

## SOME EXACT AND HEURISTICS ALGORITHMS FOR SINGLE VEHICLE ROUTING PROBLEM WITH PICKUPS AND DELIVERIES

R. V. GINDULLIN

gramiz@mail.ru

Ufa State Aviation Technical University, Russia

Submitted 2014, June 10

**Abstract.** A problem for constructing the shortest cyclic route that ensures homogeneous cargo is delivered from producers to consumers by a transport vehicle of limited capacity is considered. Formalizations in the form of quadratic Boolean programming, linear integer and linear Boolean programming problems are given. Comparative analysis of the efficiency of four exact algorithms is performed. The problem of finding the minimal admissible capacity of the transport vehicle is considered as an auxiliary problem. Dependence of the length of the optimal route on the capacity of the transport vehicle is studied experimentally.

**Key Words:** vehicle routing; optimization; logistic.

We consider the following problem. A transport vehicle is to deliver a homogeneous cargo from producers to consumers via the cyclic route. We assume that all destinations are numbered. We use  $a^i$  ( $i = 1, \dots, n$ ) to designate both availability of the cargo at the point of production and the demand for it at the point of consumption –  $a_i > 0$  for the production point and  $a_i < 0$  for the consumption point. We assign the base, i.e., the point of origin where the transport vehicle should return after it delivers all the cargo, number 0,  $a_0 \geq 0$ . We assume

that  $\sum_{i=0}^n a_i = 0$  (the cargo is balanced). Find the route

of minimal length. Problems of this type that were later called Vehicle Routing Problems (VRPs) were first stated in [1]. Note that for  $a_0 = n$ ,  $a_1 = \dots = a_n = -1$ , the stated problem is the travelling salesman problem, which is NP-hard. This means the problem involved also belongs to this class. Note that exact polynomial algorithms of solving NP-hard problems have not been proved to exist to date (the NP problem). For this reason, we restrict ourselves to problems of small dimension (not more than 20 destinations). From the four algorithms involved, we chose the most efficient in terms of the analysis of the dependence of the value of the objective function on the capacity of the transport vehicle. In [2], several heuristic algorithms are given for the stated problem. We consider the problem of finding

the minimal admissible capacity of the transport vehicle as an auxiliary problem.

*Problem 1.* Find  $S$ , i. e., the minimal admissible capacity of the transport vehicle, if each destination except for the base is to be visited only once.

The problem is formalized as follows. Suppose  $X$  is the Boolean permutation matrix of dimension  $(n + 1) \times (n + 1)$ :  $X_{ij} = 1$  if and only if the destination number  $j$  is the  $i$ th point visited in the cycle,  $i, j = 0, \dots, n$ . Find Boolean variables  $X_{ij}$  such that

$$\sum_{i=0}^n X_{ij} = 1 \quad (j = 0, \dots, n) \quad (1)$$

$$\sum_{j=0}^n X_{ij} = 1 \quad (i = 0, \dots, n) \quad (2)$$

$$X_{00} = 1 \quad (3)$$

$$0 \leq \sum_{i=0}^k \sum_{j=0}^n a_j X_{ij} \leq S \quad (k = 0, \dots, n) \quad (4)$$

$$S \rightarrow \min. \quad (5)$$

(1)–(5) is the linear integer programming problem. The following proposition holds.

*Proposition 1.*  $\max |a_i| \leq S < 2 \max |a_i|$ , and for any  $\varepsilon > 0$  there exists the number  $n$  and the sequence  $a_0, \dots, a_n$  such that  $S > (2 - \varepsilon) \max |a_i|$  (this is a stronger version of the results of [2]).

*Proof.* The lower bound is obvious. We use the designations  $a^+ = (|a| + a)/2$ ,  $a^- = (|a| - a)/2$ . If  $\max |a^+| = \max |a^-|$  i. e., the maximal cargo at the point of

production equals the maximal demand at the point of consumption, we can no longer consider them since if we place them at two first places in the sequence, delivery from one point to another can be done by the transport vehicle with the capacity  $\max|a_i| < 2 \max|a_i|$ , and the transport vehicle is left empty after that. Going on with it, we obtain a cargo sequence such that  $\max|a^+| \neq \max|a^-|$ . We prove that the transport vehicle of capacity  $\max|a^+| + \max|a^-| < 2 \max|a_i|$  is sufficient for such a cargo sequence (we can consider that the maximum in the right-hand side is taken over the elements of the initial sequence). Indeed, at the first step, we can load the transport vehicle with any positive cargo. Suppose at some instant the load of the transport vehicle is not greater than  $\max|a^-|$ . If there are still points of production with the cargo left in them, the remaining capacity is sufficient to load any of these cargos. Otherwise, if the load of the transport vehicle is greater than  $\max|a^-|$  the transport vehicle has enough cargo to unload at any consumption point. Thus, while there are still unprocessed points, the process can be continued, which is what we wanted.

