Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 47 (2020) 291–298

Transportation Research Procedia

www.elsevier.com/locate/procedia

22nd EURO Working Group on Transportation Meeting, EWGT 2019, 18-20 September 2019, Barcelona, Spain

Models and algorithms for network design in urban freight distribution systems

Lucas Rebouças Guimarães[a]*, Bruno de Athayde Prata[b]. Jorge Pinho de Sousa[c]

[a]Instituto Federal do Espírito Santo, lucas.guimaraes@ifes.edu.br, Vila Velha, Brazil
[b]Universidade Federal do Ceará, baprata@ufc.br, Fortaleza, Brazil
[c]Faculdade de Engenharia da Universidade do Porto / INESC TEC, jorge.p.sousa@inesctec.pt, Porto, Portugal

Abstract

Central areas of large cities offer in general many advantages to their inhabitants. Typically, a large number of products, services, and opportunities are available in those urban zones, thus increasing life quality. Unfortunately, these benefits are associated with increasing transportation activities that can cause serious problems, such as traffic congestion, excessive energy consumption, and pollution. This paper aims at presenting a new transport system that consists of transporting freight in long-haul passenger vehicles. Two mixed integer mathematical programming models are presented: one for total cost minimization and the other for the travel time minimization. The problem under study was considered as a multi-commodity network flow problem with time windows, multi transport-lines, and multiple vehicles. Three heuristics based on mixed integer programming (MIP) were designed to solve it: size reduction, LP-and-fix, and a combination of these two procedures. The proposed approaches were validated in a case study designed around the intercity passenger transport system, in Ceará, Northeast of Brazil. Several operational scenarios were evaluated, taking into account the available freight capacities. The developed MIP heuristics produced high-quality solutions, in reasonable computational times, with the LP-and-Fix algorithm outperforming the other approaches.

Keywords: City logistics; multi-commodity network problem; mixed integer linear programming.

* Corresponding author. Tel.: +55-27-99901-1150; fax: +55-27-3149-0700.
E-mail address: lucas.guimaraes@ifes.edu.br

10.1016/j.trpro.2020.03.101

1. Introduction

City logistics is permanently posing new challenges to government and local authorities, different types of companies, carriers, and citizens. In the vast literature in this area, there are numerous reports on practices in city logistics from all around the world, trying to identify new trends, gaps, and perspectives, in an attempt to develop and structure a set of general theoretical and practical approaches. These approaches are important to understand the evolution of city logistics as a scientific field, and encompass review studies, classification schemes, and exploratory approaches. Studies include: classification reviews (Anand et al 2012, Benjelloun et al, 2010, and Ambrosini et al, 2002); trends and evolution perspectives (Benjelloun et al. 2008, and Taniguchi et al, 2010); and concept development approaches (Ehmke, 2012, Thompson et al. 2001, and Taniguchi et al, 2001).

In this context, multi-commodity flow problems consist in moving different types of goods in a network, from a set of supplier nodes to a set of customer nodes, aiming to optimize some network system attributes (Kennington, 1978). The approach developed in this work is based on a new multi-commodity network flow model, with time windows, multi-transport lines, and multiple vehicles. The associated business model consists in transporting freight in long-haul passenger vehicles (buses), as a way to increase transport efficiency and to reduce operational costs and times (two objective functions solved separately), while decreasing the environmental and social impacts resulting from transport traffic.

Several approaches were developed in the literature to deal with the basic multi-commodity network design problem and its variants. Kim et al. (2011) consider the fixed-charge capacitated multi-commodity network design problem with turn penalties. Kleeman et al. (2012) developed a multi-objective evolutionary algorithm to solve the multi-commodity capacitated network design problem. Masri et al. (2015) present a multi-start variable neighborhood search to solve a capacitated single path multi-commodity flow problem. Wei et al. (2016) worked on the multi-source single-path multi-commodity network flow problem and developed an effective simulated annealing based metaheuristic to minimize the network total transportation cost. Lin (2017) presents a class of multi-commodity flow problems with commodity compatibility relations among commodities used at each network node.

