

23. *Kureychik V.M., Kulakov A.A.* Podsystema planirovaniya sverkhbol'shikh integral'nykh skhem trekhmernoй integratsii s uchetom vzaimnogo teplovogo vliyaniya elementov [The scheduling engine ultralarge integrated circuits three-dimensional integration taking into account the mutual thermal influence of elements], *Programmnye produkty i sistemy* [Software Products and Systems], 2014, No. 1, pp. 131-136.
24. *Kureychik V.M., Barinov S.V., Gladkov L.A.* Geneticheskie algoritmy razbieniya skhem s uchetom vremennykh zaderzhek [Genetic algorithms partitioning schemes based on temporary delays]. Taganrog: Izd-vo YuFU, 2014, 350 p.

Статью рекомендовал к опубликованию д.т.н., профессор Б.К. Лебедев.

**Чернышев Юрий Олегович** – Донской государственный технический университет; e-mail: myvnn@list.ru; 344000, г. Ростов-на-Дону, пл. Гагарина, 1; тел.: 88632738582; кафедра автоматизации производственных процессов; профессор.

**Венцов Николай Николаевич** – e-mail: vencov@list.ru; кафедра информационных технологий; доцент.

**Chernyshev Yuriy Olegovich** – Don State Technical University; e-mail: myvnn@list.ru; 1, Gagarin street, Rostov-on-Don, 344000, Russia; phone: +78632738582; the department of automation of productions; professor.

**Ventsov Nikolay Nikolaevich** – e-mail: vencov@list.ru; the department of information technologies; associate professor.

УДК 658.512.2.011.5

**Л.А. Гладков, Н.В. Гладкова**

### **ГИБРИДНЫЙ АЛГОРИТМ РЕШЕНИЯ ДИНАМИЧЕСКИХ ТРАНСПОРТНЫХ ЗАДАЧ\***

*Рассматривается новый подход к решению динамических транспортных задач. Показана актуальность и важность решения подобного рода задач для повышения эффективности перевозок и развития транспортной инфраструктуры. Отмечено, что особый интерес представляют некоторые классы задач маршрутизации автотранспорта, в частности транспортная задача с ограничением по времени. Выделены особенности статической и динамической транспортных задач. Приведено графическое представление динамической транспортной задачи. Выполнена математическая постановка задачи маршрутизации автотранспорта. Определена функция оценки качества получаемых решений. Разработана методика кодирования решений для использования их в генетическом алгоритме. Предложены новые модификации операторов кроссинговера и мутации, направленные на повышение разнообразия текущей популяции и преодоление локальных оптимумов. Описана последовательность выполнения операций и приведена структура разработанного алгоритма. Описана структура и принципы работы менеджера событий, который позволяет повысить эффективность работы алгоритма, за счет организации обработки вновь поступающих заявок. На основе проведенного анализа установлено, что для эффективного подобных задач необходима разработка новых методов, позволяющих осуществлять динамическое изменение параметров алгоритма и, при необходимости видоизменять структуру алгоритма. Приведен пример структуры вектора управляющего воздействия на параметры генетического алгоритма со стороны нечеткого логического контроллера. Также описана структура вектора входных параметров для нечеткого логического контроллера. Проведены серии вычислительных экспериментов для анализа и сравнения каче-*

\* Работа выполнена при финансовой поддержке РФФИ (проект № 14-07-00829).

ства получаемых решений, а также определения лучших значений управляющих параметров алгоритма. На основании анализа сделаны выводы о достоинствах и недостатках предложенного алгоритма.

*Задачи маршрутизации автотранспорта; динамическая транспортная задача с ограничением по времени; эволюционные вычисления; гибридные интеллектуальные методы.*

**L.A. Gladkov, N.V. Gladkova**

### **HYBRID ALGORITHM FOR SOLVING VEHICLE ROUTING PROBLEMS WITH A TIME WINDOWS**

*The article discusses a new approach to the dynamic transportation problems. The urgency and importance of addressing such problems to increase the efficiency of traffic and transport infrastructure. It is noted that of particular interest are certain classes of vehicle routing problems, in particular transport task with a time limit. The features of static and dynamic transport problems. The graphic representation of the dynamic transportation problem. A mathematical formulation of the vehicle routing problem. We define the function evaluation of the quality of the solutions obtained. The method of coding solutions for use in the genetic algorithm. Proposed new modifications crossover and mutation operators, aimed at increasing the diversity of the current population and overcome local optima. Described sequence of operations, and shows the structure of the developed algorithm. The structure and principles of the event manager, which allows you to improve the efficiency of the algorithm, through the organization of the processing of incoming orders again. Based on the analysis found that the effectiveness of these tasks necessary to develop new methods that enable the dynamic change of algorithm parameters, and modify the structure of the algorithm if necessary. An example of the structure of the vector control action on the parameters of the genetic algorithm by the fuzzy logic controller. the structure of the vector of the input parameters for the fuzzy logic controller is also described. We conducted a series of numerical experiments to analyze and compare the quality of the solutions, as well as to determine the best values of the control algorithm parameters. Conclusions based on the analysis of the advantages and disadvantages of the proposed algorithm*

*Vehicle routing problems; dynamic vehicle routing problem with time windows; evolutionary calculations; hybrid intellectual methods.*

**Введение.** Перевозка людей и грузов – это важная составляющая современной экономики. По статистике на долю транспортных перевозок в экономически развитых странах приходится порядка 10–15 % национальных расходов. Сокращение издержек на перевозку грузов дает огромный экономический эффект. Поэтому разработка новых алгоритмов, позволяющих повысить эффективность решения транспортных задач является актуальной и востребованной задачей.

