

УДК 665.9

DOI: 10.15587/2313-8416.2015.38743

ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ КАЧЕСТВА КЛАСТЕРИЗАЦИИ РАЗНООБРАЗНЫХ НАБОРОВ ДАННЫХ С ПОМОЩЬЮ МОДИФИЦИРОВАННОГО АЛГОРИТМА ХАМЕЛЕОН

© Т. Б. Шатовская, А. А. Заремская

В статье рассмотрены результаты работы модифицированного алгоритма Хамелеон. Иерархический многоуровневый алгоритм состоит из нескольких этапов: построение графа, огрубление, разделение и восстановление. Главной целью работы является исследование качества кластеризации различных наборов данных с помощью набора комбинаций алгоритмов на разных этапах работы алгоритма и улучшения этапа построения через оптимизацию алгоритма выбора k при построении графа k -ближайших соседей

Ключевые слова: кластеризация, алгоритм Хамелеон, построение графа, связность, k -ближайших соседей, иерархическая кластеризация

In this work results of modified Chameleon algorithm are discussed. Hierarchical multilevel algorithms consist of several stages: building the graph, coarsening, partitioning, recovering. Exploring of clustering quality for different data sets with different combinations of algorithms on different stages of the algorithm is the main aim of the article. And also aim is improving the construction phase through the optimization algorithm of choice k in the building the graph k -nearest neighbors

Keywords: clustering, algorithm Chameleon, building the graph, connectivity, k -nearest neighbors, hierarchical clustering

1. Введение

В статье рассмотрены результаты работы модифицированного алгоритма Хамелеон. Иерархический многоуровневый алгоритм состоит из нескольких этапов: построение графа, огрубление, разделение и восстановление. На каждом их этапов могут быть использованы различные подходы и алгоритмы. Главной целью работы является исследование качества кластеризации разных наборов данных с помощью набора комбинаций алгоритмов на разных этапах алгоритма Хамелеон.

2. Литературный обзор

Анализ данных приобретает все большую и большую значимость в современном мире. В различных областях человеческой деятельности, постоянно возникает необходимость решения задач анализа, прогноза и диагностики, выявления скрытых зависимостей и поддержки принятия оптимальных решений. Актуальность данных задач обуславливается бурным ростом объема информации, развитием технологий ее сбора, хранения и организации в базах и хранилищах данных (в том числе интернет-технологий), вследствие чего точные методы анализа информации и моделирования исследуемых объектов зачастую отстают от потребностей реальной жизни. На данный момент является актуальной проблема разработки универсальных и надежных методов и подходов, пригодных для обработки информации из различных областей. В качестве базиса для таких исследований могут служить технологии и подходы математической теории распознавания и классификации и математическое моделирование.

Существует множество различных методов, которые могут быть применены для решения постав-

ленной задачи. Также существует ряд проблем для имеющихся методов. Среди таких проблем можно выделить следующие:

а) проблема обоснования качества результатов анализа. Для различных выборок и данных различные методы оценивания результата могут давать лучший результат;

б) во многих областях, а особенно в медицине, имеющиеся данные зашумлены;

в) проблема анализа большого числа разнотипных факторов;

г) нелинейность взаимосвязей; наличие пропусков, погрешностей измерения переменных.

Так как для различных наборов данных различные методы показывают наилучшие результаты, то для каждого отдельного набора данных необходим некий критерий выбора наилучшего метода.

3. Сравнение алгоритмов кластеризации

Существует много различных методов кластеризации и каждым из них можно получить различные разбиения исходного множества. Выбор определенного метода зависит от типа желаемого результата. Производительность метода с определенными типами данных зависит от характеристик сервера и технических возможностей программного обеспечения, размера множества.

Методы по способу обработки данных делятся на: Иерархические методы: Агломеративные методы AGNES (Agglomerative Nesting), Дивизимные методы DIANA (Divisive Analysis); Неиерархические методы. Итеративные. Методы по способу анализа данных делятся на четкие и нечеткие. По количеству применений алгоритмов кластеризации методы делятся на методы с одноэтапной

кластеризацией и методы с многоэтапной кластеризацией. Методы по возможности расширения объема обрабатываемых данных делятся на масштабируемые и немасштабируемые. По времени выполнения кластеризации делятся на потоковые (on-line) и не потоковые (off-line).