The problem under study can be represented by a graph with a set of nodes (passenger transport terminals) and arcs (passenger transport lines). Decisions are related to the number of goods to be moved between the nodes, by using the arcs in such a way that total costs or operation times are minimized. Some initial surveys show that the passenger vehicles baggage compartments generally have significant empty spaces. Therefore, the transport efficiency can increase by filling these spaces (possibly through optimization models based on the bin packing problem or similar models) with loads that have the same destination of the vehicle. This feature will be the basic principle for the design of the solutions proposed in this work. However, it should be noted that not all kinds of products can use this transport solution. Goods that do not need special cares, not offering particular risks and having small dimensions are surely candidates for the services proposed here.

The system developed in this paper will hopefully directly contribute to a decrease of the transport operations costs and times of some stakeholders (shippers and carriers) and, indirectly, to a traffic reduction in intercity highways and inside cities, with positive environmental and social impacts.

As multicommodity network design problems are NP-hard, the resulting model is also expected to be NP-hard, making it impossible in practice to solve it to optimality for large instances by using an exact method. Therefore, in order to guarantee acceptable computational times, three heuristic approaches were developed: a reduction heuristic called "size reduction" (Fanjul-Peyro and Ruiz 2011); a relaxation heuristic known in the literature as "LP-and-fix" (Toledo et al, 2015) and a combination of these two heuristics. These approaches were developed using the Concert technology for C++, and the CPLEX optimization environment. The first computational results with a set of realistic instances are quite promising.

Some results of this research work will hopefully have a considerable practical impact. In particular: a transport system network was designed, integrating urban freight and passengers transport; a mathematical programming model was developed along with techniques for solving it; and the proposed approach was applied to some real-world problems, and assessed in real case studies. As far as we know, no other research initiative has adequately dealt with these concerns.

The remainder of this paper is structured in five sections as follows: section 2 presents the mathematical model of the multi-commodity network flow problem with multi-transport lines, multiple vehicles and time windows, while section 3 presents the solution approaches for the proposed network flow problem. Section 4 presents the case study. Finally, section 5 presents a set of conclusions and some suggestions for future work.

2. Mathematical model

In general terms, this problem can be represented by the following mathematical model, that can be viewed as a mono-objective multi-commodity network flow problem with multiple transport lines and time windows. Two objective functions that will be solved separately are considered in this work. In this approach, the extensions of the original model are performed through a graph $G = (V, E)$, that accommodates the following problem features.

Each edge E has: a variable cost related to the specific transported product amount; a lower bound and an upper bound for volume per freight; a departure time, a travel duration and an arrival time, for each vehicle from the transport line; and a capacity related to each vehicle from the different transport lines. Each vertex V has: an offer/demand freight value for each kind of product. Some details of the problem data structures are presented here, to facilitate the understanding of the model parameters.

The *edges* structure contains information related to the following attributes: origin transport line (t); destination transport line (o); origin vehicle (v); destination vehicle (q); commodity (k); origin node (i); destination node (j); cost (c); upper capacity bound (u); lower capacity bound (l); departure time (p); travel duration (h); and arrival time (d). All the possibilities of movements will be represented by additional edges (as it is also the case of transshipments).

The *capacity* structure stores the information related to the following attributes: destination transport line (o); destination vehicle (q); origin node (i); destination node (j); capacity (a). It should be noted that the capacities are shared among the commodities, so that (a) represents the total amount of cargo that can be transported by the vehicle, while (u), in the edges structure, represents the total amount of each commodity that can be transported in that arc. However, it should be noted that both data structure (edges and capacity) are associated with the set E.

Finally, the *vertex* structure stores three attributes: commodity (k); node (i); offer / demand (b). The offer and demand nodes have different signs: "+" means offer, and "-" means demand. The data structure vertex is associated with the set V.

