(Replication and Extension of)

Exact Solutions to the Double Travelling Salesman Problem with Multiple Stacks

By:
Yash S. Gokhale (Andrew ID: ysg)

Submission Date:
26th November 2019

Course:
Product and Supply Chain Optimization (06635)

Original Work:
Petersen H., Archetti C. and Speranza M., ‘Exact Solutions to the Double Travelling Salesman Problem using Multiple Stacks’.

TABLE OF CONTENTS

I. Description of Paper

Preface

The original paper by Petersen et. al. puts out a formulation of optimizing the pickup and delivery route of a vehicle. The paper puts up a number of constraints, that is, it assumes that the pickup and delivery regions of the product is far apart, and thus, it is not feasible to individually deliver the items and thus, must be done at a stretch. Moreover, the vehicle is also assumed to follow a LIFO policy, which implies that the last item loaded must be unloaded first.

Motivation:

Intermodality causes a huge problem in the modern transportation plan, due to the ever-increasing traffic and road congestion. The problem also finds applications in pickup and delivery freight transportation and in multi-modal transportation, where the mode of delivery is railroad or air. Moreover, the amount of time and money invested in delivering goods through intermodality adds to the planning problems, reducing its applicability. Also, the model mentioned in the paper does not permit repacking or vertical stacking. The motivation for this is , that, it may not always be possible to repack the items due to logistical issues and vertical stacking might not be allowed, owing to the risk of damage.

Novelty :

The highlight of the model proposed is that as opposed to a traditional travelling salesman problem, using the methodology, one can simultaneously chalk out the path of two regions (pickup and delivery). Apart from this, the paper also gives the stacking strategy, incase one has the dilemma of the method of stacking items, also following the LIFO (Last In First Out) approach. The paper puts forth 3 main models, all of which have different formulations, but helps one in achieving the same result, with different number of binary variables.

Approach:

The paper deals with 4 models, 3 of which I have implemented in this report. The three models are precedence, row precedence and flow model. Brief description of these models are as follows: (x is the standard tour variable)

For Precedence Model, there is no subtour elimination constraint explicitly mentioned. However, a new variable y(i,j) is introduced, which would equal 1, in the case the item i is handled before j. Also, the precedence model is a modular model, with comparatively easier to implement.

For Row Precedence Model, there are 2 subtour constraints. One of the subtour constraints takes care of the subtours in the delivery and pickup path, whereas, the other constraint is focussed on implementing the correct placement of items, so as to follow LIFO. It involves the introduction of a new variable , w(i,j,r), which equals 1, when both i,j are placed in the same row r.

Flow model has a similar behavior as the row precedence model, with 2 conditions catering to the subtour elimination. A new variable v is introduced, which is a combined notation for the row assignment and comparison of 2 different items.

Thus, the general approach followed to obtain a solution is:

1) The flow is balanced across each node,ie., for every entering flow, there must be an exit flow. This is addressed by introducing a binary variable x(i,j), which indicates if that arc (i,j) is present in the tour, then it will be set to 1.
2) Transitivity is imposed, so as to ensure that if item i is handled before j and item k is before item j, then item i must be handled before k (or vice versa).
3) LIFO condition is imposed differently for different models
4) Additional constraints are added depending upon the type of variables introduced.

Literature Review:

The aforementioned paper was published in 2010, and this section aims to explain its position as compared to its contemporary literature.

1) The paper under consideration, [1], aims to model a problem which is similar to [2]. Both the papers aim to solve a double travelling salesman problem with LIFO, however [2] has the ability to additionally solve FIFO (First In First Out). On the other hand, [1] proposes 3 new models, viz.

Precedence Model, Flow Model and TSPIP (TSP with Infeasible Paths). Hence, as compared to [2], this paper has a simpler model to solve the problem but is unable to solve for FIFO, without modifying the constraints.
2) [3] mentions a method which involves applying a branch and cut algorithm to the DTSP with LIFO loading. As compared to the mentioned model in [1], it contains several chunks of inequality constraints which are used within the B&C algorithm. The advantage of following [3] is that it approaches the solution faster. It can solve 50 nodes within reasonable computation time, whereas the given paper takes sufficient computation time for even 30 nodes.
3) [4] tackles a problem which is significantly different from the one mentioned above. In [4], the problem involves multiple vehicles as opposed to one, and is also permitted to have a 2-dimensional loading. The complexity of this problem is even higher, as each vehicle has a total weight capacity. The paper is a MILP, whereas this paper is a MIP. Thus, in all, [4] addresses a much generalized problem, which can be reduced to the [1]. [1] will fail to take into account the model in [4] and thus, would fail, and can be termed weaker as compared to it.
4) [5] is the paper that forms the basis of this paper. [5] was the paper to formulate the idea of a Double Travelling Salesman Problem with Multiple Stacks; however, it provides a metaheuristic approach to the solution, as opposed to the exact solution in [1].Thus, [1] has a more robust approach in leading to the optimal solution.
5) [6] is a subset of the above mentioned paper. The paper considers only 2 stacks, as opposed to a generalized model with ‘n’ stacks. Thus, [1] is a far superior model as it takes into account all possibilities of stacking.

