Comparing Different Approaches on the Door Assignment Problem in LTL-Terminals

Boris Naujoks[1], Annette Chmielewski[2]

[1] Log!n GmbH, Wilhelmstr. 45
58332 Schwelm, Germany
boris.naujoks@login-online.de
[2] Chair for Transportation and Logistics, TU Dortmund University,
44221 Dortmund, Germany
chmielewski@vsl.mb.tu-dortmund.de

Abstract. The work at hand yields two different ways to address the assignment of inbound and outbound doors in less-than-truckload terminals. The considered optimization methods stem from two different scientific fields, which makes the comparison of the techniques a very interesting topic. The first solution approach origins from the field of discrete mathematics. For this purpose, the logistical optimization task is modeled as a time-discrete multi-commodity flow problem with side constraints. Based on this model, a decomposition approach and a modified column generation approach are developed. The second considered optimization method is an evolutionary multi-objective optimization algorithm (EMOA). This approach is able to handle different optimization goals in parallel. Both algorithms are applied to ten test scenarios yielding different numbers of tours, doors, loading areas, and affected relations.

Keywords. Door Assignment Problem, Column Generation Approach, Multi-objective evolutionary algorithm approach

1 Introduction

In less-than-truckload (LTL) terminals, arriving trucks have to be assigned to inbound doors and to suitable time slots for unloading. Simultaneously, waiting trucks have to be allocated to outbound doors. During a limited number of hours, shipments from all incoming trucks are unloaded, sorted according to their destination/relation[1], transported to the right outbound door and loaded on the outgoing truck. The first and most important optimization goal is to minimize the total distance for transshipping the units. This leads to a meaningful reduction in operational costs. The second and minor aim is to minimize the waiting time for each truck.

[1] The term "relation" is used throughout the text as an equivalent for a destination. It origins from the German logistics vocabulary that uses this term to specify a certain transport offered between a source and a sink.

Dagstuhl Seminar Proceedings 09261
Models and Algorithms for Optimization in Logistics
http://drops.dagstuhl.de/opus/volltexte/2009/2187

The transportation request of one customer in the LTL-sector normally comprises between 1 and 10 pallets. These do not suffice to fill the load area of a whole truck (up to 33 pallets). Therefore, the transportation of LTL-shipments is organised via transportation networks:

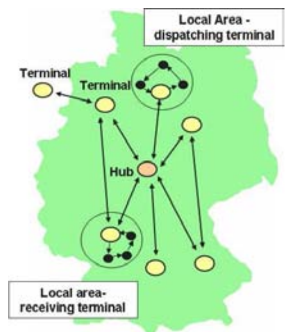

Fig. 1. Elements of a transportation Network in the LTL sector

These consist of several freight forwarding terminals which are spread evenly over the region or country (Germany: between 40 an 45 terminals), which are connected by long distance relations. A local region is assigned to each terminal. The core element of a terminal is the transshipment building with several doors (Germany: depending on the size of the terminal between 40 and 100 doors). The doors can be separated in inbound and outbound doors, which are used for loading (outbound doors) and unloading (inbound doors) trucks.

During a day of operation, all shipments coming from one local area are collected on local tours. These tours arrive according to a certain timetable with earliest arrival time at the appropriate freight forwarding terminal. Some trucks will be needed for further transport services and therefore have to leave the terminal at a certain point of time or at least as soon as possible.

While unloading, all shipments are first placed on a common buffer area. The different shipments on the buffer area are then consolidated according to their long distance relation in the underlying transportation network. In a subsequent process, the shipments are transported inside the building to those doors where the corresponding relation is loaded. A special buffer area for loading is assigned to each outbound door. Finally, all shipments in front of one door are transported inside the truck for the long distance transport.

The optimization task **door assignment in LTL-terminals** consists of different levels with different time conditions:

1. The doors have to be separated in inbound and outbound doors.
2. The relations in the underlying transportation network have to be assigned to the outbound doors.

3. An optimal allocation of resources is to be found (workers, scanners, forklifts, etc.).
4. The waiting times for the trucks are to be minimized.

The remainder of the paper is organized as follows: The subsequent section describes a complex optimization model, which is later on used in the decomposition and column generation approach for optimizing the described optimization task. After that, an evolutionary algorithm approach is presented that treats the task with a much easier inner model. Results received with both approach on a common set of ten test scenarios are presented in the final section.

