# Finding an initial plan of transport resources FTL allocation in a special VRP problem using linear programming methods

**Petr Skobelev[1], Alexander Lada[2], Igor Mayorov[3]**
**[1]Institute of the Control of Complex Systems of Russian Academy of Science, Samara, Russia**
**[2]Samara State Technical University, Samara, Russia**
**[3]Smart Solutions, Ltd, Samara, Russia**
**petr.skobelev@gmail.com, ladalexer@gmail.com, mayorov@smartsolutions-123.ru**

## ABSTRACT

The special VRP problem of transport resources allocation for freight transportation companies that deliver cargo via FTL business model was considered. It was admitted that each real freight transportation company operates in a real time and needs to react on incoming events adaptively reallocate available resources. For this purposes multi agent systems are well proved and used in many modern freight companies. But it was admitted that there is a possibility to improve a quality optimization level by using the stable time period during night hours when no new events come into the system and there is a time for using classic optimization approach in special sub problem of finding the initial allocation plan. By using expert human real logistic scheduling knowledge for a long time period the essential set of limitations to this initial allocation plan problem was defined. The problem was formalized similar to the classical assignment problem of linear programming. Acyclic and cyclic cases of the problem were considered. It was shown that the acyclic case of the problem could be reduced to the assignment problem easily but for the cyclic case it requires to exclude important resource to order matching condition. Finding the exact solution of the initial plan problem was proposed by using the Hungarian method, which is well proved exact method. It was also shown that this method couldn't be applied in case of real time scheduling, because even in static cyclic case it is impossible to support resource to order matching condition for next future orders, but it can be applied as an addition to the multi agent approach.

**Keywords**: transportation logistics, VRP, FTL, linear programming, the assignment problem, the Hungarian method, multi-agent technology.

## 1. INTRODUCTION

The freight optimization problem (Vehicle Routing Problem, VRP), first described in [1], is one of the most urgent and important tasks of the modern theory of optimization. Classification of optimization transportation logistics problems is described in [2, 3]. VRP tasks classification and review with proposed solutions is also given in [4]. In this paper, the special VRP task of trucks to orders allocation for large cargo transportation companies that has its own fleet more than 30 truck units is formalized and proposed for solving. Transportation companies of this kind are widespread in countries with a large territory and long roads (Russia, USA, Canada, etc.). These companies perform an interregional cargo transportation via FTL (Full Truck Load) business model. FTL model is characterized by the direct customers contracts on a whole truck reservation, which eliminates the need to take into account the volume of cargo and build consolidated routes. This simplification motivates to introduce a more accurate method to solve the task without using heuristics. Various models of the organization of cargo transportation by FTL model are given in [5]. In this paper we also take into account the time windows of truck arrival to the loading point, so the task we are dealing with belongs to VRPTW (Vehicle Routing Problem with Time Windows), which are explored in [6]. Additionally we consider that some orders require only special truck and trailer with additional equipment installed: refrigerator, sheathing for transportation of tires, lifters, etc. Restrictions on the maximum length of the trip is not included, because big trucking companies can afford to adaptively change drivers during the trip, bringing them to the trucks via other transport, such as aircraft. Also there is a need to choose most advantageous orders from the set in terms of prosecution of the trip, as in general, the number of demands exceeds the number of trucks and truck after the execution of the order don't return to the base but continue to move to a new order from the previous unloading place until it receives the order with unloading near its base. That's because unlike most standard VRP applications, where the return to the base is obligatory after each trip, in the task we are dealing with this condition is not rigidly defined. It generated dynamically during the problem solving process. And finally we have to keep in mind the real time factor. It means that after getting some initial truck to order allocation the data are beginning to change over time because new orders come in, delays in carrying out the previously planned orders happen, some trucks are no longer available because of unpredictable maintenance work and so on. Considering the task as a real time problem, according to [10] orders and resources are presented as a network of requirements (orders) and opportunities (resources). But in practice it is always possible to distinguish some period of time, when the network remains stable (no new order appears, none of the parameters of an existing one changed, no new truck appears and no existing one became unavailable). In practice, this happens because at the end of the working day transport managers finalize and fix the part of the schedule that has to be executed on the next day. By this action, they assign some trucks to orders but the rest part remains unassigned because there is a time for making decision. During the night before the next working day, there is a time to optimally assign this rest part according to some basic optimization criteria. Therefore, it means that the global problem of every day transport company resource management consists of 2 parts:

Construct initial plan based on available orders, taking into account the basic set of criteria.

