

УДК 004.415.2.043
DOI: 10.12737/22118

Т.В. Карлова, Н.М. Кузнецова, А.Ю. Бекмешов

ПРИНЯТИЕ УПРАВЛЕНЧЕСКОГО РЕШЕНИЯ С ПОМОЩЬЮ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ ВЫБОРА ОПТИМАЛЬНОГО СОСТАВА НАЗЕМНОГО ОБЩЕСТВЕННОГО ТРАНСПОРТА

Представлена структура автоматизированной системы, позволяющей, найти оптимальный по времени маршрут с учётом работы сети наземного общественного транспорта.

Ключевые слова: автоматизация, навигация, принятие решения, общественный транспорт, приложение для смартфонов.

T.V. Karlova, N.M. Kuznetsova, A.Yu. Bekmeshov

MANAGEMENT DECISION MAKING USING AUTOMATED SYSTEM OF OPTIMUM STRUCTURE CHOICE FOR GROUND-SURFACE PUBLIC CONVEYANCES

The article is devoted to the application of a special software complex for the definition of an optimum set of means of urban public ground-surface conveyances.

In the paper are presented the existing software solutions meant for the definition of an optimum route, their basic merits and demerits are revealed, the methods for the solution of a problem in search of an optimum set of means of ground surface public conveyance taking into account a maximum amount of factors affecting a motion speed are offered.

The authors have presented a structure of an automated system for a choice of an optimum structure in ground surface public conveyances and also there are shown fragments of a corresponding program code and database queries.

In the paper particular attention is paid to the working modes of an appendix presented, in particular, for the situation in the absence of customer's device connection to the basic computer system.

Key words: automation, navigation, decision-making, public conveyances, app to smartphone.

Введение

Современные популярные приложения навигации наземного городского общественного транспорта позволяют определить местоположение и время до прибытия на заранее заданную остановку автобусов, троллейбусов, трамваев, а также маршрутных такси. Данные приложения функционируют как системы мягкого реального времени. Кроме того, они позволяют наблюдать за перемещением общественного транспорта. На рис. 1 представлен вид части окна приложения навигации наземного общественного транспорта [1].

Класс программ построения оптимального маршрута позволяет определить путь, на который будет затрачено минимальное

время. Данное программное обеспечение позволяет проложить маршрут как для случая поездки на автомобиле, так и для случая использования общественного транспорта. Причём построение маршрута для автомобиля производится с учётом дорожно-транспортной ситуации, в то время как для случая использования общественного наземного транспорта пробки, ремонтные работы дорожного полотна и скорость движения самого средства не учитываются. На рис. 2 представлен вид части окна приложения построения оптимального маршрута до пункта назначения [2].

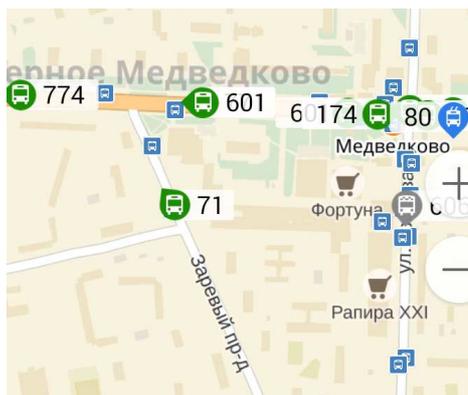


Рис. 1. Вид части окна приложения навигации наземного общественного транспорта

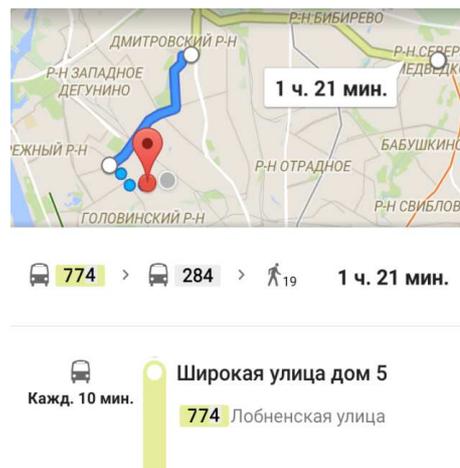


Рис. 2. Вид части окна приложения построения оптимального маршрута до пункта назначения

Статья посвящена оптимизации выбора маршрута до пункта назначения при использовании наземного общественного транспорта с учётом максимального количества факторов, влияющих на скорость движения.