Точные методы решения, позволяют решать задачи только с небольшим количеством клиентов (не более 50-ти). Для решения практических задач большой размерности отражающих существующие экономические реалии необходимо разрабатывать новые подходы, базирующиеся на использовании новейших достижений математики, компьютерных новых, информационных технологий. Одним из наиболее перспективных современных научных трендов является использование интегрированных подходов, лежащих на стыке различных наук. В частности, для решения сложных многокритериальных задач оптимизации с успехом используют различные методы вычислительного интеллекта, в том числе эволюционные методы поиска, нечеткие математические модели, методы обучения на основе нейросетевых моделей и т.д.

С практической точки зрения одной из наиболее интересных классов транспортных задач (или задач маршрутизации автотранспорта) являются задачи с ограничением по времени. Транспортные задачи подразделяются на статические и динамические. В статической транспортной задаче все параметры заданы изначально и не изменяются во время ее решения. В динамической транспортной зада-

че условия могут изменяться в процессе решения. Это свойство приобретает дополнительное значение в связи с развитием средств мобильной связи и навигации, дающих возможность отслеживать маршрут автомобиля и передавать указания в процессе движения. Поэтому задача разработки алгоритмов решения задач с ограничением по времени и с учетом динамики изменения запросов клиентов, является актуальной.

**Постановка задачи.** В нашем случае, под динамической транспортной задачей будем подразумевать задачу с возможностью изменения положения клиентов в процессе движения транспорта по маршруту [1–3]. Данную динамическую задачу можно представить как серию последовательных статических подзадач. При этом количество обслуживаемых запросов от клиентов задается заранее, также предусмотрена возможность появления запросов по ходу решения задачи. Каждая подзадача обладает своими уникальными начальными условиями и количеством клиентов, требующих обслуживания. Начальным условием является положение транспортного средства во время получения динамического запроса. Рассматриваемая динамическая транспортная задача может быть представлена в виде графа (рис. 1).

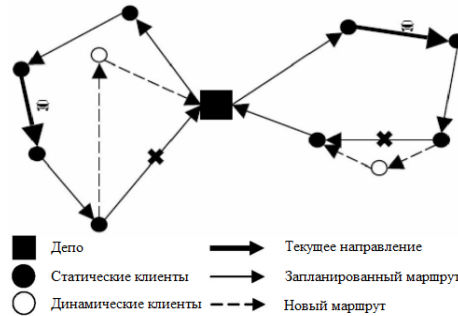


Рис. 1. Графическое представление динамической транспортной задачи

Задача маршрутизации автотранспорта с ограничением по времени (VRPTW) может быть представлена в виде графа  $G = (X, U)$ ,  $X = n + 1$ , где:  $n$  – множество вершин, равное числу клиентов (вершины  $1, 2, \dots, n$ ), а также исходное депо, откуда начинают и где заканчивают свой маршрут все автомобили (вершина  $n + 1$ );  $U$  – дуги, соединяющие вершины графа [4]. Основные цели: минимизировать общее количество задействованных транспортных средств и общее расстояние, покрытое всеми транспортными средствами [5, 6].

Тогда целевая функция рассматриваемой транспортной задачи может быть записана в следующем виде

$$F = \sum_{k \in V} \sum_{(i, j) \in U} c_{ij} x_{ij}^k \rightarrow \min$$

при соблюдении всех заданных ограничений [6]. В данном выражении используются следующие обозначения:

$c_{ij}$  – стоимость перевозки груза по маршруту  $(i, j)$ ;

$V$  – число автомобилей одного типа (грузоподъемности  $q$ ),  $k \in V$ ;

$x_{ij}^k$  – переменная, принимающая значения  $\{0, 1\}$  и характеризующая направление движения автомобиля:  $x_{ij}^k = 1$  – если автомобиль движется от клиента  $i$  к клиенту  $j$ ;  $x_{ij}^k = 0$  – если автомобиль движется в обратном направлении.

**Описание алгоритма.** Кодирование решений в разработанном гибридном алгоритме выполняется следующим образом. Предполагается, что количество транспортных средств совпадает с количеством маршрутов, тогда решение можно

представить в виде последовательности клиентов, которых поочередно обслуживает рассматриваемое транспортное средство. Маршрут должен начинаться и заканчиваться в депо (клиент 0).

В процессе формирования начальной популяции используется оператор инициализации. Все клиенты выбираются на основе вероятностных правил, что обеспечивает необходимое разнообразие решений в популяции [7–10].

В общем виде процедура формирования начальной популяции может быть описана следующим образом.

$P_1$

|          |   |    |   |   |   |
|----------|---|----|---|---|---|
| $O_{11}$ | 2 | 5  | 3 | 9 | 0 |
| $O_{12}$ | 7 | 10 | 1 | 0 |   |
| $O_{13}$ | 4 | 8  | 6 | 0 |   |

Рис. 2. Пример кодирования маршрута для 3 транспортных средств и 10 клиентов

1. Если все клиенты обслужены, то переход к пункту 5, в противном случае выбираем клиента  $k^*$  из числа еще не обслуженных.