Modificate the initial plan according to incoming real-time events. At this stage, it is necessary to take into account the total set of criteria that are usually not supported by standard methods.

For solving the second part of the problem, the multi-agent technology [7] is used that is well proved nowadays. There are good results for their applicability to the real transportation companies scheduling, described for example in [8] and [9]. In this paper we focus on the first part of the problem and develop the method of constructing the initial plan of orders to trucks assignment, taking into account the most significant factors that use humans (heads of logistics departments) when they construct it by hand, by formalizing these factors in the form of mathematical constraints. During the multi-agent system introduction [8], we analyze and summarize the empirical knowledge of the transportation companies staff that was accumulated by them for a long period of time and define a set of practical constraints, which can be used for the mathematical formulation of the linear programming problem:

Time of arrival at an order loading point, which is calculated as the time of a truck release plus the empty driving time for the truck, must be less than the right edge of the order loading time window, so earlier arrival is permissible, but to be late is not acceptable.

An empty driving time to an order loading point should be less than 500 kilometers, taking into account the fact that the average speed of trucks accepted as 50 km/h, the empty driving time should not exceed 10 hours.

Truck idle time, which is calculated as the left edge of the order loading time window minus empty driving time to the order for the truck, must be less than 24 hours. So if the truck has enough time to drive to the loading point, but it additionally has to be idle more than a day, such an assignment is not acceptable.

## 2. THE PROBLEM DEFINITION

Let's assume that we have a set of orders Oi i = 1, N, each order is characterized by a geographical point of loading and unloading with time window [TOsi; TOfi] when point is available. There is a set of resources which are trucks with trailers Rj j = 1, M, each resource is characterized by its initial location geographical point and the time of its release from that location TRfj, which corresponds to the previous executed order unloading location and time or truck's base. For any truck Rj the empty driving time Dij is known for each order Oi. Each order Oi occupies a whole truck Rj with trailer that satisfies the order constraints, so the truck Rj can match or mismatch to the order Oi. All orders are considered as equal that means we can assign order Oi to truck Rj or just skip the order Oi without any penalties from the order Oi customer (in practice these orders will be resold to another external carrier company 3PL). The goal of the problem is to assign all M resource to orders where the total empty driving time will be minimum, with a maximum quantity of assigned orders Q and constraints for acceptable assignment are fulfilled:

| | |
|---|---|
| $\sum_{i,j} D_{ij} \to \min, Q \to N$ | (1) |
| $\begin{cases} TRf_j + D_{ij} < TO_{fi} \\ D_{ij} < 10 \\ TOs_i - TR_{fj} - D_{ij} < 24 \end{cases}$ | (2) |

## 3. THE PROPOSED METHOD FOR SOLVING THE PROBLEM

For solving the problem, we propose two stages. At first stage, we construct the matrix of acceptable assignments for defining all possible assignments satisfying the given constraints of the problem. At the second stage, the matrix is reduced to the classic assignment problem, which can be solved by one of the linear programming methods.

### 3.1. Construction of acceptable assignments matrix

We construct the matrix in which the rows correspond to order $O_i$ and the columns correspond to resources $R_j$. In each cell that corresponds to $O_iR_j$ assignment, we set empty driving time $D_{ij}$ that takes for the truck $R_j$ to move from its current location taking into account its release time $TRf_j$ to the order $O_i$ loading point, but only if the truck $R_j$ matches the order $O_i$ and the condition of inequalities system (2) is satisfied, otherwise the cell remains empty.

For clarity, let's consider examples for the matrix construction for the special (acyclic) and general (cyclic) problem cases.

