Journal of Industrial and Systems Engineering
Vol. 2, No. 1, pp 1-15
Spring 2008

Door Allocations to Origins and Destinations at Less-than-Truckload Trucking Terminals

Vincent F. Yu[1], Dushyant Sharma[2], Katta G. Murty[3*]

[1]Department of Industrial Management, National Taiwan University of Science and Technology, Taiwan
vincent@mail.ntust.edu.tw

[2]Equity Trading Lab, Morgan Stanley, 1585 Broadway Ave, New York, NY 10036
Dushyant.Sharma@MorganStanley.com

[3]Department of Industrial and Operations Engineering, University of Michigan, USA
murty@umich.edu

ABSTRACT

For an LTL (Less-than Truckload) carrier, the allocation of doors at a consolidation facility to outbound trailers assigned to various destinations, and to inbound trailers in the continuous stream arriving from various origins, has a significant impact on its operations, and on the nightly man-hours needed for consolidation. In the past literature door allocations to destinations of outbound trailers are determined using deterministic mathematical models based on average volumes of shipments between origin-destination pairs. The online nature of allocation of doors to inbound trailers is either ignored or simple rules like FCFS (First Come First Served) are assumed that do not take advantage of the data on the trailer's actual contents readily available at the time of its arrival. In reality the actual shipment volume between any origin-destination pair varies significantly from day to day. Due to this wide variation destination door allocations that are optimal for the average volume tend to be far from optimum for most nights. Also, very simple on-line policies for door allocation to each inbound truck at the time of its arrival based on its actual contents can significantly reduce the man-hours needed to consolidate its contents.

In this paper we develop a new model that uses such an on-line policy for door allocations to inbound trailers, and determines doors to allocate to destinations to minimize the expected man-hours for consolidating freight nightly taking the random variation in freight volumes into account. Computational results on data from an actual facility indicate that the man-hour requirement can be reduced by over 20% compared to current practice.

***Keywords*:** Cross-docking, Less-than-truckload freight terminal, Man-hours for consolidation

1. INTRODUCTION

Freight transportation is a $ 580 billion industry in U.S.A. in annual revenue in 2001, of which trucking has an 80% role (Bureau of Transportation Statistics 2003). **Less-than-truckload (LTL)**

---

* Corresponding Author

carriers are trucking firms that specialize in moving smaller shipments, typically weighting less than 10,000 pounds. Since they commonly use 28-foot trailers that can carry approximately 20,000 pounds, LTL carriers have to consolidate multiple customer orders onto the same trailer, thus leading to an important freight consolidation issue.

Most national LTL carriers use a **hub-and-spoke network** for transporting freight. In such a network, terminals of an LTL carrier are classified into two types: **End-of-line terminals (EOLTs)** and **Intermediate Consolidation Terminals (ICTs)**. All inbound freight to an EOLT comes through an ICT with which it is affiliated, and all outbound freight from that EOLT goes out through such an ICT. In reality an EOLT may be affiliated with a different ICT for freight going to different destinations and coming from different origins.

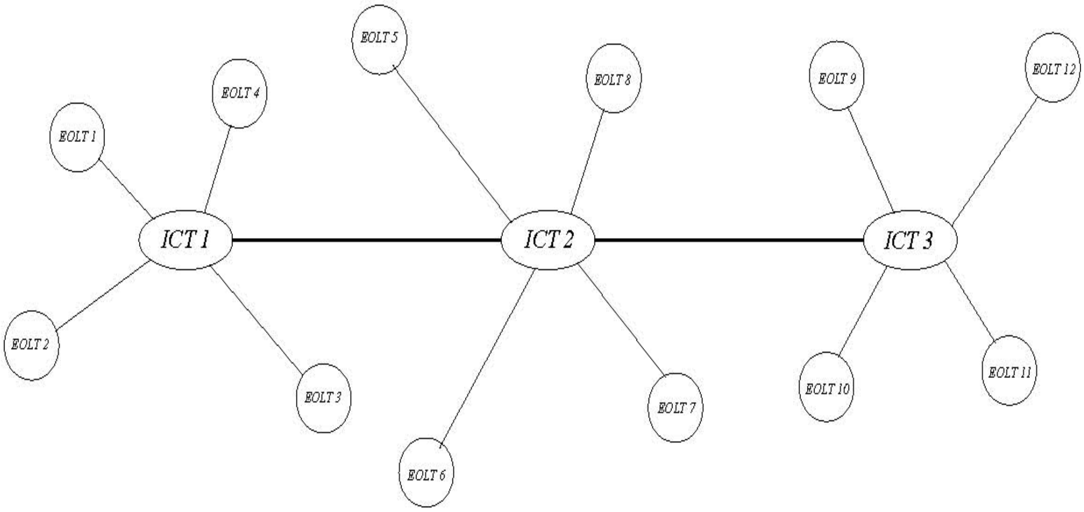

Figure 1. Example of a hub-and-spoke network used by LTL carriers.