Dataset

As cited in the report, the datasets were imported from: http://www.mat.ucm.es/~gregoriotd/dtspmsEn.htm. The data is in the form of .tsp files, with each dataset having 2 files, one for pickup and the other for delivery. It has the data of (x,y) coordinates of each node in pickup and delivery. The cost function is the euclidean distance between two points, calculated as follows:

$$C(i,j) = \sqrt{(xi - xj)^2 + (yi - yj)^2}$$

Thus, for every dataset, there are 2 values of Cost, one for the pickup and the other for delivery.

II. Replication of Results

For the replication of results, I implemented the 3 models given in the paper, namely, Precedence Model, Row Precedence Model and Flow Model. The fourth model, TSPIP, was not replicated, owing to the iterative complexity, after discussion with the instructor. The replication was done in ‘GAMS Studio’, with all test runs performed on an Intel Core i7, 8th Generation.

Challenges faced:

The primary challenge I faced was to get familiarized with the GAMS Syntax. For the precedence model, it was difficult to implement the condition that i and j are distinct, as the domain was differently set for each of the binary variables (as x includes the depot, whereas y started from the first customer). However, after trial and error, I managed to figure out that this constraint can be individually implemented for every sub-constraint.
Moreover, I reformulated some of the dimensions of the binary variables as opposed to those mentioned in the paper, ie. x^T(i,j) as x(i,j,T). This simplified the output and also made the solution clear for the reader.
For the formulation involving subtour elimination, I had the difficulty of implementing loops which incorporated both the pickup and delivery. I managed to obtain a feasible solution for the same using two different loops and thus eliminating subtours in the pickup and delivery. However, the implementation of loops for LIFO condition was unsuccessful and thus, was ultimately relaxed to obtain a feasible solution.

Progress

I was able to successfully replicate the Precedence model to satisfactory accuracy with a similar runtime.However, in case of the row precedence and flow model, the feasible solution was obtained with a satisfactory accuracy, however, after the relaxation of the LIFO Constraint. Hence, the double travelling salesman problem was solved without subtours in all the three cases; however, I had trouble in incorporating the LIFO condition for the latter two. Thus, my plan of action is to work on it further, and introducing cuts, which would render unwanted paths as infeasible.

Results

The GAMS Studio was used to implement the three models as follows:

Model 1: Precedence Model

Model 2: Row Precedence Model without LIFO

Model 3: Flow Model without LIFO

Instance	n	Opt	UB	Gap(Model 1)	Time/s (Model 1)	Opt (Model 2)	Time/s (Model 2)	Opt(Model 3)	Time/s(Model 3)
R05_2x4	18	502	502	0	<1	499	1.6	499	3
R06_2x4	18	693	693	0.035	1.3	606	<1	668	3.1
R07_2x4	18	487	487	0	4	461	2.3	478	2.7
R08_2x4	18	641	641	0	1.8	603	1.6	622	2.3
R09_2x4	18	557	557	0.067	1.02	546	1.4	546	2
R05_2x5	22	547	547	0.050	35	522	1.8	521	2.7
R06_2x5	22	775	775	0.044	105	654	1.1	733	3
R07_2x5	22	546	546	0.052	35	516	3	515	3.6
R08_2x5	22	668	668	0.062	161	621	2	621	2.9
R09_2x5	22	609	609	0.037	5	597	2	597	2
R05_3x4	26	566	566	0.019	7	560	2	560	7.34
R06_3x4	26	748	748	0.063	18	738	2	738	5.3
R07_3x4	26	557	557	0.046	49	537	3	537	7.1
R08_3x4	26	687	687	0.052	4	673	3	672	6.3
R09_3x4	26	668	668	0.048	9	664	2.1	665	6.2

Comparison with Literature

I was able to achieve a sufficient level of accuracy for the precedence model, when compared to literature. However, the level of accuracy was lower for the row precedence model, as the condition of LIFO was relaxed and only the standard pickup and delivery regions were solved for an optimal cost. As the condition of LIFO was also relaxed for the Flow Model, the optimal solution achieved was the same as that of row precedence model.

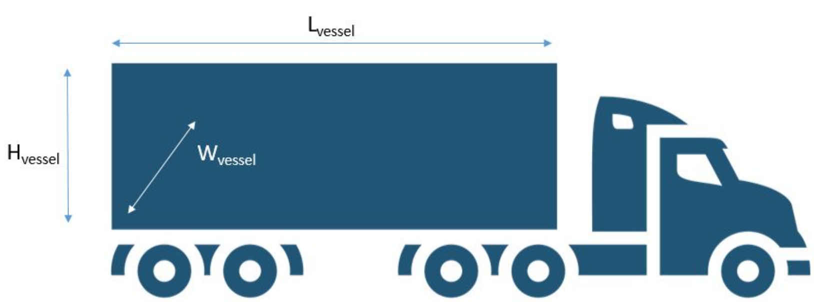
Also, the optimal value obtained was lower than that for precedence model. Commenting upon the run time of the model, the row precedence model gave a much faster result as compared to the Flow model. As for the precedence model,the run time was higher due to the LIFO constraint, which inherently eliminated the subtours.

