module. *XXI Century: Resumes of the Past and Challenges of the Present Plus*, 2023, vol. 12, no. 4 (64), pp. 89–102. (in Russian)
25. Yurtaev A.G., Stepanov M.F. Solution to the problem of selecting a component base in the design of electronic avionics blocks. *Mathematical Methods in Technologies and Technics*, 2025, no. 2, pp. 170–174. (in Russian)


## Автор

**Юртаев Александр Григорьевич** — аспирант, Саратовский государственный технический университет имени Гагарина Ю.А., Саратов, 410054, Российская Федерация, https://orcid.org/0009-0002-4146-4322, agyurtaev@mail.ru

## Author

**Alexander G. Yurtaev** — PhD Student, Yuri Gagarin State Technical University of Saratov, Saratov, 410054, Russian Federation, https://orcid.org/0009-0002-4146-4322, agyurtaev@mail.ru

*Статья поступила в редакцию 10.07.2025*
*Одобрена после рецензирования 08.11.2025*
*Принята к печати 26.01.2026*

*Received 10.07.2025*
*Approved after reviewing 08.11.2025*
*Accepted 26.01.2026*