At each EOLT freight from various customers (possibly with different destinations) is received during the day. This freight is packed on pallets so that each pallet contains freight to a single destination only. The pallets are loaded into a trailer and taken to an ICT where it is sorted and repacked into separate trailers by destination. This sorting and repacking operation at ICT is called **freight consolidation operation**, or **crossdocking**.

The cost of the labor-intensive freight consolidation operation accounts for about 15-20% of the operational costs of an LTL carrier. It is very complex because dock workers must quickly process a large amount of freight in a short time window. Since the margin of an LTL carrier is very low, consolidating freight efficiently is critical for the carrier's profitability. In addition, there are many other benefits resulting from efficient freight consolidation such as higher on-time delivery rate and customer satisfaction. This paper deals with the issue of improving the efficiency of freight consolidation operations of hub-and-spoke LTL carriers by providing better door allocations to destinations (a decision made once in a period of about six months to a year) and door allocations to the stream of inbound trailers (a decision made in real-time).

In mathematical programming, it is well recognized that if the data elements in a model are subject to random fluctuations, the optimum solution of the model with data elements replaced by their average values will be a very poor solution for the original problem. This point has been emphasized in a recent article in ORMS today (Savage *et al*. 2006), which describes several systematic errors that arise from making decisions using average data. We provide a simple example here to illustrate this point. Suppose the model for a problem to be solved daily is a linear program: minimize cx, subject to $Ax \geq b$, $x \geq 0$ in which the cost and RHS constant vectors c, b are random with expected values $\bar{c}$, $\bar{b}$. Let $\bar{x}$ be an optimum solution to the deterministic linear program: minimize $\bar{c}x$, subject to $Ax \geq \bar{b}$, $x \geq 0$. $\bar{x}$ will actually be infeasible for the original problem on most days, and even if feasible it is likely to be far from optimal. That is why in stochastic linear programming $\bar{x}$ is not considered as a candidate solution for the original problem.

The main contribution of this paper is the development of a model incorporating the uncertainty in data using actual data as it exists instead of averages. The model also accommodates online allocation of incoming trailers. To the best of our knowledge, the models in previous literature are based on average data and ignore the online allocation policy or assume a simple policy such as **First Come First Serve** (FCFS). We present two heuristic methods to solve the problem based on local search and genetic algorithms. We provide computational tests using data collected from a terminal and compare it with the current practice at the terminal. Our results show a 20% improvement over current practice, which is significant for this industry.

The rest of the paper is organized as follows. We give an overview of the daily operations at an ICT and outline important facts affecting the door allocation decisions in the remainder of this section. Section 2 provides a review of the previous literature on this topic. An outline of our approach is given in Section 3. We describe an online policy for door allocation to inbound trailers in Section 4. We formulate the destination door allocation problem in Section 5 and also describe two heuristic methods for solving this problem. We present a computational study of the model in Section 6 and a summary of our work in Section 7.

1.1 Daily Operations at an ICT

The daily operations at ICTs happen during a short time window of the day. For most carriers, drivers from EOLTs bring their trailers to the ICT in late evening. These inbound trailers form a continuous arriving stream at the ICT beginning around 10 PM and lasting up to around 1 AM. As these inbound trailers arrive, pallets are unloaded from them using forklifts, which are then loaded to proper outbound trailers in the freight consolidation operation. At the ICT, this freight consolidation operation goes on at a frenetic pitch every night usually between 10 PM and 3 AM, with peaks occurring in the middle of this time window. The room where this takes place is called the **dock**, which has many doors where trailers can be parked, all around its perimeter.

Upon arrival at an ICT, the driver of each inbound truck first stops at the check point to receive instructions from greeters on where to park the trailers. The driver may be asked to deliver the trailers directly to a door or park them in the yard. After the trailers are parked, the driver becomes a **dock worker** and starts unloading any inbound trailers ready for unloading (either her/his own, or those brought by some other drivers earlier), under the supervision of the **freight operation supervisor (FOS)** at the ICT. The driver continues this unloading activity until the FOS instructs her/him to drive back outbound trailers to her/his home EOLT. Once a dock worker begins unloading an inbound trailer, s/he is solely responsible for unloading all of the freight in the trailer.

When an inbound trailer is completely unloaded, a **hostler** moves the emptied trailer to the **yard** and the door becomes available for unloading another inbound trailer. When an outbound trailer is full, a hostler will replace it with an empty trailer to collect additional freight for the same destination.

To determine where to unload a trailer, the FOS examines the content of each inbound trailer sequentially. For each inbound trailer, the FOS will go over the trailer contents and estimate which loading doors will receive most of the freight from the trailer. The FOS will then try to allocate an appropriate unloading door, near or between these loading doors, to the trailer in order to save time on moving freight on the dock. When all unloading doors are allocated to an inbound trailer, the remaining trailers are scheduled to wait in the yard.

