AN OPTIMIZATION MODEL FOR THE PACKAGE ASSIGNMENT PROBLEM FOR ONLINE RETAILERS

Cenk Çalışkan
Department of Strategic Management and Operations
Woodbury School of Business
Utah Valley University
800 W. University Pkwy
Orem, UT 84058
(801) 863-6487
cenk.caliskan@uvu.edu

ABSTRACT

In this research, we consider the problem of optimally assigning packages to shipment methods that is usually encountered by online retailers. In fulfillment centers, customer orders are assembled into physical packages that are ready to be shipped after picking, packing and inspection operations. Each package has its size and promised delivery date as attributes, and each shipment method has the carrier name, cutoff time, expected delivery time, capacity and service type (next day, second day, etc.) as attributes. The problem is then assigning the packages to shipment methods so that the promised delivery dates for packages will be satisfied, capacities of shipment methods will not be exceeded, and total shipment costs will be minimized. A special case of this assignment problem is the generalized assignment problem (GAP), which is known to be NP-hard. We propose a Lagrangian relaxation based solution method for the problem. We also propose intuitive, easily implementable decision rules that can provide good heuristic solutions.

Keywords: assignment problem, generalized assignment, Lagrangian relaxation, branch-and-bound, package assignment.

INTRODUCTION

Online retailers operate large warehouses or distribution centers that are used to receive the goods from suppliers, store them until they are shipped to customers and fulfill customer orders by picking the individual items, assembling them into packages and shipping the packages. These facilities are sometimes called *fulfillment centers*. A typical customer puts an order for a number of items online and given a promised delivery date, usually based on the shipping option selected by the customer. These orders are then placed in a queue; and for each order, the items are picked from their locations in the warehouse by order pickers and put on a conveyor belt. Then, items on the conveyor belt are assembled into packages by order assemblers. After visual inspection, packages are weighed and compared against their computer-generated expected weight to make sure that they contain correct items and there are no missing or extra parts or items. If they pass all checks and inspections, the packages are closed and they become ready for the assignment of shipment methods. A shipment method is a record with the following attributes: name of the package carrier, the type of service (next day, second day, etc.), the cutoff time for loading onto the vehicle, capacity of the vehicle and expected delivery time. Each carrier may have multiple vehicles leaving the fulfillment center on a given day, each having a different cutoff time. Packages with different service types may be shuttled to the carrier's consolidation center by the same vehicle, but they are still considered different shipment methods since their expected delivery times are different. The rates (per weight or per volume) differ among carriers and depending on the type of service.

The problem is to assign a shipment method to each package so that each package's promised delivery date will be met, capacity of shipment methods will not be violated and the total shipping costs will be minimized. Even though the customer selects a particular service level when placing the order, there is still opportunity for the retailer to optimize shipping costs by appropriately choosing from eligible shipment methods. Besides, retailers sometimes offer free shipping as an incentive, in which case there is no service level selected by the customer and the retailer has to optimize the shipping costs.

A special case of this assignment problem is the classical generalized assignment problem (GAP). Packages in this assignment problem become tasks, and shipment methods become agents of the GAP. The GAP is known to be NP-hard, and there are many optimizing or heuristic approaches to solve it. Nauss (2003), for example, proposes a specialized branch-and-bound algorithm for the GAP. There are also variants of the GAP, such as multilevel generalized assignment problem, which is also NP-hard. Laguna et al. (1995) proposes a tabu search algorithm for the multilevel generalized assignment problem, whereas Osorio & Laguna (2003) propose logic cuts within the context of the classical branch-and-bound algorithm. Yano & Newman (2001) consider assignment of containers to trains dynamically between a depot and a destination, but their model is deterministic and does not allow heterogenous containers and trains. Powell (1996) considers a stochastic assignment problem in which trucks are assigned to loads, but it also has a spatial dimension, i.e. the agents (the trucks) are moving over a network, and repositioned full or empty to take advantage of potential future demand.

PROBLEM FORMULATION

The package assignment problem that we study in this research is essentially the classical assignment problem with added side constraints. The classical assignment problem is a special case of the minimum cost network flow problem which can be solved efficiently as an LP. Because its constraint matrix is totally unimodular, it results in an integer optimal solution when the LP is solved. However, when there are side constraints, the problem loses its total unimodularity, so it has to be solved as an integer program.