В последнее время ведутся активные разработки новых алгоритмов кластеризации, способ-

ных обрабатывать сверхбольшие базы данных. В них основное внимание уделяется масштабируемости. Разработаны алгоритмы, в которых методы иерархической кластеризации интегрированы с другими методами. К наиболее актуальным алгоритмам относятся: BIRCH, CURE, CHAMELEON, ROCK [1]. Сравнение данных методов представлено в табл. 1 [2].

Таблица 1

Сравнение алгоритмов кластеризации

Название алгоритма	Большие объемы данных	Устойчивость к шуму	Масштабируемость	Сложность	Определяет количество кластеров	Кластеры произвольного размера и плотности
BIRCH	+	+	+	$\theta(n \log n)$	+	-
CURE	+	+-	-	$\theta(n^2 \log n) - \theta(n^2)$	-	+-
CHAMELEON	+	+-	+	$\theta(mn + n \log n + n^2 \log m)$	+	+
ROCK	+	-	+	$\theta(\max(n^2 m, n^2 \log n))$	-	-

4. Модифицированный алгоритм Хамелеон

Хамелеон – это новый иерархический алгоритм который преодолевает ограничения существующих алгоритмов кластеризации. Данный алгоритм рассматривает динамическое моделирование в иерархической кластеризации. В нем можно выделить следующие стадии:

Построение графа. Хамелеон представляет объекты посредством часто используемого графа *k*-ближайших соседей (*k*-nearest neighbor graph). Каждая вершина в данном графе представляет один объект данных. Между вершинами существует ребро если один объект является одним из *k* ближайших соседей второго объекта. Граф *k*- ближайших соседей содержит концепцию что радиус смежности объекта определяется плотность региона в котором данный объект находится. Это позволяет выявлять естественные кластеры.

На следующем шаге строится очередь из последовательно уменьшенных гиперграфов – стадия огрубления (Coarsening Phase). Для огрубления графов может быть применено несколько существующих алгоритмов. На каждом уровне огрубления огрубление заканчивается как только размер результирующего огрубленного графа уменьшился в 1.7 раз.

На третьей стадии выполняется разделение огрубленного графа таким образом чтоб было удовлетворено ограничение баланса и оптимизирована функция разделения.

На четвертом шаге выполняется восстановление графа. Разделение огрубленного графа проецируется на следующий уровень исходного графа и выполняется алгоритм улучшения разделения (partitioning refinement algorithm).

На последней итерации Хамелеона определяется показатель схожести между каждой парой кластеров [3].

Рассмотрим детально каждый этап модифицированного алгоритма.

5. Построение графа

В данной работе рассмотрено 2 вида графов: симметричный *k*-nn граф и ассиметричный *k*-nn граф. При построении графа для каждой пары объектов измеряется «расстояние» между ними – степень похожести.

Евклидово расстояние. Наиболее распространенная функция расстояния. Представляет собой геометрическим расстоянием в многомерном пространстве:

$$p(x, x') = \sqrt{\sum_i^n (x_i - x'_i)^2}, \tag{1}$$

Квадрат евклидова расстояния. Применяется для придания большего веса более отдаленным друг от друга объектам. Это расстояние вычисляется следующим образом:

$$p(x, x') = \sum_i^n (x_i - x'_i)^2. \tag{2}$$

Расстояние городских кварталов (манхэттенское расстояние). Для этой меры влияние отдельных больших разностей (выбросов) уменьшается (т. к. они не возводятся в квадрат).

$$p(x, x') = \sum_i^n |x_i - x'_i|. \tag{3}$$

Расстояние Минковского.

$$p(x, x') = \left(\sum_{k=1}^n |x_{ik} - x_{jk}|^p \right)^{\frac{1}{p}}. \tag{4}$$

Расстояние Чебышева. Это расстояние может оказаться полезным, когда нужно определить два объекта как «различные», если они различаются по какой-либо одной координате.

$$p(x, x') = \max(|x_i - x'_i|). \quad (5)$$

Степенное расстояние. Применяется в случае, когда необходимо увеличить или уменьшить вес, относящийся к размерности, для которой соответствующие объекты сильно отличаются. Степенное расстояние вычисляется по следующей формуле:

$$p(x, x') = r \sqrt{\sum_i^n (x_i - x'_i)^p}. \quad (6)$$

где r и p — параметры, определяемые пользователем. Косинусальная схожесть (cosine similarity)

$$p(x, x') = \frac{\sum_i^n (x_i \times x'_i)}{\sqrt{\sum_i^n (x_i)^2} \times \sqrt{\sum_i^n (x'_i)^2}} + 1. \quad (7)$$

Единица добавлена во избежание отрицательных значений [4].