#### 3.1.1. Construction of acceptable assignment matrix example for the acyclic case

The set of orders with loading points and its time windows relative to initial time $T_0$=0 is given in table 1:

| | Loading point | *TOs* | *TOf* |
|---|---|---|---|
| $O_1$ | Moscow | 2 | 4 |
| $O_2$ | Samara | 18 | 22 |
| $O_3$ | Ekaterinburg | 38 | 40 |

Table 1: set of orders

The set of resources with initial location point and release time relative to initial time $T_0$=0 is given in table 2:

| | Location point | *TRf* |
|---|---|---|
| $R_1$ | Moscow | 1 |
| $R_2$ | Samara | 6 |
| $R_3$ | Ekaterinburg | 12 |

Table 2: set of resources

The driving time between each location is given in table 3:

| | Moscow | Samara | Ekaterinburg |
|---|---|---|---|
| Moscow | 1 | 13 | 24 |
| Samara | 13 | 1 | 9 |
| Ekaterinburg | 24 | 9 | 1 |

Table 3: driving time between each location

Each resource $R_j$ is considered suitable for order $O_i$. It is also assumed that each order execution time (the time that it needs to drive from loading to unloading point) exceeds the latest order loading start time. That is why we call that case acyclic, because none of the resource has time to execute more than one order. For each potential assignment $O_i$ to $R_j$ we check satisfaction to the condition of inequalities system (2):

| | |
|---|---|
| $O_1R_1 = \begin{cases} 1+1 < 4 \\ 1 < 10 \\ 2-1-1 < 24 \end{cases}$ | $O_1R_2 = \begin{cases} 6+13 < 4 \\ 13 < 10 \\ 2-6-13 < 24 \end{cases}$ |
| $O_2R_1 = \begin{cases} 1+13 < 22 \\ 13 < 10 \\ 18-1-13 < 24 \end{cases}$ | $O_2R_2 = \begin{cases} 6+1 < 22 \\ 1 < 10 \\ 18-6-1 < 24 \end{cases}$ |
| $O_3R_1 = \begin{cases} 1+24 < 40 \\ 24 < 10 \\ 38-1-24 < 24 \end{cases}$ | $O_3R_2 = \begin{cases} 6+9 < 40 \\ 9 < 10 \\ 38-6-9 < 24 \end{cases}$ |

$$O_1R_3 = \begin{cases} 12+24 < 4 \\ 24 < 10 \\ 2-12-24 < 24 \end{cases}$$

$$O_2R_3 = \begin{cases} 12+9 < 22 \\ 9 < 10 \\ 18-12-9 < 24 \end{cases}$$

$$O_3R_3 = \begin{cases} 12+1 < 40 \\ 1 < 10 \\ 38-12-1 < 24 \end{cases}$$

The system of conditions satisfies to the following assignments: $O_1R_1$; $O_2R_2$; $O_2R_3$; $O_3R_2$. The acceptable assignment matrix for the case is given in table 4:

| | $R_1$ | $R_2$ | $R_3$ |
|---|---|---|---|
| $O_1$ | 1 | | |
| $O_2$ | | 1 | 9 |
| $O_3$ | | 9 | |

Table 4: the acyclic case acceptable assignment matrix

### 3.1.2. Construction of acceptable assignment matrix example for the cyclic case

In the previous example, we assumed that none of the resource has time to execute more than one order, because all the orders loading time windows [*TOs; TOf*] were densely allocated, and any order execution time always exceeds them. Now we consider the general cyclic case where the time windows in the given set of orders are widely allocated because we include future orders with we know with a high possibility level (e.g., orders from regular customers, usually known with good accuracy for a week or even a month in advance). Therefore, each resource has chance to execute later orders after earlier orders execution. It should be clear that in this case, location and time of the release for each resource would change during the problem solving according to $O_i$ unloading points. The acceptable assignment matrix will have a different structure. Let's consider how it will happen by another example:

The set of orders with loading and unloading points and its time windows relative to initial time $T_0$=0 is given in table 5:

| | Loading point | Unloading point | *TOs* | *TOf* |
|---|---|---|---|---|
| $O_1$ | Moscow | Samara | 1 | 2 |
| $O_2$ | Samara | Ekaterinburg | 23 | 28 |
| $O_3$ | Ekaterinburg | Moscow | 50 | 52 |

Table 5: set of orders

The set of resources with initial location point and release time relative to initial time $T_0$=0 is given in table 6:

| | Location | *TRf* |
|---|---|---|
| $R_1$ | Moscow | 0 |
| $R_2$ | Samara | 0 |
| $R_3$ | Ekaterinburg | 0 |