Let the set of packages to be assigned be P, and the set of shipment methods, S. We denote the cost of assigning package $i \in P$ to shipment method $j \in S$, c_{ij}. We also denote whether or not package i is assigned to shipment method j, as the binary variable x_{ij}. All packages must be assigned to exactly one shipment method. This translates into the following assignment constraint:

$$\sum_{j=1}^{S} x_{ij} = 1 \qquad \forall i \in P. \tag{1}$$

We denote the size (in lbs.) of package $i \in P$ as s_i, and the capacity (in lbs.) of shipment method $j \in S$, r_j. Total amount of packages assigned to a shipment method is constrained by its capacity, which results in the following constraint:

$$\sum_{i \in P} s_i x_{ij} \leq r_j \qquad \forall j \in S. \tag{2}$$

We denote the promised delivery time of package $i \in P$ as t_i^p, and the expected delivery time of shipment method $j \in S$, t_j^d. The package has to be shipped by its promised delivery time, so the following constraint has to be added to the model:

$$t_j^d x_{ij} \leq t_i^p x_{ij} \qquad \forall i \in P \text{ and } \forall j \in S. \tag{3}$$

The time at which package $i \in P$ becomes available for shipment is denoted by t_i^a, and the cutoff time for shipment method $j \in S$ to be assigned to a package is denoted by t_j^c, Then, the following inequality must hold:

$$t_i^a x_{ij} \leq t_j^c x_{ij} \qquad \forall i \in P \text{ and } \forall j \in S. \tag{4}$$

The problem can then be formulated as a binary integer linear programming model as follows:

$$\text{[P1]} \qquad \min z = \sum_{i \in P} \sum_{j \in S} c_{ij} x_{ij} \tag{5}$$

$s.t.$

$$\sum_{j \in S} x_{ij} = 1 \qquad \forall i \in P \tag{6}$$

$$\sum_{i \in P} s_i x_{ij} \leq r_j \qquad \forall j \in S \tag{7}$$

$$t_j^d x_{ij} \leq t_i^p x_{ij} \qquad \forall i \in P \text{ and } j \in S \tag{8}$$

$$t_i^a x_{ij} \leq t_j^c x_{ij} \qquad \forall i \in P \text{ and } j \in S \tag{9}$$

$$x_{ij} = 0 \textit{ or } 1, \qquad \forall i \in P \text{ and } j \in S. \tag{10}$$

If $t_i^a = 0$ and $t_i^p \geq \max_{j \in S} t_j^d \ \forall i \in P$, which means all packages are available at the beginning of the planning horizon and the promised delivery times of all packages are later than the latest expected delivery time among all shipment methods, Eqs. 8 and 9 become redundant and can be dropped from the model. Then, the package assignment problem reduces to the classical GAP. Since the classical GAP is a special case of the package assignment problem and it is NP-hard, the package assignment problem is also an NP-hard problem.

LAGRANGIAN RELAXATION

If we relax Equation 7, the resulting problem is a many-to-one assignment problem in which each package is assignable to a subset of the shipment methods, where the subsets are defined by Equations 8 and 9. We define the Lagrange multipliers λ_j for $j \in S$ corresponding to Equation 7. Let X be the set of x that satisfy Equations 6-10. We then relax Equation 7 and bring it into the objective function, resulting in the following Lagrangian subproblem or Lagrangian function:

$$L(\lambda) = \min_{\lambda \geq 0} \{ \sum_{i \in P} \sum_{j \in S} c_{ij} x_{ij} + \sum_{j \in S} \lambda_j (\sum_{i \in P} s_i x_{ij} - r_j) : x \in X \}.$$

This can be re-written as:

$$\text{[P2]} \qquad L(\lambda) = \min_{\lambda \geq 0} \{ \sum_{i \in P} \sum_{j \in S} (c_{ij} + \lambda_j s_i) x_{ij} - \sum_{j \in S} \lambda_j r_j : x \in X \} \tag{11}$$

Let $\delta_j = \sum_{i \in P} s_i x_{ij} - r_j$ for $j \in S$. Then, Equation 7 can be re-written in the form $\delta \leq 0$.