2 Optimization model and analysis

A time-discrete multi-commodity network structure was implemented to model the logistical task:

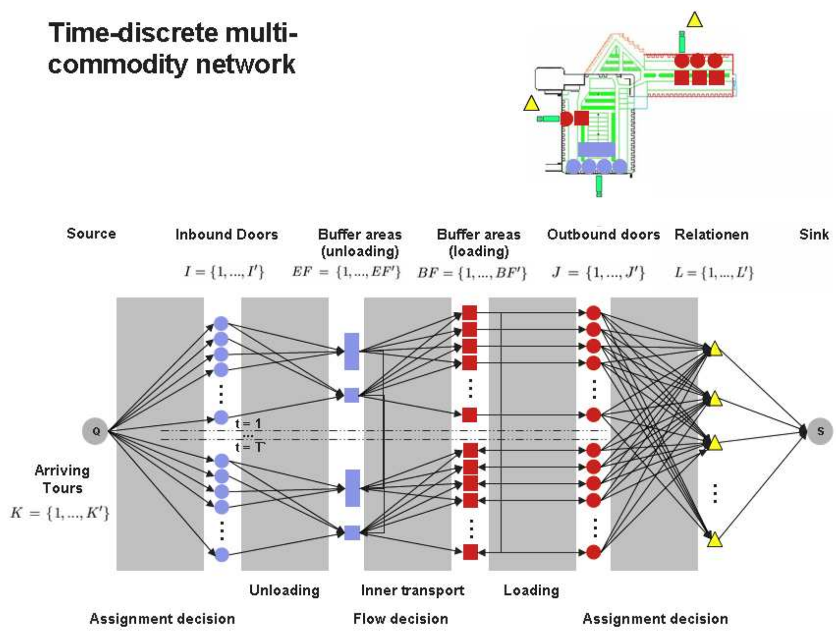

Fig. 2. Time-discrete multicommodity-network structure

The layout elements (inbound doors, buffer areas for unloading and for loading, outbound doors) are represented by separate node layers. Inner transport activities are modeled by arcs, connecting the different node elements. If a certain transport is not allowed, the according arc is deleted. For indicating the distances in meter in the LTL-terminal, costs are assigned to each arc.

The different incoming transport vehicles (= tours) and their load is modeled by commodities flowing on the arcs. Therefore, a common source is introduced where the k tours are arriving. The last node layer is used to model the l outgoing vehicles (=relations) in the system, to which the different goods have to be transported to. To indicate the time, several time slices are introduced to the model and the resulting network structure is doubled for each time slice t.

If there is a positive flow of commodities of one incoming tour k on a certain arc of the arc layer one, connecting the source with the inbound door i, this indicates that the according tour has been assigned to the inbound door i. To assure that the unloading process is proceeded without any interruptions at just one door, the following restrictions are implemented:

$$\sum_{i=1}^{I'} \delta_k^i = 1, \quad \text{and} \quad \gamma_t^k + \gamma_{t+i}^k - \sum_{j=t+1}^{t+i-1} \gamma_j^k \leq 1, \tag{1}$$

$\forall\, k \in K. \forall\, i \in \{2, ..., T' - 1\}, \forall\, t \in \{1, ..., T' - i\}$. The binary variables δ_k^i are used to check if a positive flow has been noted on the arc connecting the source node with inbound door i. This variable can just take value 1 for one inboud door i. In the second restriction, certain patterns are described, that are not allowed. For example the unloading pattern 1|0|0|1|...|1 is not allowed as there is an interruption of the unloading process in time slices 2 and 3.

If there is a positive flow of any commodity on an arc connecting outbound door j and relation node l, this means that the relation l is assigned to the outbound door j. Here, it is also just possible to load a relation at one door. In addition, the loading doors can just be occupied by one relation during the whole operation period to avoid mixing loads for different long distance relations:

$$\sum_{l=1}^{L'} \theta_l^j \leq 1, \qquad \forall\, j \in J. \tag{2}$$

Several additional restrictions exist for describing rules for unloading, inner transport and loading in the arc layers $2 - 4$ (e.g. time table of tours, resource and buffer area capacities, demand of relations for commodities of one tour) which will not be explained in detail.