To prove the second proposition, for some  $\varepsilon > 0$ , we consider the following sequence of weights  $-(k+1)$  cargos equal  $(+k)$  and  $k$  cargos equal  $(-k-1)$ . Obviously,  $S = 2k$ . The relation  $2k / \max\{|a_i|\} = 2k / (k+1) > (2 - \varepsilon)$  for a sufficiently large  $k$ .

Note that the lower bound is attained for the sequence  $(1, 1, -2)$ .

*Problem 2.* Boolean quadratic formalization. We have the transport vehicle of the admissible capacity  $S$  and the matrix  $C_I$  of the distances between the points. Under the hypotheses of problem 1, we should arrange transportation so that the route length is minimal.

Note that when the points are permuted according to matrix  $X$  from problem 1, the matrix of distances has the form  $C_X = X C_I X^T$ . The subscript  $I$  in the initial matrix  $C_I$  indicates the unity matrix. The problem can be formalized as follows. Find the Boolean matrix  $X$  such that conditions (1)–(4) hold and

$$\sum_{i=0}^{n-1} (C_X)_{i,i+1} + (C_X)_{n,0} \rightarrow \min. \quad (6)$$

(1)–(4), (6) is the problem of Boolean quadratic optimization. In the next section, we construct the linear integer formalization of problem 2 with a significantly increased dimension.

*Problem 2.* Linear integer formalization. One knows a whole number of linear formalizations of

the travelling salesman problem (see [4] for review). The modification of the technique proposed in a number of works (for in-stance, in [3]) suits us most of all.

We introduce Boolean variables  $Y_{ij}$  ( $i, j = 0, \dots, n$ ) that equal 1 if and only if the  $j$ -th destination is the next destination after the  $i$ -th one in the route of the transport vehicle. The following conditions are met:

$$\sum_{j=0}^n Y_{ij} = 1 \quad (i = 0, \dots, n) \quad (7)$$

(the transport vehicle leaves each destination for the only destination),

$$\sum_{i=0}^n Y_{ij} = 1 \quad (j = 0, \dots, n) \quad (8)$$

(the transport vehicle arrives in each destination from the only destination),

$$Y_{00} = 0 \quad (9)$$

(the transport vehicle leaves the zero destination for some other destination; for other destinations, this will follow from the subsequent restrictions).

We introduce integer variables  $v_i$  ( $i = 1, \dots, n$ ) such that:

$$1 \leq v_i \leq n, \quad (10)$$

$$(v_i - v_j) + n Y_{ij} \leq n - 1 \quad (i, j = 1, \dots, n). \quad (11)$$

The following proposition holds.

*Proposition 2.* For any cycle satisfying the problem's hypothesis, there exists the solution of system (7)–(11). Inversely, any solution of system (7)–(11) gives the cycle that includes all destinations. The variables  $v_i$  has the sense of destination numbers as they are visited in the chain that includes all destinations but for the base.

*Proof.* If we remove zero destination (the base) from a cycle and give the variables  $v_i$  the sense indicated above, all conditions (7)–(11) will obviously hold.

Now suppose variables  $Y_{ij}$  ( $i, j = 0, \dots, n$ ), ( $i = 1, \dots, n$ ) satisfy conditions (7)–(11). If  $Y_{ij} = 0$ , condition (11) is met for any values  $v_i$  that satisfy condition (10). We construct a directed graph, with its vertices marked by the numbers  $1, 2, \dots, n$ , and the presence of the arc  $(i, j)$  means  $Y_{ij} = 1$ . It follows from  $Y_{ij} = 1$  and condition (11) that  $v_i - v_j \leq -1$ ; i.e., there are no cycles in the constructed graph. It follows from condition (7) (or (8)) that  $(n+1)$  variables  $Y_{ij}$  ( $i, j = 0, \dots, n$ ) are 1, while it follows from condition (9) that exactly  $(n-1)$  variables  $Y_{ij}$  ( $i, j = 0, \dots, n$ ) are 1. Thus, the respective undi-