Because of time pressure, most of the time, the FOS can not revise door allocations to inbound trailers that were already made, and when making an allocation for an inbound trailer s/he does not have enough time to consider the contents of other inbound trailers.

1.2 The Door Allocation Decision at an ICT

Typically, the dock at an ICT has between 100 and 150 doors. The following are important facts governing the door allocation decision making process:

F1: There is uncertainty in the number of inbound trailers that will arrive at an ICT from an origin, their arrival times, their contents and the time needed to unload them. The algorithm used to make door allocation decision to inbound trailers has to take this uncertainty into account.

F2: The volume of traffic from an origin to a destination through an ICT (in terms of the number of pallets, or even the number of inbound trailers) varies significantly from day to day. Also, data on the actual volume in a day is not usually available accurately until all the trailers arrive at the ICT.

F3: The industry practice is to allocate one or more dedicated doors to each destination. These doors are therefore called **destination doors** (or **outbound doors**, **loading doors**, **stack doors**, etc.), and leave these destination door allocations fixed for periods of 6 months to a year.

F4: LTL terminals use different types of material handling equipment for consolidation.

The reason for F3 is the following: the consolidation activity with a limited time window takes place at a feverish pitch, so it is helpful to keep destination door allocations fixed so that dock workers can memorize them and consolidate the pallets quickly and efficiently. Normally, destination door allocations are only changed when new destinations start appearing or when there are significant shifts in freight flow patterns.

Regarding F4, the most commonly used equipment now-a-days is the forklift. So our model and the data in our numerical examples are based on the forklift as the material handling equipment used for consolidation.

Doors remaining after destination doors are allocated are called **inbound doors** (or **unloading doors**, **strip doors**, etc.) and used for parking inbound trailers. The productivity of freight consolidation operation is measured by: **(Total Number of Pallets Consolidated)/(Total Man-Hours Spent)**. Since all arriving pallets have to be handled, we determine the optimal door allocations that will help minimize the total man-hours spent on consolidation. The number of man hours spent is closely determined by the time spent on forklift moves between unloading and loading doors. Therefore, we minimize the total time spent on forklift moves during the consolidation operations.

2. REVIEW OF PREVIOUS WORK

We could find only four publications and a conference proceedings paper devoted to improving the freight consolidation in OR literature in the last 20 years.

Tsui and Chang (1990, 1992), and Bermudez and Cole (2000) formulate the problem of simultaneously allocating both inbound and outbound doors to trailers as either a bilinear program, or a quadratic assignment problem. The objective is to minimize a weighted distance measure between inbound and outbound doors. Here the weights are based on average volume between various (origin, destination) pairs. Peck (1983) also proposes a similar model to minimize the dock workers travel time for consolidation, with an additional constraint on the time span for the operation. Bartholdi and Gue (2000) also propose a similar model to minimize a combination of the cost of moving freight and the cost of delays due to congestion in the dock, subject to an additional door pressure (a measure of the amount of freight flow in an area) constraint. All these models use average data. Due to the prevailing uncertainty any plans made for average volumes may turn out to be poor for the day. Also due to the facts F1 and F2, they cannot get actual data for the night for their model.

Gue (1999) developed a model that addresses the effect of real-time allocation (with limited look-ahead) of inbound trailers by FOS on the choice of destination door allocation. The effect of real-time scheduling is captured indirectly by a linear programming model, that approximates the cost of consolidation for a given destination-door allocation. A local exchange based search algorithm is then used to identify the destination door assignment with the best objective value under the linear programming model. The resulting candidate solutions are further evaluated using a simulation model to identify the best solution. The model is conceptually more realistic. However, the facts F1 and F2 are not addressed directly in the model.

3. OUTLINE OF OUR APPROACH

Here is a summary of our approach:

i. **Simple Online policy for Inbound Trailers:** Assuming that the door allocations to destinations is given and fixed, we develop a very simple on-line policy to be used daily for making door allocations to inbound trailers as they arrive. At the moment an inbound trailer arrives, if all inbound doors are occupied, it is sent to the yard to be parked temporarily. If there are open inbound doors at that time, the on-line policy computes the man-hours needed to consolidate freight in it if it is parked at each of those open doors, and selects the best among them to park this trailer.

Also, when an inbound door becomes available, and there is no inbound trailers in the arriving stream, the on-line policy decides which of the inbound trailers parked in the yard to move to that door for unloading, using a similar approach.

ii. **Destination-door allocation problem:** Given the on-line policy to be used for allocating doors to inbound trailers daily, this is a one-time decision to determine the door allocations to destinations to minimize the expected number of man-hours needed for consolidation nightly taking the random variation in freight volumes into account.