Table 6: set of resources

The driving time between each location is given in table 3. Each resource $R_j$ is considered suitable for order $O_i$. For each potential assignment $O_i$ to $R_j$ we check satisfaction to the condition of inequalities system (2) and if the assignment is possible, we'll continue to consider the further assignment to the remaining orders taking into account the relocation of the resource $R_j$:

$$O_1R_1 = \begin{cases} 0+1 < 2 \\ 1 < 10 \\ 1-0-1 < 24 \end{cases} \Rightarrow$$

$$O_1R_2 = \begin{cases} 0+13 < 2 \\ 13 < 10 \\ 1-0-13 < 24 \end{cases}$$

$$O_2R_2 = \begin{cases} 0+1 < 28 \\ 1 < 10 \\ 23-0-1 < 24 \end{cases} \Rightarrow$$

$$O_2R_3 = \begin{cases} 0+9 < 28 \\ 9 < 10 \\ 23-0-9 < 24 \end{cases} \Rightarrow$$

$$O_3R_1 = \begin{cases} 0+24 < 52 \\ 24 < 10 \\ 50-0-24 < 24 \end{cases}$$

$$O_1R_1O_2 = \begin{cases} 14+1 < 28 \\ 1 < 10 \\ 23-14-1 < 24 \end{cases} \Rightarrow$$

$$O_1R_3 = \begin{cases} 0+24 < 2 \\ 24 < 10 \\ 1-0-24 < 24 \end{cases}$$

$$O_2R_2O_3 = \begin{cases} (23+9)+1 < 52 \\ 1 < 10 \\ 50-23-9-1 < 24 \end{cases}$$

$$O_2R_3O_3 = \begin{cases} (23+9)+1 < 52 \\ 1 < 10 \\ 50-23-9-1 < 24 \end{cases}$$

$$O_3R_2 = \begin{cases} 0+9 < 52 \\ 9 < 10 \\ 50-0-9 < 24 \end{cases}$$

| | |
|---|---|
| $O_1R_1O_2R_1O_3 = \begin{cases} (23+9)+1 < 52 \\ 1 < 10 \\ 50-23-9-1 < 24 \end{cases}$ | |
| $O_2R_1 = \begin{cases} 0+13 < 28 \\ 13 < 10 \\ 23-0-13 < 24 \end{cases}$ | |
| $O_3R_3 = \begin{cases} 0+1 < 52 \\ 1 < 10 \\ 50-0-1 < 24 \end{cases}$ | |

The acceptable assignment matrix for the case is given in table 7, where the columns $O_1R_1$; $O_1R_1O_2R_1$; $O_2R_2$; $O_2R_3$ are presented resources location $R_1$, $R_2$ and $R_3$ after a possible execution of orders $O_1$, $O_1$ then $O_2$ and $O_2$:

| | $R_1$ | $R_2$ | $R_3$ | $O_1R_1$ | $O_1R_1O_2R_1$ | $O_2R_2$ | $O_2R_3$ |
|---|---|---|---|---|---|---|---|
| $O_1$ | 1 | | | | | | |
| $O_2$ | | 1 | 9 | 1 | | | |
| $O_3$ | | | | | 1 | 1 | 1 |

Table 7: the case acceptable assignment matrix

### 3.2. The optimal assignment search method

After we found acceptable assignment matrix, we can solve the finding initial plan problem by searching such resources to orders assignment sequence in the matrix in which the total empty driving time for all trucks in the sequence will be minimum with the maximum quantity of assigned orders (1).

It can be seen that the assignment matrixes we found in the above examples are similar to the matrix, which formalizes one of the standard problem of linear programming – the assignment problem. As shown in [12] for the assignment problem, in some special cases (the acyclic case), it is possible to find the exact solution, which is the purpose of this work. It is known that the assignment problem is solvable in polynomial time. There are traditional solving methods for the task solution (e.g. Hungarian algorithm [13]) that have an asymptotic complexity of *O (n3)* that is more than enough even with a large dimension of the matrix in real transportation company cases.