Задача выбора оптимального состава наземного общественного транспорта с целью прибытия в пункт назначения за минимальный промежуток времени с учётом скорости движения

Входными параметрами для автоматизированной системы выбора оптимального состава наземного общественного транспорта являются:

- пункт отправления;
- пункт назначения.

Внутренними параметрами целевого алгоритма принятия решения являются:

- выбранная область наземной транспортной сети (выбор осуществляется исходя из входных параметров);
- количество путей из пункта отправления в пункт назначения (поиск путей осуществляется исходя из входных параметров и выбранной области наземного транспорта);
- количество маршрутов наземного общественного транспорта (МНОТ), имеющих общие дуги с путями из пункта отправления в пункт назначения;
- количество и скорость единиц наземного транспорта (ЕНТ), следующих по МНОТ в данное время.

С помощью перечисленных входных и внутренних параметров определяется

оптимальный маршрут с учётом дорожно-транспортной ситуации.

Для определения оптимального маршрута применяется совокупность последовательно используемых алгоритмов:

- алгоритм выбора области наземной транспортной сети исходя из входных параметров;
- алгоритм нахождения количества путей из пункта отправления в пункт назначения;
- алгоритм нахождения МНОТ, частично или полностью следующих по определенным ранее путям;
- алгоритм определения скорости ЕНТ, следующих по МНОТ в данной время;
- алгоритм нахождения совокупности маршрутов следования (СМС), каждый из которых состоит из последовательности определенных ЕНТ;
- алгоритм расчёта времени для каждого СС (с соответствующей вероятностью).

Входными данными для каждого последующего алгоритма являются выходные предыдущих.

В процессе функционирования перечисленных алгоритмов необходимо использование:

- спутниковых данных;
- информации из базы данных (БД) городской наземной транспортной сети;
- данных о дорожно-транспортной ситуации.

Вся перечисленная информация должна быть актуальной и достоверной.

Таким образом, приложение имеет клиент-серверную архитектуру. Кроме того, оно является распределенным.

На рис. 3 представлена архитектура автоматизированной системы выбора оптимального состава наземного общественного транспорта.



Рис. 3. Архитектура автоматизированной системы выбора оптимального состава наземного общественного транспорта

Как показано на рис. 3, автоматизированная система выбора оптимального состава наземного общественного транспорта получает от смартфона входные данные.

Далее выполняется последовательность алгоритмов, в ходе выполнения большинства из которых производятся обращения к вспомогательным серверам:

- спутниковой навигации;
- контроля запросов к БД городской наземной транспортной сети;

– работы с данными о дорожно-транспортной ситуации.

Чаще всего запросы поступают к серверу БД городской транспортной сети. К серверу спутниковой навигации модули автоматизированной системы обращаются в двух случаях:

- при выборе области наземной транспортной сети;
- при определении скорости ЕНТ.

Обращение к серверу работы с данными о дорожно-транспортной ситуации

происходит только при расчёте времени МС.

Важно отметить, что при работе приложения контроль над процессами синхронизации серверов не происходит, однако корректность работы данных процессов важна при выборе оптимальных составов наземного общественного транспорта.

Пример работы приложения

Рассмотрим работу приложения на конкретном примере.

Допустим, пассажиру необходимо добраться от пункта отправления «Остановка «Широкая ул., д.5» до пункта назначения «Остановка «Платформа Моссельмаш».

В городской наземной транспортной сети не существует МНОТ, который бы позволил пассажиру добраться до пункта назначения без пересадок.

В распоряжении пассажира совокупность МНОТ: автобусы №31, №774, №278, №601, №284, №92, №215, троллейбус №80.

Выходными данными являются:

– совокупность наиболее оптимальных составов наземного общественного транспорта;

– показатель вероятности для каждой совокупности наборов.

При этом в данной ситуации пассажир может сделать пересадку сразу на нескольких остановках. Путь от пункта отправления до пункта назначения представлен на рис. 4 в виде связанного ориентированного графа [3; 4]. Остановки в данном случае являются вершинами графа.

Как показано на рис. 4, пассажир может воспользоваться несколькими МС:

– №31→№284→№215;

– №278→№92;

– №601→№284→№215 и т.д.

Перед автоматизированной системой поставлена задача поиска для пассажира оптимального по времени СМС с учётом скорости движения потока.

