Educational Administration: Theory and Practice
2023,29(04) 3545 - 3553
ISSN:2148-2403
https://kuey.net/

Research Article

Optimization Techniques Of A Transportation Problem Involving Automobile Carriers

Anushaka Sharma[1], Rajpal[2*],

[2]Research Scholar, Department of Mathematics, School of Basic and Applied Sciences, Raffles University,Neemrana – 301705. Rajasthan. India. Email: anushakasharma0185@gmail.com
[2*]Associate Professor, Department of Mathematics, School of Basic and Applied Sciences, Raffles University, Neemrana – 301705. Rajasthan. India. Email: Rajpal@rafflesuniversity.edu.in

Citation: Rajpal, et al (2023), Optimization Techniques Of A Transportation Problem Involving Automobile Carriers *Educational Administration: Theory and Practice, 29(4),* 3545 - 3553
Doi: 10.53555/kuey.v29i4.8176

ARTICLE INFO **ABSTRACT**

We analyze a pragmatic distribution challenge that arises within the automotive sector, involving the loading of automobile transporters with vehicles (including automobiles, trucks, and more) prior to their delivery to dealerships. Both calculating the routes that auto-carriers must follow along the road network and establishing an acceptable load for each carrier are essential components in resolving the issue. Our approach includes an iterative local search method that employs mathematical modelling and enumeration approaches for loading and multiple inner local search strategies for routing. This optimization solves the problem. It is possible to achieve significant cost savings with relatively little computational work, according to extensive data collected from real-world examples.

Keywords: vehicle scheduling, loading, auto-transportation, and repeated neighborhood search

1. Introduction

Given its substantial workforce and revenues of 710 billion euros in 2008, it is indisputable that the automotive industry holds a significant position within modern economies. Globally, 912.7 million automobiles were in operation in 2008. Of these, 334.8 million were in Europe, 283.2 million in the Mexico, United States and Canada, and an additional 33 million were dispersed across other regions (see ANFIA 2010, EUROSTAT 2010).

The quantity of automobiles sold is still rather high, even if the global economic crisis of 2008 occurred. There were 18.4 million new motor vehicle registrations in Europe in 2009, with 16.4 million cars and 2 million industrial and commercial vehicles. Also, that year, 12.8 million new motor vehicles were registered in North America, with 6.6 million cars and 6.2 million trucks and other commercial vehicles making up the total. China, Russia, India, and Brazil all saw very large rises.

Optimization methods work well in the automotive industry because of the high volume of sales and the big swings in the market. One of the most critical logistical challenges in this industry is getting autos to dealers, but this is also a great potential for optimization. The subject matter of this paper is this.

In most cases, logistics providers are relied upon rather than the vehicle manufacturers themselves when it comes to product delivery. These businesses get the cars from the manufacturers, put them in storage, and then send them to the dealers when the dealers place an order. Specialized vehicles, known as auto-carriers or auto transporters, make the deliveries. These trucks typically have a tractor and a trailer, with the latter having upper and lower loading platforms. Figure 1 depicts a standard European autocarrier with four loading platforms that can accommodate seven vehicles. The typical number of loading platforms for normal autocarriers is four, though they can range from one to two. While all of the trucks in Figure 1 are the same type, in reality, loadings sometimes include a fleet of different vehicles.

The size and weight of the cars have a significant impact on the auto-loading carrier's capability. Since auto-carriers cannot accommodate vehicles in a side-by-side configuration, the width is a completely irrelevant feature compared to the length, height, and shape. Vehicle carriers often have specialized loading equipment to enhance their carrying capability. The top loading platforms, for instance, can be rotated and/or vertically translated. The length of the platforms can be adjusted by extending both the upper and lower ones. Loading auto-carriers typically begins in the back, and unloading without rearranging freight is typically mandatory, as the last-in-first-out (LIFO) policy is enforced. To clarify, the vehicle positioned at the lower rear portion of the cargo will be emptied initially, as illustrated in Figure 1.

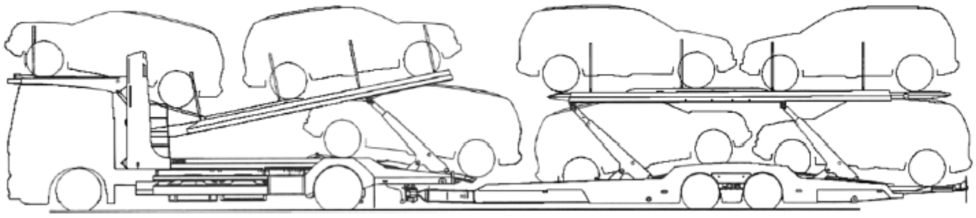


Figure 1: Pictured Here Is an Auto-Carrier Hauling Seven Vehicles

In addition to maximum height, weight, and length, transportation regulations establish a variety of other loading standards. These regulations vary from country to country (The U. S. D. T; 2004). Consequently, putting the vehicles onto an auto-carrier is a difficult undertaking akin to a specific two-dimensional loading issue characterized by a multitude of complex constraints.

Routing becomes a challenge due to the dispersion of dealers over considerable distances; also, a single dealer order rarely completely occupies the capacity of an integer number of vehicle carriers. This is why businesses have little choice but to combine orders from various dealers into a single vehicle carrier. Because of this, finding the best route for auto-carriers is likewise not easy.

