AMSI VACATIONRESEARCH SCHOLARSHIPS 2020–21

Get a Thirst for Research this Summer

Pallet-Packing Vehicle Routing Problem (PPVRP)

via Quadratic Programming

Ponpot Jartnillaphand

Supervised by Dr. Hoa Bui

Curtin University

Vacation Research Scholarships are funded jointly by the Department of Education, Skills and Employment and the Australian Mathematical Sciences Institute.

Contents

Abstract

This research explores Pallet-Packing Vehicle Routing Problem (PPVRP) via Quadratic Programming. The PPVRP is the integration of packing and routing problems. We present a formulation for a packing problem with packing conditions using quadratic programming. The model takes into account the profit and expenditure of distributing two different items in the same vehicle, which is often considered as hard loading constraints in the classical vehicle routing problems. Schedules of items that must be loaded into each vehicle are computed in the packing stage. Then, these schedules are used to find an optimal path of each vehicle by using Travel Saleman Problem (TSP). So, the process comprises two steps: (1) assign delivery requests to vehicles (packing), and (2) find the best route for each vehicle (routing).

Introduction

The Pallet Packing Vehicle Routing Problem (PPVRP) indicates the real-world problems which distribution companies have been facing. The objective of PPVRP is to determine the overall usage of vehicles being used for distributing all items to their destinations. Basically, the PPVRP comprises by two separate problems: (1) packing and (2) routing. According to the packing features, all items are feasibly stacked into appropriate pallets for distribution. Thereafter, all pallets are determined to be subsequently loaded into each vehicle with the limitation of capacity. Therefore, the goal of PPVRP is to determine the packing conditions and minimising the number of vehicles using for distribution.

At the first stage of the classical delivery processes, decision makers consider the optimal path of routes. The perspectives of loading conditions are normally considered as side constraints at this stage. Then, after determining the routes, the second stage is that the schedules are forwarded to the warehouses where items must be packed into pallets and subsequently loaded onto the appropriate vehicles. However, during this second stage, if packing arrangements cannot be identified. For example, they are not feasible and optimal. The optimal set of routes must be reconsidered or needed additional vehicles. To address these issues, the PPVRP is introduced to eliminate overloaded assumptions [3]. Figure 1 reveals the process of PPVRP. As a result, sufficiency and accuracy of decision making is guaranteed.

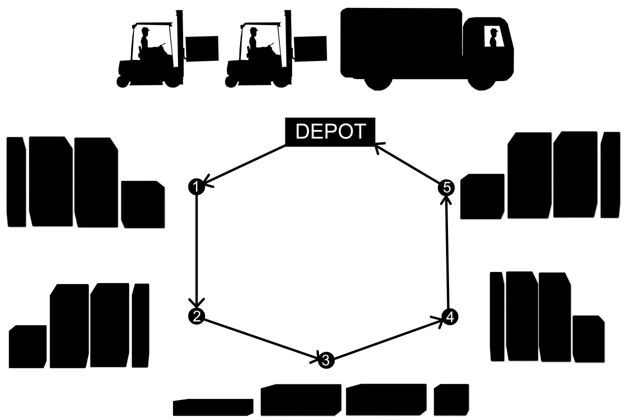

Figure 1: Illustration of a feasible packing arrangement under the PPVRP model

In literature, there are many research papers considering negative impacts of delivering items together. To eliminate the negative effects of each item with another items, extra constraints are constructed in the model. Examples may be found in grocery industry and delivering companies. There may be negative relation on two items due to their specification, such as chemical and fragile items. These items cannot be distributed or packed together to avoid the negative impacts. However, most papers only consider severe negative alignments that items cannot be packed together. In reality, a majority of cases indicates that items can be traveled together. Specifically, some items only require more expenditure on special packing or delivery services to reduce the negative impact of items being distributed together. To the best of our knowledge, the good alignments of packing two different items together have not been published in the field of research papers. For these reasons, our packing problem for PPVRP is formulated as Mixed-Integer Quadratic Programming.

By introducing the PPVRP, packing problem is considered first before determining the set of routes in routing problem. The model is a modification of the classical quadratic knapsack problem. If two items are distributed in the same vehicle, there will be an extra profit or penalty due to the specification of each item, as well as the profit of distributing two items. Furthermore, formulating the packing process as a quadratic programming allows us to consider the negative and positive alignments between two items as the lost and opportunity into the profit of distribution companies. The objective of this model is to maximise the total profit of delivering two items together. After schedules are determined in the previous stage, a standard Travel Saleman Problem (TSP) is used to find the best route for each vehicle to visit all given destinations.