6. Огрубление графа

В процессе стадии огрубления строится последовательность меньших графов, каждый с меньшим количеством узлов. Огрубление графа может быть достигнуто различными способами [5, 6].

Случайное паросочетание. Узлы посещаются в случайном порядке. Если узел u не был включен в паросочетание, то беспорядочно выбирается один из его смежных узлов, который также не включен в паросочетание.

Паросочетание из тяжелых ребер (HEM). Узлы снова посещаются в случайном порядке. Однако теперь, вместо беспорядочного сочетания узла u с одним из его смежных немаркированных узлов, мы сочетаем u с узлом v таким, что вес ребра (u, v) максимален для всех доступных ребер (наиболее тяжелое ребро).

Модифицированный алгоритм паросочетания тяжелых ребер (Modified Heavy Edge Matching – HEM*) В данном алгоритме ребра посещаются в случайном порядке. Для каждой вершины u которая инцидентна выбранной вершине пусть $W(v, u)$ будет суммой весов ребер u и которые соединяют вершину u и вершины инцидентные v . В HEM* v объединяется с такой вершиной u где $W(v, u)$ максимально.

Паросочетание из наиболее тяжелых ребер (heaviest-edge matching). В алгоритме паросочетания наиболее тяжелых ребер ребра сортируются по убыванию веса и затем производится сочетание наиболее тяжелых ребер с теми которые еще не были объединены.

Модифицированное паросочетание из наиболее тяжелых ребер HEM*+. Порядок посещения вершин не фиксирован. Вершина v для каждой своей смежной вершины u , не зависимо от того, имеет ли u пару, либо ещё нет, вычисляется значение $t(v, u)$. Это значение равно сумме весов ребер, ведущих из вершин v и u в их общие смежные вершины, плюс вес

ребра $\{v, u\}$. V может выбрать u только тогда, когда $t(u, v)$ максимально для вершин.

Паросочетание легких ребер (Light Edge Matching (LEM)) В данном алгоритме при выборе ребра (u, v) вместо ребра с наибольшим весом выбирается такое ребро, у которого вес минимален.

Паросочетание из тяжелых клик (HCM). Узлы посещаются в случайном порядке. Немаркированный узел u сочетается с его немаркированным смежным узлом v так, что плотность ребер мультиузла, созданного объединением u и v , максимальна среди всех возможных мультиузлов, включающих u и другой немаркированный сосед узла u .

Сочетание тяжелых треугольников. Heavy-triangle matching (HTM). HTM сочетает 3 узла одновременно, выбирая случайным образом из необъединенных узлов узел и сочетая его с двумя его соседями таким образом чтоб сумма весов трех ребер соединяющих узлы была максимально среди всех вариантов соседей выбранного узла.

Сочетание тяжелых схем (Heaviest Schema Matching HSM). Ребра сортируются по их весу и вершины с максимальным весом ребер сочетаются первыми. Вершины с инцидентными ей вершинами сочетаются не 1 раз, а пока вес между ними достаточно велик.

Сочетание гиперребер (Hyperedge Coarsening HEC). Ребра изначально сортируются по убыванию веса и возрастанию длины. Затем ребра посещаются в полученном порядке и для каждого ребра вершины объединяются если они еще не были объединены.

Видоизмененное сочетание гиперребер (Modified Hyperedge Coarsening MHEC). На первом шаге данного алгоритма выполняется огрубление путем сочетания гиперребер HEC. На втором шаге вершины каждого ребра для которого сочетание не произошло объединяются.

