Integrating Production and Transportation Scheduling in a
Make-to-Order Environment

Peter L. Jackson
School of Operations Research and Industrial Engineering
Cornell University, Ithaca, NY 14853-3801
pj16@cornell.edu

Pan Chen
School of Operations Research and Industrial Engineering
Cornell University, Ithaca, NY 14853-3801
pchen@orie.cornell.edu
Abstract
A company in the building products industry decides not to build a finished goods warehouse and, instead, manufactures and ships all products to order. The production and shipping schedules are identical and transportation costs are high. Constraints include order due dates, and production and vehicle capacities. Decisions include vehicle loading and routing, carrier choice, and production date. Two different heuristic approaches are presented to solve the combined production and transportation scheduling problem. Results from simulation show that substantial improvements over the human expert’s solution are possible.
1 Introduction

1.1 Overview

This paper considers a special type of integrated production and transportation scheduling problem, one in which finished goods inventory is not allowed. In manufacturing and distribution environments, finished goods inventories are frequently used to decouple the production planning problem from the transportation scheduling problem. However, in the example environment considered here, the firm has made a decision not to maintain a central warehouse and it does not manage the inventories of product held in the field. Consequently, the production facility is scheduled on a make-to-order basis and the production schedule is closely aligned with the shipping schedule. In the example environment, a small degree of decoupling between production and transportation decision is achieved through the due date quotation rule: customers are quoted shipping dates equal to the order date plus a few days (the standard production lead time). The problem is of general interest because it represents an extreme form of inventory reduction, accomplished through a flexible system of production and transportation. The integrated production and transportation scheduling problem is a challenging optimization problem. We present the problem with many of the particular features of the example environment because these features are not uncommon in distribution systems and because these practical concerns guided our selection of optimization approaches. Furthermore, by closely matching our approaches to the example environment we can legitimately compare the performance of these optimization-based solutions with that achieved by the company’s expert human scheduler.

The problem tackled in this paper is unique in the literature of integrated production and transportation scheduling: the combination of zero inventories, production capacity constraints, and shipping due dates combine to make for a difficult scheduling problem. Problem size and complexity together with practical concerns over computation time lead us to consider multi-stage solution approaches. As is common in the field, there are two natural choices for a multi-stage approach: route-first and schedule-second (RFSS), and schedule-first and route-second (SFRS). The latter approach appears to offer the greatest opportunity for optimization. We develop a separate implementation for each approach and compare them empirically with the performance of an expert human scheduler.

Novel aspects of our two approaches include the following: (1) the Route-First-Cluster-Second vehicle routing algorithm reported in Beasley (1983) is extended to include an improvement phase, called Recombinant Route-First-Cluster-Second; (2) seed routes used in the SFRS approach are time-dated to reflect the fact that different regions of the country should be visited in a periodic fashion; (3) the optimization algorithm used to generate seed routes includes constraints to spread deliveries over time and encourage
periodicity; and (4) the daily scheduling problem is dramatically simplified by treating it as a two period
problem (“schedule now or schedule later”) and using the periodic delivery plan to estimate the shipment
delay penalties in the second period.

1.2 Problem Statement

A company in the building products industry with a single production facility and a nationwide customer
base decides not to build any finished goods inventory. Instead, the company manufactures and ships all
products to order. Hence, the daily production schedule coincides with the shipping schedule. Customer
demand and demand locations are highly variable. Historical demand and location information are available.
For each customer order, the company quotes a due date that allows a few days of lead time before the ordered
products are shipped from the factory. Hence, there is a short time-window in a frozen period for scheduling.

On the production side, we assume that the production flow times are less than one day. For experimental
purpose, the production facility is modeled as a single capacitated production line. In the actual application,
multiple production lines were modeled. Shop-floor-level production sequencing issues are ignored. That is,
we assume feasible production schedules exist at the floor level, as long as the overall production capacity
is not exceeded. All orders have company-issued due dates, by when the products should be produced and
shipped from the factory. On the transportation side, locations are distinguished at the 5-digit zipcode level.
There may exist multiple orders for the same location. There are multiple vehicle types. Typically, the
company would have to make choices among carriers that have different cost structures and side constraints.
For example, the company has to pay for the back-haul of its own fleet, but does not pay for the back-haul
of any contract carrier.

The size of the problem is relatively large when compared to most other problems reported in literature.
On any given day, there may be 10,000 outstanding orders and several thousand different customer locations.

Four types of decisions are made on a daily schedule basis:

(i) When should an order be produced and shipped?

(ii) On which route should the order be shipped?

(iii) How should the stops on each route be sequenced?

(iv) Which type of vehicle should be assigned to each route?

In this paper, we explore optimization techniques to solve the scheduling problem. In section 2, we review
the existing literature in related fields. In section 3, we describe two different optimization approaches:
the Route-First-Schedule-Second approach and the Schedule-First-Route-Second approach. In section 4 we examine and compare the performance of the two approaches, using the results of an human expert scheduler as a benchmark. Section 5 concludes the paper.

2 Literature Review

Our problem is a transportation-cost-driven problem. All four decisions listed in the previous section are transportation related. Hence, our problem is closely related to the Vehicle Routing Problem (VRP) and its variants. Surveys of this literature are available in Bodin and Golden (1981), Bodin et al. (1983), Assad (1988), Desrochers et al. (1990), Laporte (1992), Fisher (1995) and Al-Fawzan and Al-Sultan (1996). We highlight a number of papers that are directly related to our problem.

2.1 The Vehicle Routing Problem (VRP)

The classical VRP is a deterministic, single-period problem with no time constraints. In this problem, a set of nodes with known demand and location is given. A fleet of trucks, from a single depot, with known capacity is used to serve these nodes. The objective is to find a set of delivery routes that minimizes the total distance traveled. When there is only one truck and no capacity constraint, the VRP is equivalent to the Traveling Salesman Problem (TSP).

