

[https://doi.org/10.51885/3134-8009\\_IJS\\_2026\\_1\\_2](https://doi.org/10.51885/3134-8009_IJS_2026_1_2)

XFTAP 73.31.75

## КЛАСТЕРЛЕУ АЛГОРИТМДЕРІН ПАЙДАЛАНА ОТЫРЫП, ТАСЫМАЛДАУ МАРШРУТТАРЫН ҚАЛЫПТАСТЫРУ

## ФОРМИРОВАНИЕ РАЗВОЗОЧНЫХ МАРШРУТОВ С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ КЛАСТЕРИЗАЦИИ

## CREATING DELIVERY ROUTES USING CLUSTERING ALGORITHMS

В.Н. Вдовин <sup>1</sup>, А. Х. Машекенова <sup>1</sup>, А.Р. Дуйсен <sup>1\*</sup>, Suresh Alapati <sup>2</sup>

<sup>1</sup>«Д. Серікбаев атындағы Шығыс Қазақстан техникалық университеті» КеАҚ, Өскемен қ., Қазақстан

<sup>2</sup>Kyungsung University, Busan, Korea

\*Жауапты автор: Дуйсен Аялым Ринатовна, e-mail: arduisen@mail.ru

### Түйінді сөздер:

кластерлеу, тасымалдау маршруттары, k-means алгоритмі, DBSCAN алгоритмі, көлік қызметі аймағы, «тармақтар мен шекаралар әдісі».

### ТҮЙІНДЕМЕ

Кластерлеу-жеткізу нүктелерін олардың сипаттамаларына негізделген белгілі бір белгілер бойынша топтастыру арқылы жеткізу нүктелерін топтарға (кластерлерге) алдын ала біріктіру процесі. Нәтижесінде маршрут нүктелердің барлық жиынтығы бойынша емес, әр кластердің ішінде бөлек жоспарланады, бұл тапсырманың күрделілігін айтарлықтай төмендетеді. Зерттеудің мақсаты жеткізу объектілерін кластерлеудің оңтайлы әдісін анықтау және оның көмегімен қаланың сауда желісіне қызмет көрсету кезінде тасымалдау маршруттарын әзірлеу болып табылады. Бұл мақалада DBSCAN және k-means алгоритмдерінің жұмысы қарастырылады. Салыстырмалы талдау DBSCAN жоғары нүктелік тығыздықта тұрақты нәтиже бермейтінін көрсетті, ал k-means әдісі айқын кластерлерді көрсетеді. Кластерлердің оңтайлы саны «шын-тақ» және «силуэт» әдістерімен анықталады. Python бағдарламалау тілінде алгоритмдерді енгізу және кейіннен «тармақтар мен шекаралар әдісі» қолдану автомобильдердің маршруттардағы жүгірісін қысқартуға және дайын өнімді жеткізуге көлік шығындарын азайтуға мүмкіндік береді.

### Ключевые слова:

кластеризация, развозочные маршруты, алгоритм k-means, алгоритм DBSCAN, зона транспортного обслуживания, «метод ветвей и границ».

### АННОТАЦИЯ

Кластеризация представляет собой процесс предварительного объединения точек доставки в группы (кластеры), путём группировки точек по определённым признакам, основываясь на их характеристиках. В результате, маршрут планируется не по всей совокупности точек, а отдельно внутри каждого кластера, что существенно снижает сложность задачи. Целью исследования является определение оптимального метода кластеризации объектов доставки и разработка с его помощью развозочных маршрутов при обслуживании торговой сети города. В данной статье рассмотрена работа алгоритмов DBSCAN и k-means. Сравнительный анализ показал, что DBSCAN не



обеспечивает устойчивого результата при высокой плотности точек, тогда как метод k-means формирует четкие кластеры. Оптимальное количество кластеров определено методами «локтя» и «силуэта». Реализация алгоритмов на языке программирования Python и последующее использование «метода ветвей и границ» позволяют сократить пробег автомобилей на маршрутах и снизить транспортные расходы на доставку готовой продукции.

---

**Keywords:**

clustering, delivery routes, k-means algorithm, DBSCAN algorithm, transport service area, "Branch-and-bound (BnB)" method.

---

**ABSTRACT**

Clustering is the process of organizing delivery points into groups (clusters) by certain attributes based on their characteristics. As a result, the route is planned not by the whole set of points, but in parts within each cluster, which significantly reduces the complexity of the problem. The aim of the study is to identify the optimal clustering method of delivery objects and use it to create delivery routes for the city's retail network.

In this article, the work of the DBSCAN and k-means algorithms is considered. A comparative analysis showed that DBSCAN does not provide a stable result with a high density of points, whereas the k-means method demonstrates clear clusters. The optimal number of clusters is determined by the "elbow" and "silhouette" methods. The implementation of algorithms in the Python programming language and the subsequent use of the "Branch-and-bound" method will reduce the mileage of cars on routes and reduce transportation costs for the delivery of finished products.

---

**КІРІСПЕ**

Тасымалдауды жоспарлаудың негізгі кезеңдерінің бірі – маршруттарды құруға дейін жүзеге асырылатын жеткізу нысандарын кластерлеу болып табылады. Кластерлеу әдістерін қолдану тарату маршруттарын қалыптастыру мәселесін тиімді шешуге мүмкіндік береді. Тауарларды жеткізудің ұтымды маршруттарын әзірлеу мәселесі қаланың сауда желісіне қызмет көрсету кезінде ерекше өзектілікке ие. Бұл – ондаған, тіпті жүздеген жеткізу нүктелері, көлік паркінің шектеулі мүмкіндіктері, шарттық міндеттемелердің уақытылы орындалуы сияқты факторлармен байланысты. Ұтымды маршруттарды қалыптастыру көлік шығындарын қысқартуға және тауарды уақтылы жеткізуді қамтамасыз етуге мүмкіндік береді (Бочкарев А.А., Клочков В.Н., 2008; Овсянникова Г.А., Шошин Д.А., 2014; Mokhammad R. және т.б., 2021).

