A Solution to the Subtask of Initial Distribution of Transport Resources in a Special Optimization FTL Transportation Problem in Real-time Using the Hungarian Algorithm

P. O. Skobelev1* and A. N. Lada2

1Samara State Technical University, Russia; petr.skobelev@gmail.com
2Research and Production Company “Smart Solutions”, Samara, Russia

Abstract

Background/Objectives: The article considers the special problem of distributing transport resources in real time for freight transportation companies that carry loads according to the Full Truck Load (FTL) business model. Methods/Statistical analysis: To solve the real-time factor problem multi-agent technologies have been used, which are successfully applied in many modern automation systems in transportation companies. During the research a large amount of statistical data obtained as a result of the operation of multi-agent automation systems were processed. Findings: The possibility to improve the quality of their resulting decisions was revealed. It was proposed to divide the problem solving process into 2 stages: the initial stage of resource allocation (static) and the real-time stage (dynamic). It was suggested to apply the Hungarian method at the static stage and the existing multi-agent method at the dynamic stage. To solve the problem at the static stage, it was brought to the classic assignment problem that can be solved by the Hungarian algorithm that has proven itself as a good accurate method. The article showed the impossibility of solving the problem by the same method at the dynamic stage, since even in the special case of the dynamic stage one cannot determine the correct resource for the next order. As a result, a new hybrid method for solving this problem has been developed. The results of effectiveness of this method were obtained using actual data on transportation companies as compared to the existing one, and significant efficiency gains were demonstrated. Applications/Improvements: The new method can be used in existing multi-agent automation systems in transportation companies to improve the quality of solutions.

Keywords: Assignment Problem, Full Truck Load (FTL), Hungarian Algorithm, Multi-agent Technology, Real Time

1. Introduction

The problem of transportation optimization (Vehicle Routing Problem, VRP), first described in1, is one of the most urgent and important problems of modern optimization theory. Classification of optimization problems of transport logistics is given in2,3. An overview and classification of VRP problems with the proposed methods of solution is given in1. In this work we formalize and offer to solve a special VRP problem of trucks distribution for large transportation companies, having in its management the fleet of long-haul trucks numbering more than 30 units. Such transportation companies which are widespread in countries with a large territory and length of roads (Russia, the USA, Canada, etc.) carry out interregional transportation according to the FTL scheme. An FTL-transportation feature has direct contracts with customers to reserve a truck as a whole, eliminating the need to take into account the volume of cargo and build consolidated routes. Such simplification motivates to get a more
accurate method to solve it without the use of heuristics. Various models of the organization of cargo transportation by the FTL scheme are given in\(^7\). In this study, time windows of the truck arriving for shipment are further taken into account, which leads to variations of VRPTW problem (Vehicle Routing Problem with Time Windows) considered in\(^6\). Further, constraints dependent on trailer type (additional equipment in it, such as the refrigeration unit, the frame for transportation of tires, jacks, etc.) are taken into account. Limitations on the maximum length of the run are not included, because the large trucking companies can afford to adaptively change drivers during the travel, bringing them to the trucks by other modes of transportation, for instance, by aircraft. It is necessary to “choose” the most advantageous order from the existing set of orders in terms of the continuation of the run, because in general, the number of orders exceeds the number of trucks and a truck does not return to the base after the execution of the order, it continues to move to a new order from the previous unloading point until it receives the order of unloading near the depot. That is, unlike most standard VRP tasks, where the return to the depot is the obligatory end point of the route, in this setting up the problem, this condition is not strictly specified, but generated dynamically in the process of task solution. In the future, it will be required to take into account the “real time” factor, when on receipt of the decision the initial data begin to change over time, new orders are secured, there are delays in carrying out the previously planned orders, trucks are not available for scheduling because of scheduled and unscheduled work on their maintenance.