4. ONLINE POLICY FOR ALLOCATING DOORS TO INBOUND TRAILERS

This policy assumes that doors allocated to all the destinations are fixed and given. All remaining doors are **unloading doors**; these are the doors to be used for parking inbound trailers for consolidation.

If there are some free unloading doors at the time of arrival of an inbound trailer, this policy selects the best among them to park this trailer to minimize the man-hours needed to consolidate the freight in it.

Let r denote the number of pallets inside this trailer, and j_k the door allocated to the destination of the k^{th} pallet for k = 1 to r. Let t_{ab} denote the forklift travel time in minutes from door a to door b plus the average pallet unloading time at door a and the average pallet loading time at door b, for every pair of doors a, b in the terminal. The data matrix t_{ab} is calculated at the terminal by actual observations, and the data r, j_k become available from the trailer's contents when it arrives, and the destination door allocations.

Then the consolidation time (in man-minutes) for this trailer if it is parked at unloading door a is $T_a = \sum_{k=1}^{r} t_{aj_k}$; and the unloading door to allocate to this trailer is the one attaining the minimum in { T_a : a is a free unloading door at the arrival time of this trailer}.

This policy is myopic in the sense that it is only guaranteed to minimize the processing time of the inbound trailer being considered, but may worsen the processing time of future arrivals. However, the average processing time of a trailer is between 30 to 40 minutes; and it is very difficult to predict how many trailers will arrive in that time, or their contents because of the uncertainties in the arrival process mentioned earlier. Further the time to store and retrieve a trailer from the yard is significant (5-10 minutes depending on the equipment used). The limited number of doors available and the storage overhead make any policy that reserves open doors for future arrivals impractical. For these reasons we found that this myopic policy is a good compromise among those that can be implemented.

If all unloading doors are occupied at the time of arrival of an inbound trailer, then it is temporarily parked in the yard. Inbound trailers parked in the yard are brought to the dock for consolidation, when an unloading door, say d, becomes available and there is no inbound trailer in the arriving stream to take it at that time. At that time, let I denote the set of all inbound trailers parked in the yard. For $i \in I$, let T_{ia} denote the man-minutes to consolidate the trailer i if it is parked at any unloading door a, computed as above. Let T_{i}^{*} = min { T_{ia} : a is any unloading door whether it is free at this time or not}, minimum unloading time of inbound trailer i at the best possible unloading door for it.

For each $i \in I$, compute the ratio $R_{id} = T_{id} / T^*_{i}$. Then bring the inbound trailer i^* that attains the minimum in min{ R_{id} : $i \in I$ } to unload at door d at this time. Break ties arbitrarily. This criterion selects the inbound trailer to unload among those in the yard at this time, as the one that compares most favorably with the best door for unloading it.

5. DESTINATION-DOOR ALLOCATION PROBLEM (DDAP)

In the DDAP, we are interested in finding an allocation of doors to destinations such that each door is allocated to at most one destination, and this allocation helps to minimize the expected man-hours spent on consolidating freight daily. We formulate a scenario-based stochastic model, which we now describe in detail.

5.1 Mathematical Model for DDAP

In order to describe our model, we first present some notation. Let

J = the set of all outbound destinations.
D = the set of all doors in the ICT.
$n = |J|$, $m = |D|$. It is assumed that $m > n$.

Definition 1. A **scenario** s contains the information on inbound trailers and freight arriving at an ICT for freight consolidation in a night. The information includes a set of inbound trailers, denoted by I_s, that will be consolidated at the ICT during that night, arrival time of each trailer in I_s, and contents of each trailer in I_s.

A scenario is therefore all the relevant data on the inbound trailers arriving in a night to implement the online policy for inbound trailers. Let

K = the set of all possible destination-door allocations.

δ = the policy used at the ICT to allocate inbound trailers to unloading doors, note that there can be many different policies for allocating unloading doors to arriving inbound trailers. We discussed one of these in Section 4.

s = a symbol used to denote a scenario.

P = the conceptual probability distribution of scenarios, which is unknown.

$F(s, \tau, \delta)$ = the total man hours spent in moving all pallets from inbound trailers to outbound trailers in a night when the scenario for that night is s, the destination-door allocation is τ, and the FOS allocates unloading doors to inbound trailers according to policy δ.

$f_\tau = E_p(s, \tau, \delta)$, the expected value function of $F(s, \tau, \delta)$ with respect to the probability distribution P of scenarios for $\tau \in K$ when δ is the policy used to allocate unloading doors to inbound trailers.