2. Перебираем клиентов в уже существующем решении, чтобы вставить клиента  $k^*$  в текущий маршрут данного решения. При возможности такой вставки выбираем вставку с минимальным добавлением расстояния. При существовании двух вариантов вставки с одинаковым добавлением расстояния, выбираем вариант с меньшим простоем транспорта.

3. Если невозможно вставить нового клиента в уже существующий маршрут, то создаем новый маршрут, который будет включать клиента  $k^*$ .

4. Переходим к пункту 1.

5. Маршрут построен.

Оператор инициализации позволяет не только строить, но и дорабатывать решения, в которых доля клиентов уже обслужена.

В качестве операторов отбора используется комбинация стандартных вариантов: пропорциональный, равновероятный и элитный отбор.

В алгоритме используется модифицированный оператор кроссинговера. Поясним его работу на примере. Родители  $P_1$  и  $P_2$  выбираются из популяции  $P$ . В каждом родителе случайным образом выбирается 2 гена (маршрута) и удаляются из генома родителей. Оставшиеся гены образуют новое решение. Затем удаленные гены вставляются в это решение таким образом, чтобы минимизировать ЦФ. Если местоположение вставки не будет найдено, создается новый маршрут.

Например, из хромосом  $P_1$  и  $P_2$  случайным образом выбираются маршруты 3 и 1 соответственно (рис. 3).

|   |          |    |    |   |   |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
|---|----------|----|----|---|---|---|----------|---|----|---|---|--|----------|---|---|---|---|--|---|----------|---|---|---|---|--|----------|---|---|---|---|---|----------|---|---|----|---|--|
| $P_1$   | $P_2$    |    |    |   |   |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
| <table border="1" style="border-collapse: collapse;"> <tr><td><math>O_{11}</math></td><td>2</td><td>5</td><td>3</td><td>9</td><td>0</td></tr> <tr><td><math>O_{12}</math></td><td>7</td><td>10</td><td>1</td><td>0</td><td></td></tr> <tr><td><math>O_{13}</math></td><td>4</td><td>8</td><td>6</td><td>0</td><td></td></tr> </table> | $O_{11}$ | 2  | 5  | 3 | 9 | 0 | $O_{12}$ | 7 | 10 | 1 | 0 |  | $O_{13}$ | 4 | 8 | 6 | 0 |  | <table border="1" style="border-collapse: collapse;"> <tr><td><math>O_{11}</math></td><td>9</td><td>7</td><td>8</td><td>0</td><td></td></tr> <tr><td><math>O_{12}</math></td><td>6</td><td>3</td><td>2</td><td>5</td><td>0</td></tr> <tr><td><math>O_{13}</math></td><td>1</td><td>4</td><td>10</td><td>0</td><td></td></tr> </table> | $O_{11}$ | 9 | 7 | 8 | 0 |  | $O_{12}$ | 6 | 3 | 2 | 5 | 0 | $O_{13}$ | 1 | 4 | 10 | 0 |  |
| $O_{11}$  | 2        | 5  | 3  | 9 | 0 |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
| $O_{12}$  | 7        | 10 | 1  | 0 |   |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
| $O_{13}$  | 4        | 8  | 6  | 0 |   |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
| $O_{11}$  | 9        | 7  | 8  | 0 |   |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
| $O_{12}$  | 6        | 3  | 2  | 5 | 0 |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |
| $O_{13}$  | 1        | 4  | 10 | 0 |   |   |          |   |    |   |   |  |          |   |   |   |   |  |   |          |   |   |   |   |  |          |   |   |   |   |   |          |   |   |    |   |  |

Рис. 3. Родительские хромосомы  $P_1$  и  $P_2$

Затем также случайным образом в этих маршрутах выбираются по 2 разрезающие точки (рис. 4).



Рис. 4. Выбор точек разреза

Клиенты, находящиеся между выбранными точками удаляются из всех маршрутов. В данном примере это клиенты 4, 8 и 7.

|                 |    |   |                 |                 |    |                 |   |   |   |
|-----------------|----|---|-----------------|-----------------|----|-----------------|---|---|---|
| O <sub>11</sub> | 2  | 5 | 3               | 9               | 0  | O <sub>11</sub> | 9 | 0 |   |
| O <sub>12</sub> | 10 | 1 | 0               | O <sub>12</sub> | 6  | 3               | 2 | 5 | 0 |
| O <sub>13</sub> | 6  | 0 | O <sub>13</sub> | 1               | 10 | 0               |   |   |   |

Рис. 5. Родители P<sub>1</sub> и P<sub>2</sub> после удаления клиентов

Затем удаленные клиенты вставляются в позиции, для которых целевая функция принимает минимальное значение (рис. 6).

|                 |    |   |   |   |                 |                 |   |    |   |   |  |
|-----------------|----|---|---|---|-----------------|-----------------|---|----|---|---|--|
| C <sub>1</sub>  |    |   |   |   |                 | C <sub>2</sub>  |   |    |   |   |  |
| O <sub>11</sub> | 2  | 5 | 3 | 9 | 0               | O <sub>11</sub> | 9 | 7  | 4 | 0 |  |
| O <sub>12</sub> | 10 | 8 | 1 | 0 | O <sub>12</sub> | 6               | 3 | 2  | 5 | 0 |  |
| O <sub>13</sub> | 6  | 7 | 4 | 0 | O <sub>13</sub> | 1               | 8 | 10 | 0 |   |  |

Рис. 6. Решение, полученное после применения кроссинговера

Результат выполнения оператора кроссинговера показан на рис. 7.