Fisher (1995) classifies three generations of VRP algorithms. The first generation algorithms are greedy heuristics. Most of these heuristics fall into two categories: the route construction heuristics, and the route improvement heuristics. The route construction can be done in just one phase as in Clarke and Wright (1964) and Altinkemer and Gavish (1991), or it can be done in two phases as in Gillett and Miller (1974) and Beasley (1983). The approach used in Gillett and Miller (1974) is known as the cluster first - route second approach; it groups the different demand points into different clusters first, and then solves a TSP for each of the clusters to determine the sequence of the stops. The approach in Beasley (1983) is known as the route first - cluster second approach. In this approach, a TSP involving all demand points is solved. Then a partition algorithm is used to divide the grand TSP tour into different clusters. A TSP is then solved for each cluster. The route improvement heuristics are derived from the k-optimal heuristics used in solving TSPs. Lin and Kernighan (1973), Christofides and Eilon (1969), Russell (1977) and Golden and Stewart (1985) offer different algorithms based on different choices of the k value.

The second generation VRP algorithms consist of mathematical programming approaches. Fisher and Jaikumar (1981) introduced a generalized assignment approach. In this approach, the assignment of customer
to vehicle \(k\) is solved by an Integer Program (IP). Instead of solving a TSP for each vehicle route, the route cost in the objective function is estimated by some linear approximation of the cost of adding each stop to a seed route. The IP selects the best seed route to assign each customer. Bramel and Simchi-Levi (1995) offer a different way of selecting the seeds. A set-partition approach was introduced in Balinski and Quandt (1964). Agarwal et al. (1989) applied column generation techniques to overcome the computational difficulty in the original set-partitioning formulation.

The third generation algorithms are based on the concepts of metaheuristics. Algorithms involving Simulated Annealing are available in Alfa et al. (1991) and Osman (1993) while algorithms using Tabu Search can be found in Osman (1993), Taillard (1993) and Gendreau et al. (1994).

Generally, the first generation algorithms are easy to implement and they run relatively fast. The second generation algorithms yield improved results over the first generation algorithms, but they potentially take a long time. Furthermore, they exhibit a lack of robustness in practice (i.e. some parameters and approximations are problem-dependent). The third generation algorithms have produced some very good results, but they typically take very long to execute.

2.2 Variants of the VRP

There is no time constraint in the classical VRP. If we require that each delivery in the VRP has to be made within a time window, defined by the earliest time and the latest time when the delivery is allowed, then the problem becomes a Vehicle Routing Problem with Time Windows (VRPTW). If the time window is one sided, i.e., with only a deadline, then the problem becomes a Deadline Vehicle Routing Problem (DVRP). Both the VRPTW and the DVRP are deterministic, single-period problems as in the case of the VRP. The solution approaches to the VRPTW/DVRP are derived from the VRP algorithms. Greedy heuristics are available in Solomon (1987) and Savelsebergh (1985) for the VRPTW, and in Nygard et al. (1988) and Thangiah et al. (1993) for the DVRP. Mathematical programming formulations can be found in Foster and Ryan (1976), Desroschers et al. (1992) or Fisher et al. (1997). Potvin et al. (1996) and Taillard et al. (1997) use metaheuristics to solve the problem.

The Inventory Routing Problem (IRP) extends the VRP to a multi-period scenario in which the demands are stochastic. Customers at the demand points consume the product at known rates and have known storage capacities for the product. The objective of the IRP is to minimize the delivery costs while attempting to make sure that no customer runs out of the commodity at any time. The IRP is sometimes known as the Allocation / Routing Problem. A survey of this problem can be found in Ball (1988). The solution approaches
to the IRP are similar to the VRP approaches. Federgruen and Zipkin (1984) use the generalized assignment approach proposed in Fisher and Jaikumar (1981). Dror and Levy (1986) introduce two algorithms that are essentially the combination of the Clarke & Wright method and the Lin & Kernighan method in the VRP. One important feature of the IRP is that the assignment of customer-to-date becomes a decision variable. Essentially, one has to solve three assignment problems: customer-to-date, customer-to-cluster and customer-to-sequence within a cluster. Dror and Ball (1987) and Dror and Levy (1986) together give two algorithms. The first is a two-stage algorithm which will either solve the customer-to-date problem first, and then solve a VRP, or it will assign customer to a date and a vehicle first, and then solve a TSP. The second algorithm tries to solve all three assignment problems together. The computational results in the paper indicates that the first algorithm is better. Herer and Levy (1997) introduce a variation of IRP called Metered Inventory Routing Problem (MIRP) to take inventory holding cost into consideration. The solution applies a concept called temporal distance that is introduced in Herer (1996).

Despite the amount of work in IRP, there is little in the literature on the coordination of production scheduling with transportation scheduling. Chandra and Fisher (1994) are an important exception. They solve the Production Scheduling and Distribution problem (PSD), which has the following scenario. There is a single production plant producing a number of products over time. A finished goods inventory of the products is maintained at the plant. The products must be distributed by a fleet of trucks to a number of retail outlets by certain due dates. Early delivery is allowed, but no backorder is allowed. The objective is to minimize the transportation cost and the inventory holding cost. The solution in the paper treats the production problem as a capacitated lot-sizing problem and the transportation problem as a standard vehicle routing problem. Essentially, the algorithm solves the two problems separately, and then tries to combine routes with underutilized trucks using inventory or minor modifications to the production schedule.

### 2.3 Relation to the Problem Under Study

Since the transportation cost is significant in the problem studied in this paper, the classical VRP is clearly an important part of the problem. This problem also involves due dates, as in the case of the VRPTW. But the VRPTW is a single-period problem, while the problem studied is a multi-period problem. Furthermore, coordination of transportation scheduling with production scheduling is an important part of the problem studied, but non-existent in the VRPTW. Nonetheless, the VRPTW literature offers valuable techniques for dealing with time contraints in a transportation problem. The IRP considers inventory, but not production scheduling. More importantly, the demands in IRP are stochastic at the time of scheduling. In this problem,
the demands are deterministic. However, the two-stage algorithms used in the IRP provide some suggestions for the problem studied here. The PSD is the closest to the problem considered here. The key difference between the two is that PSD still allows inventory. The due dates are hard constraints in the PSD, but they are soft in the problem studied here. Another difference is that carrier choice becomes an additional side constraint in this problem. Hence, the problem considered in this paper is new to the literature. Solution approaches will be presented in the following section.