III. Extension of Results

The paper assumes that all the packages to be delivered are identical. However, for all practical purposes, it is possible that different customers prefer different items from the same company. In order to address this issue, I have extended the results of the paper to a case where the items are not identical.

Methodology

Consider a vehicle container of dimensions LxBxH with r rows. As each of the items are of different dimensions, there is also a constraint on the permissible total length and breadth of the item. The height of the item is not under consideration as the model does not permit vertical stacking. For simplicity, we can assume that the height of the item is always less than that of the truck.



The base model was considered as the precedence model, with new constraints in addition to the original constraints in the model.

Using the same notation as in the precedence model of [1], additional parameters introduced are:

Parameter Lt(i), which is the length of each individual item

Additional constraints are:

$\sum_{i \to I} L(i) * z(i,r) \le Lt,\ r \leftarrow R$ ..(1)

Equations 1 ensures that the objects are fitted in a way that it fits in the truck.

Implementation

The additional constraints were added to the pre-modelled GAMS algorithm of precedence model. Based on the general truck vessel configuration, the length of truck vessel was chosen to be 8 units,10 units and 12 units.

In case of non-similar items, the alignment of items would play a part. For practical situations, if the items do not fit the truck, additional time and effort would be spent. Thus, an additional constraint needs to be incorporated.

(General consideration):

Item	1	2	3	4	5	6	7	8	9	10	11	12
L(i), (units)	1	2	3	1	2	1	1	3	2	4	3	1

Tabulated Results for all the cases are:

	8 unit length		10 unit length		12 unit length	
Instance	Opt	Gap	Opt	Gap %	Opt	Gap %
R05_2x4	502	8	502	0.44	502	0.44
R06_2x4	745	10	694	3.7	694	3.7
R07_2x4	506	9.8	504	9.3	504	9.3
R08_2x4	669	9.8	660	8.6	660	8.6
R09_2x4	603	9.7	593	1	593	1
R05_2x5	-	-	583	9.4	582	10

R06_2x5	-	-	809	9.8	805	9.8
R07_2x5	-	-	554	5.2	564	8.8
R08_2x5	-	-	703	8.9	680	8.1
R09_2x5	-	-	654	8.6	640	7.1
R05_3x4	569	4.7	598	9.2	568	4.7
R06_3x4	803	9.5	790	8.4	809	9.7
R07_3x4	588	9.8	572	8.9	567	9.1
R08_3x4	751	9.8	749	9.7	708	5
R09_3x4	705	6.9	667	2.3	691	5

Motivation & Outcome of Extension

The market for custom-made goods is on the rise. Based on customer preferences, companies tend to create personalized products. The personalization prompted me to incorporate this additional complexity in the existing LIFO model. Thus, the new model has to follow these conditions:

1) No subtour in Pickup/Delivery
2) LIFO order followed
3) Row allotment followed

From the tabulated data, it can be observed that item dimension affect the allotment strategy. In case of an 8 item length, for 3 rows (which implies 24 capacity), the 2*5 configuration does not yield a feasible solution. However, for the 3*4 configuration, a satisfactory solution is obtained. Thus, the lesson learnt here is that, in spite of having lesser number of items to be delivered, the item dimensions led the solution to infeasibility. On the other hand, a different configuration of 3*4 for 12 items had a feasible solution.

References

[1] Petersen, H., Archetti C. and Speranza M., ‘Exact Solutions to the Double Travelling Salesman Problem with Multiple Stacks’, Wiley Periodicals Inc. Networks, Vol. 56(4),229-243,2010

[2] F. Carrabs, R. Cerulli, and J.-F. Cordeau, An additive branch and-bound algorithm for the pickup and delivery traveling salesman problem with LIFO or FIFO loading, INFOR 45 (2007), 223–238.

[3]J.-F. Cordeau, M. Dell’Amico, and M. Iori, Branch-and-cut algorithm for the pickup and delivery traveling salesman problem with FIFO loading, University of Modena and Reggio, Emilia, 2009.

[4]M. Iori, J. Salazar-González, and D. Vigo, An exact approach for the vehicle routing problem with two-dimensional loading constraints, Transportation Sci 41 (2007), 253–264.

[5] H.L. Petersen and O.G. Madsen, The double travelling salesman problem with multiple stacks—formulation and heuristic solution approaches, Eur J Oper Res 198 (2009),139–147.

[6] F. Carrabs, R. Cerulli, and Speranza M., ‘A branch and bound algorithm for the double travelling salesman problem with two stacks’,Networks 2013.