In solving the problem of finding the distribution of orders for trucks in real time, orders and resources are presented as the network of needs (orders) and opportunities (resources)\(^7\). It is believed that a prolonged period of time (more than an hour) can be always identified, when the network remains unchanged (no new order is added, none of the parameters of an existing order is changed, no new truck appears and no existing truck disappears). In practice, this is because at the end of the working day the transport company supervisory operators finish their work and record the part of the plan to be executed on the next day, in which part of orders is recorded for trucks, and another part remains unrecorded, as there is still time to make a decision on them. During the night before the start of the next business day it is possible to distribute them optimally. Therefore, global task of the transport company work control is divided into 2 parts:

- Constructing an initial basic plan for orders available at the initial time, with account of a basic set of fundamental criteria.
- Modification of the initial plan for real-time events, when it is necessary to make decisions on the situation by reference to the extended set of criteria which are poorly formalized and are not supported by standard methods.

Multi-agent technologies have shown themselves to good advantage in solving the second part of the problem\(^6\). There are good results of their applicability to real transport facilities, as described, for example, in \(^9,10\). In this paper we will focus on the development of a method of constructing the initial basic plan to distribute orders for trucks accounting for the main, most significant factors influencing this distribution that the heads of logistics departments use in constructing these plans by hand, as well as their formalization in terms of mathematical constraints. As a result of the introduction of multi-agent system,\(^9\) by analyzing and summarizing the accumulated knowledge of the employees of transportation companies, a set of practical constraints was determined, which can be used for mathematical formulation of the linear programming problem:

1. The time of arrival to the point of loading of the order, calculated as the time of the truck offloading plus the value of the length of the empty run, must be less than the extreme value corresponding to the right edge of the loading window of this order, i.e. it is acceptable to arrive earlier, but not to be late.
2. The empty run to the point of loading of order must be less than 500 km, given that the average speed of the truck movement is assumed equal to 50km/h, the duration of the empty run should not exceed 10 hours.
3. The value of the idle time duration, which is calculated as the initial value corresponding to the left edge of the loading window of the order minus the time of arrival at the point of loading of the order by any truck, must be less than 24 hours, i.e., if the truck has time to get to the point of loading, but additionally forced to idle more than a day, such an assignment is not permissible.

### 2. Statement of the Problem

Suppose we have a set of orders \(O, \ i = 1, N\), each order is characterized by a point and a time window of arrival.
for loading and unloading \([T_{Os}, T_{Of}]\), when the point is available. We have a set of resources which are trucks with trailers \(R, j = 1, M\), each of which is characterized by the point of the initial location and the time of release from this point \(TRf_j\), which corresponds to the time and place of unloading the previous order performed by it or the depot. For any truck \(R\) the duration of the empty run \(D_{ij}\) to any order \(O_i\) is known. For each order \(O_i\) there is a need for a separate trailer truck \(R\) satisfying the constraints on the trailer type, i.e. \(R\) truck can either be appropriate, or not appropriate for the order \(O_i\). All orders are treated equally and one can withdraw any order for the truck without any penalties by the customer (in practice, these orders will be resold to another foreign 3PL transportation company). We need to find such an assignment of all \(M\) resources for orders, in which the total empty run would be minimal, at the maximum number of assigned orders \(Q\) and meeting the conditions of the assignment feasibility:

\[
\sum_{i,j} D_{ij} \rightarrow \min, Q \rightarrow N
\]  

\[
\begin{cases} 
TRf_j + D_{ij} < T_{Of_i} \\
D_{ij} < 10 \\
T_{Os_i} - TRf_j - D_{ij} < 24 
\end{cases}
\]  

### 3. The Problem Solving Method

To solve this problem, it is proposed to divide the assigned task into 2 parts. In the first part the feasible assignments matrix is built to determine the space of solutions satisfying the given constraints of the problem. In the second part the matrix is brought to the classic assignment problem where the search for the best assignment is performed by one of the linear programming methods.