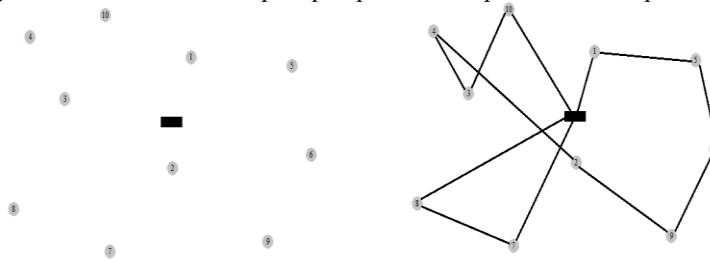


Рис. 7. Начальное расположение и результат выполнения оператора кроссинговера

В алгоритме также предложен модифицированный оператор мутации. В данном случае используется мутация обмена, когда случайным образом выбираются две позиции и меняются местами (рис. 8).

|                 |   |   |    |   |   |                 |                 |   |    |   |   |   |
|-----------------|---|---|----|---|---|-----------------|-----------------|---|----|---|---|---|
| O <sub>11</sub> | 3 | 2 | 10 | 0 | → | O <sub>11</sub> | 3               | 2 | 10 | 0 |   |   |
| O <sub>12</sub> | 4 | 8 | 6  | 5 |   | 0               | O <sub>12</sub> | 6 | 8  | 4 | 5 | 0 |
| O <sub>13</sub> | 9 | 7 | 1  | 0 |   | O <sub>13</sub> | 9               | 7 | 1  | 0 |   |   |

Рис. 8. Пример работы оператора мутации

В общем виде последовательность выполнения операций можно описать следующим образом:

1. Задание параметров алгоритма.
2. Формирование начальной популяции решений размера N. Применение оператора инициализации N раз.
3. Расчет значения ЦФ для каждого решения в популяции.
4. Сортировка решений в порядке возрастания ЦФ.
5. Присваиваем переменной t значение 0.
6. Применение оператора селекции для выбора решения.
7. Выполнение генетических операторов
  - 7.1 Выполнение оператора кроссинговера с вероятностью P<sub>k</sub>.

7.2 Выполнение оператора мутации (ОМ) к решениям, полученным после применения кроссинговера с вероятностью  $P_m$ .

8. Перерасчет значений ЦФ для каждого решения.

9. Определяем вид оператора отбора.

10. Выполнение оператора редукции.

11. Если прошло 20 итераций, то изменяем координаты одной вершины, иначе переход к следующему пункту.

12. Если  $t$  меньше  $T$ , то заменяем исходную популяцию новой, увеличиваем значение переменной  $t = t + 1$ , возврат к п. 5; в противном случае, переход к п. 12.

12. Если показатели эффективности эволюции неудовлетворительные и требуется регулировка параметров, то выполняется динамическая подстройка параметров с использованием НЛК [11–17].

13. Если пройдено заданное число поколений или достигнут другой критерий останова, то переход к п. 14, в противном случае, возврат к п. 2.

14. Завершение работы алгоритма.

Обобщенная схема алгоритма представлена на рис. 9.

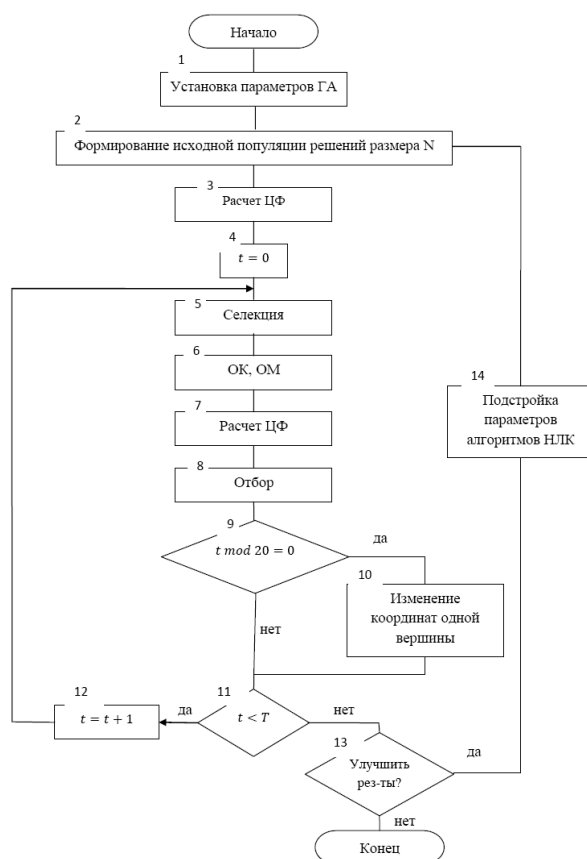


Рис. 9. Схема работы алгоритма

Для повышения эффективности обработки динамических запросов клиентов, в разработанный гибридный алгоритм был добавлен так называемый менеджер событий. Цель использования менеджера событий состоит в том, чтобы связать

воедино временные интервалы, отведенные на решение каждой статической задачи, и обеспечить последовательное обслуживание новых заявок сразу после их поступления.