One of the most prominent players in Italy's car delivery business commissioned us to write this paper detailing the algorithm we created to solve a real-world problem for them. Here is an explanation of the main issue:

One of the most prominent players in Italy's car delivery business commissioned us to write this paper detailing the algorithm we created to solve a real-world problem for them. Here is an explanation of the main issue:

Combining two NP-hard issues from the domains of loading and routing generates a complex combinatorial issue. Furthermore, the issues we tackle are of an enormous magnitude: as the examples demonstrate, eighty auto transporters deliver 800 autos to two hundred merchants daily. We opted for a heuristic approach since logistics companies require answers discovered in minutes, However, even for scenarios involving 100 clients, existing accurate techniques for the simpler CVRP (capacitated vehicle-routing problem) require days of computations.

We implemented eight different inner local search algorithms into our iterative local search method. These methods always call a loading algorithm whenever they need to assess a route's loading capability. An approximated model of the initial two-dimensional problem is used to solve this approach through an implicit enumeration technique.

We present a mathematical model and strategy for solving the loading subproblem, followed by a nontrivial heuristic for solving the total problem, in this study. The final algorithm is the first of its kind to simultaneously manage loading and routing for auto-carrier transportation, hence providing comprehensive solutions. It provides an accurate vehicle-platform assignment for each auto-carrier in order to give a realistic loading and delivery sequence. Features that take advantage of the problem's complex structure are included, and their efficacy is demonstrated through comprehensive testing on real-world examples. Our system is easily transferable to other markets, even if we primarily focus on the Italian market.

2. Description of the Problem

Here, "vehicle" refers to any object being transported, "auto-carrier" to a truck that transports autos, and "dealer" to a location where delivery is made. The following is a description of the very involved input that we are given:

Network: The full graph $G=(N, E)$. *Here, the collection of vertices is* N={0,1, 2,...,n} *and E is the well-defined collection of edges that connect each pair of vertices. The depot is represented by vertex 0, while the n dealers to be served are represented by the vertices* {0,1, 2,...,n}. *The* (i, b) *symbol represents the edge that joins vertices I and j, and the corresponding routing cost is* c_{ib}. $(i, b \in N)$. *In addition to being symmetric, the cost matrix also satisfies the triangle inequality.*

Fleet: We are presented with a diverse fleet of auto carriers, consisting of a set T of different sorts of auto carriers. Every type of auto-carrier, denoted as t, belongs to the set T and is composed of Pt loading platforms. Its maximum weight capacity is denoted as W_t. *Each type t has its own set of auto-carriers.*

In cases when there is no room for ambiguity, we designate auto-carriers and their respective types using index t, and vehicles and their related models using index k.

Given the typical intricacy of modern practical routing difficulties, the idea of a route is likewise rather intricate, and this is especially true for the situation we are addressing. Here, a route is defined using the triplet, $\langle R, S, t\rangle$.

- $R \subseteq N$ is the order in which to visit the dealers along the way.
- $S_i \subseteq M_i$ is the group of cars that will be sent to the dealer $i \in R$ with help of this route, and $S = S_i \cup \ldots \cup S_{|R|}$ is going to be supplied in its entirety.
- "t" denotes the auto-carrier type of the route. As one travels along a path, $\langle R, S, t \rangle$ an auto-carrier of type t departs from the depot with autos in S, makes stops at the dealers in the order specified by R before heading back to the depot empty-handed. To calculate the cost of a route, add up all the edges it goes through. To determine the sequence of vehicle deliveries along the route, we also employ a function f: $S \rightarrow \{1,\ldots, |R|\}$ in what follows. Every vehicle k ($k \in S$) that R's initial dealer must have (k)=1, For each car that the second R-dealer desires, $f(k)$=2, etc.

A route $\langle R, S, t \rangle$ meets the following criteria, it is deemed load viable.

i. As shown as, the sum of all automobiles in S must not be heavier than the auto-carrier t's maximum permissible mass (z).
ii. The vehicles in S can be loaded onto the Pt platforms of auto-carrier t in a two-dimensional fashion, if they do not overlap, the loading platforms completely back (which can be rotated, translated, or extended), and stay under the maximum length and height allowed for cargo according to the regulation.
iii. According to the LIFO regulation, when you go to a dealer $i \in R$, all vehicles in the range of S-i can be removed from the vehicle transporter without delay. As a result, there will be no need to transfer vehicles that will be visited at later stages along the road.

Condition I is a common capacity constraint, thus checking it is simple. However, solving the two-dimensional loading problem we covered earlier makes checking conditions (ii) and (iii) a challenging task. In §4, we outline the loading problem in detail and show how we model it and the algorithms we employ to solve it, to keep the study concise.

Our combinatorial problem is described as follows: Finding the most cost-effective routes that nevertheless satisfy all the following requirements is the objective of solving the auto-carrier transportation issue (A-CTP): each route must be load practical, all dealer requests must be satisfied, and no more than Kt auto-carriers of type t should be used.

3. A Survey of the Literature

An integral part of the A-CTP is the process of loading vehicles onto auto-carrier platforms and then directing those vehicles throughout the road network. We are proud to say that our method is the pioneering optimization solution to this issue. Our research indicates that existing optimization algorithms for auto-carrier transportation focus on either loading or routing alone, with highly basic loading policies. Here we will first review the key findings in the relevant literature, and then we will explain how our algorithm is unique.

Focusing on the loading subproblem, Agbegha, Ballou and Mathur (1998) detail the best methods utilized by corporations in the American market when delivering vehicles through auto-carriers Agbegha (1992). To represent the auto carrier's LIFO priority among slots, they use a loading network, which they represent using a predetermined array of slots. Considering the loading network and any other pairwise incompatibilities. This leads to a branch-and-bound approach, which they use to resolve a nonlinear NP-hard assignment problem. No routing algorithm is suggested by them.