Зерттеу нысаны ретінде нан-тоқаш өнімдерін өндіретін және оларды қаланың 460 сауда нүктесіне меншікті көлік құралдарымен жеткізетін кәсіпорын таңдап алынды. Жеткізу нүктелерінің саны аз болған жағдайда оңтайлы маршрутты қалыптастыру міндеті салыстырмалы түрде оңай шешіледі. Алайда нүктелер саны артқан сайын, оларды айналып өту мүмкін комбинацияларының айтарлықтай көбеюіне байланысты бұл міндет едәуір күрделене түседі. Кластерлеу әдістерін қолдану жеткізу нүктелерін географиялық жақындығы белгісі бойынша топтастыру арқылы аталған мәселені тиімді шешуге мүмкіндік береді. Нәтижесінде маршрут барлық нүктелер жиынтығы бойынша емес, әрбір кластер шегінде жеке жоспарланады, бұл есептің күрделілігін айтарлықтай төмендетеді.

**ЗЕРТТЕУ МАТЕРИАЛДАРЫ МЕН ӘДІСТЕРІ**

Оңтайлы маршруттарды әзірлеу автомобильдік тасымалдауды жоспарлаудың маңызды міндеті болып табылады. Бұл көлік шығындарының өнімді өткізу өзіндік құнында елеулі үлеске ие болуымен түсіндіріледі. Ұтымды маршруттарды құру

мәселесін маршрутизация шешеді, соның нәтижесінде кәсіпорынның көлік шығындары қысқарып, клиенттерге қызмет көрсету сапасы артады. Тасымалдауды жоспарлаудың негізгі кезеңдерінің бірі – маршруттарды құруға дейін жүзеге асырылатын жеткізу нысандарын кластерлеу болып табылады. Практикалық міндеттердің көпшілігінде көлік паркінің біркелкі жүктелуін және жалпы жеткізу уақытының қысқаруын қамтамасыз ету үшін тұтынушыларды әртүрлі маршруттар бойынша теңгерімді бөлу қажет.

Деректерді кластерлеудің бес негізгі математикалық әдісі талданды. Әрбір әдіс өнімді жеткізудің сауда нүктелерін кластерлеуге қолдану мүмкіндігі тұрғысынан қарастырылды (Aravind c. R., 2023; Тюрин а. Г., Зуев И. О., 2014; Семенов Ю. Н., Семенова О. С., 2016). Спектрлік және агломеративті кластерлеу алгоритмдері, сондай-ақ гаусс қоспалары моделі биологияда және әлеуметтік ғылымдарда тиімдірек қолданылатындықтан, әрі қарайғы зерттеу үшін екі алгоритм таңдалды: k-means (немесе k-орташа) және DBSCAN – Density-based spatial clustering of applications with noise (Апельцин Л., 2023).

Кластерлеуге деректерді дайындау, оның ішінде деректерді тазалау және географиялық координаталарды алу 2GIS әрі көмегімен жүзеге асырылды. Бұл кіріс деректерінің жеткілікті жоғары дәлдігін қамтамасыз етуге мүмкіндік берді, ол кластерлік алгоритмдердің дұрыс жұмыс істеуінің және кейінгі маршрутизацияның негізгі шарты болып табылады.

DBSCAN алгоритмі арқылы кластерлерді қалыптастыру кезінде k-distance plot графигін пайдалана отырып екі параметр анықталды: кластер радиусы және кластердегі нүктелердің ең аз саны.

K-MEANS алгоритмін қолдану барысында алдымен табиғи географиялық кедергілерді (ертiс және үлбi өзeндерi) ескеру үшiн нүктелер қызмет көрсету аймақтары бойынша сегменттелдi. Кейiн «локоть әдiсi» (elbow method) және «силуэт әдiсi» (silhouette method) арқылы кластерлердiң оңтайлы саны анықталды.

«Локоть әдісі» әртүрлі k мәндері үшін кластерлеу жүргізіп, кластерлер санына байланысты кластерішілік дисперсияның жиынтық мәнінің графигін құруды көздейді. Кластерішілік дисперсия (немесе объектілер мен олардың центроидтары арасындағы қашықтықтардың квадраттарының қосындысы) кластерлердің жинақтылық деңгейін көрсетеді: дисперсия неғұрлым төмен болса, кластерлер соғұрлым «реттелген» және «біртекті» болады.

«Силуэт әдісі» объектілердің кластерлерге қаншалықты жақсы бөлінетінін бағалайды. Силуэт коэффициенті неғұрлым жоғары болса, объект өз кластеріне соғұрлым жақсы сәйкес келеді.

Python бағдарламалау тілінде код әзірленді, бұл процесті автоматтандыруға және шешімнің ауқымдылығын (масштабталуын) қамтамасыз етуге мүмкіндік берді.

Маршрутизацияның соңғы кезеңі маршрут бойындағы сауда нүктелерін аралау ретін анықтау болып табылады. Бұл міндетті шешу үшін жоғары дәлдігімен танымал «тармақтар мен шектер әдісі» (branch and bound) таңдалды. Осы әдіс бойынша барлық есептеулер ms excel ортасында жүргізілді. Симметриялық матрицаны пайдалану және маршрут ұзындысының жергілікті төменгі шектерін есептеу арқылы тұтынушыларға қызмет көрсетудің оңтайлы кезектілігі анықталды.

## **НӘТИЖЕЛЕР ЖӘНЕ ОЛАРДЫ ТАЛҚЫЛАУ**

Иерархиялық емес шешімді табу қажет болған және кластерлер саны алдын ала белгісіз жағдайда нүктелердің тығыздығына негізделген, шуылдарды ескеретін кеңістіктік кластерлеу алгоритмін қолдану орынды. Бұл алгоритм ағылшын тілінде density-based spatial clustering of applications with noise деп аталады және қысқаша dbscan ретінде белгілі.

Алгоритм нүктелердің үш түрін ажыратады: негізгі (ядролық), шекаралық және шуылдық нүктелер. Негізгі (ядролық) нүктелер – осы нүктеден қолжетімді барлық нүктелермен бірге кластерді қалыптастырады.  $p$  нүктесі негізгі болып саналады, егер  $\varepsilon$ -окрестігінде кемінде  $\text{minpts}$  (minpts) нүкте орналасса ( $U_\varepsilon(x) = \{u \in U: \rho(x, u) \leq \varepsilon\}$ ).

Бұл жағдайда  $|u_\varepsilon(x)| \geq \text{minpts}$  шарты орындалып, аталған нүктелер  $p$  нүктесінен тікелей жетімді болып табылады.  $q$  нүктесі (шекаралық)  $p$  нүктесінен тікелей жетімді деп есептеледі, егер ол  $p$  нүктесінен  $\varepsilon$ -ден аспайтын қашықтықта орналасса.  $q$  нүктесі  $p$  нүктесінен жетімді болып табылады, егер  $p_1, p_2, \dots, p_n$  нүктелерінен тұратын жол бар болып, мұнда  $p_1 = p$  және  $p_n = q$ , ал әрбір  $p_{i+1}$  нүктесі  $p_i$  нүктесінен тікелей жетімді болса (кугаевских а. В., 2022).