3 Optimization Approaches

In making the four decisions that we listed at the end of section 1, we consider three types of costs:

(i) Transportation cost

(ii) Lateness cost

(iii) Overtime production cost

The lateness cost and the overtime production cost depend more on the decision of when to produce an order, while the transportation cost depends more on the decision of cluster, sequence and carrier choice. At the same time, the costs are interrelated. For example, suppose that the production plant is in the Midwest of the United States. The shipping cost to California would be significant. If there is only one order in California available and due today, one solution is to send a separate truck today to avoid any lateness cost. An alternative solution is to take the lateness penalty now, with the anticipation that there will be another order in California in the near future. If such an anticipation is correct, then the transportation cost will be reduced by shipping both of the California orders on the same truck. At the same time, however, the decision on when to produce the California order also affects the production capacity constraint.

Obviously the problem studied is difficult. Even the simplest VRP without any time constraint or carrier choice is known to be NP-Hard. This problem has added new dimensions. Trying to solve all four decision problems in one step would be too ambitious. Therefore, two-stage optimization approaches are proposed. The first one is called the Route-First-Schedule-Second (RFSS) approach and the second one is called the Schedule-First-Route-Second (SFRS) approach. In the first approach, the routing decision is made first, then schedules are made from the available routes. In the second approach, historical data are used in a planning step and the two stages in the first approach are reversed. Note that the meaning of “route” is broader here than in the VRP context. In our problem, it involves both vehicle routing and carrier choice.
3.1 The Vehicle Routing Sub-Problem

In both optimization approaches that we explore, the VRP appears as a sub-problem. In the application environment studied, the VRP is complicated by carrier choice decisions, vehicle capacity constraints, and limitations on route length and number of stops per route. We are also interested in relatively short computation times. We chose to use a first generation heuristic for the VRP because of its speed and modeling flexibility.

The algorithm we developed is a variation on the Route First-Cluster Second (RFCS) algorithm for the VRP described by Beasley (1983). In that algorithm, the cities to be visited are first sequenced into a single tour. Let $\Gamma = (0, 1, \cdots, N)$ denote the grand tour formed in this step, where 0 denotes the shipping depot and there are $N$ cities to visit. For this step, we chose to sequence the cities to minimize the total length of the tour, i.e., solve a TSP. We use a combination of heuristics for the TSP: Farthest Insertion followed by 2-Opt for small problems and a variation on the Sweep algorithm for large problems.

The RFCS algorithm proceeds to break the tour into subtours of the form $(0, s, s+1, \cdots, t-1, t)$ where $(s, s+1, \cdots, t-1, t)$ is a subsequence of $\Gamma$. Given the restriction that all subtours are subsequences of $\Gamma$, this clustering portion of the algorithm can be formulated and solved efficiently using Dynamic Programming (DP) as follows.

Let $c(s, t)$ denote the cost of serving subtour $(0, s, s+1, \cdots, t-1, t)$ using a single vehicle. Let $f(t)$ denote the minimum cost of serving cities $(t, t+1, \cdots, N)$, assuming that $t$ is the first city to be visited after the depot and that the order of visiting cities on any subtour follows the order given by $\Gamma$. Because costs are additive, this function obeys the following DP recursion:

$$f(s) = \min_{s \leq t \leq N} \{c(s, t) + f(t + 1)\}, s \in \Gamma;$$

(1)

$$f(N + 1) = 0.$$  

(2)

Note that if there is a restriction on the maximum number of cities per route, say, $L$, then the minimization in (1) can be resolved using at most $L$ comparisons. With such a restriction, the DP can be solved in $O(LN)$ calculations, exclusive of the effort required to compute the $c(s, t)$ function. The cluster DP computation is essentially linear in the problem size.

As noted by Beasley, the power of this approach is the modeling flexibility afforded by the cost function $c(s, t)$. In our application, we incorporated carrier choice by letting

$$c(s, t) = \min_{k \in K} c_k(s, t)$$

in our application, we incorporated carrier choice by letting
where $K$ is the set of possible carriers and $c_k(s,t)$ is the cost of serving subtour $(0, s, \cdots, t)$ using carrier type $k$. The carrier types differ in term of vehicle capacity, mileage cost, per-stop cost, maximum mileage, maximum number of stops per route, and whether or not the return-to-depot portion of the trip is included in the cost. Vehicle constraints (capacity, mileage, and stops) were modeled using penalty functions. Some experimentation was required to find appropriate penalty weights.

In preliminary experiments, the RFCS algorithm was unable to match the performance of an expert human scheduler in solving the VRP. This is due in part to the poor performance of the Sweep heuristic for solving large TSPs. However, significant gains are possible by embedding the RFCS algorithm in an iterative scheme of recombining nearby routes and resolving the VRP for each recombined cluster. Figure 1 outlines the logic.

![Figure 1: Recombinant Route First - Cluster Second Algorithm](image)

For each route, we compute the geographical centroid of the cities on the route, and label the route with the azimuth formed between the centroid, the shipping depot, and true north. We maintain a list of routes sorted by azimuth. Each route in this list is considered in turn (sweep fashion). Next we consider combining this route with a single nearby route: one of the next $A$ routes, in azimuth sequence, for some small number $A$ ($A = 5$ in our experiments). The RFCS algorithm is applied to this combined set of cities. If there is a
cost improvement then the combined route is replaced by the results of the RFCS algorithm; otherwise, the original routes are restored.

The Recombinant RFCS algorithm is an effective approach to solving the VRP. It is relatively easy to implement. It has the modeling flexibility of the RFCS algorithm and it is faster than the second or the third generation VRP algorithms. As we will show in the computational results section, it achieves good transportation cost performance.