We were unable to reproduce Agbegha's (1992) findings when we attempted to mimic contemporary European auto-carriers. It seems that the adaptability of these carriers has been much enhanced by loading equipment research over the past 20 years; they can now readily accommodate the transportation of one or two huge trucks or as many as twelve compact vehicles.

The case study conducted by Perboli, Tadei, and Della Croce (2002) focuses on an Italian auto shipping company. Using an integer linear programming (ILP) framework, they offer a general heuristic that considers both loading and routing considerations. Their complex, multi-day, profit-maximizing problem is sure to test their mettle. Partially because of the immense complexity, they use two relaxations. First, they start by making the loading subproblem easier. When determining the equivalent length of each auto-carrier, the lengths of the loading platforms and any applicable loading equipment are added together, plus a constant. Then, they sort the cars into various loading classes based on their shapes. Then, they multiply the original length of each vehicle by a factor that is dependent on the loading class, and that's the equivalent length of the vehicle. As a result, there is only one capacity constraint that represents the loading problem: the total length of the related cars must not exceed that of the corresponding auto-carrier. By grouping all possible destinations together, they further reduce the routing burden. First, they implement a system that limits auto-carriers to loading vehicles within their own cluster. Then, to make it even better, they add support for vehicles from surrounding clusters. They distribute auto-carriers to various clusters rather than creating routes. These two caveats prevent us from using their system to learn specifics like platform vehicle loading and auto-carrier routing.

To solve a vehicle carrier transportation issue in the American market, Miller (2003) investigates the loading and routing components, creates a greedy heuristic, and then optimizes the search between and within routes. To simplify things, he imposes certain constraints. He doesn't figure out how far it is to go from one place to another for the routing section. He assumes that automobiles are placed directly into the two

platforms, models the auto carrier as two flat loading platforms, and ignores technical constraints that might impede the assignment of vehicles to certain platforms. To reframe it, the loading problem is now a two-bin packing problem.
A case study of the distribution of auto-carriers in Venezuela is discussed by Cuadrado and Griffin (2009). In the medium-term, they use an ILP model to handle the issue of auto-carrier fleet size optimization, and in the short-term, they use a two-phase heuristic that is directly based on Tadei, Perboli, and Della Croce to solve the problem of daily trip and load assignment to the transport units (2002).
In their study of the American market, Jin et al. (2010) compared the two main modes of vehicle transportation: road and train. They use an ILP formulation to a business model that aims to reduce total distribution expenses. They group dealers into regions and then think about shipping cars by putting them in designated depots or ramps (automotive distribution centers). They address issues related to the placement of facilities; nevertheless, they do not give specific instructions on how to load automobiles or direct auto-carriers. Lin (2010) analyses the computational behavior of the model on multiple instances by replicating it from Agbegha, Ballou, and Mathur (1998).
In order for the loading subproblem to be considered an A-CTP, it must have a considerable impact on the routing subproblem. To comply with the CVRP's two-dimensional loading limitation, weighted two-dimensional rectangles must be loaded into squares that have the same dimensions. The incorporation of two-dimensional constraints transforms it into a CVRP variant that is extremely challenging (Iori, Salazar González, and Vigo 2007). Gendreau et al. (2006) for furniture distribution, Doerner et al. (2007) for timber distribution, and Hoff et al. (2009) for real-world distribution hurdles are some examples of recent research that have investigated loading and routing issues that are not related to the A-CTP (mineral water distribution). The loading of auto-carriers, split deliveries, and heterogeneous fleets are all beyond the capabilities of any method that we are aware of. Iori and Martello have recently conducted research on routing and loading technologies (2010).
Toth and Vigo (2002) and Golden, Raghavan, and Wasil (2007) are two works that we recommend to readers interested in vehicle routing in general (2008). The second one has an extensive overview of the topic of routing issues with split deliveries (Archetti and Speranza 2008).

4. The Loading Problem and Its Resolution

To ascertain the load feasibility of a specific route $\langle R, S, t\rangle$ in accordance with conditions (i)-(iii), we propose a methodology that can be applied to this objective. Realistically solving the initial two-dimensional loading problem is famously challenging. The basis of our approximate modelling, which we use in conjunction with their loading network approach. Collaborating with the logistics company, we tested the reliability of our approximate modelling. Loadings involving cars of the same model are called homogeneous loadings, whilst loadings involving vehicles of various models are called heterogeneous loadings.
Algorithm 1 lays out our loading technique, which we call check load. Two simple checks and a more sophisticated combinatorial method form its basis.

Algorithm

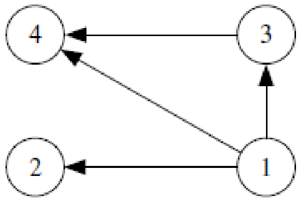
Input: Route $\langle R, S, t\rangle$
Output: feasible when the route satisfies load feasibility; not feasible otherwise.
if $(\sum_{k\in S} w_k > W_t)$, *afterwards, return not feasible* $f_t = \sum_{k\in S} 1/d_{\{kt\}}$
If (loading is uniform), then
if $(f_t > 1)$ afterwards, return not feasible
else return would be not feasible.
else
if $(f_t > f_{sup})$ then return infeasible
else carry out precise method
return information regarding feasibility supplied by the precise process
end
end