The DDAP is to find a $\tau^* \in K$ such that $f(\tau^*) \le f(\tau)$ for every $\tau \in K$, i.e., we want to solve the following optimization problem:

$$\min\{ f(\tau) : \tau \in K \} \quad (1)$$

Problem (1) is difficult to solve exactly, because of the following reasons:

i. It is not possible to find a closed form representation of the objective function $f(\tau)$, and even calculating its value for a given τ exactly is not possible. For any given destination-door allocation τ, we can only estimate $f(\tau)$ by running a simulation using a prepared representative set of sample scenarios.

ii. The number of possible destination-door allocations is very large.

Therefore, it is impractical to develop an exact solution method for solving Problem (1). In the following sections, we will describe two approximate solution methods for getting good solution for it.

These approximate methods are based on a finite sample set of scenarios from the probability distribution P that maybe generated by

– data of daily occurrences collected over a representative period of time, or

– a realistic sample set of scenarios generated by a simulation model for the arrival process of inbound trailers.

We use the symbol S to denote the sample set of scenarios generated.

5.2 Solution Methods

For each scenario $s \in S$, destination-door allocation τ, and inbound trailer assignment policy δ, the function value of $F(s,\tau,\delta)$ can be calculated easily. Using these values for each scenario $s \in S$, we can estimate the value of $f(\tau)$ as the sample mean $\overline{f}(\tau)$ given by

$$\overline{f}(\tau) = \frac{\sum_{s \in S} F(s,\tau,\delta)}{|S|}$$

However, it is impractical to evaluate $\overline{f}(\tau)$ for every $\tau \in K$ to find the optimum τ^* due to the large number of possible destination-door allocations. Therefore, we develop heuristics to find τ^* approximately. We describe below two heuristic approaches for solving the problem that we found to be effective in our study.

5.3 Approach 1: Local Search Method

Local search methods have been successfully used to get good solutions for many hard combinatorial optimization problems. To develop a local search method, for each solution y say, we have to define a **neighborhood** of it which is a set of solutions obtained by making local exchanges in y. This defines a **neighborhood structure** on the set of all solutions. Once a neighborhood structure is defined, the local search method starts at an initial solution y^0, looks for a better solution in the neighborhood of y^0. If such a solution y' is found the method repeats this process with y'. If the neighborhood of y^0 contains no solution better than y^0 the method terminates with y^0 as the local optimum for the problem. We present below our local search method for DDAP. Let

I_s = the set of inbound trailer arrivals in scenario s, for $s \in S$.

$I = \bigcup_{s\in S} I_s$, set of all inbound trailer arrivals.

P_{ij} = number of pallets in trailer i whose destination is j, for every inbound trailer $i \in I$ and destination $j \in J$.

Algorithm 1 (Local Search)

Step 1 Find an initial destination-door allocation $\tau^0 \in K$. Evaluate $\overline{f}(\tau^0)$ as discussed above. The algorithm may start with the destination-door allocation currently used at the ICT, or one generated randomly.

Step 2 Identify a better destination-door allocation in the neighborhood of τ^0. We create a linear approximation to the objective function $\overline{f}(\tau)$ to search for a better allocation. Let

$\overline{D} \in D$ = the set of doors that are allocated to destinations by τ^0.

$\overline{D}^c = D \setminus \overline{D}$.

$\delta_i = D \setminus \overline{D}$ = the door allocated to trailer i by policy δ at the time it is unloaded, for each inbound trailer $i \in I$.

$c_{dj} = \sum_{i\in I} p_{ij} t_{\delta_i d}$, for each destination $j \in J$ and door $d \in \overline{D}$, cost of processing all pallets going to destination j when it is assigned to door d and each inbound trailer $i \in I$ is assigned to door δ_i.

For any destination-door allocation τ over the set $\overline{D}$, we associate a 0-1 vector $x^\tau = (x^\tau_{dj})$ such that

$$x^\tau_{dj} = \begin{cases} 1, & \text{if door } d \in \overline{D} \text{ is allocated to destination } j \in J \text{ by } \tau; \\ 0, & \text{otherwise.} \end{cases}$$

Then it can be verified that the sample mean objective value of τ^0, $\overline{f}(\tau^0)$, is equal to $\sum_{d\in\overline{D}}\sum_{j\in J} c_{dj}x_{dj}^{\tau^0}$. For other values of τ, we can use the function $g(x^\tau)=\sum_{d\in\overline{D}}\sum_{j\in J} c_{dj}x_{dj}^{\tau}$ as a linear approximation for $\overline{f}(\tau)$, to search for a destination-door allocation τ using the same set of destination doors $\overline{D}$ used by τ^0, which may be superior to the present τ^0. But as $\tau\neq\tau^0$, $\overline{f}(\tau)$ and $g(x^\tau)$ may not be equal.

We keep the inbound door allocations to trailers in I fixed as represented by δ_i, and look for destination-door allocation τ among those using the same set of destination doors $\overline{D}$ used by τ^0, that minimizes the linear approximation $g(x^\tau)$.