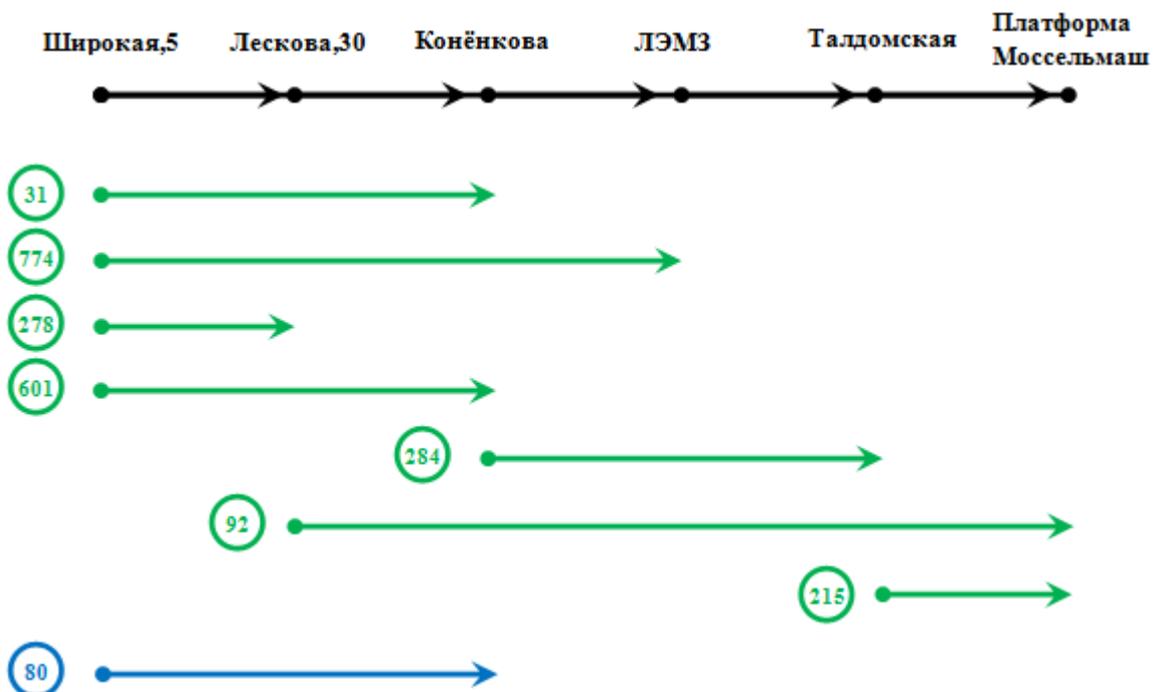


Рис. 4. Путь пассажира от пункта отправления до пункта назначения и совокупность МНОТ

Для решения поставленной задачи необходимо применение нескольких научных дисциплин. Главный математический аппарат составляют:

- теория графов [4];
- комбинаторика [5];

- теория конечных автоматов;
- теория вероятности и математическая статистика;
- теория принятия решений.

Решение задачи нахождения оптимального состава наземного общественного транспорта с учётом скорости в условиях отсутствия постоянной связи со вспомогательными серверами

Возможна ситуация, при которой отсутствует связь с одним или несколькими вспомогательными серверами, в связи с чем информация теряет актуальность. Вероятность ошибки при принятии решения выбора оптимального набора ЕНТ возрастает.

Для последовательности алгоритмов, применяемых в автоматизированной системе выбора оптимального состава

наземного общественного транспорта, недостаточно информации, поэтому в данной ситуации необходимо применение дополнительного инструмента, а именно ведение БД статистики запросов пользователей и дальнейший ее анализ.

На рис. 5 представлена архитектура автоматизированной системы в условиях отсутствия постоянной связи со вспомогательными серверами.



Условные обозначения:

→ данные (входные, выходные);

↔ запрос;

Рис. 5. Схема работы с БД статистики запросов наборов ЕНТ

Как показано на рис. 5, автоматизированная система получает запрос от смартфона пользователя. Запрос обрабатывает алгоритм анализа прежних запросов пользователя, который, в свою очередь, обращается к БД статистики запросов пользователей.

Алгоритм анализа прежних запросов может работать в двух режимах:

- режим поиска путём простой сортировки по времени выборки соответствующих данному пользователю и маршруту (пунктам отправления и назначения) кортежей и выбора первого из них;
- режим использования дополнительного механизма расчёта оптимального

пути с применением метода Монте-Карло [6 - 8].

В таблице 1 представлены поля главной таблицы БД статистики запросов пользователей.