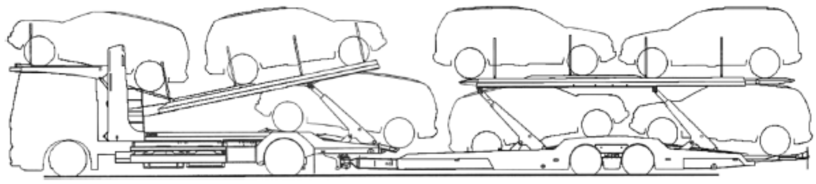
The total weight of the vehicles is the initial point of calculation to determine if the load is practicable; if it is, the load is infeasible. In any other case, we calculate the fill index, f_t, as part of the second fast check.

$$f_t = \sum_{k\in S} \frac{1}{d_{kt}}$$

When the loading is uniform, every vehicle is loaded $k \in S$ possess an identical load index d_{kt}. Based on what we know about load index, it follows that the maximum d_{kt} this truck transporter can accommodate such vehicles. Hence, we return feasible if $f_t \leq 1$ and impossible in any other case. As an illustration, consider Figure 1's homogenous loading: we have
$f_t = \sum_{k\in S} \frac{1}{7} = 1$ Moreover, the burden is manageable. The data acquired by computation f_t is extremely practical for homogeneous loadings but only approximate when applied to heterogeneous loadings.

Additionally, heterogeneous loading can be impracticable, when $f_t \leq 1$ and viable in situations where $f_t > 1$. After an empirically determined maximum value fmax, we rule out all possible heterogeneous loadings. Once the business has completed the preliminary setup with the, we set $f_{max} = 1.2$.





2(a) **2(b)**

Figure 2 The Auto-Carrier Type's Precedence Graph in (b)

As seen in Figure 1, Figure 2 displays the auto-precedence carrier graph. (a). The auto-carrier is shown in Figure 2 in a reduced dimension for clarity's sake (b). While all platforms are preceded by the rear lower platform (which we designate as 1), the rear upper platform precedes just the higher front platform. The transit method involves lowering platforms 3 and 4 to meet with the traffic law's maximum height requirement. Whenever a dealer comes to collect vehicles from platform 1, they raise platform 3. Because of this, two choices emerge: I raising platform 4, which would open up platform 1 for trucks on platform 2 to be emptied; or (ii) lowering the back part of platform 3, which would open up platform 3 for cars to offload. If Platform 3 is raised before Platform 2 is unloaded, raising it again will fix the emptying of Platform 2. The last step in unloading cars from platform 4 is to drive them through platform 3. As a result, there is clearly no order or precedence between platforms two and four.
Assuming the cars were to be put directly onto the platforms, ensuring feasibility would have been as easy as making sure their combined length did not surpass the platform's length. The loaded cars can still spin, though, because loading devices are accessible on every platform (by lifting up their front or rears). A greater number of vehicles can be accommodated by reducing the vehicle's consumption of the platform length in this manner.

5. Findings from Computer Models

Our algorithms were implemented in CCC and executed on a Windows XP machine with a Pentium Dual-Core processor, 2.70 GHz, and 2 GB of RAM. Using examples taken from the logistics company's actual problem, we evaluated the algorithms. We obtained 23 occurrences, or one for every working day, in July 2009 when we were considering the daily distributions from the company's main depot. There were four different kinds of auto-carriers in the original fleet we looked at, but we ended up keeping just the first two since they handle nearly all of the deliveries (98 percent). In the first, there are four loading platforms that can hold 15.1 tonnes of weight, whereas in the second, there are just two platforms that can hold 6 tonnes of weight. There are a total of 723 vehicle kinds, with 14 auto-carrier type 1 loading classes and 8 auto-carrier type 2 loading classes. Using GIS-based software, for each set of vertices I j, we determined the shortest distances (in km) and entered them into the cost matrix cij.
We have made the instances publicly available at www.or.unimore.it/A-CTP to promote future research on this vital topic. We took precautions to protect individuals' privacy by omitting personally identifiable information from the publicly accessible data.

5.1. Evaluation considering Real-World Business Solutions

With so many potential interruptions to regular life, it's hard to draw any reasonable comparisons to the company's industrial solutions. So, we used the company's daily-used heuristic method to build the initial daily plan and tested it on accessible instances to get a fair appraisal of our ILS. This allowed us to compare options that were independent of potential subsequent interruptions and related to the start of the working activity. After 1,500 CPU seconds, we stopped the optimal ILS configuration that employed the enumeration tree and fathoming criterion 1 to perform the comparison.
Assuming a fill load of 1.1 or less makes loading practicable, the logistics company's technique considers it infeasible otherwise. Following the initial area clustering of dealers using the company's algorithm, routes are sequentially developed using a constructive greedy heuristic. Each region is visited by an auto carrier. The route will pass through a residential area if the auto-carrier departs before it is fully loaded. To avoid storing the full matrix c_{ij} in memory, the cost of a route is calculated after the route has been established. Much like with actual transportation problems, altering the routes provided by the software necessitates substantial human involvement.