The main aim of the logistical task is to minimize the inner transport distances when transshipping and to minimize the waiting times of the trucks. Therefore, the following objective function is introduced:

$$\begin{aligned} min \ \sum_{k=1}^{K'} \sum_{t=1}^{T'} \Bigg(& \sum_{i \in A_{2t}} B^t * x_{arc_i}^k * \left(\frac{{arc_i}^{length} * c_{unl}}{g_{unl}} \right) + \\ & \sum_{i \in A_{3t}} x_{arc_i}^k * \left(\frac{{arc_i}^{length} * c_{tra}}{g_{tra}} \right) + \sum_{i \in A_{4t}} x_{arc_i}^k * \left(\frac{{arc_i}^{length} * c_{loa}}{g_{loa}} \right) \Bigg) \end{aligned} \tag{3}$$

The different blocks represent the resulting transport distances or working minutes produced by unloading transports, inner transports or loading transports. Actions in late time slices are punished by introducing an increasing factor on the first block.

The resulting mathematical model is NP-hard as it incorporates an integer multi-commodity flow problem and a quadratic assignment problem. Tests by CPLEX have shown, that for larger problem instances, no solution was found in reasonable time and solution quality (neighborhood to theoretical optimum). Therefore, in the next section, a solution approach using the concepts of column generation and decomposition is introduced and applied to different test scenarios.

3 Decomposition and column generation approach

The idea of decomposing the problem leads to the introduction of routings for each tour k in the system:

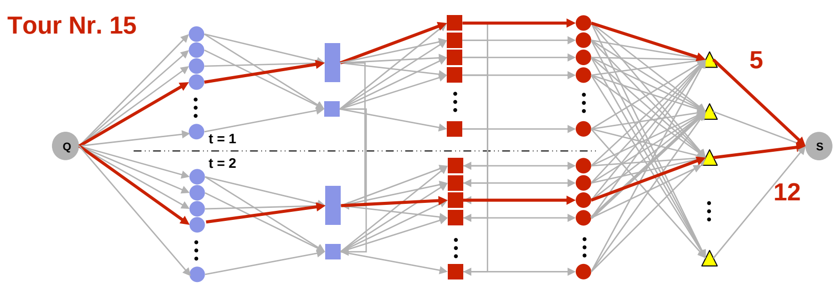

Fig. 3. Decomposing the problem by introducing routings for commodities of one tour

A feasible routing - as indicated in the figure - represents the flow of all commodities of a certain tour through the time-discrete multi-commodity network implemented in section 2. All restrictions, concerning the assignment decisions, the loading and unloading processes, the inner transport, the time table and the resource capacities in the system are valid again. But as the solution is chosen for just one tour, the model has a lower complexity.

The column generation concept allows to generate feasible routings individually for each tour and to chose different routings from a pool of feasible routings by using a binary choice model. This model consists of binary variables indicating if a certain routing for a tour has been chosen or not. Additional constraints are necessary to check when all routings are imposed in the same network, capacity restrictions on the arcs for example are still not violated.

The relaxed choice model is used as master problem in the column generation process. The pricing problem for creating new good routings for the different tours is identical to the mathematical model from the section above, just being reduced to one tour. As long as new routings with negative reduced costs exist, the column generation process is continued and dual variables in the master problem are updated. When the process ends, a branch-and-bound algorithm is proceeded by using all routings having been created and identified so far by the column generation process before.

When comparing the new decomposition and column generation process with the performance of CPLEX, feasible solutions are found earlier and the quality of the solutions, i.e. the objective function value (which should be minimized) is much lower. The following figure visualizes this aspect for two different test scenarios:

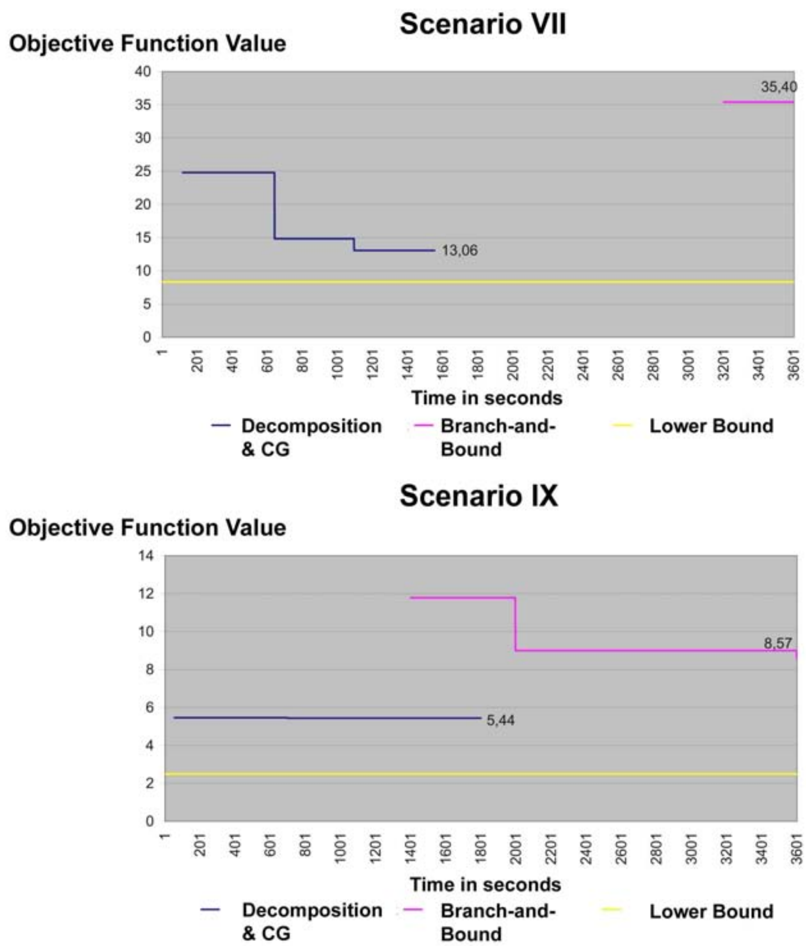

Fig. 4. Comparing speed and soluton quality of the new approach with CPLEX

4 Evolutionary algorithm approach

The evolutionary algorithm chosen for the door-assignment problem at hand is a multi-objective EA to handle both objectives (transshipment distances and waiting times) of the application at the same time. Here, the concept of Pareto dominance comes into play. A solution one is said to dominate ($\prec$) a solution two, if and only if (iff) all components of the fitness function f of solution one are not greater that the corresponding components of solution two and really smaller in at least one component:

$$x \prec y :\Longleftrightarrow \forall i : f_i(x) \leq f_i(y) \quad \wedge \\ \exists j : f_j(x) < f_j(y)$$

The set of non-dominated solutions is called the Pareto set of solutions while the corresponding pictures under function f are called the Pareto front.

To define the objectives, the representation of the individuals of the EA have to be introduced. A candidate solution I implements an array of lists. Consider the array of lists for the ith solution $I_i = [G_{i1}, G_{i2}, \dots, G_{im}]$, where m denotes the number of doors. Each list $G_{ij} = [K_{ij1}, \dots, K_{ijk}]$ represents one door. These list entries represent tours. Each tour K_{ijl} consists of an array with four integer values:

$$K_{ijl} = [\text{tourNumber}, \text{doorNumber}, \text{startTime}, \text{endTime}].$$

To accelerate the function evaluation, two additional arrays to store times and doors were implemented.

The first objective function f_1 describes the distances inside the transshipment building each pallet has to be transported:

$$f_1(I_i) := \sum_{j=1}^{m} \sum_{l=1}^{k} \sum_{r=1}^{s} d(G_{ij}(P_{ijlr}), G_{id}(P_{ijlr})),$$

with P_{ijlr} being r-th pallet of tour K_{ijl} at door G_{ij} with destination door G_{id} ($d \in \{1, \dots, m\}$). The function d describes the distance inside the transshipment building from one door to another. Here, the resources needed for the operations inside the transshipment buildings are ignored.

The second objective function f_2 displays the waiting time for each truck:

$$f_2(I_i) := \sum_{j=1}^{m} \sum_{l=1}^{k} t_w(K_{ijl}),$$

with function $t_w(K_{ijl})$ being the difference between the point of time the unloading of truck of the corresponding tour is started and the arrival time at the transshipment building. For reasons of simplicity, we neglected a detailed description of all constraints that can be derived from the problem description. Of course, all constraints are represented in our algorithm for the task.

The rules of an (1+1)-EA for MCO were published earlier by Bartz-Beielstein et al. [1]. If the new individual is not dominated by any member of the archive, the new individual becomes a member of the archive and the parent for the next generation. While updating the archive, dominated archive members are deleted. If the individual is dominated by a member of the archive, the old parent individual proceeds to the next generation.

In addition to the first and simple $(1+1)$-EA for MCO, a multi-membered $(\mu+\lambda)$-EA for MCO was presented by Chmielewski et al. [2]. Here, a population of μ individuals is to be initialized before the evolution loop is entered. Within a simple selection procedure, all given individuals are sorted according to the rank assigned by *non-dominated sorting* [3]. Afterwards, the next parent population is filled with μ individuals featuring minimum rank. If the set of individuals with the last processed rank does not fit into the population completely, the population is filled with individuals of this last rank randomly. This random choice of individuals to fill the coming parent population is the main difference to common and ore sophisticated approaches like NSGA-II [4,3] or SMS-EMOA [5].