3.2 The Route-First-Schedule-Second Approach

3.2.1 The Algorithm

The first approach to the zero-inventory production and vehicle scheduling problem is straightforward: first assign all orders to vehicle routes to minimize transportation costs; then assign vehicle production loads to production days to minimize customer delays and to smooth production loads. This approach is called Route-First-Schedule-Second (RFSS).

The input to the first step in the algorithm is simply a list of all outstanding orders, aggregated by zipcode location. The Recombinant RFCS algorithm is used to organize these orders into separate vehicle routes. Observe that transportation costs are considered in detail throughout this step, but customer due dates and production considerations are ignored.

The output of the first step is a set of vehicle routes, denoted by $V$. The second step of RFSS is to assign these vehicle routes to production days using a Mixed Integer Linear Program (MILP). Let $T$ denote the set of production days and let

$$X_{v,t} = \begin{cases} 
1 & \text{if vehicle route } v \text{ is assigned to day } t \\
0 & \text{otherwise,}
\end{cases}$$

for all $v \in V$ and $t \in T$. We specify a customer delay penalty of $p_{v,t}$ for scheduling vehicle route $v$ on day $t$. Many penalty functions are possible within this framework. For our experiments, we chose

$$p_{v,t} = (t - t_v)^+$$

where $t_v$ is the earliest due date of all orders assigned to vehicle route $v$.

Production constraints can be modeled in considerable detail with this approach. For experimental purposes, we focus on a single measure of production capacity. Let $a_v$ be the total number of production units required by the orders on vehicle route $v$. Then $\sum_{v \in V} a_v X_{v,t}$ is the production capacity required on day $t$. 

11
Denote by $l_t$ and $u_t$ the minimum and the maximum desired production (in production units) on day $t$. Let $O_t$ denote production in excess of $u_t$ on day $t$, and let $U_t$ denote the shortfall of production below $l_t$ on day $t$. Let $M$ denote an arbitrarily large number. The production assignment problem can be written as:

$$
\text{Minimize} \quad Z = \sum_{v \in V} \sum_{t \in T} \sum_{i \in T} p_{v,t} X_{v,t} + \sum_{t \in T} M(O_t + U_t) \quad (3)
$$

Subject to

$$
l_t \leq \sum_{v \in V} a_{v,t} X_{v,t} - O_t + U_t \leq u_t \quad \forall t \in T, \quad (4)
$$

$$
X_{v,t} \in \{0, 1\} \quad \forall v \in V, \quad t \in T, \quad (5)
$$

$$
O_t \geq 0 \quad \forall t \in T, \quad (6)
$$

$$
U_t \geq 0 \quad \forall t \in T. \quad (7)
$$

Note that we have chosen to dualize the production capacity constraints. These are not hard constraints. Furthermore, the formulation can be extended easily to capture the costs of over- and under-production.

### 3.2.2 Rolling Horizon Implementation

The RFSS approach is intended to be implemented in a rolling horizon fashion. Only the vehicle routes assigned to the next production day are implemented. The entire problem is re-solved each day using all outstanding orders except those committed for production and shipment on the current day. The reason for this is that new orders are arriving continuously in time and the arrival of new orders can significantly change the economics of the production and transportation plan.

### 3.2.3 Shortcomings of the RFSS Formulation

There are a number of objections that can be raised a priori to the RFSS approach as we have formulated it. Its performance in practice is considered in a later section. The first concern is that the role of the order due dates is somewhat haphazard. Consider the situation in which a single order on a vehicle route is late. The due date penalty function then assigns a high priority to this vehicle route as a whole, perhaps forcing it into the production schedule much earlier than required by the other orders on the vehicle. A simple swap of this order with another order having a later due date on a different vehicle could increase the scheduling flexibility, with little increase in transportation cost.

A second objection to the formulation is that it ignores the stochastic nature of the problem. An expert scheduler seeing an order in a remote location may simply refuse to schedule that order on any vehicle for a few days. The expert, through experience, knows that other orders near to that remote location are likely to materialize in the near future. By waiting for these orders to materialize, a more economical routing solution
may become available. In the RFSS approach, routes are created to cover all order locations and there is nothing in the production scheduling problem formulation that explicitly recognizes the value of delaying a scheduled shipment.

A third objection to the formulation is that because the production schedule is created with whole vehicle loads, it can be difficult to create smooth schedules, particularly as other production constraints are added to the formulation. An expert scheduler, on the other hand, has the capability to design vehicle loads to satisfy production constraints. The RFSS approach ignores such opportunities.

These are the considerations which suggest that an alternative approach to the problem might be useful.

3.3 The Schedule-First-Route-Second Approach

The second approach to the zero-inventory production and vehicle scheduling problem places a greater emphasis on mathematical programming techniques: first, assign orders to production days to simultaneously optimize transportation costs, vehicle loads, and production capacity; then re-optimize the assignment of orders to vehicle routes to minimize transportation costs. This approach is called Schedule-First-Route-Second (SFRS) because the first step yields a production schedule and the second step yields a vehicle routing plan.

The first step in the SFRS approach is to use a MILP to assign orders to production days. The immediate difficulty in formulating this approach lies in capturing transportation routing costs. Following Fisher and Jaikumar (1981), a natural approach is to allow the MILP to choose from a set of seed routes and to charge a fixed cost for inserting any given order into one of these routes. To compute the true transportation routing cost, the sequence of delivering these orders on the resulting routes would also have to be specified. However, sequencing is ignored in this step in order to reduce the problem size. Even so, the MILP resulting from such an approach can turn out to be a large scale assignment problem. For the application environment studied, the MILP could easily require 250,000 binary variables (5000 outstanding orders × 10 seed routes per order × 5 production days). Rather than attempt to solve such a large problem directly, we have opted to further decompose and approximate the assignment problem. In particular, the set of outstanding orders is divided into geographical regions and a rotation schedule is used. Only a subset of the regions is assigned to any particular production day. Since the same region may be visited more than once in a production cycle, orders must yet be assigned to specific production days. However, the problem size is much smaller. A further size reduction is achieved by formulating the assignment decision as a two-period problem: either produce and ship the order today, or produce and ship the order at the next opportunity. We heuristically
approximate the tradeoffs involved in that decision.