rected graph is a tree. It follows from conditions (7) and (8) that each vertex of the graph has in-degree and out-degree not greater than 1. Thus, the constructed graph is a simple chain. Since the number of the values  $v_i$  is the same as the number of vertices in the graph and if there is the arc  $(i,j)$  by the condition  $v_i - v_j \leq -1$ , different vertices are matched to different values  $v_i$ ,  $v_i - v_j \leq -1$  for the adjacent vertices, which means that  $v_i$  are indeed the numbers of vertices in the order of passing along the chain. By condition (7), the cargo is delivered to the destination that corresponds to the first vertex of the chain from some place. It can be the base or the destination that corresponds to the last vertex of the constructed chain. However, the second case contradicts condition (9) since in this case the only unity in the zero row should be at the zero place. A similar case takes place for the last vertex of the chain. Thus, we constructed the cycle of the required form.

We introduce the Boolean variables  $Z_{sk}$  that are 1 if and only if  $v_s \leq k$  ( $s, k = 1, \dots, n$ ). The restrictions hold

$$Z_{sk} \geq (1 + k - v_s)/(2n) \quad (s, k = 1, \dots, n) \quad (12)$$

(if the right-hand side is positive, then  $Z_{sk} = 1$ ).

$$\sum_{s=1}^n \sum_{k=1}^n Z_{sk} = n(n+1)/2. \quad (13)$$

It follows from conditions (12) and (13) that  $Z_{sk} = 0$  for  $v_s > k$ .

The conditions of overload and nonnegativity of the cargo weight in the transport vehicle when each destination is visited have the form

$$0 \leq a_0 + \sum_{s=1}^n Z_{sk} a_s \leq S \quad (k=1, \dots, n). \quad (14)$$

The objective function is

$$\sum_{i=0}^n \sum_{j=0}^n Y_{ij} c_{ij} \rightarrow \min. \quad (15)$$

We describe the variables of the stated problem:  $Y_{ij}$  are Boolean, their number is  $(n+1)^2$ ;  $Z_{sk}$  are Boolean, their number is  $n^2$ ; and  $v_i$  are integer, their number is  $n$ . The total number of restrictions is of the order  $O(n^2)$ .

*Problem 2.* Linear Boolean formalization. Like in [5] we introduce Boolean variables that equal 1 if and only if  $j$ -th destination is the next destination after the  $i$ -th one in the step  $k$  of the route of the transport vehicle. The following conditions are met:

$$\sum_{k=1}^{n+1} \sum_{i=0}^n X_{ij}^k = 1 \quad (j=1, \dots, n), \quad (16)$$

$$\sum_{k=1}^{n+1} \sum_{j=0}^n X_{ij}^k = 1 \quad (i=1, \dots, n), \quad (17)$$

$$\sum_{i=0}^n \sum_{j=0}^n X_{ij}^k = 1 \quad (k=1, \dots, n+1), \quad (18)$$

$$\sum_{i=1}^n X_{i0}^{n+1} = 1, \quad (19)$$

$$\sum_{j=1}^n X_{0j}^1 = 1, \quad (20)$$

$$\sum_{i=0}^n X_{ij}^k = \sum_{p=0}^n X_{jp}^{k+1} \quad (j=0, \dots, n; k=1, \dots, n). \quad (21)$$

*Proposition 3.* Any solution of system (16)–(21) gives the cycle that includes all destinations exactly once (Hamiltonian cycle).

*Proof.* Condition (20) means that there is exactly one destination  $i_1 \neq 0$ , so  $\sum_{j=1}^n X_{0i_1}^1 = 1$ . And let's

create chain  $(0, i_1, \dots, i_k)$  by induction, so  $X_{i_s i_s}^s = 1$  ( $s=1, \dots, k, k < n, i_0 = 0$ ). Condition (16) and equality means that  $X_{i_k i_k}^k = 1$ . Using that in the

condition (21) results in  $\sum_{p=0}^n X_{i_k p}^{k+1} = 1$ , i. e. there is

exactly one destination  $i_{k+1}$  for which  $X_{i_k i_{k+1}}^{k+1} = 1$  is true. Check that  $i_{k+1} \notin \{i_1, \dots, i_k\}$ . If opposite is true, then there is  $i_s$  ( $s < k+1$ ) for which  $X_{i_k i_s}^s = 1$ .