The assignment problem formulation in terms of linear programming is following. Let's consider $O$ as variety of orders which contains $N$ elements and $R$ as variety of resources which contains $M$ elements. Variable $x_{ij}$ represents the assignment of $O_i$ to $R_j$, taking the value 1 if the resource $R_j$ is assigned to the order $O_i$, and 0 otherwise. $D(i,j)$ is the empty driving time from $R_j$ to $O_i$. The objective function and constraints for the task as follows:

| | |
|---|---|
| $\sum_{i \in O, j \in R} D(i,j) x_{ij}$ | (3) |
| $\sum_{j \in R} x_{ij} = 1, i \in O$ | (4) |
| $\sum_{i \in O} x_{ij} = 1, j \in R$ | (5) |
| $x_{ij} \geq 0, i, j \in O, R$ | (6) |

Depending on the number $N$ and $M$, the equations (4) and (5) will be replaced by inequality equations. If $M > N$, then some resources remain unoccupied, otherwise some orders will stay unassigned.

Consider how it is possible to reduce this paper problem to the assignment problem and solve it, for example, by the Hungarian algorithm. We begin by considering the special acyclic class of the problem, as described in above example 1, where any order execution time exceeds any order loading start time and none of the resource has time to perform more than one order. This class of the problem is simply reduced to the assignment problem, even if we assume that not all resources fit all orders and there are some $n<=N$ and $m<=M$ where $O_n$ doesn't fit $R_m$. But when we consider a general cyclic class of the problem, as described in above example 2, where location and release time of each truck change during the task definition, it is also possible to reduce it to the assignment problem, but in that case we have to remove the resource to order matching condition, assuming that any truck fits any order. Taking this assumption, for reducing to the assignment problem we need to convert acceptable assignment matrix to a new one, where we'll generalize all possible assignments and won't consider the particular truck to a certain order assignment, so the matrix in table 7 will be converted to the matrix in table 8:

| | $R_1$ | $R_2$ | $R_3$ | $RO_1$ | $RO_2$ | $RO_3$ |
|---|---|---|---|---|---|---|
| $O_1$ | 1 | | | | | |
| $O_2$ | | 1 | 9 | 1 | | |
| $O_3$ | | | | | 1 | |

Table 8: the assignment problem matrix for cyclic case

In the matrix columns after specific resources $R_1$, $R_2$ and $R_3$, we have columns with undefined resources $RO_1$, $RO_2$ and $RO_3$, which correspond to unloading points $O_1$, $O_2$ and $O_3$. By using this approach we cannot determine which specific resource will arrive at the next order loading point, this will be evaluated during the solving process. That is why it is impossible to take into account the resource to order matching condition for these undefined resources, but for the first group columns with exact resources, that condition is possible to check. When for this matrix the assignment problem will be solved by the Hungarian algorithm, for all assignments kind a $RO_iO_j$, we will evaluate the resource, which was previously assigned to the order $O_i$. It is also important to note that before solving the matrix by the Hungarian algorithm, the empty rows and columns must be excluded from the matrix. Also cells with blank values with correspond to unacceptable assignments should be filled with large values, greater than the maximum value in the matrix. If the calculated assignment sequence contains an assignment witch is not acceptable, it should be deleted from the sequence, leaving the assigned order as not assigned, and the resource as free.

As a result of the problem solving by the described above method, we obtain an exact solution, but with the assumption that any resource fits any order. To overcome this assumption and to obtain an exact solution taking into account this condition is not possible. However, based on industrial multi-agent planning system use experience [8], it is not necessary, because further the initial plan will be converted to agent base structure. After that, it will be modificated according to real-time events, by using the multi-agent approach [7]. If the order agent has received the initial assignment to the resource, which is not suitable for it, the order agent will adaptively change the resource at the other suitable one, taking into account possible changes that has happened by this time.

## 4. COMPUTATIONAL RESULTS

We performed several experiments to test the described methods by using the real data from our client companies. Each of them uses multi agent system [8]. The first one company PROLOGICS [15] has 140 resources and about 25 new orders per day. The second one company MONOPOLY [16] has 300 resources and about 76 new orders per day. The third one company LORRY [17] has 680 resources and about 240 new orders per day. The goal of the experiments was to compare multi-agent method, witch is real time oriented, with the Hungarian method in the task of initial plan construction. We run the system [8] on each client data snapshots and stopped it just after the initial plan was created. Based on the same data snapshots we created assignment problem matrixes and solved it by the Hungarian algorithm implementation [18]. For getting more clear picture the experimental data was snapped from the real client data in a different time during working month. It was clustered by the density (% of empty cells) in the assignment matrix because in general it varies from 5% to 95%. The results of the experiments are shown in Table 9. These experiments were run on workstation with a 3.4 GHz Intel Core i7-4770 CPU with 8Gb of RAM under Windows 8.1.