Lemma 1. *For any $\lambda \geq 0$, the value $L(\lambda)$ of the Lagrangian function is a lower bound on the optimal objective function value z^* of problem P1.*

Proof: Let x^* be an optimal solution to problem P1 and $\hat{x}$ be an optimal solution to problem P2. Then,

$$z^* = cx^* \geq cx^* + \lambda(sx^* - r) \geq c\hat{x} + \lambda(s\hat{x} - r) \tag{12}$$

The first part of this inequality is because x^* is a feasible solution to problem P1, and the second part is because $\hat{x}$ is an optimal solution to problem P2. □

Solving the Relaxed Assignment Problem

The relaxed problem is a simple assignment problem in which each package has to be assigned to a shipment method. For each package $i \in P$, there is a set $S_i \subset S$ which includes the shipment methods assignable to package i. S_i is the set of shipment methods that satisfy the constraints 8 and 9. In other words, $S_i = \{j \in S \mid t_j^d \leq t_i^p \text{ and } t_i^a \leq t_j^c\}$.

Lemma 2. *Let x^* be an optimal solution to the relaxed assignment problem. Then, for all $i \in P$, $x^*_{ij} = 1$ for $j = \text{argmin}_{k \in S_i}\{c_{ij}\}$ and $x^*_{ij} = 0$ for any other $j \in S_i$.*

Proof: Assume that x^* is not optimal. Then, for some $i \in P$, there exists a shipment method $k \in S_i$ for which $c_{ik} < c_{ij}$. But this contradicts the fact that c_{ij} is the minimum cost shipment method in S_i. □

We therefore obtain the optimal solution to the relaxed problem by simply assigning each package to its lowest cost shipment method.

Solving the Lagrangian Subproblem

We obtain an immediate lower bound for problem P1 when we solve the relaxed assignment problem for any value of λ. But in order to obtain the tightest bounds, we need to maximize the Lagrangian function, which is also called the Lagrangian subproblem or Lagrangian dual:

$$L^* = \max_{\lambda \geq 0} L(\lambda). \tag{13}$$

In order to solve the Lagrangian dual, we use a subgradient optimization procedure. We use an adaptation of the Newton's method in which we update the Lagrange multipliers in a relatively large step towards the optimal solution along the subgradient direction. The initial value of the multipliers are set to zero, and at every iteration, they are updated as follows:

$$\lambda_j^{t+1} = \max\{0, \lambda_j^t + \theta_t \delta_j\} \qquad \forall j \in S \tag{14}$$

where θ_t is the step size at iteration t. In order to make sure that the procedure will converge, the step size is chosen as follows (see, for instance, Ahuja et al. (1993)):

$$\theta_t = \frac{\sigma_t[UB - L(\lambda^t)]}{||\delta||^2} \tag{15}$$

where $||\delta|| = (\sum_i \delta_i^2)^{0.5}$ is the Euclidean norm of δ and σ_k is a scalar that is chosen strictly between 0 and 2. The initial value of σ is 2, and it is halved once the Lagrangian objective function fails to increase after a long series of iterations. UB is the surrogate for the optimal Lagrangian objective function value, L^* and it is chosen as the best upper bound, i.e. the minimum objective function value of the feasible solutions found thus far in the algorithm.

Note that during the branch and bound algorithm, the iterations of the subgradient method may be stopped whenever $L(\lambda) \geq UB$ as that means we cannot obtain a better feasible solution than the best existing solution from this branch and bound node. We describe the branch and bound algorithm in the next section.

THE BRANCH AND BOUND ALGORITHM

At the beginning of the branch and bound algorithm, we solve problem P2 with $\lambda = 0$. If the resulting solution is feasible, than it is an optimal solution to problem P1 as well. If it is not feasible, we execute the subgradient procedure to solve the Lagrangian dual. If the resulting solution is feasible, we might have obtained the optimal solution. In order to determine if that is the case, we check the optimality conditions as follows.

Lemma 3 (Optimality Conditions). *An optimal solution (x^*, λ^*) to the Lagrangian subproblem $L(\lambda)$ (problem P2) is also an optimal solution to the package assignment problem (problem P1) if and only if:*

(i) $sx^* \leq r$
(ii) $\lambda^*(sx^* - r) = 0$ (Complementary slackness)

Proof: Condition (i) implies that x^* is a feasible solution. Since $L(\lambda)$ is a lower bound for the optimal cost of problem P1, we have: $cx^* \geq L(\lambda^*) = cx^* + \lambda(sx^* - r)$. By condition (ii), we have $cx^* = L(\lambda^*)$. □

If the solution is feasible after the subgradient procedure, we keep it as a candidate for optimal solution and set its total cost as the upper bound (UB). We apply branching to the solution of the relaxed problem before the subgradient procedure (which was not feasible). If the solution is infeasible, we apply branching to this solution.