5 Comparison

The two approaches are applied to ten different test scenarios. The logistical dimension of the scenarios is increased successively. The testing phase starts with quite unrealistic scenarios of 14 and 25 doors in total and less than 20 tours and 10 relations to get first experiences concerning the solution quality and solution time. Middle-sized terminals in Germany have between 42 and 64 doors (scenarios V - VIII). The layout information (distances, areas, etc.) of scenarios IX and X have been taken from real freight forwarding companies belonging to the class of large terminals in Germany:

Scenario	Logistical System							
	Time Slices	Inbound Doors	Outbound Doors	Total number of doors	Areas Unloading	Areas Loading	Relations	Tours
I	6	8	6	14	1	6	6	6
II	6	4	10	14	1	10	6	12
III	10	11	14	25	1	14	10	20
IV	6	15	10	25	1	10	10	40
V	6	17	25	42	2	25	20	40
VI	8	22	20	42	2	20	20	50
VII	6	25	39	64	2	39	25	40
VIII	8	10	54	64	1	54	20	40
IX	4	40	60	100	2	60	30	30
X	6	20	60	80	2	60	40	80

Fig. 5. Logistical dimension of the test scenarios

An amount of 300.000 fitness function evaluations have been spend for the EA approach. The resulting calculation time of one run of the corresponding EA was approx. one second. In case of the application of the EA, this is constant for all considered scenarios, because the driving force for the calculation time are the genetic operators. These have to be applied for a constant number of times. For each scenario, 10 optimization runs have been performed and the best distance values were considered next to the corresponding waiting times (see [2]).

Comparing the results received, it must be admitted that the evolutionary algorithms perform rather bad on these benchmarks with respect to distances. Although being quite fast within real calculation time, the quality of the received distances is not as good as the ones received with the approach from discrete mathematics. Only the first two scenarios are exceptions, but these are unrealistic scenarios with only small numbers of doors and tours. In all other cases, the column generation approach performs much better than each of the EA approaches.

Nevertheless, it is remarkable that the waiting times are much better for the EA approaches. This trade-off origins from the multi-objective approach followed by the EA in contrast to the single-objective one of the column generation approach. To improve the distance values received with the EA a single-objective EA considering only on the distance values could be implemented. But this would mean to neglect the waiting times again like in the column generation approach.

Comparing both EA approaches, the $(1 + 1)$-EA clearly outperforms the multi-membered approach with respect to the considered distance. Considering the waiting times, the statement from above can be repeated: better distances mean worth waiting times. This was observed comparing the column generation and the EA approaches as well and gives clear evidence for the trade-off between the two objectives. Multi-membered EA performing worth than the simple $(1+1)$-EA may be due to the stochastic approach considered for selection. The multi-membered approach may be improved using some better secondary ranking criterion.

References

1. Bartz-Beielstein, T., Chmielewski, A., Janas, M., Naujoks, B., Scheffermann, R.: Optimizing door assignment in LTL-terminals by evolutionary multiobjective algorithms. In Fogel, D.B., et al., eds.: Congress on Evolutionary Computation (CEC 2006) within IEEE World Congress on Computational Intelligence (WCCI 2006), Piscataway NJ, IEEE Press (2006)
2. Chmielewski, A., Naujoks, B., Janas, M., Clausen, U.: Optimizing the Door Assignment in LTL-Termials. Transportation Science **43** (2009) 198–210
3. Deb, K., Agrawal, S., Pratab, A., Meyarivan, T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. In Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J., Schwefel, H.P., eds.: Proceedings of the Parallel Problem Solving from Nature VI Conference, Paris, France, Springer. Lecture Notes in Computer Science No. 1917 (2000) 849–858
4. Deb, K., Agrawal, S., Pratab, A., Meyarivan, T.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. KanGAL report 200001, Indian Institute of Technology, Kanpur, India (2000)
5. Emmerich, M., Beume, N., Naujoks, B.: An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. In Coello Coello, C.A., et al., eds.: Evolutionary Multi-Criterion Optimization. Third International Conference, EMO 2005. Volume 3410 of Lecture Notes in Computer Science., Springer, Berlin (2005) 62–76