The model parameters are given by: $c_{tovqkij}$ – unit variable cost from arc (i,j) to transport each product k from the transport line t, vehicle v to transport line o, vehicle q (the flow starts on line t, vehicle v and continues in line o, vehicle q – this rule is valid for all parameters and variables with the index *tovqkij*); a_{oqi} – the capacity of arc (i,j) and transport line *o, vehicle* q; b_{ik} – demand from node i by product k; $l_{tovqkij}$ – lower bound of flow from the arc (i,j) to each product k from the transport line t, vehicle v to transport line o, vehicle q; $u_{tovqkij}$ – upper bound of flow from the arc (i,j) to each product k from the transport line t, vehicle v to transport line o, vehicle q; $p_{tovqkij}$ – departure time for arc (i,j) to each product k from the transport line t, vehicle v to transport line o, vehicle q; $h_{tovqkij}$ – duration time for arc (i,j) to each product k from the transport line t, vehicle v to transport line o, vehicle q; $d_{tovqkij}$ – arrival time for arc (i,j) to each product k from the transport line t, vehicle v to transport line o, vehicle q.

The decision variables of the model are: $y_{tovqkij}$ – flow from the arc (i,j), to each product k from the transport line t, vehicle v to transport line o, vehicle q; $x_{tovqkij}$ – indicates whether the arc (i,j), to each product k from the transport line t, vehicle v to transport line o, vehicle q is used; $w_{tovqkij}$ – indicates the total displacement time of the freight along an arc (i,j), to each product k from the transport line t, vehicle v to transport line o, vehicle q.

$$\min \sum_{(t,o,v,q,k,i,j)\in E} c_{tovqkij}\, y_{tovqkij} \tag{1}$$

$$\min \sum_{(t,o,v,q,k,i,j)\in E} w_{tovqkij} \tag{2}$$

subject to:

$$\sum_{j:(t,o,v,q,k,i,j)\in E} y_{tovqkij} - \sum_{j:(t,o,v,q,k,i,j)\in E} y_{tovqkji} = b_{ik} \text{ ,} \forall\, i,k \ \in V \tag{3}$$

$$\sum_{j:(t,o,v,q,k,i,j)\in E} y_{tovqkij} \le a_{oqij} \text{ ,} \forall\, (o,q,i,j) \in E \tag{4}$$

$$l_{tovqkij} \le y_{tovqkij} \le u_{tovqkij} \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{5}$$

$$y_{tovqkij} \le \left(\sum_{(i,k)\in V} b_{ik} \right) x_{tovqkij} \text{ ,} \forall\, (t,o,v,q,k,i,j) \ \in E \tag{6}$$

$$w_{tovqkij} = p_{tovqkij} + h_{tovqkij} \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{7}$$

$$x_{tovqkij} \le y_{tovqkij} \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{8}$$

$$x_{tovqkij} \times d_{tovqkij} \le w_{tovqkij} \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{9}$$

$$y_{tovqkij} \ge 0 \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{10}$$

$$w_{tovqkij} \ge 0 \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{11}$$

$$x_{tovqkij} \in \{0,1\} \text{ ,} \forall\, (t,o,v,q,k,i,j) \in E \tag{12}$$

$$a_{oqij} < \sum_{(t,o,v,q,k,i,j)\in E} \left(u_{tovqkij} - l_{tovqkij} \right) \text{ ,} \forall\, (o,q,i,j) \in E \tag{13}$$

$$\sum_{(i,k)\in V} b_{ik} = 0 \tag{14}$$

Objective function (1) represents the network operational costs, while objective function (2) represents the travel times, both to be minimized. Equations (3) guarantee the flow conservation on the nodes. Equations (4) guarantee that the flow in the arcs does not exceed their capacity. Equations (5) define the upper and lower bounds on the arc flows. Expressions (6) guarantee the inexistence of flow in arcs where vehicle time windows are incompatible. Equations (7) define the time spent in the arcs, while equations (8) force binary variable to have the value "1" when the arc is being used. Expressions (9) guarantee the time compatibility between vehicles from different transport lines. Expressions (10) and (11) express the non-negativeness of the flow and time variables, while binary variables (12) are used to define the time compatibility of each vehicle. Expressions (13) consider the capacities of vehicles for each arc, and expressions (14) guarantee that the total offer will be sufficient to satisfy all the demand.