*Epsilon* ( $\text{eps}$ ,  $\varepsilon$ ) параметрі окрестік радиусын білдіреді, ал  $\text{minpoints}$  ( $\text{min\_samples}$ ) – кластердегі нүктелердің ең аз санын көрсетеді.

Кәсіпорын қызмет көрсететін сауда нүктелерін кластерлеу үшін python бағдарламалау тілінде DBSCAN алгоритмін қолдана отырып код әзірленді (1-сурет).

```
import pandas as pd
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import DBSCAN
from sklearn.neighbors import NearestNeighbors
import matplotlib.pyplot as plt
from sklearn.metrics import silhouette_score

file_path = "final_merged_with_route.csv"
df = pd.read_csv(file_path)

df = df.dropna(subset=['Longitude', 'Latitude', 'Отгружено количество', 'Возврат количество', 'Всего вес, кг'])

features = df[['Longitude', 'Latitude', 'Отгружено количество', 'Возврат количество', 'Всего вес, кг']]

scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)

neighbors = NearestNeighbors(n_neighbors=10)
neighbors_fit = neighbors.fit(features_scaled)
distances, indices = neighbors_fit.kneighbors(features_scaled)

distances = sorted(distances[:, 3], reverse=True)
plt.plot(distances)
plt.title('K-distance plot')
plt.xlabel('Points')
plt.ylabel('Distance to 4th nearest neighbor')
plt.show()

dbscan = DBSCAN(eps=0.7, min_samples=15)
df['Cluster_DBSCAN'] = dbscan.fit_predict(features_scaled)

print(f"Количество кластеров: {len(set(df['Cluster_DBSCAN']))} - (1 if -1 in df['Cluster_DBSCAN'].values else 0)")
print(f"Количество точек с шумом: {(df['Cluster_DBSCAN'] == -1).sum()}")

silhouette = silhouette_score(features_scaled, df['Cluster_DBSCAN'])
print(f"Silhouette Score: {silhouette}")

plt.scatter(df['Longitude'], df['Latitude'], c=df['Cluster_DBSCAN'], cmap='viridis')
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.title('DBSCAN Clustering of Delivery Points')
plt.colorbar(label='cluster')
plt.show()
display(df.head())
```

**1-сурет.** DBSCAN алгоритмінің коды ( $\text{eps} = 0,7$ ;  $\text{min\_samples} = 15$  кезінде)

Ескерту – Python бағдарламасында автор құрастырған

Код жазу кітапханаларды импорттаудан басталады. DBSCAN алгоритміне деректер келесі құралдар арқылы енгізілді:

*Pandas* – кестелік деректерді өңдеуге арналған бағдарламалық кітапхана;

*StandardScaler* – деректерді стандарттайды (орташа мәнді нөлге, стандарттық ауытқуды бірге келтіру);

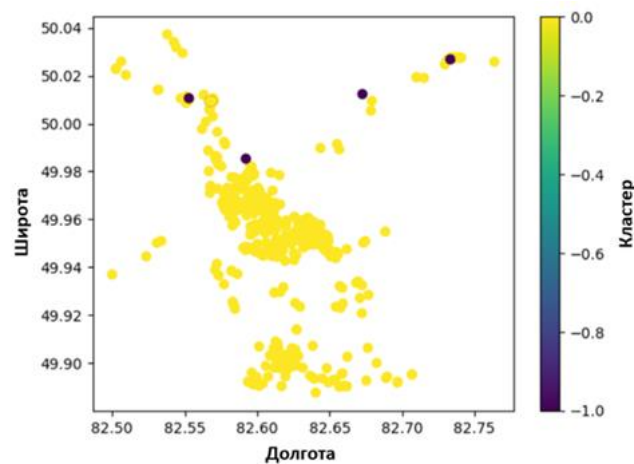
*Matplotlib.pyplot* – кластерлерді визуализациялауға арналған;

*Silhouette\_score* (силуэт коэффициенті) – кластерлеу сапасын бағалау метрикасы.

Одан әрі сауда нүктелері туралы ақпаратты қамтитын csv-файл жүктеліп, кластерлеу жүргізілетін белгілер таңдап алынды. Белгілерді масштабтау DBSCAN алгоритмінің дұрыс жұмыс істеуі үшін қажет, себебі ол деректер шкаласына сезімтал. Кейін алгоритм үшін  $min\_samples = 10$  және  $eps = 2$  параметрлері орнатылып, кластерлер мен шуылдық нүктелер санын есептеу командалары енгізілді. Шуылдық нүктелер «-1» белгісімен анықталады.

Кластерлеу сапасын бағалау үшін мәні -1 мен 1 аралығында өзгертін силуэт коэффициенті (*silhouette score*) қолданылды. 1-ге жақын мән кластерлердің айқын бөлінгенін көрсетсе, 0-ге жақын мән олардың бір-бірімен қабаттасуын білдіреді, ал теріс мән ықтимал қателіктің бар екенін көрсетеді. Соңғы кезеңде кластерлер *longitude* және *latitude* координаталары бойынша визуализацияланды. Кодтың соңғы жолында кластерлеуден кейінгі датафреймнің алғашқы бес жолы көрсетілді.

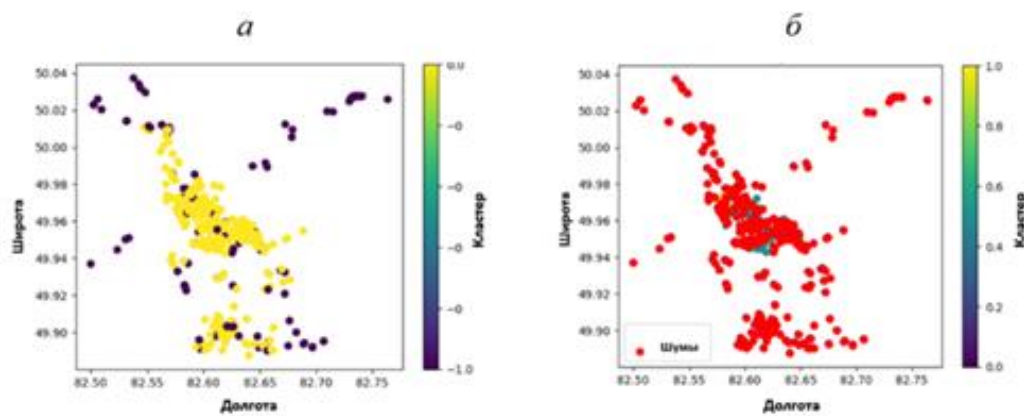
DBSCAN алгоритмі 7 шуылдық нүктесі бар 1 кластерді қалыптастырды (2-сурет). Кластердің силуэт коэффициенті 0,81-ге тең.