We branch as follows. We pick shipment method $n \in S$ with the highest capacity violation. We then find package $p \in P$ that is assigned to n with the highest weight. We create two branches: (1) we set $x_{pn} = 1$; (2) we set $x_{pn} = 0$. Clearly, these two branches are mutually exclusive and divide the search region in two.

We then select an unexamined branch and bound node and repeat the same procedure until all of the created branches are fathomed, at which time we will have found the optimal solution to problem P1. If a branch has a lower bound greater than the best UB, that branch is fathomed because it cannot yield a better solution. If a branch yields a feasible solution and satisfies Lemma 3, then that branch is fathomed as well, as we can't find a better solution by further branching from that node. We update the best UB if this feasible solution has a lower total cost.

DECISION RULES

The Lagrangian relaxation based branch and bound method that we propose in this paper is expected to be computationally efficient. Lagrangian subproblems do not require any algorithm to solve them, since the optimal solutions are determined in a straightforward manner, as we describe in Section . However, it is also beneficial to have some intuitive and easily implementable decision rules. It will be interesting to compare the performance of the branch and bound algorithm with that of the decision rules. The following are the decision rules that we propose in this paper.

Greedy: This is the simplest of the decision rules. We simply assign each package as it becomes available to the cheapest shipping method available that has enough capacity and satisfies the promised delivery date of the package.

Look ahead: We determine a possible sequence of orders after adding processing time to each order that depend on the number and type of items in each package. We pick the first n packages in the list, add the packages that are already available to be shipped and sort them according to some priority rule (e.g., we can sort by size), and greedily assign them to shipment methods as in greedy assignment. While assigning the packages, we don't make the actual assignment for the packages that aren't ready yet, but we reduce the capacities of the shipment methods as if the assignment actually happened. This reserves the capacity for the possible future arrivals. In the next round of assignments, we repeat the procedure but we use the actual capacities.

Buffer: We wait until n packages are ready to be shipped. We then assign any package immediately if the cutoff time is near for the latest feasible shipment method for the package. Then we sort the packages by some priority rule (again, we can use size here) and then we greedily assign them to the cheapest available shipment method that does not violate the promised delivery date.

CONCLUSION

In this research we propose a Lagrangian relaxation based branch and bound algorithm to solve the package assignment problem that is encountered by major online retailers. We use Lagrangian relaxation to obtain a tight lower bound at each branch and bound node. Lagrangian relaxation lower bounds are usually tighter than the lower bounds generated by simply relaxing the integrality constraints. In this problem, Lagrangian relaxation creates a trivial problem which has an immediate solution, so we expect a fast solution procedure overall. We describe the details of the algorithm in this paper. We also describe a few intuitive decision rules that can be used to heuristically solve the problem in a dynamic setting. We will test the computational efficiency of the exact branch and bound method as well as the solution quality of the heuristic decision rules in future research. We will also compare our algorithm with general integer programming solvers such as CPLEX.

References

Ahuja, R., Magnanti, T., & Orlin, J. (1993). *Network flows: Theory, algorithms and applications*. Englewood Cliffs, NJ: Prentice-Hall.

Laguna, M., Kelly, J. P., González-Velarde, J. L., & Glover, F. (1995). Tabu search for the multilevel generalized assignment problem. *European Journal of Operational Research*, 82(1), 176–189.

Nauss, R. M. (2003). Solving the generalized assignment problem: An optimizing and heuristic approach. *INFORMS Journal on Computing*, 15(3), 249–266.

Osorio, M. A. & Laguna, M. (2003). Logic cuts for multilevel generalized assignment problems. *European Journal of Operational Research*, 151(1), 238–246.

Powell, W. B. (1996). A stochastic formulation of the dynamic assignment problem with an application to truckload motor carriers. *Transportation Science*, 30(3), 195–219.

Yano, C. A. & Newman, A. M. (2001). Scheduling trains and containers with due dates and dynamic arrivals. *Transportation Science*, 35(2), 181–191.