Statement of Authorship

- Ponpot Jartnillaphand developed the theory behind this report, the code in Python using CPLEX package, generated the numerical experiments, reported and interpreted the results, and wrote this report.
- Dr. Hoa Bui developed the theory behind this report, supervised the work, assisted with the mathematical models and scenarios, and proofread this report.

1 Problem description and formulations

1.1 Problem description

This research focuses on a delivering problem of distributing different types of from one starting location to different destinations by using a limited number of vehicles. The destination of each item is according to delivering requests. Each item is selected to be distributed by an appropriate vehicle with the limitation of capacity. Each vehicle can complete more than one delivery request. Therefore, it can travel to multiple destinations. There are some extra profits when two appropriate items are distributed together. Additionally, extra expenditures for distributing items with special specification, such as fragile and chemical items, are taken into account as well. The company also has to pay for using each vehicle (eg. hiring trucks, drivers, etc.) along with the usual cost of transportation (eg. petrol). The delivering requests are forwarded to the warehouses where all items are packed into pallets and loaded into vehicles. Then, the optimal route for each vehicle is determined at this stage to distribute all scheduled items to their destinations. The objective of this problem is to maximise the total profit of all the selected items and to minimise the extra expenditures and the usage of vehicles with limited capacity. Therefore, the distribution process comprises two steps: (1) determining delivery requests for all selected items to be assigned into appropriate vehicles and (2) finding the best route for each vehicle to visit all given destinations.

1.2 Mathematical formulations

Let $\mathcal{K} := \{1, \ldots, K\}$ be the set of all vehicles, $\mathcal{N} := \{1, \ldots, N\}$ the set of all items (with $K, N \in N^*$), and $\mathcal{M} := \{v_1, \ldots, v_M\}$ the set of destinations ($M \in N^*$ and $v_i \in R^2$ for $i = 1, \ldots, M$, each vector v_i represents the coordinators location). The distance between each location, d_{ij}, is calculated by Euclidean norm in R^2. Let $\mathcal{Q}$ be the set of delivery requests, $\mathcal{Q} \subset \mathcal{N} \times \mathcal{M}$. Note that there is a $1-1$ correspondent map between $\mathcal{N}$ and $\mathcal{Q}$ (or $|\mathcal{M}| = |\mathcal{Q}|$), but some requests may have the same destinations. The profit of delivering item $i \in \mathcal{N}$ is $q_i > 0$, and the profit of distributing two items $i, j \in \mathcal{N}$ in one vehicle is P_{ij}. Note that P_{ij} can take value negative and positive value. We denote $Q := [P_{ij}]$ the $N \times N$ matrix whose $(i, j)^{\text{th}}$ element is P_{ij}, with all diagonal entries zero. It is easily seen that Q is symmetric. Each item i has fixed weight and volume w_i and v_i respectively and each vehicle k has a limitation in terms of weight c_k and volume v_k.

1.2.1 Packing Model

Regarding the packing problem, all items must be packed into each vehicle before distribution. To determine which items should be distributed by which vehicles, P_{ij} is introduced to consider the positive and negative alignments of distributing several items together. Profitability increases if the appropriate parcels are distributed together. Meanwhile, it can also decrease as there are extra payments of safety or protection for the items with specific types. This type of problems can be formulated as Quadratic Programming, where the objective function is quadratic and constraints are linear. This formulation is not only commonly seen in logistic companies, but also appeared in many other fields. Therefore, Quadratic Programming is introduced to minimise the most appropriate expenditure and to maximise the benefits of all the distributed items.

The binary variables $x_{ik} \in \{0,1\}$ ($i \in \mathcal{N}$, $k \in \mathcal{K}$) are defined as the decision variables. They control whether item i is selected to be distributed by vehicle k or not (i.e. $x_{ik} = 1$). When item i and j are selected to be delivered together by vehicle k, then $x_{ik}x_{jk} = 1$. This is a trigger allowing us to accommodate the negative and positive relations between two items as the lost and opportunity into the total profit for the companies. In addition, when a vehicle is used, there is some cost associating with the use of these vehicles. Therefore, we introduce an additional decision variable $e_k \in \{0,1\}$. If vehicle $k \in \mathcal{K}$ is used for the delivering process, then $e_k = 1$. The model is stated as follows

$$\max \qquad \sum_{\substack{i \in \mathcal{N} \\ k \in \mathcal{K}}} q_i x_{imk} + \frac{1}{2} \sum_{\substack{i,j \in \mathcal{N} \\ k \in \mathcal{K}}} P_{ij} x_{ik} x_{jk} - \sum_{k \in \mathcal{K}} d_k e_k \tag{Packing 1}$$