3. Resolution Approach

Three heuristic algorithms were developed to solve this multi-commodity network flow problem with time windows, multi-transport lines and multiple vehicles. A preliminary step of these algorithms consists in reading a data structure composed by the three matrices described in the previous section: edges, capacity, and vertex. In the next sections the three heuristics developed in this work are described.

3.1. Size reduction heuristic

According to Fanjul-Peyro and Ruiz (2011), a reduction heuristic explores the topology of the problem and reduces some problem attributes without making its solution infeasible. The authors obtained good results applying a set of metaheuristics based on a size-reduction of the original assignment in the unrelated parallel machines scheduling problem. The method produced solutions of very good quality, in a short amount of time.

The developed approach makes reductions analyzing the pairs of origin (i) / destination (j) nodes that are repeated throughout the data structure. In this way, the procedure does not impair the operation of the network. The number of nodes analyzed by the algorithm will depend on a pre-defined reduction "percentage". The reduction criteria are based on first analyzing costs and then time. The algorithm traverses all the edges of the network and creates a list containing the edges with repeated pairs (i, j) of origin and destination nodes, so that the edges are ordered decreasingly in terms of costs and time. The reduction is made using the given percentage, starting by the edges in the top of the list.

Therefore, the size reduction heuristic will list all the origin and destination nodes that are repeated (creating a list of repetitions that stores the edges positions in the matrix edges) over the network, and eliminating from the data structure (specifically from the edges matrix) those that have the largest costs and times. Thus, the possibility of making the network operation unfeasible by the reduction of its edges decreases. In this way, the procedure allows larger reduction percentages without making the problem unfeasible.

3.2. LP-and-fix heuristic

According to Toledo et al. (2015), heuristics based on mathematical programming are, in general, an efficient way to solve mixed integer problems. The "relaxation and fix" heuristic described in this section has, as the above procedure, a very simple operating mechanism. It consists in relaxing some integer or binary decision variables, (easily) solving the relaxed model, then fixing the values of those variables according to some rule or criterion, and finally, solving the resulting problem.

In the context of our problem, the *LP-and-fix* heuristic was implemented relaxing the value of the binary decision variables x (that indicate the presence or not of a flow in a given edge) letting them have values between "0" and "1", and then solving the resulting, relaxed problem, using optimization. After that, the algorithm fixes the values of variables with a 0 value to 0, and those with a 1 value to 1, and then solves the resulting problem using the same optimization tool. This relaxed problem is easily solved, as it has more freedom of choice in terms of allocation levels between the network edges. Then, by fixing the variables to "0" or "1", we significantly reduce the search procedure, making the optimization easier. The two other sets of decision variables, y (the amount of goods moved in each edge) and w (the amount of time spent by each edge) are already continuous variables.

3.3. Hybrid algorithm: combining the size reduction heuristic and the LP-and-fix heuristic

A third approach was developed to solve the multi-commodity network flow problem with time windows and multiple transport lines. It is an algorithm that hybridizes the "size reduction" and the "LP-and-fix" heuristics. In a simple way, this approach consists in first reducing the number of edges by a pre-established percentage (in the tests of this work we have used a reduction of 20%).

Then, the algorithm will take into consideration the matrix edges obtained by the previous step, jointly with other data structures (capacity and vertex) and apply the LP-and-fix algorithm to solve the problem with the relaxed binary decision variables x. Then, the value of these variables are fixed to "1" or "0", according to a pre-established criterion, and the problem is solved again.

4. Case study

According to the Brazilian Institute of Geography and Statistics, the state of Ceará, located in the northeast of Brazil has a geographical extension of 148.920.472 km^2, 184 municipalities, and a population of more than 8.842.000 inhabitants. The intercity passenger transport in the State of Ceará is organized in seven operational areas.

Each area is operated by one or more passenger transport companies, providing lines that depart from the main city of the area to the specific destinations in the area. There are six transport companies operating in the state.