Таблица

Запросы (DEMANDS) БД статистики

Имя	Тип данных	Обозначение
Идентификатор запроса	Счётчик	ID_DEMAND
Пользователь	Текстовое	USER
Пункт отправления	Текстовое	DEPARTURE_POINT
Пункт назначения	Текстовое	DESTINATION_POINT
Набор оптимальных ЕНТ	Текстовое	SET_OPTIMAL_ENT
Время в пути, с	Время	TOTAL_TIME
Соответствующая погрешность	Проценты	ACCORD_FAULT

Ключевым полем таблицы является поле «Идентификатор запроса».

Режим простого поиска предполагает написание запроса выбора кортежей по полям «Пользователь», «Пункт отправления», «Пункт назначения», сортировки по возрастанию по полю «Время в пути», выбора первого кортежа.

На языке SQL [9] данные действия представлены в виде примера программного кода (имя пользователя – «Natalia», пункт отправления – «Shirokaya5», пункт назначения – «Mosselmash»):

```
SELECT *FROM DEMANDS WHERE
USER='Natalia' AND DEPARTURE_POINT='Shirokaya5' AND
DESTINATION_POINT='Mosselmash'
ORDER BY TOTAL_TIME ASC.
```

Режим использования дополнительного механизма расчёта оптимального пути предполагает применение специального метода расчёта выходящих параметров исходя из среднего значения входящих параметров и их соответствующей дисперсии.

Данный режим алгоритма включает использование дополнительных расчётов:

- расчёт среднего значения входящих параметров;
- расчёт соответствующих математического ожидания и дисперсии;
- расчёт выходящего параметра (оптимального набора ЕНТ) и соответствующей погрешности.

Метод Монте-Карло предполагает генерацию множества случайных значений входящих параметров с соответствующими значениями математического ожидания и дисперсии с целью нахождения оптимального значения выходящего параметра (с соответствующим отклонением) из полученного множества результатов генерации [6 - 8].

Важно отметить, что при использовании режима применения метода Монте-Карло при получении результата пользователь должен быть осведомлён о том, что расчёт производился в условиях отсутствия постоянной связи со вспомогательными спутниками.

Процесс формирования БД статистики запросов пользователей

Таблица «Запросы» БД статистики обновляется при каждом новом запросе пользователя тогда, когда связь со вспомогательными серверами не нарушена. При каждом обращении пользователя ответ не только поступает на смартфон, но и записывается в таблицу «Запросы».

Таким образом, чем чаще пользователь обращается к автоматизированной системе, тем точнее будет ответ в условиях отсутствия связи со вспомогательными серверами.

Архитектура приложения в условиях отсутствия постоянной связи со вспомо-

гательными серверами может иметь два вида:

– БД статистики расположена в автоматизированной системе – так, как это представлено на рис. 5;

– БД статистики расположена на смартфоне пользователя.

Первый вариант архитектуры используется в случае применения режима использования метода Монте-Карло, так как информация из БД необходима для соответствующего алгоритма автоматизированной системы. Запросы данных должны проходить локально, без нагрузки на внешнюю телекоммуникационную сеть. Кроме того, данный алгоритм анализа прежних запросов пользователей предполагает наличие больших вычислительных мощностей, которые рационально размещать «рядом» с автоматизированной системой.

Возможно размещение БД статистики и на смартфоне пользователя, при котором не понадобится наличие связи с самой автоматизированной системой. Решение будет принято локально. Для реализации подобной архитектуры требуется наличие

Программно-аппаратная реализация

Приложение имеет архитектуру «тонкого клиента» [10], согласно которой клиентом выступает смартфон пассажира, а сервером – автоматизированная система выбора оптимального состава наземного общественного транспорта.

Автоматизированная система выбора оптимального состава наземного общественного транспорта, в свою очередь, является клиентом сразу для нескольких серверов: спутниковой навигации, контроля запросов к БД городской наземной транспортной сети, работы с данными о дорожно-транспортной ситуации.

Кроме того, особенности применяемых в автоматизированной системе алгоритмов позволяют использовать при решении поставленной задачи технологии параллельных вычислений [11].

Важно отметить, что работа автоматизированной системы должна удовлетворять требованиям режима мягкого реального времени. Другими словами, принятие

определенного объема постоянной памяти на смартфоне для размещения БД статистики. Выбор оптимального набора ЕНТ возможен только в режиме простого поиска.