But this contradicts (16).

Using this we'll get chain  $(0, i_1, \dots, i_{n-1})$  which includes all destinations. Because of (17) and (19)  $X_{i_n 0}^{n+1} = 1$ . So, we have a cycle with right properties.

$$0 \leq \sum_{k=1}^p \sum_{j=0}^n \sum_{i=0}^n X_{ij}^k a_i \leq S \quad (p=1, \dots, n). \quad (22)$$

$$\sum_{k=1}^{n+1} \sum_{j=0}^n \sum_{i=0}^n X_{ij}^k c_{ij} \rightarrow \min. \quad (23)$$

Problem (16)–(23) is linear Boolean problem with  $(n+1)^3$  variables. The total number of restrictions is of the order  $O(n^2)$ .

## COMPUTATIONAL EXPERIMENT

(1) We performed comparative analysis of the efficiency of using three algorithms to solve Problem 2

- Boolean quadratic (we used MATLAB with the cplex extension);
- linear integer (we used MATLAB with the cplex extension);
- linear Boolean (we used MATLAB with the cplex extension);

• branch and bound with the initial approximation constructed by the “first suitable” principle.

We randomly generated 100 examples for each of the dimensions  $n = 4, \dots, 18$ . The Table 1 gives the average number of iterations. The Table 2 gives the average time.

Table 1

Average number of iterations it takes various algorithms to solve the problem

$n$	Linear integer	Boolean quadratic	Branch and bound	Linear Boolean
4	137	15,8	14,3	1,24
5	$1,4 \cdot 10^3$	107	47,7	20,6
6	$1,47 \cdot 10^4$	841	199	54,2
7	$2,66 \cdot 10^5$	$7,46 \cdot 10^3$	911	185
8	$9,31 \cdot 10^6$	$6,67 \cdot 10^4$	$4,18 \cdot 10^3$	736
9		$6,54 \cdot 10^5$	$1,92 \cdot 10^4$	$1,98 \cdot 10^3$
10		$6,98 \cdot 10^5$	$1,01 \cdot 10^5$	$5,79 \cdot 10^3$
11			$4,63 \cdot 10^5$	$1,19 \cdot 10^4$
12			$2,21 \cdot 10^6$	$3,08 \cdot 10^4$
13			$1,22 \cdot 10^7$	$4,73 \cdot 10^4$
14				$1,11 \cdot 10^5$
15				$1,39 \cdot 10^5$
16				$2,67 \cdot 10^5$
17				$9,72 \cdot 10^5$
18				$1,03 \cdot 10^6$

Table 2

Average time it takes various algorithms to solve the problem

$n$	Linear integer, sec	Boolean quadratic, sec	Branch and bound, sec	Linear Boolean, sec
4	0,0686	0,0463	0	0,0306
5	0,235	0,123	0,000156	0,0466
6	0,856	0,123	0,00140	0,0721
7	13,5	0,496	0,00764	0,151
8	616	3,88	0,0414	0,294
9		41,1	0,248	0,573
10		490	1,61	1,42
11			8,97	2,99
12			52,0	7,75
13			333	14,3
14				36,2
15				53,5
16				131
17				478
18				856

Minimum number of destinations for which at least one example was not solved during 1 hour:

- $n=14$  for branch and bound algorithm;
- $n=9$  for linear integer programming;
- $n=20$  for linear Boolean programming;
- $n=11$  for Boolean quadratic programming.

Maximum number of destinations for which at least one example was solved during 1 hour:

- $n=15$  for branch and bound algorithm;
- $n=14$  for linear integer programming;
- $n=28$  for linear Boolean programming;
- $n=10$  for Boolean quadratic programming.

For  $n \geq 13$ , the branch and bound algorithm is almost inapplicable, with the linear Boolean algorithm being the most efficient.

(2) We analyzed how the increased capacity of the transport vehicle influences the route length, i.e., the objective function. To solve Problem 1, we used MATLAB with the cplex extension. Then, we applied the linear Boolean algorithm to solve Problem 2 for the minimal capacity of the transport vehicle, i. e., the solution to Problem 1, and for the increased capacity of the transport vehicle, which was 1.1; 1.2;...; and 2.5 times greater than the minimal one. We formed 100 examples for each of 4–12 destinations and averaged the lengths. Fig. 1 gives some results.