Сочетание лучшего (первого) выбора (First Choice Coarsening FCC). Вершины посещаются в случайном порядке. Каждая вершина объединяется с ближайшим соседом не смотря на то был ли он объединен или нет.

7. Разделение графа

На данном этапе выполняется разделение огрубленного графа таким образом, чтоб было удовлетворено ограничение баланса и оптимизирована функция разделения (mincut). Разделение может быть выполнено как одновременным разделением на k частей так и последовательно с помощью рекурсивной бисекции.

Покоординатное разбиение (Coordinate Nested Dissection (CND)). Метод основан на рекурсивном делении пополам сети по наиболее длинной стороне.

Деление сети с использованием кривых заполняющих пространство (Space-filling Curve Techniques) На этапе инициализации выбирается квадрат, в который попадают все имеющиеся элементы множества. Квадрат маркируется для заполнения кривой. Делится на 4 подквадрата и так пока квадрат содержит не более 1 объекта. По полученной кривой составляется отсор-

тированный список элементов множества. Далее список делится на необходимое количество классов.

Алгоритм возрастающего графа (GGP) Алгоритм начинает с одного узла и наращивает область вокруг этого первого узла динамически, пока не будет включена половина узлов (или половина общего веса узлов).

Алгоритм возрастающего графа с учетом выгод (GGGP). В данном алгоритме узлы изначально отсортированы по убыванию веса сепаратора, после шаги алгоритма повторяют алгоритм возрастающего графа.

Уровневое ячеечное разбиение (Levelized Nested Dissection – LND) Выбирается начальная периферийная вершина v и помечается номером 0. Для каждой вершины вычисляется расстояние до $v0$ используя BFS (breadth-first search) алгоритм начиная с вершины $v0$ и помечается. Когда половина вершин уже помечена граф разбивается на два подграфа.

Seed-Growth bisection. В начале работы данного алгоритма выбирается 2 несмежных равных подмножества из множества вершин графа. Далее постепенно из оставшихся по одной вершины добавляются поочередно к каждому их подмножеств. На каждом шаге для добавления выбирается вершина ближайшая к подмножеству.

8. Восстановление графа и улучшение разделения

Разделение огрубленного графа проецируется на следующий уровень исходного графа и выполняется алгоритм улучшения разделения (partitioning refinement algorithm) для улучшения целевой функции не нарушая ограничение баланса.

Kernighan-Lin Algorithm (KL). KL алгоритм основан на понятии веса – величины, которая определяет выигрыш от перемещения вершины из одного

подмножества в другое. Вес рассчитывается для каждой вершины как количество соединений вершины с другим подмножеством, минус количество соединений с подмножеством, в котором вершина находится. Пока есть вершины с положительным весом, алгоритм меняет вершины с максимальным весом местами с вершинами из другого подмножества.

Fiduccia and Mattheyses. По существу схема работы алгоритма такая же, но за один его шаг перемещается только одна вершина, вместо обмена пары вершин, после этого для каждой вершины пересчитывается вес.

Граничный KL и граничный FM (Boundary KL and Boundary FM) Эти алгоритмы в точности повторяют шаги выполняемые в KL и FM с тем единственным отличием что вместо расчета выгоды для всех пар (u, v) в графе для KL или всех u и всех v для FM, рассчитываются выгоды только для граничных вершин (т. е. вершин сопряженных с вершинами из другого класса).

Ключевым шагом является поиск пары подклассов которые наиболее похожи.

Относительная связность и относительная плотность. Этот метод комбинирует относительную связность и относительную плотность, для объединения выбирается пара кластеров которые максимизируют полученную функцию.

Схожесть кластеров(Cluster Similarity) Данная мера основана на произведении количества ребер, которые соединяют два класса по отношению к количеству ребер в меньшем класса [5, 6].

9. Результаты экспериментов

Эксперименты проводились на различных известных существующих и смоделированных выборках. Результаты представлены в табл. 2.