Задача заключается в разделении рабочего дня на  $n_{ts}$  (*time slices*) отрезков времени, каждый из которых равен величине  $\frac{T}{n_{ts}}$ , где  $T$  – длина рабочего дня.

В процессе принятия решения мы откладываем принятие новых заявок до конца этого отрезка времени. Благодаря этому у нас есть возможность оптимизировать новые входные заказы за дискретное число шагов (рис. 10).

Каждый отрезок времени представляет собой статическую задачу, в которой транспортное средство должно обслужить всех известных клиентов. Менеджер событий выполняется в последовательности решения алгоритма на этом отрезке времени. Решение об обслуживании новой заявки может быть принято в рамках расширенного времени обслуживания –  $t_{ac}$  (*advanced commitment time*), что позволяет водителю реагировать на новые заказы еще до его обработки.



Рис. 10. Разделение рабочего дня

Первая статическая задача, назначенная на первый отрезок времени, состоит из всех заказов, оставшихся от предыдущего рабочего дня. Время отключения  $t_{co}$  (*time cut-off*) – параметр, управляющий временем, в течение которого могут быть получены новые заказы, но при этом останутся не обслуженные клиенты. Все заказы, полученные после времени  $t_{co}$ , интерпретируются как клиенты, которые не были обслужены за текущий день и переносятся на следующий рабочий день.

Следующая статическая задача рассматривает все заказы, полученные в течение следующего отрезка времени, а также еще не обслуженные водителями заказы. В нашем представлении, каждое транспортное средство  $k$  начинает свой маршрут от места последнего обслуженного им клиента. Время отсчета начинается с момента завершения обслуживания предыдущего клиента. А объем груза определяется объемом, освободившимся в результате обслуживания предыдущих  $k$  клиентов.

В конце каждого отрезка времени, выбираются наилучшее решение и заказы

со временем обработки  $\frac{T}{n_{ts}} + t_{ac}$ , обслуживаются соответствующими транспортными средствами. Транспортное средство, которое доставило весь свой груз, возвращается в депо.

Задача нечеткого логического контроллера в данном алгоритме заключается в том, чтобы, используя получаемые значения оценочных величин, соответствующим образом динамически изменять параметры генетического алгоритма, а также при необходимости принимать решение о выполнении дополнительных генетических операторов для того, чтобы избежать преждевременной сходимости [18].

Вектор управления, оказывающий воздействие на динамику эволюции решений, описывается следующим образом:

$$U^t = (\Delta p_c^t, \Delta p_m^t, \delta^t),$$

где  $\Delta p_c^t$  и  $\Delta p_m^t$  – изменение значений вероятности выполнения операторов кроссинговера и мутации, а  $\delta^t$  – параметр определяющий момент переключения эволюции с одной фазы на другую [19, 20].

В состав параметров поступающих на вход НЛК входят оценка эффективности эволюции на предыдущем шаге  $\gamma$ , текущее время эволюции  $t$ , промежуток времени  $\Delta t$  в течении которого не наблюдалось улучшения значений целевой функции:  $X_t = (\gamma t - 1, t, \Delta t)$ .

**Вычислительные эксперименты.** Для реализации предложенного гибридного алгоритма решения динамической транспортной задачи была разработана программа на языке высокого уровня C++. Программа предназначена для работы с операционными системами Windows XP, 7, 8.

Для определения эффективности предложенного алгоритма проводились вычислительные эксперименты, в ходе которых выполнялось сравнение результатов, показываемых данным алгоритмом с результатами известных тестовых задач (бенчмарков).

Задача первого этапа тестирования состояла в подборе параметров для начальных задач. Для этого программа запускалась с различными вариантами начальных значений. Результаты исследования зависимости числа поколений от вероятностей операторов кроссинговера и мутации, представлены в табл. 1.

Таблица 1

**Определение оптимальных значений ОМ и ОК**

| Вероятность ОК | Вероятность ОМ |      |      |      |
|----------------|----------------|------|------|------|
|                | 20 %           | 40 % | 60 % | 80 % |
| 20 %           | 4366           | 4287 | 4210 | 4315 |
| 40 %           | 4241           | 4326 | 4163 | 4235 |
| 60 %           | 4349           | 4301 | 4256 | 4154 |
| 80 %           | 4341           | 4235 | 4215 | 4193 |

Также проводились исследования зависимости времени работы алгоритма от числа итераций и размера популяции (табл. 2).

Таблица 2

**Зависимость времени работы алгоритма от числа итераций**

| Количество итераций | Размер популяции | Время работы алгоритма |
|---------------------|------------------|------------------------|
| 10                  | 50               | 0.006                  |
| 10                  | 100              | 0.01                   |
| 200                 | 200              | 0.514                  |
| 300                 | 300              | 1.86                   |
| 300                 | 400              | 2.83                   |
| 300                 | 2000             | 15.63                  |
| 300                 | 6000             | 54.01                  |
| 300                 | 8000             | 80.91                  |

Исходя из приведенной таблицы, можно сказать, что лучшие результаты достигаются при 300 итерациях и популяции размером от 300 до 400 решений.