| Problem matrix | | |
|---|---|---|
| *N* orders | *M* resources | % of empty cells |
| 25 | 140 | 5 |
| 25 | 140 | 15 |
| 25 | 140 | 30 |
| 25 | 140 | 50 |
| 25 | 140 | 70 |
| 25 | 140 | 85 |
| 25 | 140 | 95 |
| 76 | 300 | 5 |
| 76 | 300 | 15 |
| 76 | 300 | 30 |
| 76 | 300 | 50 |
| 76 | 300 | 70 |
| 76 | 300 | 85 |
| 76 | 300 | 95 |
| 240 | 680 | 5 |
| 240 | 680 | 15 |
| 240 | 680 | 30 |
| 240 | 680 | 50 |
| 240 | 680 | 70 |
| 240 | 680 | 85 |
| 240 | 680 | 95 |

| Multi-agent method | | |
|---|---|---|
| *N* orders | KPI (1) | Execution time, sec |
| 25 | 35 | 0,014 |
| 25 | 33 | 0,016 |
| 25 | 43 | 0,012 |
| 25 | 49 | 0,009 |
| 25 | 73 | 0,005 |
| 25 | 147 | 0,004 |
| 25 | 341 | 0,003 |
| 76 | 82 | 0,6 |
| 76 | 86 | 0,5 |
| 76 | 93 | 0,4 |
| 76 | 111 | 0,3 |
| 76 | 127 | 0,2 |
| 76 | 208 | 0,12 |
| 76 | 483 | 0,052 |
| 240 | 242 | 39,91 |
| 240 | 240 | 34,99 |
| 240 | 245 | 29,91 |
| 240 | 255 | 21,8 |
| 240 | 285 | 13,29 |
| 240 | 431 | 6,97 |
| 240 | 980 | 2,66 |

| Hungarian method | | |
|---|---|---|
| *N* orders | KPI (1) | Execution time, sec |
| 25 | 34 | 0,001 |
| 25 | 32 | 0,001 |
| 25 | 41 | 0,001 |
| 25 | 47 | 0,001 |
| 25 | 68 | 0,001 |
| 25 | 144 | 0,001 |
| 25 | 339 | 0,001 |
| 76 | 77 | 0,001 |
| 76 | 81 | 0,001 |
| 76 | 92 | 0,003 |
| 76 | 106 | 0,001 |
| 76 | 123 | 0,001 |
| 76 | 199 | 0,003 |
| 76 | 472 | 0,001 |
| 240 | 240 | 0,011 |
| 240 | 240 | 0,014 |
| 240 | 241 | 0,01 |
| 240 | 246 | 0,01 |
| 240 | 273 | 0,008 |
| 240 | 414 | 0,007 |
| 240 | 934 | 0,006 |

Table 9: the results of computational experiments

## 5. CONCLUSIONS

In this paper, we apply the methods of linear programming for a special VRP problem for finding an initial plan of trucks to orders allocation in large transportation companies with use FTL business model. By taking into account the minimum necessary set of criteria, which were found based on human empirical knowledge, we reduce the problem, with agreed assumption, to the assignment problem that has an exact solution The Hungarian method. We also found the limitations of applicability to this method in case of switching to real time scheduling, because even if it is used for static cyclic task it is impossible to take into account truck matching condition for next future orders. Thus, we can conclude, that when shifting to the real time trucks to orders allocation, when new orders emerges or existing orders change or cancel, classical methods won't be enough, but classical and multi-agent approach combination will give a good solution which can be applicable in practice.

The study was supported by the Ministry of education and science of Russian Federation (the contract № 14.576.21.0014).

## 6. REFERENCES

[1] Dantzig G.B., Ramser J.H. The Truck Dispatching Problem // Management Science. 1959. Vol. 6, No. 1. P. 80-91.

[2] Michael Browne; Alan McKinnon; Sharon Cullinane. Green Logistics by Anthony Whiteing Published by Kogan Page, 2010.