### 3.1 The Construction of the Feasible Assignment Matrix

For the problem the feasible assignment matrix is built, in which lines correspond to orders \(O_i\) and columns to resources \(R\). The duration of the empty run \(D_{ij}\) from the point where the truck \(R\) is and the time of its release \(TRf_j\) to the point of loading the order \(O_i\) are recorded in the matrix cell corresponding to the assignment \(O_i R_j\), provided that the truck \(R\) is suitable to the order \(O\) and the system of inequalities (2) is done, otherwise the cell remains empty.

For clarity, consider examples of construction of the feasible assignment matrix for private (acyclic) and total (cyclical) problem.

#### 3.1.1 An Example of Constructing the Feasible Assignment Matrix in the Acyclic Task

A set of orders, the point of loading and the time windows relative to the initial time \(T_0 = 0\) are given in Table 1.

| Order | Loading | TOs | TOf |
|-------|---------|-----|-----|
| \(O_1\) | Moscow | 2 | 4 |
| \(O_2\) | Samara | 18 | 22 |
| \(O_3\) | Yekaterinburg | 38 | 40 |

A set of resources, their initial location and release time relative to \(T_0 = 0\) are given in Table 2.

| Resource | Location |
|----------|----------|
| \(R_1\) | Moscow |
| \(R_2\) | Samara |
| \(R_3\) | Yekaterinburg |

The duration of the route between locations is given in Table 3.

| Location | Moscow | Samara | Yekaterinburg |
|----------|--------|--------|---------------|
| Moscow   | 1      | 13     | 24            |
| Samara   | 13     | 1      | 9             |
| Yekaterinburg | 24 | 9 | 1 |
Inequality is executed for assignments: \( O_1R_1; O_2R_2; O_3R_3 \). The feasible assignment matrix for the task is given in Table 4.

### 3.1.2 The Feasible Assignment Matrix in the Cyclic Task

In the previous example, it was considered that none of the resources have time to perform more than one order, since that loading time intervals operation \([TO_i; TO_j]\) are distributed densely, and the duration of any order implementation by any truck always exceeds them. Now the general case is considered, when the assigned orders are known with a view to the future (e.g., standing orders from regular customers are usually known with good accuracy for a week or even a month in advance). Therefore, every resource has the possibility to perform later orders after executing earlier orders. It should be understood that in this formulation, the location and time of the release of each resource will change in the course of solving the problem and the feasible assignment matrix will have a different view:

A set of orders with their loading and unloading points with time windows relative to the initial time \( T_0 = 0 \) are given in Table 5.

A set of resources with their initial location and release time relative to \( T_0 = 0 \) are given in Table 6.

The duration of a rout between locations is given in Table 3. For each resource \( R_j \) the possibility of its assignment for the order \( O_i \) according to the system of inequalities (2) is checked and if the assignment is possible, the possibility of a further assignment for the remaining orders is considered, taking into account the resource \( R_j \) relocation:
will be minimal with the maximum number of assigned orders (1).

It can be seen that the obtained feasible assignment matrices in the above examples are similar to the matrices by which the standard linear programming problem is formalized – the problem of assignments. As shown in\(^1\), in some special cases (for example, acyclic tasks), you can get the exact solution to the assignment problem, which was the aim of this work. It is known that the assignment problem is polynomially solvable, the traditional solution (Hungarian algorithm\(^2\)) has an asymptotic complexity \(O(n^3)\); this is enough to solve this problem, even with large dimension matrix of feasible assignments in real problems of transportation companies.

We formulate the problem of assignments in terms of linear programming. Let \(A\) be a lot of orders, containing \(N\) elements, and \(R\) is a set of resources comprising \(M\) elements. The variable \(x_{ij}\) is the assignment \(O_i\) to \(R_j\), taking the value of 1 if the resource \(R_j\) is assigned to the order \(O_i\), and 0 otherwise. We introduce \(D(i, j)\) as the empty run length of the resource \(R_j\) to the order \(O_i\). The objective function and constraints for the problem will be:

\[
\begin{align*}
\sum_{i \in O, j \in R} D(i, j)x_{ij} \\
\sum_{i \in O} x_{ij} &= 1, i \in O \\
\sum_{j \in R} x_{ij} &= 1, j \in R \\
x_{ij} &\geq 0, i, j \in O, R
\end{align*}
\]

Depending on the number of \(N\) and \(M\) the constraints (4) and (5) of the equations will be replaced by inequalities, for example, if \(M > N\), then some resources remain vacant, on the contrary, if \(N > M\), part of the orders will remain not assigned.