Поскольку исследуемая задача является динамической, то в моменты изменения внешних условий на графике ЦФ появляются изгибы (рис. 11).



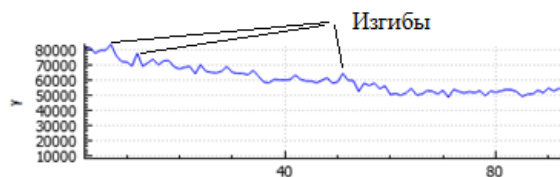


Рис. 11. График ЦФ

Это связано с тем, что при поступлении новых заявок, ЦФ существующих решений ухудшаются, т.к. прежний маршрут не учитывал новых клиентов. Через некоторое время после этого алгоритм адаптируется к изменившимся условиям и средняя ЦФ снова уменьшается.

Для оценки эффективности разработанного гибридного алгоритма, был проведен сравнительный анализ результатов, показанный с использованием НЛК и без него (рис. 12)

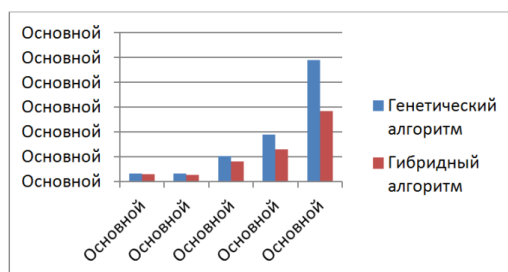


Рис. 12. Сравнение эффективности алгоритмов

Также была проведена оценка временной сложности алгоритма и построена зависимость времени работы алгоритма от числа вершин (рис. 13).

**Выводы.** Анализ полученных результатов известных алгоритмов позволяет сделать вывод о том, что оптимальные вероятности выполнения операторов кроссинговера и мутации 0.8 и 0.6, соответственно. Размер популяции 300–400 решений. Число итераций – 300. Как видно из графиков на работу гибридного алгоритма оказывает существенное процесс формирования начальной популяции.

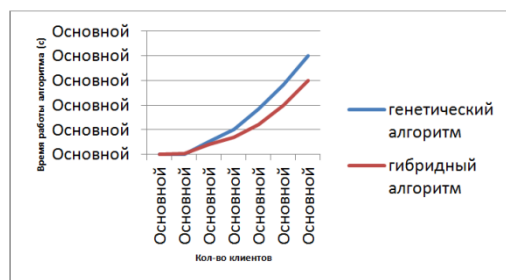


Рис. 13. График зависимости времени работы от числа вершин

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Кажаров А.А., Курейчик В.М. Классификация и критерии оптимизации задачи маршрутизации автотранспорта // Сборник трудов VII Международной научно-практической конференции "Интегрированные модели и мягкие вычисления в искусственном интеллекте". Т. 2. – М.: Физматлит, 2013. – С. 879-886.

2. *Емельянова Т.С.* Эвристические и метаэвристические методы решения динамической транспортной задачи // Перспективные информационные технологии и интеллектуальные системы. – 2007. – № 3 (31). – С. 33-43.
3. *Емельянова Т.С.* Анализ методов решения нелинейных транспортных задач. // Перспективные информационные технологии и интеллектуальные системы. – 2007. – № 1 (29). – С. 38-49.
4. *Гладков Л.А., Гладкова Н.В.* Решение динамических транспортных задач на основе гибридных интеллектуальных методов и моделей // Известия ЮФУ. Технические науки. – 2013. – № 7 (144). – С. 102-107.
5. *Гладков Л.А., Гладкова Н.В.* Особенности и новые подходы к решению динамических транспортных задач с ограничением по времени // Известия ЮФУ. Технические науки. – 2014. – № 7 (156). – С. 178-187.
6. *Гладков Л.А., Гладкова Н.В.* Гибридный алгоритм решения транспортных задач с ограничением по времени // Известия ЮФУ. Технические науки. – 2015. – № 6 (167). – С. 180-191.
7. *Емельянова Т.С.* Генетический алгоритм решения транспортной задачи с ограничением по времени // Перспективные информационные технологии и интеллектуальные системы. – 2007. – № 4 (32). – С. 43-59.
8. *Курейчик В.М., Емельянова Т.С.* Решение транспортных задач с использованием комбинированного генетического алгоритма // Одиннадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2008: Труды конференции. Т. 1. – М.: Физматлит, 2008. – С. 158-164.
9. *Гладков Л.А., Курейчик В.М., Курейчик В.В., Сороколетов П.В.* Генетические алгоритмы. – М.: Физматлит, 2010. – 365 с.
10. *Гладков Л.А., Курейчик В.М., Курейчик В.В., Сороколетов П.В.* Биоинспирированные методы в оптимизации. – М.: Физматлит, 2009. – 384 с.
11. *Herrera F., Lozano M.* Fuzzy Genetic Algorithms: Issues and Models. Technical Report DECSAI-98116, Department of Computer Science and A.I., University of Granada, 1999. – 25 p.
12. *Herrera F., Lozano M.* Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions // Soft Computing 7 (2003), Springer-Verlag, 2003. – P. 545-562.
13. *Lee M.A., Takagi H.* Dynamic Control of Genetic Algorithms using Fuzzy Logic Techniques // Proceeding of 5th International Conference on Genetic Algorithms (ICGA'93), Urbana-Champaign, IL, July 17-21, 1993. – P. 76-83.
14. *Батыршин И.З., Недосекин А.О. и др.* Нечеткие гибридные системы. Теория и практика / под ред. Н.Г. Ярушкиной. – М.: Физматлит, 2007. – 208 с.
15. *Ярушкина Н.Г.* Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.
16. *Hongbo Liu, Zhanguo Xu, Ajith Abraham* Hybrid Fuzzy-Genetic Algorithm Approach for Crew Grouping // Proceedings of the Fifth International Conference on Intelligent Systems Design and Applications (ISDA 2005), 8-10 September 2005, Wroclaw, Poland. IEEE Computer Society 2005. – P. 332-337.
17. *Гладков Л.А., Гладкова Н.В.* Особенности использования нечетких генетических алгоритмов для решения задач оптимизации и управления // Известия ЮФУ. Технические науки. – 2009. – № 4 (93). – С. 130-136.
18. *Гладков Л.А., Курейчик В.В., Курейчик В.М., Родзин С.И.* Основы теории эволюционных вычислений: монография. – Ростов-на-Дону: Изд-во ЮФУ, 2010.
19. *Gladkov L.A., Gladkova N.V., Leiba S.N.* Manufacturing Scheduling Problem Based on Fuzzy Genetic Algorithm // Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2014). Kiev, Ukraine, September 26 – 29, 2014. – P. 209-213.
20. *Gladkov L.A., Leiba S.N., Gladkova N.V., Legebokov A.A.* Parallel Genetic Algorithm Based on Fuzzy Logic Controller for Design Problems // Advances in Intelligent Systems and Computing. Proceedings of the 5th Computer Science On-line Conference 2016 (CSOC 2016), Vol. 1: Artificial Intelligence Perspectives in Intelligent Systems. – Springer International Publishing, Switzerland, 2016. – P. 147-157.