Запрос к БД будет проще за счёт наличия только одного пользователя смартфона:

```
SELECT *FROM DEMANDS WHERE
DEPARTURE_POINT='Shirokaya5'
AND DESTINATION_POINT='Mosselmash' ORDER
BY TOTAL_TIME ASC.
```

Простота SQL-запроса и отсутствие потерь времени на передачу информации по телекоммуникационной сети повысят скорость принятия решения.

Важно отметить, что в данном случае применение метода Монте-Карло исключено, так как ни один из современных смартфонов не может предоставить необходимый объём вычислительных мощностей для корректного функционирования алгоритма анализа прежних запросов пользователя в режиме использования дополнительного механизма.

решения должно производиться в минимальные сроки.

Все перечисленные факторы обуславливают необходимость применения крупных вычислительных комплексов, обладающих высоким уровнем быстродействия и позволяющих производить обработку больших массивов информации.

С другой стороны, обмен данными с пользователем приложения (пассажиром) предполагает передачу информации небольшого объёма (пункты отправления и назначения и набор МС не повлияют на пропускную способность канала между смартфоном и автоматизированной системой выбора оптимального набора средств наземного общественного транспорта).

На смартфоне не производится никаких вычислений: пользователь только передаёт входные данные для автоматизированной системы и принимает от нее выходные.

Таким образом, дополнительных требований к аппаратному обеспечению смартфона пользователя не предъявляется.

Информационная связь между смартфоном и автоматизированной системой аналогична работе технологии веб-сервиса. Причём данный веб-сервис имеет только один метод:

```
List<List<String>>getSetSMS  
(StringdepPoint, String-  
destPoint).
```

Метод `getSetSMS` возвращает структуру данных `List<List<String>>`, которая соответствует набору МС и включает соответствующую вероятность (последний член внутреннего списка `List<String>`). По-

следовательность членов общего списка `List<List<String>>` важна: данный список должен быть отсортирован по последним членам соответствующего внутреннего списка.

Входными данными для метода выступают `depPoint` пункт отправления и `destPoint` пункт назначения. Переменные метода имеют тип `String`. Важно отметить, что передаются именно названия остановок, а не их координаты. Координаты находятся в процессе работы внутреннего алгоритма автоматизированной системы, а именно алгоритма выбора области наземной транспортной сети.

Заключение

Предложенное в статье приложение, использующее автоматизированную систему выбора оптимального состава наземного общественного транспорта, может функционировать:

- как дополнение к существующим приложениям (в качестве дополнительной опции);

- как отдельное решение.

Особо полезным приложение будет при поиске оптимального пути дальнего следования, когда пассажиру приходится

производить как минимум одну пересадку. В данной ситуации произойдёт уменьшение погрешности метода за счёт большего объема статистических данных (количество ЕНТ, скорости движения ЕНТ и т.д.). Также использование приложения (и соответствующей автоматизированной системы) незаменимо в ситуации, когда пассажир не имеет информации о локальной транспортной сети общественного транспорта.

СПИСОК ЛИТЕРАТУРЫ

1. Яндекс-транспорт. Описание работы приложения . - URL: <http://android-techno.ru/yandex-transport-android>.
2. Google Карты. Описание работы приложения. - URL: <http://android-techno.ru/google-maps-android>.
3. Теория графов. - URL: <http://wikipedia.org>.
4. Харари, Ф. Теория графов / Ф. Харари. – Ленард; Едиториал УРСС, 2015. – 304 с.
5. Шахмейстер, А. Комбинаторика. Статистика. Вероятность / А. Шахмейстер. – Изд-во МЦНМО; КДУ; Петроглиф, 2014. – 296 с.
6. Кузнецова, Н.М. Программа для статистического моделирования в задачах оптимизации технических систем и процессов / Н.М. Кузнецова // Материалы к студенческой научно-практической конференции факультета метрологической информатики «Автоматизация и информационные технологии» (АИТ-2009). Первый тур: сб. тез. докл. – М.: МГТУ «Станкин», 2009. – С. 37-43.
7. Кузнецова, Н.М. Решение обратной задачи методом Монте-Карло / Н.М. Кузнецова // Материалы к студенческой научно-практической конференции «Автоматизация и информационные технологии» (АИТ-2008). Первый тур: сб. тез. докл. – М.: МГТУ «Станкин», 2008. – С.40-42.
8. Кольчужкин, А.М. Метод Монте-Карло в инженерной теории надёжности и проектировании качества / А.М. Кольчужкин, Н.М. Кузнецова // IX научная конференция МГТУ «Станкин» и Учебно-научного центра математического моделирования МГТУ «Станкин» – ИММ РАН. Программа: сб. докл. / под ред. О.А. Казакова. – М.: Янус-К»: МГТУ «Станкин», 2006. – С. 26-29.
9. Грофф, Джеймс Р. Опель SQL – Полное руководство / Р. Джеймс Грофф, Вайнберг Пол Н., Дж. Эндрю. –М.: Вильямс, 2014. – 966 с.