[3] Bertazzi, L., Savelsbergh, M., & Speranza, M. G. (2008). Inventory Routing. In Bruce Golden, S. Raghavan, & E. Wasil (Eds.), The Vehicle Routing Problem: Latest Advances and New Challenges (Vol. 43. P. 49–72). New York: Springer Science+Business Media. doi:10.1007/978-0-387-77778-8.

[4] The vehicle routing problem: state of the art classification and review De Jaegere N, Defraeye M, Van Nieuwenhuyse, KU Leuven. [online] Available at: <https://lirias.kuleuven.be/bitstream/123456789/457452/1/KBI_1415.pdf> [14 January 2015].

[5] Oleg Granichin, Petr Skobelev, Alexander Lada, Igor Mayorov, Alexander Tsarev. Cargo transportation models analysis using multi-agent adaptive real-time truck scheduling system. – Proceedings of the 5th International Conference on Agents and Artificial Intelligence (ICAART'2013), February 15-18, 2013, Barcelona, Spain. – SciTePress, Portugal, 2013, Vol. 2. P. 244-249.

[6] Hideki Hashimoto, Toshihide Ibaraki, Shinji Imahori, Mutsunori Yagiura. The vehicle routing problem with flexible time windows and traveling times. Discrete Applied Mathematics. Volume 154, Issue 16, 1 November 2006. P. 2271–2290.

[7] Skobelev P. Bio-inspired multi-agent technology in real time scheduling / 10th IFAC Workshop on Intelligent Manufacturing Systems (IMS'10). July 1-2, 2010, Lisbon, Portugal. P. 384 – 385.

[8] A. Ivashchenko, A. Lada, Mayorov I., Skobelev P., Tsarev A. Analysis of the effectiveness of multi-agent system of management of regional traffic in real time // Proceedings of the 4th multiconference on governance ICCP-2011, October 3-8, 2011, p. Divnomorskoe, Gelendzhik, Russia. Vol. 1. – Taganrog. P. 353-356.

[9] A. Vaysblat, A. Diyazitdinova, A. Ivashchenko, P. Skobelev, A. Tsarev. Organization of interactive cooperation in multi-agent control system forwarding company // Proceedings of the XII International Conference "Problems of control and modeling in complex systems" Samara: Samara Scientific Center of Russian Academy of Sciences, 2010. P. 620 - 628.

[10] Skobelev P, Vittikh V. Multiagent Interaction Models for Constructing the Needs-and-Means Networks in Open Systems // Automation and Remote Control. 2003. Vol. 64, No.1. P. 177-185.

[11] Michal Maciejewski and Kai Nagel. Towards Multi-Agent Simulation of the Dynamic Vehicle Routing Problem in MATSim. 9th International Conference, PPAM 2011, Torun, Poland, September 11-14, 2011.

[12] Malkovsky N, Granichin O, Amelin K. The allocation of resources in the context of multi-agent systems // In Proc. works XII All-Russian Conference on Control (VSPU 2014), Russia, Moscow, Institute of Control Sciences, June 16-19, 2014. P. 9003-9013.

[13] Kuhn H.W. The Hungarian method for the assignment problem // Naval Research Logistics Quarterly. 1955. Vol. 2, No. 1-2. P. 83-97.

[14] J. Munkres, «Algorithms for the Assignment and Transportation Problems», Journal of the Society for Industrial and Applied Mathematics, 5(1):32—38, 1957 March.

[15] PROLOGICS. [online] Available at: <http://eng.prologics.ru/> [14 January 2015].

[16] MONOPOLY. [online] Available at: <http://monopoly.su/> [14 January 2015].

[17] LORRY. [online] Available at: <http://en.lorry-ural.ru/> [14 January 2015].

[18] GitHub. [online] Available at: <https://github.com/catbaxter/algorithms/blob/master/hungarian/hungarian.cpp> [14 January 2015].

Published: Published: Proceedings of the 19th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI 2015), Orlando, Florida, USA, July 12-15, 2015, vol.1. – International Institute of Informatics and Systemics, – P. 16-21. ISBN 978-1-941763-24-7.

https://books.google.ru/books/about/The_19th_World_Multi_Conference_on_Syste.html?id=7Fu2DAEACAAJ&redir_esc=y