Figure 2: Functional Flow for Schedule-First-Route-Second Algorithm

Figure 2 illustrates the functional flow of this decomposition approach. The following sub-sections describe the models used in each functional block.

3.3.1 Compute Shipment Frequency (A1)

A Pareto analysis of the historical sales data for the target company revealed that a high fraction of the shipments were concentrated in a small fraction of all the locations to which shipments were made (i.e., the “80-20” rule applied to the diversity of zipcodes visited). The zipcodes of major shipping destinations were identified. These include not only regional distribution centers but also high demand customer locations as well. The purpose of steps A1-A4 is to develop a cyclical plan of visiting these so-called major zipcodes while preserving the production capacity constraints, and then use this plan as the basis for creating seed routes.

Let $\bar{T}$ denote the planned cycle time, an input to this process. Let $T = (1, 2, \ldots, \bar{T})$ be the set of production days. Some regions may be served only once in a cycle. If an order in such a region arrives just after a shipment to that region has been finalized, it must wait at least $\bar{T}$ days before it will be shipped.

Let $Z$ denote the set of major zipcodes and let $q_z$ denote the average demand per planning cycle at
zipcode $z \in Z$, measured in shipping units. Let $K$ denote average vehicle capacity, measured in shipping units. For planning purposes, $\tilde{S}_z = \lfloor q_z / K \rfloor$ is an estimate of the number of shipments that will be made to zipcode $z$ in a planning cycle, and $q_{z,s} = \lfloor q_z / S_z \rfloor$ is an estimate of the average shipment quantity that will be made on shipment $s \in \{1, 2, \ldots, \tilde{S}_z\}$ to zipcode $z$ in a planning cycle. Because of the stochastic nature of the way orders materialize, it would be wise to spread these planned shipments over the entire planning cycle. Accordingly, we define lower and upper bounds to constrain the assignment of shipments to planning days:

$$\underline{S}_{z,t} = \lfloor \tilde{S}_z \cdot \frac{t}{T} \rfloor, \quad \forall z \in Z, t \in T$$

$$\overline{S}_{z,t} = \lceil \tilde{S}_z \cdot \frac{t}{T} \rceil, \quad \forall z \in Z, t \in T$$

Figure 3 illustrates the bounding functions for the case $\tilde{S}_z = 3$ and $T = 5$. Cumulative shipments to this zipcode must lie between these two functions. The bounding functions are used in step A3.

![Figure 3: Upper and Lower Bounding Shipment Functions](image)

### 3.3.2 Define Geographical Regions (A2)

A practical technique to make the optimization problem size in A5 manageable is to divide the entire geographical region served by the production / distribution system into smaller regions and to serve only a subset of regions on any particular day. The heuristic below is a simple implementation of this idea. It divides the entire region into angular sectors of roughly equal planned production load.

Let $\tilde{R}$ denote the desired number of regions, a management parameter. We require $\tilde{R} \geq \tilde{T}$, the number of production days in the cycle. Let $R = \{1, \ldots, \tilde{R}\}$ denote the set of regions. Let $\alpha_r$ denote the angle from true north marking the sector boundary between the $r^{th}$ and $(r + 1)^{st}$ region. Let $a_z$ denote the azimuth of
major zipcode \( z \in Z \) relative to the shipping depot. Let \( Z_r \) denote the subset of major zipcodes that fall in region \( r \):
\[
Z_r = \{ z \in Z : \alpha_{r-1} \leq a_z \leq \alpha_r \}, \quad r \in R
\]
where \( \alpha_0 = 0^\circ \) and \( \alpha_R = 360^\circ \).

Let \( P_z \) denote the average demand per planning cycle at zipcode \( z \in Z \), measured in production units. Note that production units can differ from shipping units, but they will tend to be highly correlated. Let \( \bar{P} = \sum_{z \in Z} P_z \) be the total planned production load for major zipcodes over the planning cycle, and let \( P_r = \sum_{z \in Z_r} P_z \) be the planned production load per cycle in region \( r \). We choose the values \( \alpha_r, r \in R \), to equalize \( P_r \).

The value of \( \alpha_r \) can be computed in a sweep fashion. Define a function \( P(\alpha) = \sum_{z \in Z} 1_{\{\alpha, < \alpha\}} P_j \). Let \( \alpha_r \) denote the largest value of \( \alpha \) such that \( P(\alpha) < r \bar{P} / \bar{R} \) and let \( \alpha_r \) denote the smallest value of \( \alpha \) such that \( P(\alpha) \geq r \bar{P} / \bar{R} \). Set \( \alpha_r = (\alpha_r + \alpha_r) / 2 \), for \( r = 1, 2, \cdots, \bar{R} - 1 \). Taking this average avoids having a sector boundary run through a major zipcode. The computation can be performed recursively in \( \alpha \).

### 3.3.3 Plan Shipments for Production Cycle (A3)

The next step in the planning process is to plan how the major zipcodes will be served over the interval of a planning cycle. The goals of this plan are to spread out the shipments to each major zipcode (in order to minimize the expected waiting time of each arriving order), to smooth the production load across different production days, and to minimize the number of regions considered on each production day (in order to control problem size in step A5). We formulate the problem as a MILP in which the potentially conflicting goals are brought together with management penalty weights into a single objective function.

To capture the requirements of production smoothing, we define cumulative bounding functions similar to those for controlling the distribution of shipments. Let
\[
\bar{P}_z = t \cdot l, \quad \text{and} \quad \bar{P}_l = t \cdot u,
\]
where \( l \) and \( u \) are lower and upper bounds respectively on the planned daily production capacity available for serving major zipcodes (which, from Pareto analysis, are approximately 80% of total production capacity).