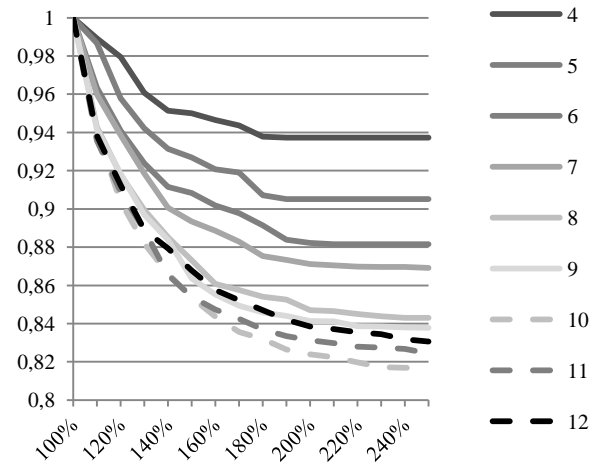


Fig. 1. Length of the optimal route depending on the growing capacity of the transport vehicle

The Tables 3 and 4 gives the average number of iterations and average time for heuristics described in [2].

There is two numbers in cells. First number in cells is average number of cycle relative to optimal one (accepted as 1). Second number is average time it takes algorithm to solve the problem. Stopping rule for random search algorithm: if after 20  $n$  iterations there was no improvement then algorithm is stopped.

Table 3  
Average path length relative to optimal and average time (seconds) it takes various algorithms to solve the problem

$n$	Linear Boolean	Random search	Perturbations (v. 1)	Perturbations (v. 2)	Prim's spanning tree	Greed search
4	1 / 0,0306	1,02 / 0,0009	1,01 / 0,0006	1,01 / 0,0009	1,02 / 0,0009	1,05 / 0,0004
5	1 / 0,0466	1,05 / 0,0007	1,03 / 0,0015	1,03 / 0,0013	1,05 / 0,001	1,08 / 0,0007
6	1 / 0,0721	1,102 / 0,0015	1,07 / 0,0012	1,08 / 0,0018	1,08 / 0,0012	1,11 / 0,0012
7	1 / 0,151	1,17 / 0,0017	1,12 / 0,0018	1,13 / 0,0020	1,09 / 0,0014	1,12 / 0,0028
8	1 / 0,294	1,24 / 0,0024	1,16 / 0,0034	1,17 / 0,0037	1,11 / 0,0015	1,14 / 0,0012
9	1 / 0,573	1,33 / 0,0037	1,17 / 0,0069	1,17 / 0,0065	1,12 / 0,0048	1,15 / 0,0025
10	1 / 1,42	1,41 / 0,0046	1,26 / 0,0199	1,25 / 0,0201	1,14 / 0,0138	1,17 / 0,0014
11	1 / 2,99	1,51 / 0,0057	1,27 / 0,149	1,25 / 0,159	1,13 / 0,0483	1,16 / 0,0031
12	1 / 7,75	1,539 / 0,0043	1,29 / 0,504	1,29 / 0,517	1,15 / 0,111	1,19 / 0,0040
13	1 / 14,3	1,65 / 0,0086	1,29 / 2,29	1,28 / 2,298	1,16 / 0,012	1,18 / 0,0059
14	1 / 36,2	1,71 / 0,0070	1,34 / 16,3	1,32 / 17,3	1,16 / 0,0144	1,17 / 0,0052
15	1 / 53,5	1,78 / 0,0079	1,37 / 39,3	1,35 / 43,1	1,17 / 3,61	1,19 / 0,0075
16	1 / 131	1,83 / 0,0090	1,37 / 339	1,35 / 325	1,17 / 0,0228	1,19 / 0,0062
17	1 / 478	1,92 / 0,0091	-	-	1,18 / 781	1,21 / 0,0063

Table 4  
Average number of iterations it takes various algorithms to solve the problem