Let us consider how we can reduce a task posed in this work to the task on assignments and solve it by the Hungarian algorithm. We begin by considering the private acyclic class of this problem as described in example 1, where the duration of all assigned orders is obviously superior to the start time of loading of the latest order, i.e., none of the resources have time to perform more than one order. This class of problems is simply reduced to a problem of assignments, even if we assume that not every resource is suitable for any order in general, since this condition can be checked in the construction of the

### Table 4. The feasible assignment matrix in the cyclic task

|     | \(R_1\) | \(R_2\) | \(R_3\) |
|-----|--------|--------|--------|
| \(O_1\) | 1      |        |        |
| \(O_2\) |        | 1 9    |        |
| \(O_3\) | 9      |        |        |

### Table 5. A set of orders

|     | Loading     | Unloading | TOs | TOf |
|-----|-------------|-----------|-----|-----|
| \(O_1\) | Moscow      | Samara    | 1   | 2   |
| \(O_2\) | Samara      | Yekaterinburg | 23  | 28  |
| \(O_3\) | Yekaterinburg | Moscow   | 50  | 52  |

### Table 6. A set of resources

|     | Location       | TRf |
|-----|----------------|-----|
| \(R_1\) | Moscow        | 0   |
| \(R_2\) | Samara        | 0   |
| \(R_3\) | Yekaterinburg | 0   |

### Table 7. The feasible assignment matrix in the cyclic task

|     | \(R_1\) | \(R_2\) | \(R_3\) | \(O_1 R_1\) | \(O_1 R_2\) | \(O_1 R_3\) | \(O_2 R_1\) | \(O_2 R_2\) | \(O_2 R_3\) | \(O_3 R_1\) | \(O_3 R_2\) | \(O_3 R_3\) |
|-----|--------|--------|--------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| \(O_1\) | 1      |        |        |             |             |             |             |             |             |             |             |             |
| \(O_2\) |        | 1 9    | 1      |             |             |             |             |             |             |             |             |             |
| \(O_3\) | 9      |        |        |             |             |             |             |             |             |             |             |             |

The resulting feasible assignment matrix is given in Table 7. Locations of the resources \(R_1\), \(R_2\) and \(R_3\) are presented in matrix columns \(O_1 R_1, O_1 R_2, O_1 R_3, O_2 R_1, O_2 R_2, O_2 R_3\) after their possible execution of orders \(O_1\), \(O_2\), then \(O_2\) and \(O_3\), respectively.

### 3.2 The Search for Optimal Assignments

Having constructed the feasible assignment matrix, it can be assumed that the problem of finding the base program is reduced to the search for such a resource assignment for the orders in the matrix, in which the total time of the empty run for all resources for all assigned orders
matrix itself. When considering the more general cyclic class tasks described in example 2, where the location and time of each truck release will change in the course of the solution, it is also possible to bring it to the task assignment, on this case, however, the condition of resources and orders compatibility must be ignored, assuming that any resource fits any order. Taking this assumption, in order to reduce to the problem of assignments it is necessary to convert the feasible assignment matrix to a new species, giving up the consideration of the whole route of a specific truck to a specific order, so the matrix in Table 7 will be converted to the form in Table 8.

Here, in the matrix columns, which go after specific resources $R_1, R_2, R_3$ and $R_4$, there are already impersonal $RO_1, RO_2$ and $RO_3$, corresponding to unloading points of orders $O_1, O_2$ and $O_3$. In this step of forming the matrix one cannot determine which specific resource will arrive to load the next order, it will be determined only in the course of solving the problem. Therefore, it is impossible to consider the resources with the orders compatibility in these cases, but for the first columns, which provide specific resources it is possible to perform this check. When the problem of assignments will be solved for this matrix, the resource assigned to the previous order $O_i$ will be calculated for all assignments of the kind $RO_0$. It is also important to note that before solving the feasibility assignment matrix by the Hungarian algorithm, one must exclude empty rows and empty columns from it, as well as fill empty values in the remaining cells (the solution of which is unacceptable) by admittedly large numbers. If the solution is obtained in which any of the resources are assigned to an invalid order, the assignment can simply be deleted, leaving the order not assigned and a free resource.