## REFERENCES

1. Kazharov A.A., Kureychik V.M. Klassifikatsiya i kriterii optimizatsii zadachi marshrutizatsii avtotransporta [Classification and criteria optimization problem of routing vehicles], *Sbornik trudov VII Mezhdunarodnoy nauchno-prakticheskoy konferentsii "Integrirovannye modeli i myagkie vychisleniya v iskusstvennom intellekte"* Proceedings of the VII International scientific-practical conference "Integrated models and soft computing in artificial intelligence". Vol. 2. Moscow: Fizmatlit, 2013, pp. 879-886.
2. Emel'yanova T.S. Evristicheskie i metaevristicheskie metody resheniya dinamicheskoy transportnoy zadachi [Heuristic and metaheuristic methods for solving the dynamic transportation problem], *Perspektivnye informatsionnye tekhnologii i intellektual'nye sistemy* [Advanced Information Technologies and Intelligent Systems], 2007, No. 3 (31), pp. 33-43.
3. Emel'yanova T.S. Analiz metodov resheniya nelineynykh transportnykh zadach [Analysis of methods for solving nonlinear transport problems], *Perspektivnye informatsionnye tekhnologii i intellektual'nye sistemy* [Advanced Information Technologies and Intelligent Systems], 2007, No. 1 (29), pp. 38-49.
4. Gladkov L.A., Gladkova N.V. Reshenie dinamicheskikh transportnykh zadach na osnove gibridnykh intellektual'nykh metodov i modeley [The decision of dynamic vehicle routing problems on the basis of hybrid intellectual methods and models], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2013, No. 7 (144), pp. 102-107.
5. Gladkov L.A., Gladkova N.V. Osobennosti i novye podkhody k resheniyu dinamicheskikh transportnykh zadach s ogranicheniem po vremeni [Features and new approaches to the decision of dynamic vehicle routing problems with time windows], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2014, No. 7 (156), pp. 178-187.
6. Gladkov L.A., Gladkova N.V. Gibridnyy algoritm resheniya transportnykh zadach s ogranicheniem po vremeni [Hybrid algorithm for solving vehicle routing problems with a time windows], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2015, No. 6 (167), pp. 180-191.
7. Emel'yanova T.S. Geneticheskii algoritm resheniya transportnoy zadachi s ogranicheniem po vremeni [Genetic algorithm for solving transportation problem with time limit], *Perspektivnye informatsionnye tekhnologii i intellektual'nye sistemy* [Advanced Information Technologies and Intelligent Systems], 2007, No. 4 (32), pp. 43-59.
8. Kureychik V.M., Emel'yanova T.S. Reshenie transportnykh zadach s ispol'zovaniem kombinirovannogo geneticheskogo algoritma [The solution of transport problems using the combined genetic algorithm], *Odinnadtsataya natsional'naya konferentsiya po iskusstvennomu intellektu s mezhdunarodnym uchastiem KII-2008: Trudy konferentsii* [Eleventh National conference on artificial intelligence with International participation KII-2008: Proceedings of the conference]. Vol. 1. Moscow: Fizmatlit, 2008, pp. 158-164.
9. Gladkov L.A., Kureychik V.M., Kureychik V.V., Sorokoletov P.V. Geneticheskie algoritmy [Genetic algorithms]. Moscow: Fizmatlit, 2010, 365 p.
10. Gladkov L.A., Kureychik V.M., Kureychik V.V., Sorokoletov P.V. Bioinspirirovannye metody v optimizatsii [Bioinspired methods in optimization]. Moscow: Fizmatlit, 2009, 384 p.
11. Herrera F., Lozano M. Fuzzy Genetic Algorithms: Issues and Models. Technical Report DECSAI-98116, Department of Computer Science and A.I., University of Granada, 1999, 25 p.
12. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions, *Soft Computing 7 (2003)*, Springer-Verlag, 2003, pp. 545-562.
13. Lee M.A., Takagi H. Dynamic Control of Genetic Algorithms using Fuzzy Logic Techniques, *Proceeding of 5th International Conference on Genetic Algorithms (ICGA'93), Urbana-Champaign, IL, July 17-21, 1993*, pp. 76-83.
14. Nechetkie gibridnye sistemy. Teoriya i praktika [Fuzzy hybrid system. Theory and practice], under the ed. N.G. Yarushkinoy. Moscow: Fizmatlit, 2007, 208 p.
15. Yarushkina N.G. Osnovy teorii nechetkikh i gibridnykh system [Fundamentals of the theory of fuzzy and hybrid systems]. Moscow: Finansy i statistika, 2004, 320 p.
16. Hongbo Liu, Zhanguo Xu, Ajith Abraham Hybrid Fuzzy-Genetic Algorithm Approach for Crew Grouping, *Proceedings of the Fifth International Conference on Intelligent Systems Design and Applications (ISDA 2005), 8-10 September 2005, Wroclaw, Poland. IEEE Computer Society 2005*, pp. 332-337.