Substitute to Constraints:

Constraint (1) enforces that for $k \in \mathcal{K}$ if any item i is delivered by k i.e. $x_{ik} = 1$, then vehicle k is active or $e_k = 1$ and the cost d_k for using k must be paid. Note that when $\sum_{i \in \mathcal{N}} x_{ik} = 0$, no request is assigned to k, then $e_k = 0$. Normally, the constraint $\sum_{i \in \mathcal{N}} x_{ik} \geq e_k$ is used to enforce $e_k = 0$ in that event, however due to the fact that $d_k > 0$ and (Packing 1) is a maximisation problem, this constraint is unnecessary

$$x_{ik} \leq e_k, \quad \forall i \in \mathcal{N}, k \in \mathcal{K}. \tag{1}$$

Constraint (2) and (3) is the standard loading constraint to ensure loading items does not exceed the capacity of the vehicle

$$\sum_{i \in \mathcal{N}} w_i x_{ik} \leq c_k, \quad \forall k \in \mathcal{K} \tag{2}$$

$$\sum_{i \in \mathcal{N}} v_i x_{ik} \leq v_k, \quad \forall k \in \mathcal{K}. \tag{3}$$

Constraint (4) captures the fact that one request is only completed by only one vehicle. In practice, there may have some other additional constraints. For instance, for some request $i \in \mathcal{N}$, only the vehicles in $K_i \subset \mathcal{K}$ can complete i. In this case, we impose constraint $x_{ik} = 0$ if $k \notin K_i$

$$\sum_{k \in \mathcal{K}} x_{ik} \leq 1, \quad \forall i \in \mathcal{N} \tag{4}$$

$$x_{ik}, e_k \in \{0, 1\}.$$

The objective function of the problem is the total profit including the profit of completing a request $i \in \mathcal{N}$ along with the profit of distributing two different items in a vehicle and the cost of using vehicles. The model (Packing 1) allows to not schedule some items in $\mathcal{N}$. However, in some delivering problems, this is not an option, i.e., all requests must be completed. The profit of completing each individual item is removed from the objective function. Model (Packing 1) can be reformulated as follows

$$\max \qquad \frac{1}{2} \sum_{\substack{i,j \in \mathcal{N} \\ k \in \mathcal{K}}} P_{ij} x_{ik} x_{jk} - \sum_{k \in \mathcal{K}} d_k e_k \tag{Packing 2}$$

Substitute to Constraints:

Constraint (1), (2), and (3) hold.

Constraint (4) is replaced by constraint (5), which guarantees for any request there is exactly one vehicle delivering it,

$$\sum_{k \in \mathcal{K}} x_{ik} = 1, \quad \forall i \in \mathcal{N} \tag{5}$$

$$x_{ik}, e_k \in \{0, 1\}.$$

We consider a special case when the cost of using each vehicle is the same, i.e., $d_k = d$ for all $k \in \mathcal{K}$. Then minimising the cost of using vehicle is equivalent to minimising the number of vehicles used for delivering. In this case, we can eliminate the decision variables e_k as follows. Let $\mathcal{J}_1 := \{(i,j) \in \mathcal{N} \times \mathcal{N} : P_{ij} < -d\}$ and $\mathcal{J}_2 := \{(i,j) \in \mathcal{N} \times \mathcal{N} : P_{ij} \geq -d\}$. For the pair $(i,j) \in \mathcal{J}_1$, the penalty of distributing them together is more than the cost of using a particular vehicle to deliver each of them. This is the case when two items in i and j cannot be distributed together in one vehicle. Therefore we need to impose a hard constraint to prevent that event:

$$x_{ik} + x_{jk} \leq 1, \quad \forall k \in \mathcal{K}. \tag{6}$$

For the pair in $\mathcal{J}_2$, packing them in one vehicle is always more favorable than hiring one more vehicle. The objective function of model (Packing 2) becomes maximising the total profit of distributing different items, as well as each individual item, in one vehicle. Therefore, we can turn all entries in $Q = [P_{ij}]$ to positive by adding a positive number $M_0 > d$ to each entry P_{ij}. This transformation does not change the nature of the problem. For that reason, we can assume that $P_{ij} > 0$ and we can rewrite the packing model as