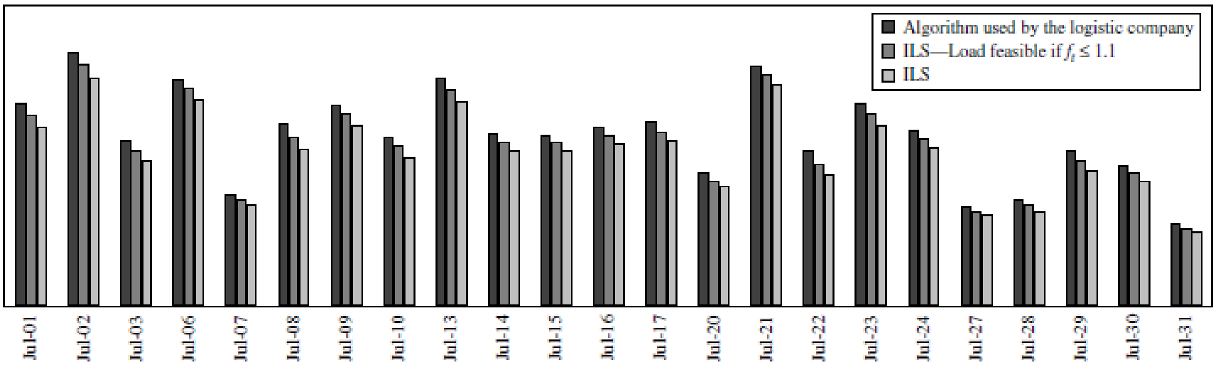


Figure 5: Evaluation of the Logistics Firm's Algorithm vs. ILS and ILS with Simplified Loading

As it turns out, the ILS consistently achieves better results than the company's algorithm, with an improvement of approximately 10.5% on average. While the company's algorithm outperforms the ILS with simplified loading in every scenario, the improvement here is 5.6% on average. The gains we've achieved appear to be attributable, therefore, to the routing and loading aspects of the issue. When compared to the company's current algorithm, the ILS clearly performs better when it comes to creating the first daily plans.
In the event that the disruption transpires in the midst of the day, when computational resources may be scarce, the organization may contemplate executing the algorithm in order to generate an updated delivery schedule. Our proposed ILS converges to satisfactory solution values within minutes of execution, which qualifies it as an acceptable option for this task. This feature is emphasized in Figure 6, which was generated by imposing an 8-hour time constraint on each instance of the ILS. Initially, the percentage mismatch between the solution value returned at the conclusion of the operation and the current solution value was ascertained. The average gap per minute for all 23 cases included in our calculations is displayed along the vertical axis in Figure 6. Approximately fifty percent of the total improvement will be realized in solutions that are generated after only five minutes of ILS execution. Approximately seventy-five percent of the overall benefit is observed within twenty-five minutes after application. Three hours of CPU time results in an almost flat curve.

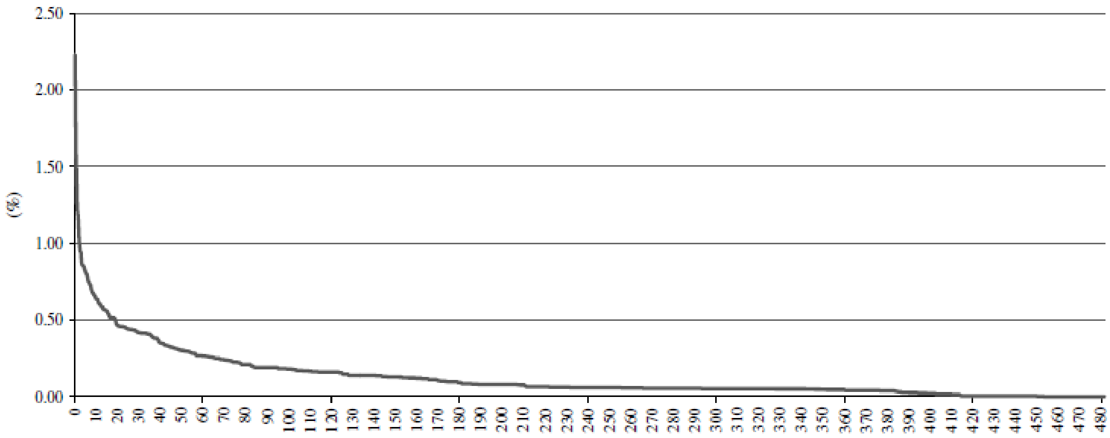


Figure 6: Changes in the Disparity in Percentage Relative to the most well-known solution within a period of eight hours

5.2. Assessment of ILS Efficiency

Following 1,500 CPU seconds of executing the ILS with the enumeration tree and understanding criterion 1 within the procedure check load, the execution was terminated. It was determined that this time constraint struck a suitable balance between the need for a practical application to execute rapidly (within minutes) and the assurance of a high-quality solution. We present the results of our analysis in Table 1.

Table 1: Results Obtained from the ILS Calculation and Comparison to a Multistate Approach