10. Chappell, David. Java Web Services First Edition/ D. Chappell, T. Jewell // O'Reilly. – 2002. – 276 pages.
1. Yandex-transport. Description of app function. - URL: <http://android-techno.ru/yandex-transport-android>.
2. Google maps. Description of app function. - URL: <http://android-techno.ru/google-maps-android>.
3. Graph Theory. - URL: <http://wikipedia.org>.
4. Kharari, F. Graph theory / F. Kharari. – Lenard; Editorial UPCC, 2015. – pp.304.
5. Shahmeister, A. Theory of Combinations. Statistics. Likelihood / A. Shahmeister. – Publishing House MCSMO; KDU; Petroglif, 2014. – pp. 296.
6. Kuznetsova, N.M. Software for statistic modeling in problems of engineering systems and processes optimization / N.M. Kuznetsova // *Proceedings to the Students Scientific – Pract. Conf. of the Faculty of metrological Informatics “Automation & Information Technologies” (AIT-2009). The first round: the collection of abstracts.* – М.: MSTU “Stankin”, 2009. – pp. 37-43.
7. Kuznetsova, N.M. Solution of inverse problem through Monte-Carlo method / N.M. Kuznetsova //
11. Антонов, А.С. Технологии параллельного программирования MPI и OpenMP / А.С. Антонов // Изд-во МГУ, 2012. – 344 с.
- Proceedings to the Students Scientific-Pract. Conf. “Automation & Information Technologies” (AIT-2008). The first round: the collection of abstracts. – М.: MSTU “Stankin”, 2008. – pp. 40-42.
8. Kolchuzhkin, A.M. Monte-Carlo method in engineering theory of reliability and quality design / A.M. Kolchuzhkin, N.M. Kuznetsova // *the IX-th Scientific Conf. of MSTU “Stankin” and Training-Scientific center of mathematical modeling of MSTU “Stankin” – IMM RAS. Program: Proceedings / under the editorship of O.A. Kazakov.* – М.: “Yanus-K”: MSTU “Stankin”, 2006. – pp. 26-29.
9. Groff, James R. Opel SQL – *Complete Guide* / R. James Groff, Weinberg Pol N., J. Andrey. –М.: Williams, 2014. – pp. 966.
10. Chappell, David. Java Web Services First Edition/ D. Chappell, T. Jewell // O'Reilly. – 2002. – 276 pages.
11. Antonov, A.S. *Techniques for Parallel Programming MPI and OpenMP* / A.S. Антонов // Publishing house of MSU, 2012. – pp. 344.

Статья поступила в редколлегию 11.07.2016.

Рецензент: д.т.н., профессор
Куликов М.Ю.

Сведения об авторах:

Карлова Татьяна Владимировна, д. соц. н., профессор Института конструкторско-технологической информатики РАН, тел.: 8-(499)-978-99-62, 8-(903)-776-90-78, e-mail: karlova-t@yandex.ru.

Кузнецова Наталия Михайловна, к.т.н., преподаватель Московского государственного технологического университета «СТАНКИН», тел.: 8 (499)-972-94-37, 8-(903)-581-80-15, e-mail: knm87@mail.ru.

Бекмешов Александр Юрьевич, к.т.н., Института конструкторско-технологической информатики РАН, тел.: +7 (499) 978-99-62, e-mail: s_bekmeshov@mail.ru.

Karlova Tatiana Vladimirovna, D.Soc., Prof. of Institute of Design-Technological Informatics of RAS, Phone: 8-(499)-978-99-62, 8-(903)-776-90-78, e-mail: karlova-t@yandex.ru.

Kuznetsova Natalia Mikhailovna, Can.Eng., Lecturer of Moscow State Technological University “Stankin”, Phone: 8 (499)- 972-94-37, 8-(903)-581-80-15, e-mail: knm87@mail.ru.

Bekmeshov Alexander Yurievich, Can.Eng., Institute of Design-Technological Informatics of RAS, Phone: +7 (499) 978-99-62, e-mail: s_bekmeshov@mail.ru.