This leads to the following bipartite assignment problem:

$$\begin{aligned}
&\text{minimize} && g(x^\tau)=\sum_{d\in\overline{D}}\sum_{j\in J} c_{dj}x_{dj}^{\tau}\\
&\text{subject to} && \\
& && \sum_{d\in\overline{D}} c_{dj}x_{dj}^{\tau}=1,\ \forall j\in J\\
& && \sum_{j\in J} x_{dj}^{\tau}=1,\ \forall d\in\overline{D}\\
& && x_{dj}^{\tau}\in\{0,1\},\ \forall d\in\overline{D},\ j\in J
\end{aligned}$$

The bipartite assignment problem above can be solved very efficiently using network flow techniques, see for example Ahuja *et al.* (1993). If x^{τ^0} is optimal to this assignment problem, go to Step 3. Otherwise let τ^1 be the destination-door allocation corresponding to an optimum solution of this assignment problem. Compute $\overline{f}(\tau^1)$.

If $\overline{f}(\tau^1)<\overline{f}(\tau^0)$, replace τ^0 with τ^1 and repeat Step 2 with it. If $\overline{f}(\tau^1)\geq\overline{f}(\tau^0)$, go to Step 3.

Step 3 Identify improvements by changing in the set of doors assigned to destinations. For each destination $j\in J$, and door $d\in\overline{D}^c$, evaluate $\overline{f}(\tau')$ where $\overline{f}(\tau')$ is the destination-door allocation resulting from changing the door allocation of destination j in τ^0 to door d.

If there exists such an exchange with positive savings, let τ'' be the destination-door allocation resulting from the exchange with the largest savings. Set $\tau^0=\tau''$ and go to Step 2 with it; otherwise go to Step 4.

Step 4 **Termination.** The present destination-door allocation τ^0 is the best destination-door allocation found by the algorithm.

The algorithm seeks two types of improvements on the current solution τ^0. In Step 2, the algorithm tries to find the best destination-door assignment using a linear approximation of the objective function. In Step 3, the algorithm searches for improvements by changing the set of doors assigned to destinations. When neither type of improvement exists, the algorithm terminates. The local search procedure above can be run multiple times using different starting solutions. This improves the chance of getting better results.

5.4 Approach 2: Genetic Algorithm

The **genetic algorithm (GA)** is a powerful and robust heuristic approach for large-scale combinatorial optimization problems (Holland 1975, Goldberg 1989, Davis 1991).

In GA, each solution of the problem is represented using some coding scheme and the resulting representation is called a **chromosome**. GA is an iterative search method that is initiated with a **population** of solutions (represented as chromosomes). Typically the starting population is generated randomly. In each iteration the population is changed by two types of steps. In one, two solutions are selected randomly from the population as **parents** to be replaced by two children generated from them by a combinatorial operation called **crossover**. In the other, one solution is selected randomly from the population, and either changed slightly (**mutation**), or replaced by another from outside (**immigration**).

A basic GA requires specification of the three components listed above: (i) Representation of solutions as chromosomes, (ii) crossover operation, and (iii) mutation/immigration operation. In addition to these components, we also use another component for GA, used in Ahuja *et al.* (2000), in which a subset of solutions from the population are selected and replaced with solutions obtained by applying a local search heuristic on them. The GA terminates after some iterations, and takes the best solution in the population at that time as the output. We next describe the components for our implementation of the GA.

5.4.1 Representation of a Solution

In the DDAP, the number of available doors, n is larger than the number of destinations m. However, for the purpose of representation as a chromosome, we introduce $n-m$ dummy destinations and number them $m+1, \ldots, n$. Using these additional destinations, we can represent a destination-door assignment as a permutation of n numbers as follows. Suppose $a_1, \ldots, a_n$ is the permutation, then the destination a_1 is assigned to door 1, a_2 is assigned to door 2, and so on with the convention that if a door is assigned a number greater than m then it has no destination assigned to it. For example, if there are 8 doors and 4 destinations, then destinations 5, 6, 7, 8 are dummy destinations. In this case the permutation (1, 4, 5, 2, 6, 3, 7, 8) represents the solution in which outbound doors are (1, 2, 4, 6) allocated to destinations (1, 4, 2, 3) in that order, and doors 3, 5, 7, 8 are inbound doors.

5.4.2 Crossover Operation

In our crossover operation, we generate a single child solution that replaces one of the parent solutions. Let $I_1 = (a_1, \ldots, a_n)$, $I_2 = (b_1, \ldots, b_n)$ be two parent solutions selected for the crossover. We use the **insert path crossover** to generate a single child solution with the property that if $a_k = b_k$ then the child has $a_k (= b_k)$ in its k^{th} position. The child solution is obtained as follows.