Instance			ILS						Multistart						
Name	n	M	n_{ac}	n_{vls}	z	sec_z	it_z	it_{tot}	n_{ac}	n_{vls}	z	sec_z	it_z	it_{tot}	%gap
09-Jul-01	228	832	88	265	**51,159**	1,254.0	1,432	1,718	89	263	51,697	1,086.6	931	1,259	1.04
09-Jul-02	221	1,139	117	295	**65,224**	1,425.5	1,104	1,161	117	293	65,738	77.8	47	899	0.78
09-Jul-03	195	737	81	240	**41,490**	1,484.9	2,358	2,380	83	229	41,924	298.0	339	1,660	1.04
09-Jul-06	243	1,063	110	292	**58,526**	1,449.1	691	722	109	296	58,756	238.3	98	606	0.39
09-Jul-07	165	629	66	186	**28,739**	1,165.3	2,369	3,045	67	189	28,961	977.5	1,375	2,058	0.77
09-Jul-08	206	810	84	231	**44,619**	1,446.9	1,654	1,713	85	240	45,143	1,079.3	935	1,276	1.16
09-Jul-09	200	941	101	266	**51,360**	595.3	539	1,365	101	257	52,085	1,043.2	727	1,020	1.39
09-Jul-10	199	803	85	240	**42,576**	981.0	912	1,365	86	232	43,021	1,164.6	593	755	1.03
09-Jul-13	244	1,030	108	297	**58,458**	1,336.9	936	1,045	108	301	58,798	233.3	118	755	0.58
09-Jul-14	227	826	89	265	**44,311**	1,183.9	1,315	1,623	90	261	44,931	1,250.7	1,011	1,188	1.38
09-Jul-15	211	729	79	243	**44,439**	1,043.3	1,677	2,423	78	240	44,845	633.5	651	1,522	0.91
09-Jul-16	206	833	88	244	**46,393**	1,335.5	1,988	2,234	89	252	46,644	123.3	137	1,581	0.54
09-Jul-17	200	801	91	245	**47,100**	1,393.9	880	942	91	244	47,452	398.2	169	635	0.74
09-Jul-20	198	707	76	233	**34,087**	1,173.3	1,637	2,118	76	238	34,516	1,176.3	1,177	1,482	1.24
09-Jul-21	209	940	105	264	**63,090**	1,374.3	879	968	106	259	63,432	1,346.0	600	658	0.54
09-Jul-22	189	614	66	214	**37,336**	926.8	1,962	3,194	66	211	37,806	872.6	1,277	2,128	1.24
09-Jul-23	251	875	92	294	**51,647**	1,215.1	1,191	1,464	93	286	52,244	302.5	210	1,020	1.14
09-Jul-24	198	811	87	242	**45,360**	426.6	392	1,303	88	240	45,678	145.1	98	1,018	0.70
09-Jul-27	162	552	58	188	**25,971**	1,166.5	3,698	4,762	60	186	26,149	360.8	908	3,684	0.68
09-Jul-28	176	556	59	197	**26,997**	1,199.4	3,063	3,857	59	192	27,294	1,097.2	2,009	2,694	1.09
09-Jul-29	221	690	75	257	**38,733**	1,488.7	2,529	2,546	76	251	39,045	791.5	1,023	1,913	0.80
09-Jul-30	204	614	65	231	**35,775**	1,379.3	2,911	3,162	66	229	36,216	245.6	352	2,131	1.22
09-Jul-31	96	272	30	104	**21,034**	162.1	2,731	25,868	30	103	21,099	309.5	3,029	14,131	0.31
Avg.	202.1	774.1	82.6	240.6	**43,670.6**	1,156.9	1,689.0	3,086.0	83.2	238.8	44,064.1	663.1	774.5	2,003.2	0.90
Total			1,900	5,533					1,913	5,492					

To gain a more comprehensive understanding of the performance of the ILS, we analyzed the perturbation style and the local search, which are the two fundamental components of the algorithm. The ILS is one of numerous metaheuristics based on multistate systems. It is customary within this family to draw comparisons between the perturbation approach and the most fundamental implementation accomplished by the usual multistate (MS) algorithm (Lourenco et al.; 2010). The MS, like the ILS, seeks solutions by the iterative implementation of the randomized closest neighbor heuristic. In contrast, when the algorithm encounters a local minimum, it develops a novel solution by commencing from the beginning.
Each local search algorithm improves the ILS's performance, as we have noticed. The average solution value is negatively affected by every removal (see column z). The average solution value rises to 43,781.1 or 43,818.2 when the 1-0 dealer move or 1-1 dealer swap is eliminated, respectively, leading to the worst possible outcomes. It is worth mentioning that we eventually disregarded other local search operators. With each removal having no detrimental effect on solution quality, the ILS can outperform the original greedy solutions and prove to be quite resilient.

5.3. Results of the Evaluation of the Packing Proposal

This paper lays out the procedure for checking load, provides a mathematical model for it, and explains how to apply three algorithms inside it. These algorithms are based on exploring an enumeration tree. The specific effects of various algorithms on the ILS's performance are detailed in Table 4. In addition to the instance data, the table has four sets of columns that indicate the outcomes of executing the ILS with the given loading method. We tried out the following setups:

- exhaustive listing of trees;
- trees catalogued using understanding criterion 1
- family tree enumeration using comprehension criterion 1 and 2.

Tabulated in Table 1 are the optimal solution values (z), total iterations (ittot), and percentage of CPU time consumed in the loading procedure for each setup (percent load). The bolded values are the optimal solution values for each case.

5.4. Effects of the Limitations on Loading

As an integrated loading and routing problem, the A-CTP is classed with similar ones. This field is being studied because a pure routing model is insufficient when there are loading limits, which can significantly impact the solution to a transportation challenge. An approximation method focusing solely on routing could be sufficient if the loading's impact is minimal. If it's big, you'll need to employ integrated loading and routing strategies. Therefore, the assessment of the loading constraints' effects has been the focus of the research by Gendreau et al. 2006; Zachariadis et al. 2012; This assessment is also significant for methods that view weigh down as a penalizing component in the objective function rather than a constraint; for instance, Erdoˇgan et al (2012); Fuellerer et al. 2009.