We have identified and listed the ten following categories of information necessary to build the problem data structure: 1. cities; 2. distances matrix; 3. vehicles schedules; 4. travel times; 5. travel periodicity; 6. routes of transport lines; 7. total capacity of the vehicles trunk; 8. type of vehicles; 9. average empty space on the vehicle trunks; and 10. goods to be transported. Information categories from 1 to 6 were obtained from ARCE (the Regulatory Agency of Public Services Delegates of the State of Ceará). Categories from 7 to 9 were obtained from one of the transport enterprises. The information for category 10 was estimated based on observation and informal information.

4.1. Assumptions

This section describes the assumptions that were taken into consideration in order to adapt the real case to our modeling framework. First, we understood that it would be impossible to deal with all the 184 sites because the number of arcs of the resulting problem would be intractable, in computational terms. The municipalities taken into account in the case study were those considered as "poles" inside each of the seven operational areas. The pole towns have in general larger bus terminals, more inhabitants, significant people flows, etc. In order to somehow increase the number of visited municipalities, and to create more transshipment points, we analyzed in more detail towns that were visited by more lines from different operational areas. Therefore, we selected the towns of Camocim, Jaguaribe, Mombaça and Pedra Branca to be part of the network. Table 1 shows the towns included in the case study, the reason for selection, and the connections of the towns with the different operational areas.

Another assumption considered here is related to the capacity of the vehicles. These capacities were estimated taking into account the information provided by one of the transport enterprises, based on the vehicle luggage rack capacity and the average free space of vehicles by trip. For the computational tests, the available capacities of vehicles were estimated using random values between 10% and 40% of the total capacity. As the total vehicle capacity and the (used) standard box measures are known, it is possible to estimate the average number of boxes that fit into a vehicle. The full capacity (97 boxes) was estimated, and then, depending on the percentage of available capacity, the following values have been computed: 40% (39 boxes), 30% (29 boxes), 20% (19 boxes) and 10% (9 boxes) of the total capacity.

Three sites were chosen to be offer nodes: Fortaleza, Sobral and Juazeiro do Norte. Fortaleza was chosen to be an offer node of *food* products, because the majority of these products that come to the state of Ceará have as destination the warehouses near the capital city, to supply the big supermarket networks. Sobral was chosen to be a *footwear* supplier, due to the fact that this city is the largest footwear production center in the state. Finally, Juazeiro do Norte has one of the largest centers of *textiles* production. Twenty kinds of products in these categories were considered for our test problem instances. The total demand of the network was estimated based on the available capacity of the vehicles. Through some preliminary tests we have checked that the system works if the total demand (positive values on the vector b in the data structure “vertex”) corresponds to approximately 10,4% of the total capacity of the vehicles in the “capacity” structure (the total of vector a).

Table 1. Summary of the towns included in the case study

Town	code	reason for selection	connections areas	Town	code	reason for selection	connection areas
Fortaleza	1	pole/transshipment	1 x 2 x 3 x 4 x 5 x 6 x 7	Sobral	11	pole/transshipment	3 x 4 x 5
Aracati	2	pole	1	Itapipoca	12	pole/transshipment	4 x 5
Russas	3	pole/transshipment	1 x 6 x 7	Iguatu	13	pole/transshipment	4 x 6 x 7
Morada Nova	4	pole	1	Juazeiro do Norte	14	pole/transshipment	3 x 4 x 6 x 7
Limoeiro do Norte	5	pole	1	Crato	15	pole/transshipment	3 x 4 x 6 x 7
Baturité	6	pole	2	Mombaça	16	transshipment	2 x 4 x 6 x 7
Quixadá	7	pole/transshipment	2 x 3 x 6 x 7	Pedra Branca	17	transshipment	2 x 3 x 4 x 7
Canindé	8	pole/transshipment	3 x 4	Camocim	18	transshipment	3 x 4 x 5
Crateús	9	pole/transshipment	3 x 4 x 7	Jaguaribe	19	transshipment	1 x 6 x 7
Tauá	10	pole/transshipment	3 x 7				

In order to better explore the case study, three scenarios were designed. These scenarios were developed varying the capacities of the vehicles. The 1st scenario is a "pessimistic" scenario, considering between 10% and 40% of the total destination vehicle capacity. The 2nd case is the "medium" scenario, with a vehicle's capacity between 20% and 40%, while the "optimistic" scenario considers values between 30% and 40%. All the scenarios were generated randomly, using uniform distributions. The resulting values were used in vector *a* of the sub-structure "capacity".