$$\max \quad \underbrace{\frac{1}{2}\sum_{\substack{i,j\in\mathcal{N}\\ k\in\mathcal{K}}} P_{ij}x_{ik}x_{jk}}_{\text{Profit of selecting two items together}} + \underbrace{\sum_{\substack{i\in\mathcal{N}\\ k\in\mathcal{K}}} q_i x_{ik}}_{\text{Profit of selecting an item}} \qquad \text{(Packing 3)}$$

$$\text{s.t.} \quad (2), (3), (5), (6) \text{ hold.}$$

1.2.2 Routing Model

In terms of the routing problem, after determining schedules of packing, the optimal route of one vehicle is considered. As the schedules cannot be change anymore, determining the optimal path for each scheduled vehicle can be done one at the time. Then, routing becomes routing problem with one vehicle. In other words, this problem can now be seen as Travel Saleman Problem (TSP). Regarding TSP, it is to calculate the shortest possible paths by distances between each destination [2]. The vehicle starts from the starting point and then return to the point where it starts. This problem is considered as a NP-hard problem as only one destination can be visited once during each route. The example of optimal route using TSP can be seen in figure 2.

2 Solution Methods

2.1 Glover's Linearisation

The formulation of Quadratic Programming is difficult to be solved due to its non-linear form. There is still no existing algorithm to solve this problem in polynomial time. However, possible solutions can still be computed with suitable solution methods, such as Linearisation, Cuttin

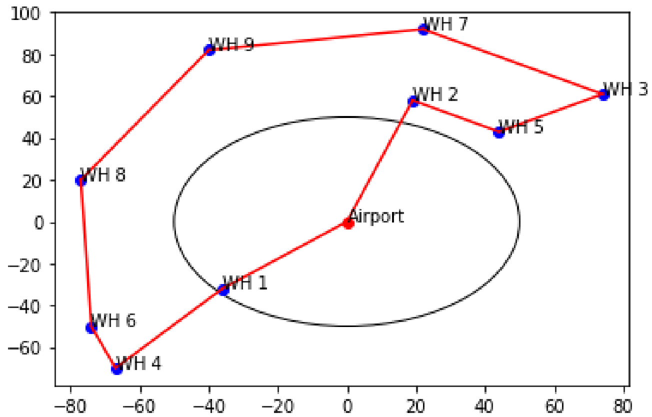

Figure 2: Example of an optimal route using TSP

Planes, and Dynamic Programming Heuristic. Regarding our packing model, It is not standard linearisation but inspired by Glover's Linearisation to address the issue of non-linear form. The Glover's Lineasisation is more concise compared to the Standard Linearisation [1]. In the formulation, the expression of $\sum_{j=1, i\neq j}^{\mathcal{N}} P_{ij} x_i x_j$ is replaced with continuous variables z_i, which are the total profit of each individual item i. Note that L_i and U_i are lower and upper bounds respectively on $\sum_{j=1, i\neq j}^{\mathcal{N}} P_{ij} x_j$. Thus, standard solvers can now be used to optimally compute objective solution. The modified model (Packing 3) with new constraints is stated as follows

$$\max \quad \underbrace{\frac{1}{2}\sum_{i\in\mathcal{N}} z_i}_{\text{the total profit of each individual item } i} + \underbrace{\sum_{\substack{i\in\mathcal{N}\\ k\in\mathcal{K}}} q_i x_{ik}}_{\text{Profit of selecting an item}} \qquad \text{(Glover's Packing)}$$

Substitute to Constraints:

Constraint (2),(3),(5), and (6) hold.

Constrain (7) is to ensure that z_i is less than or equal to the upper bound U_i when item i is selected.

$$z_i \leqslant U_i x_{ik} \quad \forall i \in \mathcal{N}, \forall k \in \mathcal{K}. \tag{7}$$

Constrain (8) is to ensure the lower bound of z_i

$$\sum_{j \in \mathcal{N}} P_{ij} x_{jk} + U_i(1 - x_{ik}) \geqslant z_i \quad \forall i \in \mathcal{N}, \forall k \in \mathcal{K}. \tag{8}$$