Table 2: The ILS's Outcomes Under Varying Stress Levels

Instance			All conditions			No condition (i) no weight			No condition (iii) no LIFO			No conditions (ii) and (iii) relaxed loading		
Name	n	M	n_{ac}	n_{vis}	z	Δn_{ac}	Δn_{vis}	$\%gap$	Δn_{ac}	Δn_{vis}	$\%gap$	Δn_{ac}	Δn_{vis}	$\%gap$
09-Jul-01	228	832	88	265	51,159	−1	−2	0.33	−1	0	−0.82	−2	3	−1.49
09-Jul-02	221	1,139	117	295	65,224	0	−19	−0.01	−1	−10	−0.38	−1	−5	−1.16
09-Jul-03	195	737	81	240	41,490	0	−7	−0.43	0	−8	−0.48	−1	−12	−1.00
09-Jul-06	243	1,063	110	292	58,526	0	2	−0.11	−2	4	−0.94	−3	13	−1.95
09-Jul-07	165	629	66	186	28,739	0	4	−0.09	−1	9	−0.94	−2	12	−1.62
09-Jul-08	206	810	84	231	44,619	1	7	0.48	−1	9	−0.61	−1	8	−0.98
09-Jul-09	200	941	101	266	51,360	0	0	0.00	−1	−7	−0.55	−2	−14	−2.16
09-Jul-10	199	803	85	240	42,576	1	0	0.00	0	0	−0.82	−1	−2	−1.51
09-Jul-13	244	1,030	108	297	58,458	0	−8	0.33	−2	−1	−0.75	−3	0	−2.07
09-Jul-14	227	826	89	265	44,311	0	−4	0.05	−1	−7	−0.67	−2	−6	−1.83
09-Jul-15	211	729	79	243	44,439	−1	2	−0.11	−2	5	−1.03	−2	−1	−2.46
09-Jul-16	206	833	88	244	46,393	−1	4	−0.39	−2	5	−0.60	−3	9	−1.49
09-Jul-17	200	801	91	245	47,100	0	3	0.13	−1	10	−0.16	−3	1	−1.36
09-Jul-20	198	707	76	233	34,087	0	2	0.23	0	−4	−0.63	−2	−3	−1.94
09-Jul-21	209	940	105	264	63,090	0	5	−0.71	−2	5	−1.24	−6	−6	−4.11
09-Jul-22	189	614	66	214	37,336	0	1	−0.08	−1	0	−0.20	−1	3	−1.04
09-Jul-23	251	875	92	294	51,647	0	−2	0.56	−1	−6	−0.74	−2	0	−1.76
09-Jul-24	198	811	87	242	45,360	0	1	−0.14	−1	−2	−0.97	−2	−2	−2.24
09-Jul-27	162	552	58	188	25,971	1	2	−0.22	0	−1	−0.47	−1	4	−2.91
09-Jul-28	176	556	59	197	26,997	0	1	−0.09	−1	−2	−0.56	−1	5	−1.25
09-Jul-29	221	690	75	257	38,733	1	1	−0.16	0	−11	−0.72	1	−5	−0.66
09-Jul-30	204	614	65	231	35,775	0	2	0.30	−1	3	−0.16	−1	−1	−0.64
09-Jul-31	96	272	30	104	21,034	0	2	−0.02	0	3	−0.01	−1	6	−0.44
Avg.	202.1	774.1	82.6	240.6	43,670.6	0.0	−0.1	−0.01	−1.0	−0.3	−0.63	−1.8	0.3	−1.65

Change from the first ILS run, expressed as a percentage of the total number of auto-carriers used, is reported for each set of columns (Δn_{ac}), variation in the extent to which services are provided (Δn_{vis}), and proportion gap from z (($\%gap$). We observe that the problem is mostly unaffected by the weight constraint. Values are typically unaffected. of Δn_{ac}, Δn_{vis}, and $\%gap$. Indeed, only a small number of vehicles are known for their exceptionally large weight, and in the vast majority of instances, the shapes of the vehicles place significant constraints on loading. In regard to the initial issue, the quantity of impractical alternatives lends credence to this idea. It turned out that five of the solutions were actually not possible when we checked them after condition I was removed.

Removing criteria (ii) and (iii) has a bigger impact, resulting in a cost decrease of approximately 1.65% and the utilization of two fewer trucks. Keep in mind that the more visits there are, the more split deliveries will be required in this instance. All solutions generated by this setup were determined to be infeasible in relation to the initial problem; specifically, 669 out of 1,858 pathways were deemed infeasible, which has a significant impact on infeasibility. Due to the destructive nature of deleting the loading component and the consequent meaninglessness of the solutions produced in relation to the A-CTP, we do not present the results that were achieved by doing so. In general, it is not feasible to apply current techniques from the CVRP literature and loosen the loading requirements; doing so would provide solutions with high infeasibility.

6. Findings and Looking Ahead

We handled an interesting real-world transportation challenge involving putting vehicles onto auto-carriers and then routing them to dealers. Using an enumeration tree and an ILS, In order to load the automobiles and route the auto-carriers, we suggested a heuristic approach.

Our proposed loading problem model accurately simulates real-world loadings and requires little in the way of processing resources to resolve. When compared to the results given by the company's present algorithm and a normal multistate technique, the overall algorithm achieves significant and consistent savings. We have thoroughly evaluated the optimal algorithm configuration. After analyzing the effects of various loading constraints, we determined that using pure CVRP algorithms and losing the loading constraints is not a viable option since it would produce solutions with high infeasibilities. Despite our presentation's emphasis on the Italian market, the idea is easily adaptable to other regions. By making the benchmark examples publicly available, we aim to inspire greater research on the subject.

The essay was written with the assumption that there are sufficient auto-carriers to fulfil all the requests for autos on that day. Our examples sparked the change because they represented a period of great upheaval in the automotive sector (July 2009). It may be necessary to delay some deliveries until the following days if the fleet size is insufficient. Reducing route costs, fines for late deliveries, and other factors are typically considered while making these selections. Decisions like these arise in the context of dynamic multiperiod routing problems, which are sure to be exciting areas of study in the years to come.