4.2. Results and discussion

The generated test problem instances were run in the IBM ILOG CPLEX version 12.6.1 optimization IDE, using the Concert technology and the Visual C++ IDE, where the heuristics were implemented. This was made to analyze the performance of the tool in processing the two objective functions (related to cost and time) separately. The following four approaches were used: *1st method* – solving the problem with CPLEX, without heuristics; *2nd method* – using the "size reduction" heuristic to reduce the problem size; *3rd method* – using the "LP-and-fix" heuristic; *4th method* – combining the two heuristics (by first using the "size reduction" procedure, and then the "LP-and-fix" heuristic).

A total of 24 tests were performed. These tests are the different combinations of the 2 objective functions, the 4 different methods, and the 3 scenarios. The objective function values, computational execution times and the relative percentage deviations (gap) are shown on the tables below – Table 2 for the cost, and Table 3 for the time.

Table 2. Case study results for the *cost* objective function

scenario	*feature*	1^{st} method	2^{nd} method	3^{rd} method	4^{th} method
	objective value	261990	261990	261990	261990
Pessimistic	execution time	177.835	192.845	48.323	58.086
	gap	0%	0%	0%	0%
	objective value	258681	258686	258681	258681
Medium	execution time	175.601	178.993	43.52	66.101
	gap	0%	0.18%	0%	0%
	objective value	258293	258293	258293	258293
Optimistic	execution time	178.344	171.174	36.127	56.13
	gap	0%	0%	0%	0%

Table 3 – Case study results for the *time* objective function

scenario	*feature*	1^{st} method	2^{nd} method	3^{rd} method	4^{th} method
	objective value	0	0	86589	95699
Pessimistic	execution time	3600	3600	74.417	84.18
	gap	0%	0%	0%	0%
	objective value	0	0	80499	81640
Medium	execution time	3600	3600	78.091	91.766
	gap	0%	0%	0%	0%
	objective value	75127	78076	75359	78279
Optimistic	execution time	3600	3600	60.187	83.714
	gap	0.62%	0.57%	0%	0%

The results on Table 2 (objective function *cost*) show that all the "methods" provided the optimal solution (except for the 2^{nd} method in the medium scenario, with an almost optimal result) – the main difference between the methods was in the execution times. In this aspect, the best results were obtained by the 3^{rd} method, followed by the 4^{th} method. The 1^{st} method was better than the 2^{nd} in the pessimistic and medium scenarios, but worse in the optimistic scenario. Moreover, Table 3 shows that, for the objective function *time*, the methods have a similar performance regarding the *gap* and the objective function value. However, in what concerns the execution time, there was a significant difference, especially between the two first methods and the other two.

5. Conclusions

In general terms, we can conclude that the overall objectives of this work have been achieved, with the proposed approach being capable of efficiently integrating urban freight with the intercity passenger network. When applied to the case study, the model and its resolution methods presented good, quite promising results. A potential decrease of the traffic congestion is also a benefit generated by this model. By exploring the developed approach, we could show that, in the case study, a total of 1600 boxes of products could be delivered just considering a few transport lines and vehicles. However, one clear weakness (limitation) of the model is the need of complex data structures. The high number of "indices" in the model reflects the existence of a large number of possible combinations, especially on the edges data structure. Therefore, it seems very difficult to develop a heuristic / metaheuristic approach to deal directly with the problem (as modeled), without the use of an optimization tool.