Таблица 2

Сравнение алгоритмов кластеризации

№ выборки	Характеристики выборки			Выбранные алгоритмы	Оценки качества
	Название	max	min		
1	Elements=36 Expectation StdDeviation Asymmetry Excess Range	0 67313,18 6,01E-07 3,31E-07 300	-2,39 13141,60 2,85E-08 7,24E-09 196,96	Граф: AKNNGraph(5) Расстояние: Euclidian Огрубление: RM Разделение: FM_GGGP Восстановление: UBoundaryFM	CS=0 RIRC=0,2233 Silhouette Index=0,0683 Conn Index= 0,1853
2	Elements = 28 Expectation StdDeviation Asymmetry Excess Range	8 601,25 1,17E-05 1,03E-06 10	0 0 5,74E-06 3,45E-07 0	Граф: AKNNGraph(4) Расстояние: Euclidian Огрубление: HEM Разделение: CND Восстановление: UBoundaryFM	CS=0 RIRC=0,3102 Silhouette Index= 0,4205
3	Elements = 765 Expectation StdDeviation Asymmetry Excess Range	7,72 23913849 3,95E-13 1,81E-15 415	0 0 1,89E-13 5,22E-16 0	Граф: AKNNGraph(4) Расстояние: Manhattan Огрубление: RM Разделение: FM Восстановление: UBoundaryFM	CS=0 RIRC=0,7081 Silhouette Index=0,2929 Conn Index=0,0001
4	Elements = 143 Expectation StdDeviation Asymmetry Excess Range	2,7 1964540 7,39E-08 7,88E-12 283	0,0048 1461,73 5,52E-11 8,48E-13 0,534	Граф: AKNNGraph(6) Расстояние: Euclidian Огрубление: RM Разделение: GGGP Восстановление: UFM	Dunn=3,4105 CS=0 RIRC=0,2287 Silhouette Index=0,4240

Результаты кластеризации данных выборок представлены на рис. 1, 2.