We start at some random position and examine the permutations I_1, I_2 from left to right in a cyclic fashion. If the entries in the position being looked at are the same, we move to the next position. Otherwise, let $\bar{a}$, $\bar{b}$ be the entries at the current position in permutations I_1 and I_2, respectively. We obtain two new solutions as follows. The first solution, say P_1, is generated from I_1, by inserting $\bar{b}$ in the current location which has $\bar{a}$, and shifting $\bar{a}$ and the following entries to the right in a cyclic way so that the entries common between I_1 and I_2 are not shifted. The second solution, say P_2, is generated in a similar way by using I_2 in place of I_1, and inserting $\bar{a}$ in the position of $\bar{b}$. We choose the solution with the smaller objective value to continue. The chosen solution is stored in a list L. If the solution P_1 has the smaller objective value then we repeat the process with P_1 (replacing I_1) and I_2 starting at the next position in the permutations representing them, otherwise we use I_1 and P_2. We continue in this fashion until all positions are considered.

If the best solution in L is better than one of the parents, it replaces the worst parent. If it is worse than both the parents, then it replaces the parent that is more similar to it. The insert path crossover is illustrated in Figure 2 with parents I_1 and I_2 and the resulting solutions P_1 and P_2 when the position being considered is 6.

I_1: 1-2-**3-4-5-**6-7-8	$\bar{a} = 6$	P_1: 1-2-**3-4-5-**8-6-7
I_2 : 6-7-**3-4-5-**8-2-1	$\bar{b} = 8$	P_2 :1-7-**3-4-5-**6-8-2

Figure 2. Insert path crossover. The common entries are shown in bold type and the current position being looked at is the 6^{th}.

5.4.3 Mutation/Immigration Operation

For immigration we select the worst solutions in the current population by new randomly generated solutions. We use a variable immigration rate in our algorithm. At the beginning, we perform immigration after every 10 trials, and then increase the number of trials between immigrations by 2 after every 200 trials.

In some iterations we change the population by applying the two-exchange heuristic to improve 20% of the population.

6. COMPUTATIONAL STUDY

For our computational study we use data on actual scenarios collected at an ICT of a national LTL carrier, with about 120 doors involving about 70 destinations. However, to keep the company data confidential, here we report on the results from computational work using simulated data patterned after the actual.

The company and previous models use average data to determine the destination-door allocations. This is done based on the ratio of freight volume going to each destination regardless of the freight's origins. Therefore, we use this ratio to generate scenarios for the computational study. The ratio of pallets going to each destination observes this ratio. The average number of inbound trailers arriving/night is about 100. We use the online policy described in Section 4 in the computational study. The online policy is superior to the policy used by the company described in Section 1.1.

All programming is done in C++ and the testing is done on a PC with single Intel Pentium 4 1.4GB CPU, 1GB RAM, and 256KB L-2 Cache.

Performance of the Local Search Heuristic (LS) and GA:

We apply LS and GA to produce a destination-door allocation and compare it with the current destination-door allocation used at the ICT during this period.

For the GA, we tested various combinations of initial population size and maximum iteration number. We noticed that small population size (10), and higher number of iterations (20,000) tended to give best results, so we report on results with the settings.

Data from 20 scenarios was used to select the destination-door allocation first, by both LS and GA separately. Then each of these destination-door allocations was fixed and its performance was evaluated over 60 additional scenarios.

Results for our algorithms refer to this destination-door allocation, and door allocations made by the policy described in Section 4. Results reported for current practice refer to the destination-door allocation in use at the ICT at the time of data collection, and the same policy (Section 4) for unloading door allocation.

For the final set of 60 scenarios, the average man-hour requirement/night for the solutions obtained by the various algorithms are:

- (LS) Local Search Heuristic: 41.7
- (GA) Genetic Algorithm: 41.6
- (CP) Current Practice: 53.4

It can be seen that the solutions obtained by both algorithms of Section 5 are about the same and about 22% better than that under current practice. Figure 3 shows a plot of the man-hours (in minutes)/night under each of these 60 scenarios separately, for each of the above solutions. It can be seen that the solutions obtained by the algorithms are superior not only on an average, but also every night.

7. SUMMARY

We discussed a new on-line approach for allocating doors to arriving inbound trailers at an ICT of an LTL carrier that takes into account all the actual input data needed for this decision (Stage 1). Using this we developed approaches for allocating doors to destinations that help to minimize the expected man-hours for consolidating freight daily (Stage 2). This Stage 2 problem is very complex because the objective function to be optimized is not available as an explicit expression, and can only be estimated by simulations.