This research has naturally generated some ideas and guidelines for future work in the domain: to develop a multi-objective approach, integrating the objective functions cost and time; to apply some variant of the tridimensional bin packing problem to optimize the use of the baggage pack of the vehicles; to develop a discrete event simulation for reducing the uncertainty related to the vehicles capacity measures and to validate the results generated by the optimization model; and to apply the model to other modes of transport as trains or airplanes.

Acknowledgements

This work is financed by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within project POCI-01-0145-FEDER-032053 and National Council for Scientific and Technological Development (CNPq) through grants 404232/2016-7 and 303594/2018-7."

References

Alvarez, A. M., González-Velarde, J. L., & De-Alba, K. (2005). GRASP embedded scatter search for the multicommodity capacitated network design problem. Journal of Heuristics, 11(3), 233-257.

Ambrosini, C., & Routhier, J. L. (2004). Objectives, methods and results of surveys carried out in the field of urban freight transport: an international comparison. *Transport Reviews*, *24*(1), 57-77.

Anand, N., Quak, H., van Duin, R., & Tavasszy, L. (2012). City logistics modeling efforts: Trends and gaps-A review. Procedia-Social and Behavioral Sciences, 39, 101-115.

Benjelloun, A., & Crainic, T. G. (2008). Trends, challenges, and perspectives in city logistics. *Transportation and land use interaction, proceedings TRANSLU*, *8*, 269-284.

Benjelloun, A., Crainic, T. G., & Bigras, Y. (2010). Towards a taxonomy of City Logistics projects. *Procedia-Social and Behavioral Sciences*, *2*(3), 6217-6228.

Crainic, T. G., Ricciardi, N., & Storchi, G. (2009). Models for evaluating and planning city logistics systems. Transportation science, 43(4), 432-454.

Ehmke, J. (2012). *Integration of information and optimization models for routing in city logistics* (Vol. 177). Springer Science & Business Media.

Fanjul-Peyro, L., & Ruiz, R. (2011). Size-reduction heuristics for the unrelated parallel machines scheduling problem. *Computers & Operations Research*, *38*(1), 301-309.

Kennington, J. L. (1978). A survey of linear cost multicommodity network flows. Operations Research, 26(2), 209-236.

Kim, J. G., Jun, H. B., & Kim, C. M. (2011). A two-phase heuristic algorithm for the fixed-charge capacitated network design problem with turn penalties. *KSCE Journal of Civil Engineering*, *15*(6), 1125-1132.

Kleeman, M. P., Seibert, B. A., Lamont, G. B., Hopkinson, K. M., & Graham, S. R. (2012). Solving multicommodity capacitated network design problems using multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, *16*(4), 449-471.

Lin, C. K. Y. (2011). A vehicle routing problem with pickup and delivery time windows, and coordination of transportable resources. *Computers & Operations Research*, *38*(11), 1596-1609.

Masri, H., Krichen, S., & Guitouni, A. (2015). A multi-start variable neighborhood search for solving the single path multicommodity flow problem. *Applied Mathematics and Computation*, *251*, 132-142.

Taniguchi, E., Thompson, R. G., & Yamada, T. (2010). Incorporating risks in city logistics. *Procedia-social and behavioral sciences*, *2*(3), 5899-5910.

Taniguchi, E., Yamada, T., & Kakimoto, Y. (2001). Models for evaluating city logistics measures. In *Proceedings of the Eastern Asia Society for Transportation Studies* (Vol. 3, No. 2, pp. 511-526).

Thompson, R. G., & Taniguchi, E. (2001). City logistics and transportation. *Handbook of Logistics and Supply-Chain Management*, *393405*.

Toledo, C. F. M., da Silva Arantes, M., Hossomi, M. Y. B., França, P. M., & Akartunalı, K. (2015). A relax-and-fix with fix-and-optimize heuristic applied to multi-level lot-sizing problems. *Journal of Heuristics*, *21*(5), 687-717.

Wei, K., Li, X., Lin, S., Yue, C., & Li, S. (2014, June). A simulated annealing based heuristic for the multi-source single-path multi-commodity network flow problem. In *2014 11th International Conference on Service Systems and Service Management (ICSSSM)* (pp. 1-6). IEEE.