**2-сурет.** DBSCAN алгоритмін іске асыру

*Ескерту – Python бағдарламасында автор құрастырған*

Аталған параметр мәндері кластерлерді қалыптастырмағандықтан, келесі итерацияларда кластер радиусы мен минималды нүктелер саны кездейсоқ таңдалды (3-сурет).



**3-сурет.** DBSCAN алгоритмінің қайталап іске асырылуы:

(а)  $eps = 0,7$ ;  $min\_samples = 15$  (б)  $eps = 0,3$ ;  $min\_samples = 15$

*Ескерту – Python бағдарламасында автор құрастырған.*

Эксперимент барысында алгоритмге әртүрлі *eps* және *min\_samples* мәндері берілді, алайда ешқайсысы нәтиже берген жоқ; айқын кластерлер қалыптаспады, олардың саны анықталған жоқ, көптеген нүктелер шуылдық ретінде белгіленді. Сол себепті, зерттеу нысаны үшін DBSCAN алгоритмін әрі қарай қолдану тиімсіз деп танылды.

Келесі кезеңде k-means алгоритмі іске асырылды. Бұл әдіс кластерлердегі нүктелердің орталықтарынан квадраттық ауытқулардың жалпы сомасын минимизациялауға бағытталған (Дюран Б., Оделл П., 1977; Ершов К.С., Романова Т.Н., 2016).

$$V = \sum_{i=1}^k \sum_{x_j \in S_i} (x_j - \mu_i)^2, \quad (1)$$

мұндағы:

$k$  – кластерлер саны;

$S_i$  – алынған  $i$ -ші кластер,  $i = 1, 2, \dots, k$ ;

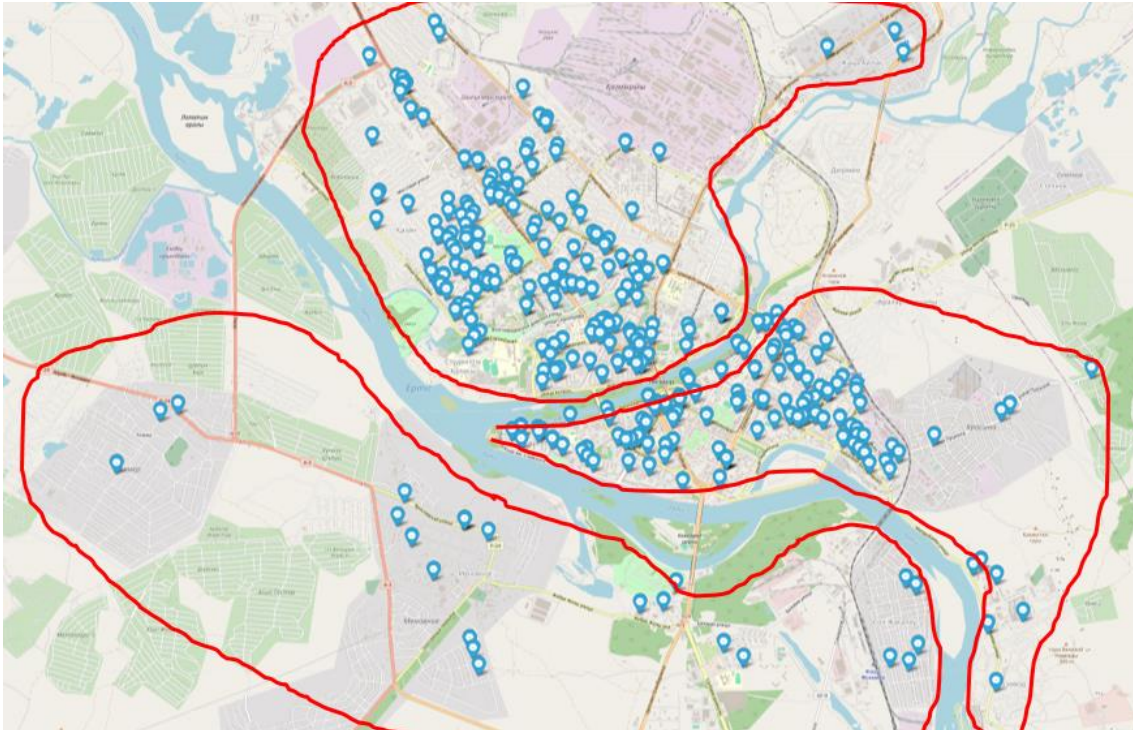
$\mu_i$  –  $S_i$  кластеріндегі  $x_j$  векторларының массалық орталығы.

Осы жағдайда метрика ретінде евклид қашықтығы қолданылды:

$$\text{dist}(x, x') = \sqrt{\sum_{i=1}^n (x_i - x'_i)^2} \quad (2)$$

мұндағы:  $(x_i - x'_i)$  – Евклид кеңістігінің бастапқы нүктесінен (исходной точки) шыққан векторлар;  $n$  –  $n$  өлшемді кеңістік.

K-MEANS алгоритмі үшін міндетті кезеңдердің бірі – нүктелерді қызмет көрсету аймақтары бойынша сегменттеу. Мұндай бөлу қаланың табиғи географиялық кедергілерін ескеруге мүмкіндік береді. Осы жағдайда аймақтардың шекарасы ретінде Ертіс және Үлбі өзендері алынды. Үш секторға бөлу – Жоғарғы (Үлбі аудан), Оң жақ (Октябрьский аймағы) және Сол жақ (КШТ) – жеткізу сауда нүктелерінің кластерлерін қалыптастыруды жеңілдетті (4-сурет).