Let \( S_z = \{1, 2, \ldots, \bar{S}_z\} \) be the set of planned shipments to major zipcode \( z \in Z \). The focus of the formulation is on the decision variables given by
\[
x_{z,s,t} = \begin{cases} 
1 & \text{if shipment } s \text{ is made to zipcode } z \text{ on day } t, \\
0 & \text{otherwise}; 
\end{cases}
\]

16
for $z \in Z$, $s \in S_z$, and $t \in T$. Auxiliary decision variables are given by

$$y_{z,t} = \begin{cases} 
1 & \text{if there is any shipment to zipcode } z \text{ on day } t, \\
0 & \text{otherwise}; 
\end{cases}$$

$$w_{r,t} = \begin{cases} 
1 & \text{if there is any shipment to region } r \text{ on day } t, \\
0 & \text{otherwise}; 
\end{cases}$$

$e_{z,t}$ is the number of shipments in excess of 1 to zipcode $z$ on day $t$;

$S_{z,t}$ is the cumulative number of shipments to zipcode $z$ by day $t$;

$P_{a,t}$ is the cumulative quantity shipped to zipcode $z$ by day $t$, measured in production units;

$E_{z,t}$ is the cumulative number of shipments to zipcode $z$ in excess of $S_{z,t}$ by day $t$;

$F_{z,t}$ is the cumulative shortfall of shipments to zipcode $z$ below $S_{z,t}$ by day $t$;

$O_t$ is the cumulative production quantity in excess of $P_t$ by day $t$;

$U_t$ is the shortfall of cumulative production quantity below $P_t$ by day $t$.

The problem can be formulated in the following MILP:

Minimize \[ Z^* = \lambda_1 \sum_{r \in R, t \in T} w_{r,t} + \lambda_2 \sum_{z \in Z, t \in T} e_{z,t} \]
\[ + \lambda_3 \sum_{z \in Z, t \in T} E_{z,t} + \lambda_4 \sum_{z \in Z, t \in T} F_{z,t} \]
\[ + \lambda_5 \sum_{t \in T} O_t + \lambda_0 \sum_{t \in T} U_t \] (8)

Subject to \[ x_{z,s,t} \leq y_{z,t} \quad \forall z \in Z, t \in T, s \in S_z, \] (9)
\[ \sum_{t \in T} x_{z,s,t} = 1 \quad \forall z \in Z, s \in S_z, \] (10)
\[ y_{z,t} \leq w_{r,t} \quad \forall z \in Z, r \in R, t \in T, \] (11)
\[ \sum_{s \in S_z} x_{z,s,t} - 1 \leq e_{z,t} \quad \forall z \in Z, t \in T, \] (12)
\[ S_{z,t} = \sum_{t' \leq t} x_{z,s,t'} \quad \forall z \in Z, t \in T, \] (13)
\[ S_{z,t} - E_{z,t} \leq \tilde{S}_{z,t} \quad \forall z \in Z, t \in T, \] (14)
\[ S_{z,t} \leq S_{z,t} + F_{z,t} \quad \forall z \in Z, t \in T, \] (15)
\[ P_{z,t} = \sum_{t' \leq t} q_{z,s} \cdot x_{z,s,t'} \quad \forall z \in Z, t \in T, \] (16)
\[ P_t \leq \sum_{z \in Z} P_{z,t} + U_t \quad \forall t \in T, \] (17)
\[ \sum_{z \in Z} P_{z,t} - O_t \leq \tilde{P}_t \quad \forall t \in T \] (18)
\[ x_{z,s,t} \in \{0,1\} \quad \forall z \in Z, s \in S_z, t \in T, \] (19)

17
\[ y_{z,t} \in \{0,1\} \quad \forall z \in Z, t \in T, \]  
\[ w_{r,t} \in \{0,1\} \quad \forall r \in R, t \in T, \]  
\[ e_{z,r}, \quad S_{z,t}, \quad P_{z,t}, \quad E_{z,t}, \quad F_{z,t} \geq 0 \quad \forall z \in Z, t \in T, \]  
\[ O_t, \quad U_t \geq 0 \quad \forall t \in T. \]  

where \( \lambda_1 - \lambda_6 \) are penalty coefficients. The first term in the objective function penalizes the number of regions considered on each production day. The second term in the objective function penalizes shipments to the same zip code in excess of 1 on the same day. The third and the fourth terms in the objective function act to spread out the shipments to the same zip code. The fifth and the sixth terms in the objective function act to smooth the production load across different production days. Constraint (9) ensures the consistency between \( x_{z,a,t} \) and \( y_{z,t} \). Constraint (10) makes sure each shipment will be served on some day. Constraint (11) forces variable \( w_{r,t} \) to be 1 whenever there exists a shipment to region \( r \) on day \( t \). Constraint (12) - (18) define the relationships among the auxiliary variables.

In our actual implementation, the MILP was solved with a linear relaxation on \( x_{z,a,t} \) and \( y_{z,t} \). However, this formulation is so tight that 99.43% of the decision variables turned out to be integers in our experiments.

### 3.3.4 Create Seed Routes (A4)

The solution to the MILP in A3 yields a set of major zipcodes intended to be served for each day in the planning cycle. In A4, the Recombinant RFCS algorithm is used to route those major zipcodes on each day. In our experiments, we used 60% of the true vehicle capacity as the vehicle capacity in this step, so that we could insert other orders into these routes in step A5. We call the routes formed in this step the seed routes.

We label each seed route with its intended service day in the cycle. Hence, the concept of seed routes is the key to link the two dimensions of the problem together: distance and time. We will take advantage of this in the next step.