2.2 Miller-Tucker-Zemlin (MTZ) Formulation

To formulate the TSP as an binary integer linear program, there are many well-known formulations. MTZ is one of the useful formulations [2]. The binary variables $x_{ij} \in 0, 1$ $(i, j \in \mathcal{M})$ are definded as the decision variables. They control whether the vehicle is selected to travel a path from node i to j or not (i.e. $x_{ij} = 1$). The dummy variables u_i represent the order of destination that the vehicle must visit. The model is stated as follows

$$\min \quad \underbrace{\sum_{\substack{i,j \in \mathcal{N} \\ i \neq j}} d_{ij} x_{ij}}_{\text{the total distances vehicle } k \text{ is traveling}} \tag{MTZ}$$

Substitute to Constraints:

Constrain (9) is to ensure that the vehicle must exactly arrive at one other node

$$\sum_{i \in \mathcal{M}, i \neq j} x_{ij} = 1 \quad \forall j \in \mathcal{M}. \tag{9}$$

Constrain (10) is to ensure that the vehicle must exactly depart to one other node

$$\sum_{j \in \mathcal{M}, j \neq i} x_{ij} = 1 \quad \forall i \in \mathcal{M}. \tag{10}$$

Constrain (11) and (12) is to ensure that there is no multiple tour to cover all nodes in the optimal route.

$$u_i - u_j + v_m x_{ij} \leqslant v_m - 1 \quad 2 \leqslant i \neq j \leqslant v_m, \tag{11}$$

$$1 \leqslant u_i \leqslant v_m - 1 \quad 2 \leqslant i \leqslant v_m. \tag{12}$$

3 Computational experiments

3.1 Scenario

We assume that there are identical vehicles for distributing all items to their warehouses. Distributing all items with same destination together can surely provide the profit. However, in some scenarios, not all items can be shipped together with others. For example, in case there is not enough capacity in vehicle k, dispatching additional vehicles is to a way to address this issue. Some items can also provide less cost when distributing with others, even though they have a different destination. Therefore, all items should be distributed by any vehicle regardless to their destination. This scenario is that which items are distributed by which vehicles This is identified by packing problem. Thereafter, the vehicles just follow the optimal routes created by using TSP to find the order of the warehouses that each vehicle will visit. The aim is to consider that which item will be distributed together.

We compute model (Glover's Packing) firstly to obtain delivery requests for the items to be packed into the vehicle. Then, we find the optimal route for that vehicle. Before starting the next vehicle, we eliminate those chosen items. The process is to run until there is no items left. In general, we cannot do this. But this is a special case with identical vehicles. So we can do it.

3.1.1 Packing Input

The instances of Quadratic Programming are randomly generated for testing and experimenting. P_{ij} contains a symmetric positive integer matrix with the random distributed value in [1,100]. We also consider the area zones. For example if the warehouse is more than 50 distance unit apart from the origin, the cost of distribution is increased. The weight and volume of each item is randomly distributed in [1, 50]. The limited capacity and volume of vehicles are also randomly distributed in [50, $\sum_{i=1}^{\mathcal{N}} c_i x_{ik}/\mathcal{NV}$] and [50, $\sum_{i=1}^{\mathcal{N}} v_i x_{ik}/\mathcal{NV}$] respectively. Note that $\mathcal{NV}$ is the number of vehicles used for division, which affects the running time of our model. We have to carefully choose it.

3.1.2 Routing Input

The instances using for routing problem are randomly generated for testing and experimenting. The location of all warehouses is randomly generated into x and y axis between [-100,100]. Then, the Euclidean Distance are used to find the distance between each warehouse. An origin of the vehicles to start is at (0,0) to which is also the point where all vehicles must return. In figure 3 below, the origin is surrounded by a circle which indicates zone 1 with a radius of 50 distance unit. The area outside the circle is zone 2, which requires more travelling cost added into P_{ij}.

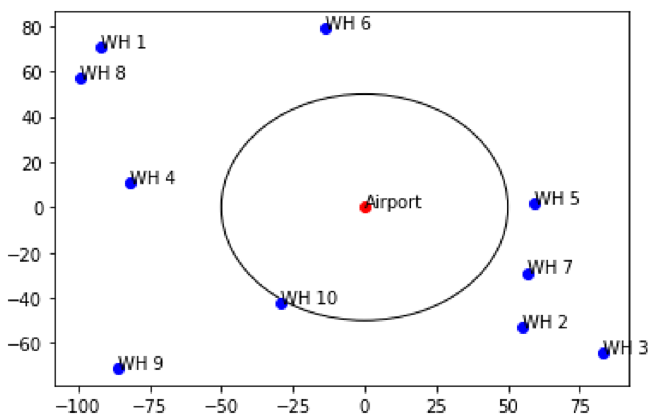

Figure 3: Demonstration of 10 warehouse (WH 1-10) locations and an origin point (Airport)

3.2 Numerical Results

Computational experiments were run on ASUS-pc with an AMD Ryzen 7 4800H Processor and RAM 16 GB. The table below reveals computational results. The number of items are in range [100,160]. The number of warehouses are in range [7,10]. Note that $\mathcal{NV}$ is equal to 2 as we would like to assume that there are two vehicles to complete each delivery request. Note that the unit of all values in the table are in second.