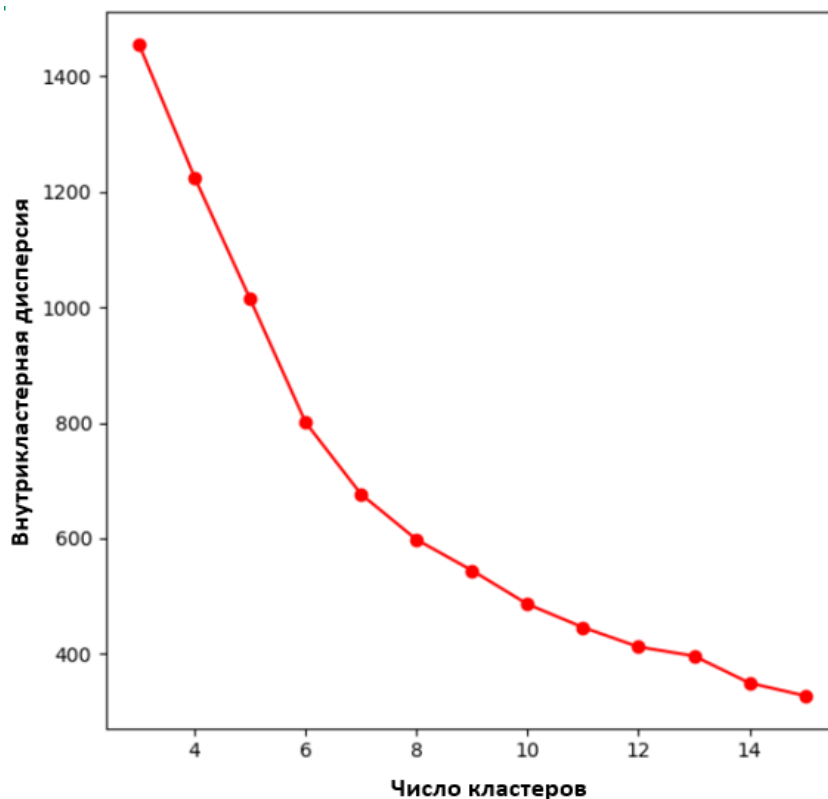
**4-сурет.** Нүктелерді қызмет көрсету аймақтары бойынша сегментациялау  
Ескерту – 2GIS API негізінде автор құрастырған

Қызмет көрсету аймақтарының шекараларын анықтағаннан кейін кластерлер санын белгілеуге болады. Санды немесе кәсіпорын қызметкерлері анықтаған маршруттар саны немесе қолда бар көлік құралдарының саны негізінде өздігінен белгілейді, немесе кластерлер санын анықтау әдістерін қолдануға болады, мысалы, elbow method («локоть әдісі») және silhouette method («силуэт әдісі»).

«Локоть әдісі» әртүрлі  $k$  мәндері үшін кластерлеу жүргізіп, кластерлер санына байланысты кластерішілік дисперсияның (inertia) жиынтық мәнін көрсететін график құруға негізделген. Кластерішілік дисперсия (немесе объектілер мен олардың центроидтары арасындағы қашықтық квадраттарының қосындысы) кластерлердің қаншалықты жинақталғанын көрсетеді. Дисперсия неғұрлым төмен болса, кластерлер соғұрлым «реттелген» және «біртекті» болады.

Әдіс келесі кезеңдерді қамтиды:

- $k$ -means алгоритмін әртүрлі  $k$  мәндері үшін іске қосу, мысалы, 1-ден 10-ға дейін;
- әрбір  $k$  мәні үшін кластерішілік дисперсияны есептеу;
- график құру:  $X$  осінде  $k$  мәндері,  $Y$  осінде – сәйкес кластерішілік дисперсия мәндері;
- графиктегі «локоть» нүктесін табу: бұл нүкте кластерлер санын арттырған кезде кластерішілік дисперсияның айтарлықтай төмендемеуі байқалатын сәт (5-сурет).



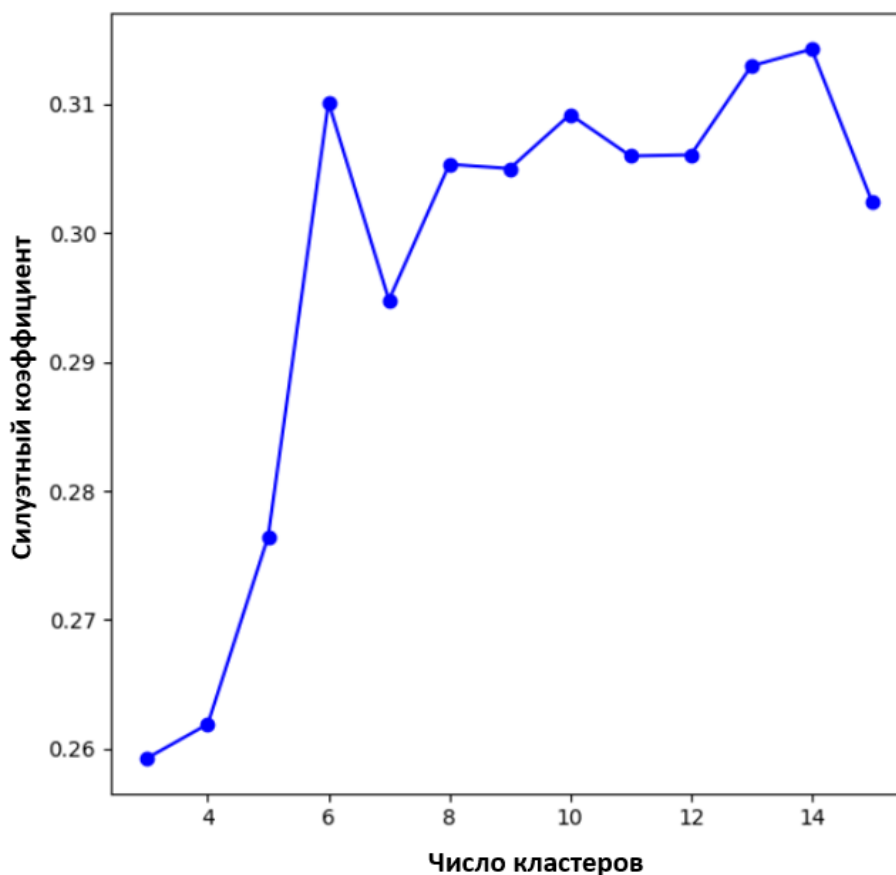
**5-сурет.** Кластерлер санын «локоть әдісі» арқылы анықтау

*Ескерту – автор құрастырған*

График мәліметтеріне сүйенсек, оңтайлы кластер саны 6...9 аралығында орналасады. Алты кластер үшін кластерішілік дисперсия 800,7, жеті кластер үшін – 676,4, сегіз кластер үшін – 597,9, тоғыз кластер үшін – 544,5 тең.