17. Gladkov L.A., Gladkova N.V. Osobennosti ispol'zovaniya nechetkikh geneticheskikh algoritmov dlya resheniya zadach optimizatsii i upravleniya [Features of use of fuzzy genetic algorithms for the decision of problems of optimisation and control], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2009, No. 4 (93), pp. 130-136.
18. Gladkov L.A., Kureychik V.V., Kureychik V.M., Rodzin S.I. Osnovy teorii evolyutsionnykh vychisleniy: monografiya [Fundamentals of the theory of evolutionary computing: monograph]. Rostov-on-Don: Izd-vo YuFU, 2010.
19. Gladkov L.A., Gladkova N.V., Leiba S.N. Manufacturing Scheduling Problem Based on Fuzzy Genetic Algorithm, *Proceedings of IEEE East-West Design & Test Symposium (EWDTS'2014). Kiev, Ukraine, September 26 – 29, 2014*, pp. 209-213.
20. Gladkov L.A., Leiba S.N., Gladkova N.V., Legebokov A.A. Parallel Genetic Algorithm Based on Fuzzy Logic Controller for Design Problems, *Advances in Intelligent Systems and Computing. Proceedings of the 5th Computer Science On-line Conference 2016 (CSOC 2016), Vol. 1: Artificial Intelligence Perspectives in Intelligent Systems.* – Springer International Publishing, Switzerland, 2016, pp. 147-157.

Статью рекомендовала к опубликованию д.т.н., профессор Л.С. Лисицына.

**Гладков Леонид Анатольевич** – Южный федеральный университет; e-mail: leo@tgn.sfedu.ru; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: 88634371625; кафедра САПР; доцент.

**Гладкова Надежда Викторовна** – тел.: 88634393260; кафедра ДМ и МО; старший преподаватель.

**Gladkov Leonid Anatol'evich** – Southern Federal University; e-mail: leo@tgn.sfedu.ru; 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: 88634371625; the department of CAD; associate professor.

**Gladkova Nadezhda Viktorovna** – phone: +78634393260; the department of DM&MO; senior teacher.

УДК 004.896

**В.В. Курейчик, А.Е. Глущенко, А.Н. Орлов**

### **ГИБРИДНЫЙ ПОДХОД ДЛЯ РЕШЕНИЯ ЗАДАЧИ 3-Х МЕРНОЙ УПАКОВКИ\***

*Работа связана с решением одной из важных оптимизационных задач – задачей трехмерной упаковки разногабаритных элементов в объеме. Задача трехмерной упаковки разногабаритных элементов является NP-сложной и NP-трудной. В статье описывается гибридный подход к решению данной задачи. Приведена постановка и ограничения задачи трехмерной упаковки. Предложен модифицированный гибридный поиск, использующий многоуровневую эволюцию и позволяющий частично решать проблему предварительной сходимости алгоритмов. Принципиальным отличием предложенного подхода является разделение процесса поиска на два этапа и применение на каждом из этих этапов различных алгоритмов. На первом этапе поиска реализуется генетический алгоритм, позволяющий производить первоначальную эффективную перестановку блоков. На втором этапе поиска реализуется быстрый эволюционный алгоритм, позволяющий улучшить предыдущие решения. Такой гибридный подход, позволяет получать наборы квазиоптимальных решений, за полиномиальное время. Описан пример работы гибридного алгоритма решения задачи трехмерной упаковки блоков в объеме. Разработан программный комплекс и реализованы на ЭВМ алгоритмы автоматизированной трехмерной упаковки на основе гибридного поиска. Проведен вычислительный эксперимент на тестовых примерах (бенчмарках). Качество упаковки, полученной, на основе гибридного подхода, в среднем на 2,17 % превосходит результаты упаковки, полученные с использованием известных алгоритмов*

\* Работа выполнена при финансовой поддержке РФФИ (проект № 16-01-00586).