### 3.3.5 Filter Orders (A5)

The seed routes for any production day are concentrated in a subset of regions defined in A2. We focus on the orders in the intended service regions, expecting to reduce the problem size. However, to avoid lateness cost, we also consider the orders that are due or that are already late. Thus, instead of considering all the available orders (as the candidates to be inserted into the seed routes), the input to the MILP consists of only two types of orders: those that are in the intended service regions for the current production day, and those that are due or overdue.
3.3.6 Select Orders for Current Production Day (A6)

Steps A1 - A4 comprise the planning phase. This should be run infrequently. Steps A5 - A6 are run daily. In particular, step A5 yields a daily production schedule and step A6 yields the corresponding vehicle routing plan. Thus, they are the “schedule” and the “route” parts of the SFRS approach and they are built on the dated seed routes created in steps A1 - A4.

We formulate a MILP to select orders for the current production day in A5. The goal of this MILP is to minimize the combined transportation cost and lateness cost. At the same time, we also penalize underutilized transportation and production capacities.

Each of the input orders could potentially be inserted into any of the seed routes. The cost parameters are estimated as follows. We estimate the transportation cost by adding the cost of using the seed routes and the cost of inserting new stops into the seed routes. If we assign orders to a seed route but choose not to serve that seed route on the current production day, then the lateness cost incurred on any order on the route is estimated by the lateness that the order will have when the region that contains the route is an intended service region again.

Denote by $V$ the set of seed routes and let $O$ be the set of input orders. Notice that the set $V$ in this step is much smaller than in the RFSS approach, where $V$ was the set of all routes. We define the following parameters:

- $C_v =$ the estimated transportation cost of using $v \in V$,
- $c_{o,v} =$ the estimated cost of inserting order $o \in O$ into seed route $v \in V$,
- $l_{o,v} =$ estimated lateness cost of scheduling order $o \in O$ on seed route $v \in V$,
- $K_v =$ the capacity of the vehicle serving seed route $v \in V$,
- $\bar{U} =$ the production capacity for the current production day,
- $q_o =$ the order size (in shipping units) of order $o \in O$,
- $p_o =$ the production required for order $o \in O$, in production units,
- $\gamma_v =$ the penalty coefficient per unit of underutilized vehicle capacity on seed route $v \in V$,
- $\lambda =$ the penalty coefficient per unit of underutilized production capacity.

The focus of the formulation is on decision variables given by

$$x_{o,v} = \begin{cases} 1 & \text{if order } o \text{ is assigned to the seed route } v, \\ 0 & \text{otherwise,} \end{cases}$$

for $o \in O$, $v \in V$. Auxiliary decision variables are given by
\[ y_v = \begin{cases} 
1 & \text{if seed route } v \text{ is selected for the current production day}, \\
0 & \text{otherwise}, 
\end{cases} \]

\[ z_{o,v} = \begin{cases} 
1 & \text{if order } o \text{ is assigned to the seed route } v \text{ that is selected for current production day}, \\
0 & \text{otherwise}, 
\end{cases} \]

for all \( o \in O, v \in V \).

We select orders for next production day using the following MILP:

Minimize \[ Z^* = \sum_{v \in V} C_v \cdot y_v + \sum_{o \in O} \sum_{v \in V} c_{\alpha,v} \cdot x_{o,v} \]
\[ + \sum_{o \in O} \sum_{v \in V} l_{\alpha,v} \cdot (1 - y_v) \]
\[ + \sum_{v \in V} \left[ \nu \cdot \left( K_v \cdot y_v - \sum_{o \in O} q_o \cdot z_{o,v} \right) \right] \]
\[ + \lambda \cdot (M - \sum_{v \in V} \sum_{o \in O} q_o \cdot z_{o,v}) \quad (24) \]

Subject to \[ z_{o,v} \leq x_{o,v} \quad \forall o \in O, v \in V, \quad (25) \]
\[ z_{o,v} \leq y_v \quad \forall o \in O, v \in V; \quad (26) \]
\[ \sum_{o \in O} q_o \cdot z_{o,v} \leq K_v \cdot y_v \quad \forall v \in V, \quad (27) \]
\[ \sum_{v \in V} \sum_{o \in O} p_o \cdot z_{o,v} \leq \bar{U}, \quad (28) \]
\[ \sum_{v \in V} x_{o,v} = 1 \quad \forall o \in O, \quad (29) \]
\[ x_{o,v}, z_{o,v} \in \{0,1\} \quad \forall o \in O, v \in V, \quad (30) \]
\[ y_v \in \{0,1\} \quad \forall v \in V. \quad (31) \]

The first two terms in the objective function are the estimated transportation costs. The third term is the lateness cost. The fourth is the cost of underutilized vehicles and the fifth is the cost of underutilized production. The first two constraints define the consistency among \( x_{o,v}, y_v \) and \( z_{o,f} \). The third constraint is the vehicle capacity constraint. The fourth is the production capacity constraint. The fifth forces each order to be assigned to exactly one route.

Due to the size of the problem, in the actual implementation, we solved the problem with a linear relaxation on \( x_{o,v} \) and \( z_{o,v} \). Constraint (26) helps to force the linear solution to be integers. In our experiment, about 85% of the decision variables turned out to be integers. We rounded up the fractional solutions by setting:
\[ x_{o,v} = \begin{cases} 
1 & \text{if } x_{o,v} \geq x_{o,v'} \quad \forall v' \in V \text{ and } v \neq v' \\
0 & \text{otherwise} 
\end{cases} \]

Thus, the decision variable \( x_{o,v} \) can be viewed as the probability of assigning order \( o \in O \) to route \( v \in V \).

Since we can assign each order to exactly one route, we assign it to the most likely route.

3.3.7 Schedule and Route Next Day Vehicles (A7)

Step A6 selects a set of orders for the current production day. Step 7 solves the routing problem on the set of orders. For this, the Recombinant RFCS algorithm is again applied. As in the previous approach, transportation costs are modeled in detail throughout this step. The difference from the previous approach is that the RFSS approach considers all outstanding orders when formulating the VRP. Under the SFRS approach, the VRP is formulated only with the orders assigned to a specific production day. It should be noted that since the VRP has smaller dimension under the SFRS approach, it might be practical to apply a second or the third generation VRP algorithm in this case.