References

[1] Agbegha GY (1992) An optimization approach to the auto-carrier problem. Unpublished doctoral dissertation, Case Western Reserve University, Cleveland, OH.

[2] Agbegha GY, Ballou RH, Mathur K (1998) Optimizing auto-carrier loading. Transportation Sci. 32:174–188.
[3] Alba Martínez MA, Cordeau J-F, Dell'Amico M, Iori M (2013) A branch-and-cut algorithm for the double traveling salesman problem with multiple stacks. INFORMS J. Comput. 25:41–55.
[4] ANFIA (Italian Association of the Automotive Industry) (2010) Studies and statistics on the international markets. Accessed January 28, 2014, http://www.en.anfia.it/index.php?moduloDview_studi_mercati_esteri.
[5] Archetti C, Speranza MG (2008) The split delivery vehicle routing problem: A survey. Golden B, Raghavan S, Wasil E, eds. The Vehicle Routing Problem2 Latest Advances and New Challenges (Springer, Berlin), 103–122.
[6] Cuadrado M, Griffin V (2009) Mathematical models for the optimization of new vehicle distribution in Venezuela (Case: Clover International C.A.). Ingeniería Industrial. Actualidad y Nuevas Tendencias 1:53–65.
[7] Doerner K, Fuellerer G, Gronalt M, Hartl R, Iori M (2007) Metaheuristics for vehicle routing problems with loading constraints. Networks 49:294–307.
[8] Erdoˇgan G, Battarra M, Laporte G, Vigo D (2012) Metaheuristics for the traveling salesman problem with pickups, deliveries and handling costs. Comput. Oper. Res. 39:1074–1086.
[9] European Commission, Mobility and Transport (2006) Freight transport logistics in Europe: The key to sustainable mobility. Accessed January 28, 2014, http://eur-lex.europa.eu/ LexUriServ/LexUriServ.do?uriDCOM:2006:0336:FIN:EN:PDF.
[10] EUROSTAT (2010) Regional transport statistics. Accessed January 28, 2014, http://epp.eurostat.ec.europa.eu/portal/page/portal/transport/data/main_tables.
[11] Fuellerer G, Doerner K, Hartl R, Iori M (2009) Ant colony optimization for the two-dimensional loading vehicle routing problem. Comput. Oper. Res. 36:655–673.
[12] Gendreau M, Iori M, Laporte G, Martello S (2006) A tabu search algorithm for a routing and container loading problem. Transportation Sci. 40:342–350.
[13] Golden B, Raghavan S, Wasil E, eds. (2008) The Vehicle Routing Problem2 Latest Advances and New Challenges. Operations Research/Computer Science Interfaces Series, Vol. 43 (Springer, Berlin).
[14] Hoff A, Gribkovskaia I, Laporte G, Loekketangen A (2009) Lasso solution strategies for the vehicle routing problem with pickups and deliveries. Eur. J. Oper. Res. 192:755–766.
[15] Iori M, Martello S (2010) Routing problems with loading constraints. TOP 18:4–27.
[16] Iori M, Salazar-González J-J, Vigo D (2007) An exact approach for the vehicle routing problem with two-dimensional loading constraints. Transportation Sci. 41:253–264.
[17] Jin M, Eksioglu SD, Eksioglu B, Wang H (2010) Mode selection for automotive distribution with quantity discounts. Networks Spatial Econom. 10:1–13.
[18] Laporte G, Semet F (2002) Classical heuristics for the capacitated VRP. Toth P, Vigo D, eds. The Vehicle Routing Problem. Monographs on Discrete Mathematics and Applications, Vol. 9 (SIAM, Philadelphia), 109–128.
[19] Lenstra HW Jr (1983) Integer programming with a fixed number of variables. Math. Oper. Res. 8:538–548.
[20] Lin C-H (2010) An exact solving approach to the auto-carrier loading problem. J. Soc. Transportation Traffic Stud. 1:93–106.
[21] Lourenço H, Martin O, Stützle T (2003) Iterated local search.
[22] Glover F, Kochenberger G, eds. Handbook of Metaheuristics.
[23] International Series in Operations Research and Management Science, Vol. 57 (Springer, New York), 320–353.
[24] Lourenço HR, Martin OC, Stützle T (2010) Iterated local search: Framework and applications. Gendreau M, Potvin J-Y, eds.
[25] Handbook of Metaheuristics. International Series in Operations Research and Management Science, 2nd ed., Vol. 146 (Springer, Berlin), 363–398.
[26] Miller BM (2003) Auto-carrier transporter loading and unloading improvement. Unpublished doctoral dissertation, Air Force Institute of Technology, Wright-Patterson Air Force Base, OH.
[27] Sinnott RW (1984) Virtues of the Haversine. Sky Telescope 68: 158–159.
[28] Tadei R, Perboli G, Della Croce F (2002) A heuristic algorithm for the auto-carrier transportation problem. Transportation Sci. 36:55–62.
[29] Toth P, Vigo D (2002) The Vehicle Routing Problem. Monographs on Discrete Mathematics and Applications, Vol. 9 (SIAM, Philadelphia).
[30] U.S. Department of Transportation, Federal Highway Administration (2004) Federal size regulations for commercial motor vehicles. Accessed January 28, 2014, http://ops.fhwa.dot.gov/freight/publications/size_regs_final_rpt/.
[31] Zachariadis EE, Tarantilis CD, Kiranoudis CT (2012) The palletpacking vehicle routing problem. Transportation Sci. 46: 341–358.