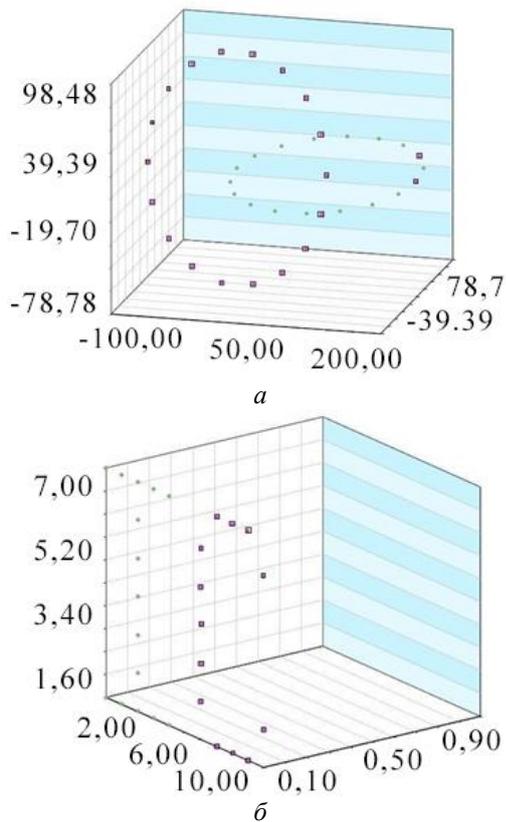


Рис. 1. Результат кластеризации: *a* – выборки 1; *б* – выборки 2

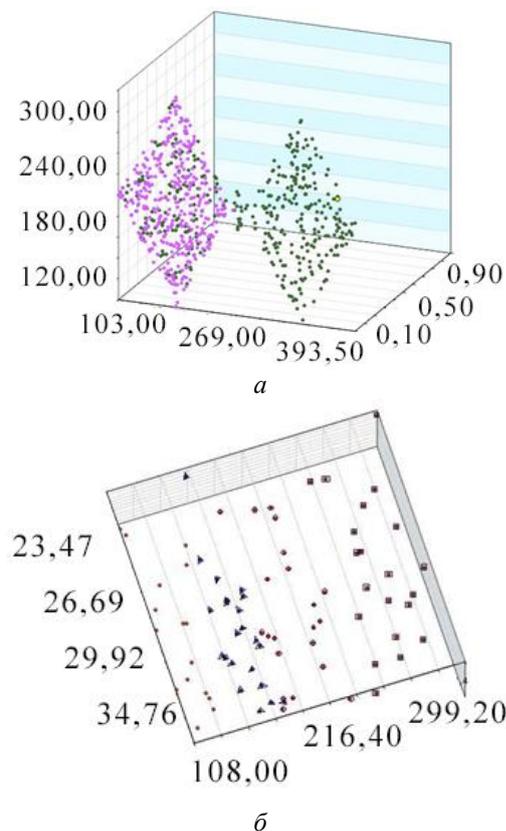


Рис. 2. Результат кластеризации: *a* – выборки 3; *б* – выборки 4

10. Выводы

На основании проведенного эксперимента можно сделать вывод что для различных выборок с разными статистическими характеристиками необходимо использовать различные алгоритмы для каждого из шагов алгоритма Хамелеон.

Литература

1. Asuncion, A. UCI Machine Learning Repository [Electronic resource] / A. Asuncion, D. J. Newman. – University of California, School of Information and Computer Science, Irvine, CA, 2007. – Available at: <http://www.ics.uci.edu/~mlern/MLRepository.html>
2. Blake, C. L., Mer, C. J. (2001): UCI repository of machine learning databases. Available at: <http://www.ics.uci.edu/~mlern/ML - Repository.html>
3. Pearson, S. An Adaptive Privacy Management System For Data Repositories [Electronic resource] / S. Pearson, M. Mont, P. Bramhall. – Trusted Systems Laboratory, Hewlett-Packard Laboratories, Bristol, UK, 2004. – Available at: <http://www.hpl.hp.com/techreports/2004/HPL-2004-211.pdf>
4. Cunningham, K. An open repository and analysis tools for fine-grained longitudinal learner data [Electronic resource] / K. Cunningham, R. Kenneth, Koedinger, A. Skogsholm, B. Leber. – Human Computer Interaction Institute, Carnegie Mellon University, 2008. – Available at: http://www.educationaldatamining.org/EDM2008/uploads/proc/16_Koedinger_45.pdf
5. Xie T. JMAPO: mining API usages from open source repositories. In: Proceedings of the International Workshop on Mining Software Repositories (MSR '06) [Electronic resource] / T. Xie, Pei. – ACM Press, New York, Shanghai, Chinapp, 2006. – P. 54–57. – Available at: <http://people.engr.ncsu.edu/txie/publications/msr06-mapo.pdf>
6. Zimmermann, T. Knowledge Collaboration by Mining Software Repositories [Electronic resource] / T. Zimmermann. – Saarland University, Saarbrücken, Germany, 2006. – Available at: <http://thomas-zimmermann.com/publications/files/zimmermann-kcsd-2006.pdf>

References

1. Asuncion, A., Newman, D. J. (2007). UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA. Available at: <http://www.ics.uci.edu/~mlern/MLRepository.html>
2. Blake, C. L., Mer, C. J. (2001). UCI repository of machine learning databases. Available at: <http://www.ics.uci.edu/~mlern/ML - Repository.html>
3. Pearson, S., Mont, M., Bramhall, P. (2004). An Adaptive Privacy Management System For Data Repositories. Trusted Systems Laboratory, Hewlett-Packard Laboratories, Bristol, UK. Available at: <http://www.hpl.hp.com/techreports/2004/HPL-2004-211.pdf>
4. Cunningham, K., Kenneth, R., Koedinger, Skogsholm, A., Leber, B. (2008). An open repository and analysis tools for fine-grained longitudinal learner data. Human Computer Interaction Institute, Carnegie Mellon University. Available at: http://www.educationaldatamining.org/EDM2008/uploads/proc/16_Koedinger_45.pdf
5. Xie, T., Pei (2006). JMAPO: mining API usages from open source repositories. In: Proceedings of the International Workshop on Mining Software Repositories (MSR '06), ACM Press, New York, China, 54–57. Available at: <http://people.engr.ncsu.edu/txie/publications/msr06-mapo.pdf>
6. Zimmermann, T. (2006). Knowledge Collaboration by Mining Software Repositories. Saarland University, Saarbrücken, Germany. Available at: <http://thomas-zimmermann.com/publications/files/zimmermann-kcsd-2006.pdf>

Рекомендовано до публікації д-р техн. наук Шамим Б. В.
Дата надходження рукопису 20.02.2015