«Силуэт әдісі» объектілердің кластерлерге қаншалықты жақсы бөлінетінін бағалайды. Силуэт коэффициенті неғұрлым жоғары болса, объект өз кластеріне соғұрлым жақсы сәйкес келеді.

6-суретте 6, 8, 9...14 кластер үшін силуэт мәні шамалы өзгеріп, 0,30 – 0,31 аралығында қалатыны көрінеді.



**6-сурет.** Кластерлер санын «силуэт әдісі» арқылы анықтау

*Ескерту – автор құрастырған*

Кәсіпорынның сауда нүктелерін k-means алгоритмі арқылы кластерлеу үшін Python тілінде код та жасалды (5-сурет). Төменде әзірленген кодтың сипаттамасы және кластерлеу нәтижелері графиктерде көрсетілген.

DBSCAN алгоритмінде болғандай, кітапханаларды импорттау келесі құралдар арқылы жүзеге асырылады: Pandas – кестелік деректерді өңдеу; matplotlib.pyplot – кластерлерді визуализациялау; silhouette\_score – кластерлеу сапасын бағалау метрикасы.

Сондай-ақ, сауда нүктелері туралы ақпаратты қамтитын CSV-файл жүктеліп, белгілер масштабталады.

«Локоть» және «силуэт» әдістерінің нәтижелеріне сүйене отырып, алгоритмге кластерлер саны  $k = 9$  деп берілді. Кластерлеу сапасын бағалау үшін silhouette score мәні енгізілді.

Одан кейін inertia командасы қолданылады – кластерішілік дисперсия есептеледі, ол «локоть әдісінде» қолданылады, сонымен қатар координаталар бойынша кластерлер графигі салынады.

Inertia енгізілген соң кластерлердің оңтайлы санын анықтау үшін график құрылады. Кодтың соңында датафрейм CSV-файлға сақталады.

Эксперименттік есептеулер әртүрлі кластерлер саны үшін жүргізілді (6-дан 10-ға дейін).

```
import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
import matplotlib.pyplot as plt
import folium
from sklearn.metrics import silhouette_score

file_path = "final_merged_with_route.csv"
df = pd.read_csv(file_path)

df = df.dropna(subset=['Longitude', 'Latitude', 'Отгружено количество', 'Возврат количество', 'Всего вес, кг'])

features = df[['Longitude', 'Latitude', 'Отгружено количество', 'Возврат количество', 'Всего вес, кг']]

scaler = StandardScaler()
features_scaled = scaler.fit_transform(features)

kmeans = KMeans(n_clusters=7, random_state=42)
df['Cluster'] = kmeans.fit_predict(features_scaled)

silhouette = silhouette_score(features_scaled, df['Cluster'])
print(f"Silhouette Score: {silhouette:.4f}")

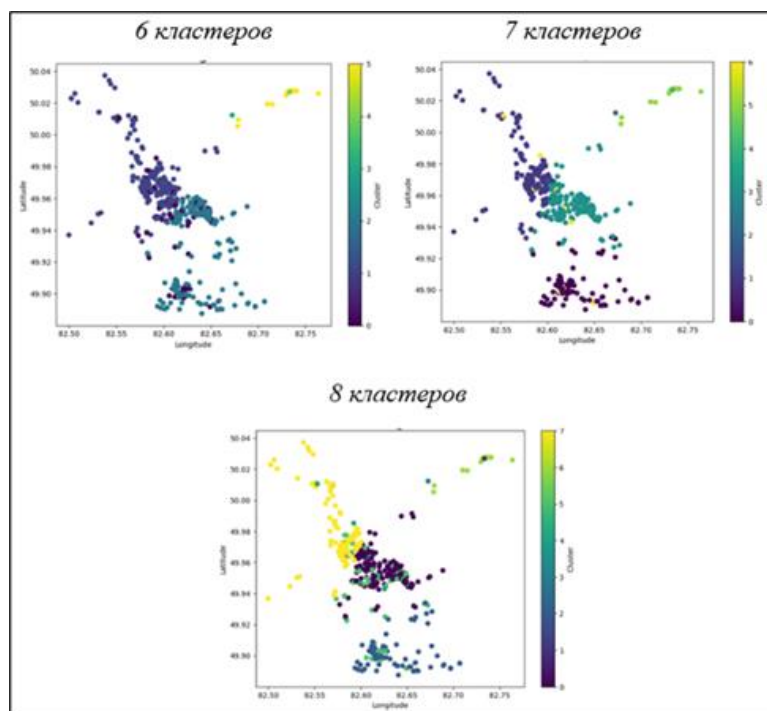
plt.scatter(df['Longitude'], df['Latitude'], c=df['Cluster'], cmap='viridis')
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.title('K-means Clustering of Delivery Points')
plt.colorbar(label='Cluster')
plt.show()

df.to_csv("final_merged_with_clusters.csv", index=False)
```

### 5-сурет. k-means алгоритміне арналған код

Ескерту – Python бағдарламасында автор құрастырған

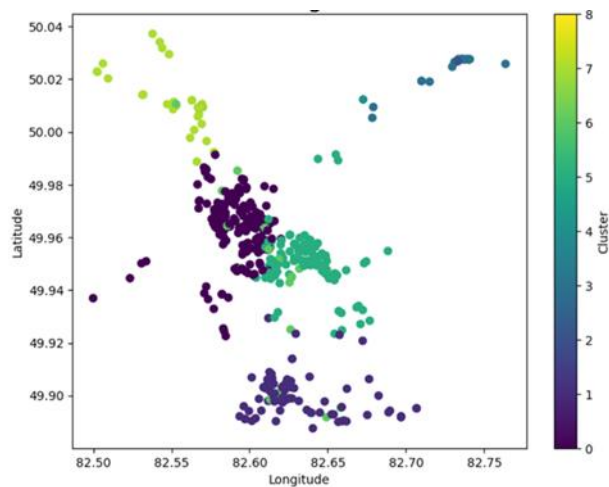
6–8 кластер қолданылған жағдайда (6-сурет) кластерлер арасындағы шекаралар айқын емес, кластерлер бір-біріне «қабаттасып» көрінеді.



### 6-сурет. Алгоритмді 6, 7, 8 кластермен іске асыру

Ескерту – Python бағдарламасында автор құрастырған

Нүктелерді бөлудің егжей-тегжейлілігі кластерлер саны 9 болған кезде байқалады (7-сурет). Силуэт коэффициенті 0,305-ке тең, ал кластерішілік дисперсия 544,47 құрайды.