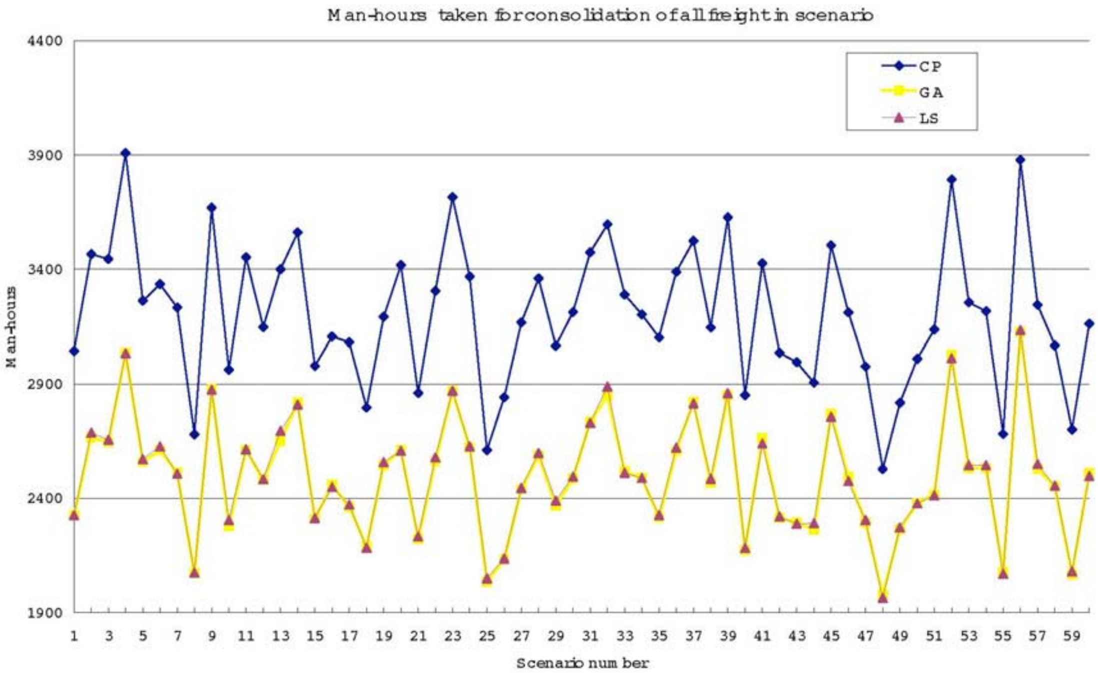

Figure 3. Man-hours taken for consolidating all freight in scenario.

Computational results at an ICT of an LTL carrier, using simulation based on observed data over a representative period of time, indicate that the man-hour requirement can be reduced by over 20% compared to current practice.

There are several potential topics for consideration in continued research. In the present system, each dock worker has the full responsibility to unload all the pallets in an inbound trailer once s/he begins to unload it, and then only moves on to unloading another inbound trailer. This system does result in some wasted labor and fuel. Treating dock Workers as a pool to unload all inbound trailers may increase the overall productivity of the consolidation operation.

The uncertainties associated with the inbound trailer arrival stream and freight contents can be reduced by using advanced communication and information technologies. Also, the capability of automated **material handling systems (MHSs)** such as automated guided vehicles, if properly used, can significantly enhance the efficiency of consolidation operations. The influence of information technology and automated MHSs on allocation policy can be taken into account in future research.

REFERENCES

[1] Ahuja R.K., Magnanti T.L., Orlin J.B. (1993), *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall.

[2] Ahuja R.K., Orlin J.B., Tiwari A. (2000), A greedy genetic algorithm for the quadratic assignment problem; *Computers and Operations Research* 27; 917-934.

[3] Bartholdi J.J., Gue K.R. (2000), Reducing labor costs in an LTL crossdocking terminal; *Operations Research* 48; 823-832.

[4] Bermudez R., Cole M.H. (2000), A genetic algorithm approach to LTL breakbulk terminal door assignment; In *Proceedings of the 2000 Industrial Engineering Research Conference*; Cleveland, Ohio.

[5] Bureau of Transportation Statistics, USA (2003), National transportation statistics.

[6] Davis L. (1991), Handbook of Genetic Algorithms; Van Nostrand, New York.

[7] Goldberg D.E. (1989), Genetic Algorithms in Search, Optimization, and Machine Learning; Addison Wesley.

[8] Gue K.R. (1999), The effects of trailer scheduling on the layout of freight terminals; *Transportation Science* 33; 419-428.

[9] Holland J.H. (1975), Adaptations in Natural and Artificial Systems; The University of Michigan Press, Ann Arbor.

[10] Peck K.E. (1983), Operational Analysis of Freight Terminals Handling Less than Container Load Shipments; *Ph.D. thesis*, University of Illinois at Urbana-Champaign.

[11] Savage S., Scholtes S., Zweidler D. (2006), Probability management; *ORMS Today* 33(1); 20-28.

[12] Tsui L.Y., Chang C.-H. (1990), A microcomputer-based decision support tool for assigning dock doors in freight yards; *Computers and Industrial Engineering* 19; 309-312.