$n$	Linear Boolean	Random search	Perturbations (v. 1)	Perturbations (v. 2)	Prim's spanning tree	Greed search
4	1,24	13,3	22,2	23,7	21,5	16
5	20,6	18,4	47,4	50,1	33,1	25
6	54,2	22,5	101	102	51,5	36
7	185	26,3	184	171	73,9	49
8	736	32,4	451	309	122	64
9	$1,98 \cdot 10^3$	38,8	$1,26 \cdot 10^3$	519	183	81
10	$5,79 \cdot 10^3$	39,2	$4,69 \cdot 10^3$	712	210	100
11	$1,19 \cdot 10^4$	39,6	$3,58 \cdot 10^4$	$1,15 \cdot 10^3$	290	121
12	$3,08 \cdot 10^4$	46,7	$1,13 \cdot 10^5$	$1,68 \cdot 10^3$	368	144
13	$4,73 \cdot 10^4$	47,9	$4,75 \cdot 10^5$	$2,44 \cdot 10^3$	415	169
14	$1,11 \cdot 10^5$	54,7	$3,33 \cdot 10^6$	$3,32 \cdot 10^3$	503	196
15	$1,39 \cdot 10^5$	56,4	$8,07 \cdot 10^6$	$4,61 \cdot 10^3$	748	225
16	$2,67 \cdot 10^5$	58,7	$2,67 \cdot 10^7$	$6,59 \cdot 10^3$	933	256
17	$9,72 \cdot 10^5$	64,7	—	—	$1,50 \cdot 10^3$	289

## CONCLUSION

We considered some optimization problems of transport logistics. We implemented alternative approaches to construct the exact solution of the problem of minimization of the cyclic route of cargo delivery from producers to consumers. The computational experiments showed that:

- the most efficient way is to reduce the problem to the model of linear Boolean programming;
- if the carrying capacity of the transport vehicle is increased more than 1.6–1.7 times as compared to the minimal admissible capacity, the length of the route is not reduced on average;
- if the number of destinations is increased (until  $n = 10$ ), the growth of the carrying capacity of the transport vehicle leads to a greater reduction of the length of the cycle;
- if number of destinations exceeds  $n = 10$  then the reduction of the length of the cycle is stopped.
- Heuristics based on Prim's spanning tree algorithm finds paths closest to the optimal ones.

## REFERENCES

1. Dantzig G. B., Ramser J. H. "The truck dispatching problem," *Management Science*, vol. 6, no. 1, pp. 80-91, Oct., 1959.
2. Bronshtein E. M., Gindullin R. V. "On one class of routing problems," *Mathematical Modelling*, 23 (6):123–132, 2011.
3. Gavish B. "A note on the formulation of the m-salesman traveling salesman problem," *Management Science*, 22 (6), 704–705, 1976.
4. Diaby M. "The salesman traveling problem: a linear programming formulation," *WSEAS Transactions on Mathematics*, 6 (6):745–754, 2007;.
5. Picard J. C., Queyranne M. "The time-dependent traveling salesman problem and its application to the tardiness in one-machine scheduling," *Operations Research*, 26:86–110, 1978.

## ABOUT THE AUTHOR

**GINDULLIN, Ramiz Vilevich**, technician and former aspirant of department of computational mathematics and cybernetics. Diploma: economist-mathematician (UGNTU, 2009).

## МЕТАДААННЫЕ

**Название:** Некоторые точные и эвристические алгоритмы для задач маршрутизации с одним транспортным средством, вывозом и доставкой грузов.

**Авторы:** Р.В. Гиндуллин<sup>1</sup>

**Организация:**

ФГБОУ ВПО «Уфимский государственный авиационный технический университет» (УГАТУ), Россия.

**Email:** <sup>1</sup>gramiz@mail.ru.

**Язык:** English.

**Источник:** Вестник УГАТУ. 2014. Т. 18, № 5 (66), С. 48–53, 2014. ISSN 2225-2789 (Online), ISSN 1992-6502 (Print).

**Аннотация:** Рассматривается задача построения кратчайшего замкнутого маршрута для перевозки однородного груза от производителей потребителям транспортным средством ограниченной вместимости. Представлены формализации в виде задач квадратичного булева, линейного целочисленного и линейного булева программирования. Выполнен сравнительный анализ эффективности четырех точных алгоритмов. Дополнительно рассмотрена задача нахождения минимальной допустимой вместимости транспортного средства. Экспериментально найдена зависимость длины оптимального маршрута от вместимости транспортного средства.

**Ключевые слова:** транспортная маршрутизация; оптимизация; логистика.

**Об авторе:**

**ГИНДУЛЛИН, Рамиз Вилевич**, техн. каф. выч. математики и кибернетики. Дипл. экон.-мат. (УГНТУ, 2009), аспирантура (там же).