As a result of solving the problem by the method described above, we will obtain the exact solution, but only on the assumption that any order fits any resource. To overcome this assumption and to obtain an exact solution based on this condition is not possible. However, as practice shows, for industrial multi-agent planning system$^9$, it is not necessary, because then the resulting initial plan will be modified according to real-time events where the multi-agent approach$^9$ will be used for the solution. In particular, if the agent of the order has received the initial assignment to a trailer truck not suitable for it, it will adaptively re-plan for the other suitable truck, taking into account possible changes in the plan due to the real-time events.

### 4. Experimental Results

We conducted several experiments to test the described methods using real data from our client companies. Each of them uses a multi-agent system$^9$. The first company ProLogics$^{13}$ has 140 resources and about 25 new orders a day. The second company MONOPOLY$^{14}$ has 300 resources and about 76 new orders a day. The third company LORRY$^{15}$ has 680 resources, and about 240 new orders a day. The purpose of the experiments was to compare the multi-agent method, based on the work in real time, to the Hungarian algorithm in the problem of constructing the initial plan. In the course of experiments, the launch of the multi-agent system was carried out$^9$ to get each customer’s data. Immediately after the initial plan had been created, the system stopped. Based on the same data sets the matrices were formed to solve the assignment problem, which were then solved by one of the implementations of the Hungarian algorithm$^{16}$. For a more detailed picture the experimental data have been received from the real data of the above mentioned 3 companies at different time during the working month. They were grouped due to the density (% of blank cells) in the assignment matrix, which generally ranges from 5% to 95%. The experimental results are shown in Table 9. It can be seen that the Hungarian algorithm gives better results and is less time consuming in almost every experiment. The experiments were performed on a workstation with a processor 3.4 GHz Intel Core i7-4770 and 8 GB of RAM running Windows 8.1.

### 5. Conclusion

In this paper we apply the methods of linear programming for a special task of building the initial VRP schedules for trucks on the orders of a large transport company, carrying out inter-regional transportation according to the FTL scheme. We propose the method of finding the exact solution with the stipulated assumption and allowing for the minimum required set of criteria, the knowledge of which were based on the experience of the
Table 9. The results of experiments

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

| Matrix to solve the assignment problem | Multiagent method | The Hungarian method |
|---------------------------------------|-------------------|----------------------|
| KPI (1) | Working time, s | KPI (1) | Working time, s |
| 35 | 0.014 | 34 | 0.001 |
| 33 | 0.016 | 32 | 0.001 |
| 43 | 0.012 | 41 | 0.001 |
| 49 | 0.009 | 47 | 0.001 |
| 73 | 0.005 | 68 | 0.001 |
| 147 | 0.004 | 144 | 0.001 |
| 341 | 0.003 | 339 | 0.001 |
| 82 | 0.6 | 77 | 0.001 |
| 86 | 0.5 | 81 | 0.001 |
| 93 | 0.4 | 92 | 0.003 |
| 111 | 0.3 | 106 | 0.001 |
| 127 | 0.2 | 123 | 0.001 |
| 208 | 0.12 | 199 | 0.003 |
| 483 | 0.052 | 472 | 0.001 |
| 242 | 39.91 | 240 | 0.011 |
| 240 | 34.99 | 240 | 0.014 |
| 245 | 29.91 | 241 | 0.01 |
| 255 | 21.8 | 246 | 0.01 |
| 285 | 13.29 | 273 | 0.008 |
| 431 | 6.97 | 414 | 0.007 |
| 980 | 2.66 | 934 | 0.006 |

implementation of industrial multi-agent system. The limited applicability of this method in the transition to the real-time planning were shown, because even when used for the static cyclic task it is not possible to take into account the matching of orders and truck trailers’ parameters. Thus, we can conclude that at the transition to the real-time planning, when the system encounters perturbations associated with the emergence of new orders or the cancellation/change of the existing orders, using only traditional methods is not sufficient, but the combination of classical and non-classical multi-agent approach will give good solution which can be applied in practice.