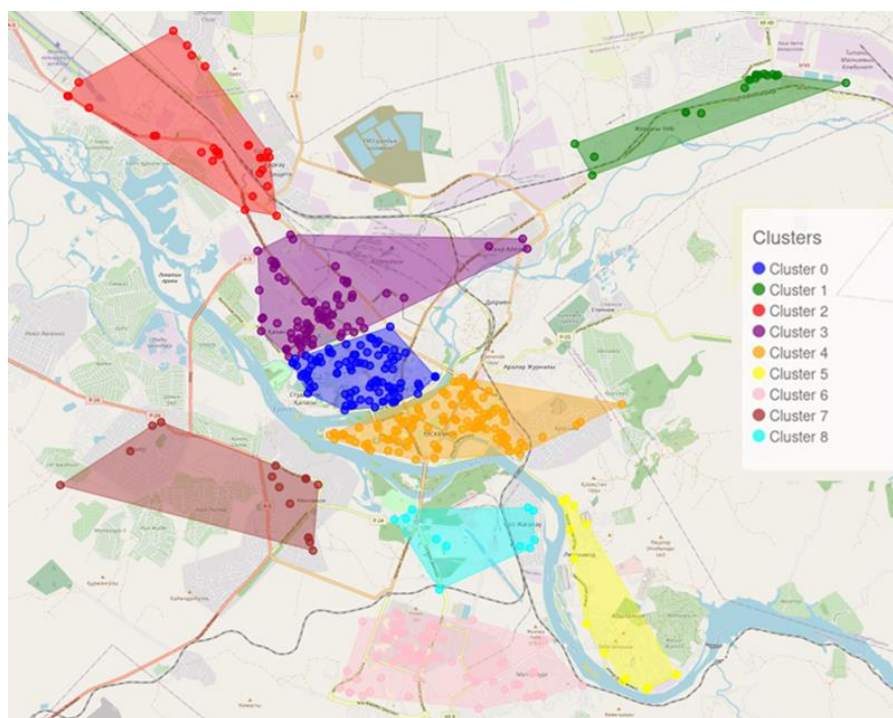


**7-сурет.** Алгоритмді 9 кластермен іске асыру

*Ескерту – Python бағдарламасында автор құрастырған*

Тоғыз кластер қолданылған жағдайда топтар арасындағы шекаралар айқын көрінеді, объектілердің тығыз орналасқан аймақтары бір-бірінен бөлінген және аралас аймақтар саны азайтылған.

k-means алгоритмі 9 ішкі жиынтықты қалыптастырды: Жоғарғы секторда – 4 кластер, Оң жақ секторда – 2 кластер, Сол жақ секторда – 3 кластер. Кластерлеу нәтижесі 8-суретте көрсетілген.



**8-сурет.** k-means алгоритмімен 9 кластерге бөлудің нәтижесі

*Ескерту – 2GIS API негізінде автор құрастырған*

Тасымалдау маршруттарды қалыптастырудың соңғы кезеңі – маршрут бойындағы сауда нүктелерін аралау ретін анықтау болып табылады. Бұл міндетті шешу үшін «тармақтар мен шектер әдісі» (branch and bound) қолданылды. Әдіс симметриялық матрицаны пайдалануды және маршрут ұзындығының жергілікті төменгі шектерін есептеуді қамтиды (Домке Э. Р., 2014; Павлов Л. А., 2020; Галяутдинов Р. Р., 2023). Осы әдісті қолдану арқылы сауда нүктелерін аралаудың оңтайлы кезектілігі анықталды, бұл маршрут бойынша орташа жүру қашықтығын 13 %-ға қысқартуға мүмкіндік берді және дайын өнімді жеткізудегі көлік шығындарын азайтуға жағдай жасады.

### ҚОРЫТЫНДЫ

Зерттеу барысында екі кластерлеу алгоритмі үшін python тілінде кодтар әзірленді. Эксперимент барысында dbscan алгоритміне әртүрлі кластер радиусы (eps параметрі) және негізгі нүктені анықтау үшін қажетті ең аз нүктелер саны (min\_samples параметрі) мәндері берілді, бірақ ешқайсысы нәтиже бермеді: айқын кластерлер қалыптаспады, олардың саны анықталған жоқ. Сол себепті, бұл алгоритм жеткізу маршруттары бойынша тығыз орналасқан нүктелерді біріктіруде жеткілікті тиімді емес деп танылды.

K-means алгоритмін қолданудың практикалық маңыздылығы мен тиімділігі көрсетілді. Бұл алгоритм кластер ішіндегі қашықтықты минимизациялады. Барлық кластерлер көлік құралдарының максималды сыйымдылық шегіне – 4000 бірлік өнімге – сәйкес келеді.

**МҮДДЕЛЕР ҚАҚТЫҒЫСЫ:** авторлар мүдделер қақтығысы жоқ екенін мәлімдейді.

**ЖАСАНДЫ ИНТЕЛЛЕКТ ТЕХНОЛОГИЯЛАРЫН ПАЙДАЛАНУ ТУРАЛЫ ХАБАРЛАМА:** ғылыми мақала авторлары мақаланы жазу барысында жасанды интеллект технологияларын пайдаланбағанын мәлімдейді.

### ӘДЕБИЕТТЕР ТІЗІМІ

- Бочкарев, А.А., & Клочков, В.Н. (2008). Методология планирования доставки мелкопартионных грузов в условиях крупного города. Вестник СГТУ, 5–15. // Bochkarev, A. A., & Klochkov, V. N. (2008). Metodologiya planirovaniya dostavki melkopartiionnykh gruzov v usloviyakh krupnogo goroda. Vestnik SGTU, 5–15. [Methodology for planning small-lot cargo delivery in a large city] (In Russ.)
- Овсянникова, Г.Л., & Шошин, Д.А. (2014). Организация автомобильных перевозок мелких партий груза на предприятии молокозавода. Молодой ученый, 11, 87–90. // Ovsyannikova, G.L., & Shoshin, D.A. (2014). Organizatsiya avtomobilnykh perevozok melkikh partiy gruzha na predpriyatii molokozavoda. Molodoy uchenyy, 11, 87–90. [Organization of road transportation of small cargo lots at a dairy plant] (In Russ.)
- Aravind, C. R. (2023). Exploring clustering algorithms: Explanation and use cases. Neptune.ai. <https://neptune.ai/blog/clustering-algorithms>
- Mokhammad, R., & Indratno, S. W. (2021). Dynamic items delivery network: Prediction and clustering. Heliyon, 7(8), e07640. <https://doi.org/10.1016/j.heliyon.2021.e07640>
- Тюрин, А. Г., & Зуев, И. О. (2014). Кластерный анализ: методы и алгоритмы кластеризации. Вестник МГТУ МИРЭА, 2, 86–97. // Tyurin, A.G., & Zuev, I.O. (2014). Klasternyy analiz: metody i algoritmy klasterizatsii. Vestnik MGTU MIREA, 2, 86–97. [Cluster analysis: methods and clustering algorithms] (In Russ.)
- Семенов, Ю. Н., & Семенова, О. С. (2016). Применение методов кластеризации при организации междугородных перевозок грузов. Вестник КузГТУ, 6, 201–204. // Semenov, Yu. N., & Semenova, O. S. (2016). Primenenie metodov klasterizatsii pri organizatsii mezhdugorodnykh perevozok gruzov. Vestnik KuzGTU, 6, 201–204. [Application of clustering methods in organizing intercity freight transportation] (In Russ.)