Warehouses \ Items	100	120	140	160	180
7	11.68	14	173.67	120.42	> 5000
8	33	47	896	246	> 5000
9	16.8	14.32	87.44	1038.77	> 5000
10	6.86	22.22	158.8	415.46	> 5000
AVG	33.668	43.508	291.182	396.13	> 5000

Table 1: Computational Results with units in second

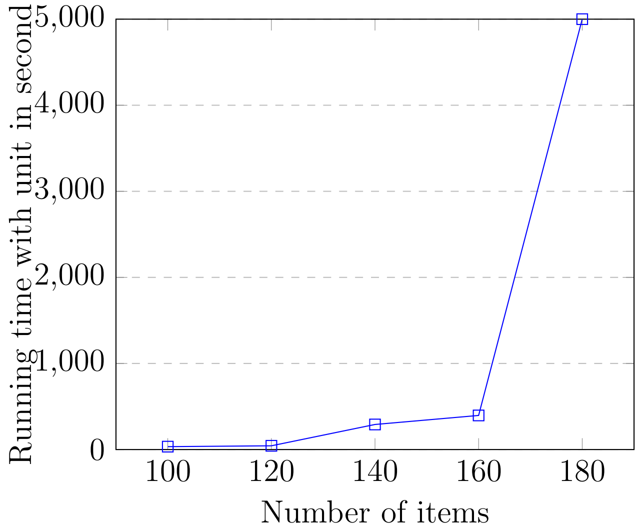

Figure 4: Line graph of average running time against number of items

As we can see in table 1, the running times are slightly fluctuated between different numbers of items and warehouses. It is undeniable that randomly generated instances are one of the reasons. The capacity of vehicles has a huge influence on computational complexity as well. The running time with small usage of vehicles is faster than big usage of vehicles. However, the average running time clearly indicates that it increases due to the number of items in distribution system. This can be clearly seen in figure 4. In terms of the limitation of our model, it can solve 160 items in distribution system with any number of warehouses.

4 Conclusion

In this research, the objective is to explore Pallet-Packing Vehicle Routing Problem (PPVRP) via Quadratic Programming. PPVRP is introduced to address the issues of classical delivery process due to infeasible packing arrangements. Two separate problems comprise PPVRP: (1) packing problem and (2) routing problem. The model of packing problem is formulated as Quadratic Programming because positive and negative alignments of distributing two items together are considered. Delivery requests are determined in packing stage. The schedules of items are forwarded to a warehouse for items to be packed and loaded onto each vehicles. The aim of routing problem stage is to identify the optimal routes of all identical vehicles. Travel Saleman Problem (TSP) is used as the route is computed for each vehicle one at the time. The model of our routing problem is Miller-Tucker-Zemlin (MTZ) Formulation. In regard to numerical results, a special scenario is introduced to test our model. All vehicles are identical. Instances for experiments are randomly generated. The running time increases as the number of items in transportation system rises. The limitation of our model is 160 items with any reasonable number of destinations.

In the future, more general scenarios must be involved to ensure whether our strategies are efficient. In some scenarios, solving two problems, packing and routing problem, separately may not be satisfying. Furthermore, other factors according to real world problem must be included as well, such as time windows of items and vehicles. There maybe some other formulations of TSP that is better than what we are using in this project. Those formulations are expected to be explored in the future.

Acknowledgement

Special thank to Dr. Hoa Bui for her constructive comments on a draft of this project. She is also a viewer, whose comments have helped to improve the project. This support is gratefully acknowledged.

References

[1] Warren P. Adams, Richard J. Forrester, and Fred W. Glover. Comparisons and enhancement strategies for linearizing mixed 0-1 quadratic programs. *Discrete Optimization*, 1(2):99 – 120, 2004.

[2] C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. *J. ACM*, 7(4):326–329, October 1960.

[3] Emmanouil E. Zachariadis, Christos D. Tarantilis, and Chris T. Kiranoudis. The pallet-packing vehicle routing problem. *Transportation Science*, 46(3):341–358, 2012.