6. Acknowledgement

This work was financially supported by the Ministry of Education and Science of the Russian Federation (contract № 14.576.21.0014, NIR RFMEFT57614X0014 unique identifier).

7. References

1. Dantzig GB, Ramser JH. The truck dispatching problem. management science. 1959; 6(1):80–91.
2. Browne M, McKinnon A, Cullinane Sh. Green logistics, Anthony Whiting, editor; Kogan Page; 2010.
3. Bertazzi L, Savelsbergh M, Speranza MG. Inventory routing. Bruce Golden, S. Raghavan, & E. Wasil, editors. The Vehicle Routing Problem: Latest Advances and New Challenges; New York: Springer Science+Business Media; 2008. doi:10.1007/978-0-387-77778-8.
4. De Jaegere N, Defraeye M, Van Nieuwenhuyse, Leuven KU. The vehicle routing problem: state of the art classification and review [Internet]. [Cited 2015 Jan 14]. Available from: https://lirias.kuleuven.be/bitstream/123456789/457452/1/ KBI_1415.pdf.
5. Granichin O, Skobelev P, Lada A, Mayorov I, Tsarev A. 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); Barcelona, Spain: SciTePress, Portugal; 2013 Feb 15–18. p. 244–49.
6. Hashimoto H, Ibaraki T, Imahori Sh, Yagiura M. The vehicle routing problem with flexible time windows and traveling times. Discrete Applied Mathematics. 2006 Nov 1; 154(16):2271–90.
7. Vittich VA, Skobelev PO. Multi-agent interaction models for building needs networking and opportunities in open systems; Moscow: Automation and Remote Control; 2003. p. 177–185.
8. Skobelev PO. Multi-agent technology in industrial applications: for the 20th anniversary of the Samara school of multi-agent systems. Mechatronics, Automation, Control. 2010; 12:33–46.
9. Ivaschenko A, Lada A, Mayorov I, Skobelev P, Tsarev A. The analysis of efficiency of application of the multi-regional traffic control system in real time. Materials of the 4th multiconference on the problems of control of MCPM-2011; S. Divnomorskoe, Gelendzhik, Russia. Taganrog: Publishing House of TIT SFU. 2011 Oct 3–8. p. 353–356.
10. Vaisblat AV, Diyazitdinova AR, Ivashchenko AV, Skobelev PO, Tsarev AV. Organization of interactive communication in multi-agent management system in transportation and shipping company. Proceedings of the XII International Conference Problems of control and simulation of complex systems; Samara: Samara Scientific Center RAS; 2010. p. 620–8.
11. Malkovskiy NV, Granichin ON, Amelin KS Allocation of resources in the context of multi-agent systems. Proceedings of XII All-Russian Conference on Control (VSPU 2014); Russia: Moscow, Institute of Control Sciences; 2014 Jun 16–19. p. 9003–13.
12. Kuhn HW. The Hungarian method for the assignment problem. Naval Research Logistics Quarterly. 1955; 2(1–2): 83–97.
13. PROLOGICS [Internet]. [Cited 2015 Jan 14]. Available from: http://eng.prologics.ru/.
14. MONOPOLY [Internet]. [Cited 2015 Jan 14]. Available from: http://monopoly.su/.
15. LORRY [Internet]. [Cited 2015 Jan 14]. Available from: http://en.lorry-ural.ru/.
16. GitHub [Internet]. [Cited 2015 Jan 14]. Available from: https://github.com/catbaxter/algorithms/blob/master/hungarian/hungarian.cpp.