- Апельцин, Л. (2023). Data Science в действии. Санкт-Петербург: Питер. // Apeltsin, L. (2023). Data Science v deystvii. Saint Petersburg: Piter. [Data Science in action] (In Russ.)
- IT-ресурс Хабр. (2023). Кластеризация в ML: от теоретических основ популярных алгоритмов к их реализации с нуля на Python. <https://habr.com/ru/articles/798331/> // IT-resurs Khabr. (2023). Klasterizatsiya v ML: ot teoreticheskikh osnov populyarnykh algoritmov k ikh realizatsii s nulya na Python. [Clustering in ML: from theoretical foundations of popular algorithms to their implementation from scratch in Python] (In Russ.)
- Кугаевских, А. В., Муромцев, Д. И., & Кирсанова, О. В. (2022). Классические методы машинного обучения: учебное пособие. Санкт-Петербург: НИУ ИТМО. // Kugaevskikh, A. V., Muromtsev, D. I., & Kirsanova, O. V. (2022). Klassicheskie metody mashinnogo obucheniya: uchebnoe posobie. Saint Petersburg: NRU ITMO. [Classical machine learning methods: study guide] (In Russ.)
- Duran, B., & Odell, P. (1977). Cluster analysis. New York: Springer. // Duran, B., & Odell, P. (1977). Klasternyy analiz. Moscow: Statistika. [Cluster analysis] (In Russ.)
- Ершов, К. С., & Романова, Т. Н. (2016). Анализ и классификация алгоритмов кластеризации. Новые информационные технологии в автоматизированных системах. // Ershov, K. S., & Romanova, T. N. (2016). Analiz i klassifikatsiya algoritmov klasterizatsii. Novye informatsionnye tekhnologii v avtomatizirovannykh sistemakh. [Analysis and classification of clustering algorithms] (In Russ.)
- IT-resource GeeksforGeeks. (2025). Elbow method vs. silhouette score: Which is better? <https://www.geeksforgeeks.org/machine-learning/elbow-method-vs-silhouette-score-which-is-better/>
- Домке, Э. Р., & Жесткова, С. А. (2014). Методы оптимизации маршрутных схем развозки грузов автомобильным транспортом: учебное пособие. Пенза: ПГУАС. // Domke, E. R., & Zhestkova, S. A. (2014). Metody optimizatsii marshrutnykh skhem razvozki грузов avtomobilnym transportom. Penza: PGUAS. [Methods for optimizing road freight distribution routes] (In Russ.)
- Павлов, Л. А., & Первова, Н. В. (2020). Структуры и алгоритмы обработки данных (2-е изд.). Санкт-Петербург: Лань. // Pavlov, L. A., & Pervova, N. V. (2020). Struktury i algoritmy obrabotki dannykh (2nd ed.). Saint Petersburg: Lan. [Data structures and processing algorithms] (In Russ.)
- Галяутдинов, Р. Р. (2023). Задача коммивояжера — метод ветвей и границ. <https://galyautdinov.ru/post/zadacha-kommivoyazhera> // Galyautdinov, R. R. (2023). Zadacha kommivoyazhera — metod vetvey i granits. [The traveling salesman problem — branch and bound method] (In Russ.)

**Авторлар туралы мәліметтер**  
**Информация об авторах**  
**Information about authors**



**Вдовин Владимир Николаевич** – т.ғ.к., аға оқытушы «Д. Серікбаев атындағы Шығыс Қазақстан техникалық университеті» КЕАҚ

**Вдовин Владимир Николаевич** – к.т.н, старший преподаватель НАО «Восточно-Казахстанский технический университет имени Д.Серикбаева»

**Vdovin Vladimir** – Cand. of Tech. Sci., Senior Lecturer, NJSC «D. Serikbayev East Kazakhstan technical university»

e-mail: [vnvdovin@list.ru](mailto:vnvdovin@list.ru) ,

ORCID: <https://orcid.org/0000-0002-8273-0212>



**Машекенова Әсия Хасеновна** – т.ғ.к., «Д. Серікбаев атындағы Шығыс Қазақстан техникалық университеті» КЕАҚ Басқарма мүшесі – академиялық мәселелер жөніндегі проректор

**Машекенова Асия Хасеновна** – к.т.н., проректор по академическим вопросам НАО «Восточно-Казахстанский технический университет имени Д. Серикбаева»

**Mashekenova Asiya** – Cand. of Tech. Sci., Member of the Board - Vice-Rector for Academic Affairs NJSC « D. Serikbayev East Kazakhstan technical university»

e-mail: amashekenova@edu.ektu.kz

ORCID: <https://orcid.org/0000-0003-3556-3033>



**Дүйсен Аяулым Ринатовна** – ғылым магистрі «Д. Серікбаев атындағы Шығыс Қазақстан техникалық университеті» КЕАҚ

**Дүйсен Аяулым Ринатовна** – магистр наук НАО «Восточно-Казахстанский технический университет им. Д. Серикбаева»

**Duisen Ayaulym** – Master of Science NJSC «D. Serikbayev East Kazakhstan technical university»

e-mail: arduisen@mail.ru



**Suresh Alapati** – Ph.D, Department of Mechatronics Engineering, Busan, South Korea

**Suresh Alapati** – Ph.D, Department of Mechatronics Engineering, Busan, South Korea

**Suresh Alapati** – Ph.D, in Mechanical Engineering, Department of Mechatronics Engineering, Busan, South Korea

e-mail: suresh@ks.ac.kr

ORCID: <https://orcid.org/0000-0002-1438-1954>