Шатовская Татьяна Борисовна, кандидат технических наук, доцент, кафедра программной инженерии, Харьковский национальный университет радиоэлектроники, пр. Ленина, 16, г. Харьков, Украина, 61166
E-mail: shatovska@gmail.com

Заремская Анастасия Александровна, кафедра программной инженерии, Харьковский национальный университет радиоэлектроники, пр. Ленина, 16, г. Харьков, Украина, 61166
E-mail: NastenkaZar@yandex.ua

УДК 656.13

DOI: 10.15587/2313-8416.2015.38836

ВПЛИВ КОМФОРТАБЕЛЬНОГО ОЧІКУВАННЯ ПАСАЖИРІВ ПРИМІСЬКИХ АВТОБУСІВ НА ЗМІНУ ЇХ ТРАНСПОРТНОЇ СТОМЛЮВАНОСТІ

© Т. М. Григорова, Ю. О. Давідіч, В. К. Доля

Досліджено закономірності зміни транспортної стомлюваності пасажирів при очікуванні сидячи приміських маршрутних автобусів. Наведені результати обробки натурних досліджень дозволили встановити вплив віку пасажирів і часу очікування на значення його показника активності регуляторних систем. З використанням статистичних методів оцінки складних об'єктів розроблено регресійну модель зміни показника активності регуляторних систем пасажирів при очікуванні сидячи приміських маршрутних автобусів

Ключові слова: транспортне обслуговування, приміське сполучення, транспортна стомлюваність, умови очікування, час очікування

The regularities of changing the transport fatigue of passengers sitting in the waiting for suburban buses are investigated. The results of field research processing allow to establish the influence of age and passenger waiting time on the value of its index activity of regulatory systems. Using statistical methods for assessing complex objects it is developed regression model of activity index change of passenger's regulatory systems sitting waiting for suburban buses

Keywords: transport services, suburban traffic, transport fatigue, waiting conditions, waiting time

1. Вступ

Підвищення якості транспортних послуг можливе тільки за рахунок істотного вдосконалення і модернізації всіх складових транспортної системи. За даними дослідників, внесок транспортної системи у валовий внутрішній продукт становить близько 10 %, що вказує на важливість галузі для подальшого розвитку нашої економіки [1]. В теперішній час, 68,9 % населення України проживає у містах та селищах міського типу [2]. Внаслідок цього, приміські перевезення займають друге місце по масовості (більше 14 % в загальних обсягах) після внутрішньоміських [1, 2]. Концентрація цих перевезень у найбільших містах, їх великий обсяг та безперервне зростання обумовлені швидким розвитком міст, інтенсивною забудовою приміських районів, створенням міст-супутників, організацією у приміських зонах місць відпочинку та спортивно-оздоровчих закладів, підвищенням матеріального добробуту і культурного рівня населення [1].

2. Постановка проблеми

Науковці визначають, що в теперішній час існує загальна потреба в конкретних методиках, що дозволяють кількісно ув'язати попит на перевезення у приміському сполученні з пропозицією транспортних послуг, оцінити вплив на нього як ціно-

вих, так і нецінових факторів. Такі методики необхідні як у теоретичному плані, так і для вирішення практичних завдань управління пасажирським транспортом на рівні транспортних підприємств і державних органів міських і обласних адміністрацій, з метою виявлення ключових факторів, від яких залежить ефективність та якість функціонування транспортної системи [1, 2]. При розвитку приміських транспортних систем недостатньо уваги приділяється вивченню технологій організації перевезень. Методи, моделі та алгоритми організації транспортного обслуговування населення приміських зон, як правило, спираються на розробки, які були виконані ще в минулому столітті, а тому не повністю враховують сучасні особливості його організації. Найбільш суттєві відмінності в організації перевезення пасажирів у приміському сполученні, в порівнянні з іншими видами сполучення, характеризуються закономірностями формування пасажиропотоків та вимогами управління режимами роботи транспортних засобів. Тому, удосконалення системи перевезення пасажирів у приміському сполученні набуває вирішального значення для сільських населених пунктів.

3. Літературний огляд

Для здійснення переміщень у приміському сполученні пасажири постійно проводять вибір між