4 Computational Results

To understand the performance of our heuristics, we ran simulations based on actual data from the target company collected over a 2-month period. We briefly describe the data set and some of our data approximations below:

- There are over 2,000 distinct zipcodes.
- There are over 10,000 individual orders over the 2-month period. We removed orders to foreign countries. Such orders are few and the associated production quantity is negligible.
- There are three carrier types. They differ in capacity, number of stops allowed per route, maximum mileage, per-stop cost, mileage cost and whether or not the company has to pay for the back-haul. Two of the three types are in the company’s own fleet. The other is a contract carrier. Contract carriers are more expensive, but can travel further than the company’s fleet vehicles. The company specifies the route but does not pay for the return trip if a contract carrier is used.
- Production quantities are measured in linear feet. The average daily production capacity is estimated at 130,000 linear feet.
• The shipping quantity is measured in shipping units as is the vehicle capacity. The original data set from the company did not contain accurate measurements of the shipping quantity with respect to the vehicle capacities. We worked closely with the company to estimate the shipping units for each product and define the vehicle capacity in terms of the shipping units. Nonetheless, the value of the shipping units is only an approximation of the actual shipping volume with respect to the vehicle capacities.

• Orders generally became available 8 to 12 days before they were due. Due to credit holds and other exceptions, many order availability dates in the data set were unrealistic. We applied conservative rules to replace outliers with more reasonable values. However, discrepancies may still exist between the order availability dates we used and the dates that were actually used by the company.

• A human expert’s schedule is part of the data set. It reflects the actual production schedule and vehicle routing schedule used by the company.

In our implementation of the SFRS approach, we set the planning cycle to be 5 days and divided the country into 5 different regions.

The simulation was conducted on a Pentium II - 300 MHZ personal computer. Most of the data manipulation is done through Microsoft Access. We also use some of the GIS features in TransCAD. MILP modeling makes use of software called LPSQL, which uses the XA optimization software library as the underlying solver. A detailed description of the LPSQL software can be found in Jackson and Rappold (1996). Figure 4 illustrates the relationship among all the software components in the simulation. A detailed description of the RFSS software system can be found in Jackson (1998)

![Diagram](image-url)

Figure 4: Simulation Software Components
Because we have many components in the simulation and we did not fully automate the simulation, it is difficult to estimate the computation time accurately. On the average, it takes approximately 30 minutes of elapsed time to simulate a day in the RFSS approach and 20 minutes of elapsed time to simulate a day in the SFRS approach. For problems of this size, such computation time is acceptable for practical applications.

We compare our results with the actual schedule used in the company. Table 1 shows the comparison.

<table>
<thead>
<tr>
<th></th>
<th>Expert</th>
<th>RFSS</th>
<th>SFRS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fleet Miles</td>
<td>513,783</td>
<td>438,951</td>
<td>533,237</td>
</tr>
<tr>
<td>Contract Miles</td>
<td>575,709</td>
<td>451,260</td>
<td>480,852</td>
</tr>
<tr>
<td>Fleet Stops</td>
<td>1,339</td>
<td>965</td>
<td>1,051</td>
</tr>
<tr>
<td>Transportation Cost</td>
<td>$1,757,558</td>
<td>$1,435,717</td>
<td>$1,620,797</td>
</tr>
<tr>
<td>Number of late orders</td>
<td>1,906</td>
<td>438</td>
<td>1,814</td>
</tr>
<tr>
<td>Number of late order days</td>
<td>12,328</td>
<td>648</td>
<td>4,667</td>
</tr>
<tr>
<td>Overtime (in linear feet)</td>
<td>377,107</td>
<td>364,627</td>
<td>21,285</td>
</tr>
<tr>
<td>Coefficient of Variation of Workload</td>
<td>0.173</td>
<td>0.158</td>
<td>0.086</td>
</tr>
<tr>
<td>Transportation Cost Savings</td>
<td>-</td>
<td>18.31%</td>
<td>7.78%</td>
</tr>
<tr>
<td>Reduction of Late Orders</td>
<td>-</td>
<td>77.92%</td>
<td>4.83%</td>
</tr>
<tr>
<td>Reduction of Late Order Days</td>
<td>-</td>
<td>99.74%</td>
<td>62.14%</td>
</tr>
<tr>
<td>Reduction in Overtime</td>
<td>-</td>
<td>3.31%</td>
<td>94.46%</td>
</tr>
</tbody>
</table>

Table 1: Comparison of Simulation Results

Expert, RFSS and SFRS refer to the human expert schedule, the result from the RFSS approach and the result from the SFRS approach, respectively. Since we could not obtain the cost of production overtime and the cost of lateness in terms of dollar values from the company, we used some physical measures. As shown in the table, RFSS results in over 18% savings in transportation costs, while SFRS saves 7.78%. RFSS also has huge reduction in term of lateness measurement. RSFR, however, has significant savings in term of overtime reduction. It is also the best in term of the production smoothing. It has the lowest coefficient of variation of workload. Figure 5 illustrates this graphically.

The three types of performance measures are also interchangeable, depending on the parameter settings in the experiment. It should be noted that some errors in estimating the shipping units and the order availability dates may have contributed to the savings we have noted over the expert scheduler. However,
Figure 5: Workload Comparison
it should be noted that both the RFSS algorithm and the SFRS algorithm dominate the expert schedule in every measurement. This indicates that both algorithms show promising improvement even in the absence of the dollar values associated with some of the physical measurements.

5 Conclusion

We have explored ways to integrate production scheduling and transportation scheduling without finished goods inventory. The optimization techniques have demonstrated promise to reduce costs without the presence of inventory. Both our heuristics have out-performed the human expert schedule in a retrospective simulation study. The Recombinant RFSS algorithm is a robust approach to the VRP that handles side constraints such as carrier choices. We believe that the SFRS approach is a superior optimization due to its ability to smooth production loads.

References


