Routing with Early Ordering for Just-In-Time Manufacturing Systems

Mingzhou Jin, Kai Liu, and Burak Eksioglu

Department of Industrial Engineering, Mississippi State University,
P.O. Box 9542, Mississippi State, MS, 39762, USA
mjin@ie.msstate.edu

Abstract. Parts required in Just-In-Time manufacturing systems are usually picked up from suppliers on a daily basis, and the routes are determined based on average demand. Because of high demand variance, static routes result in low truck utilization and occasional overflow. Dynamic routing with limited early ordering can significantly reduce transportation costs. An integrated mixed integer programming model is presented to capture transportation cost, early ordering inventory cost and stop cost with the concept of rolling horizon. A four-stage heuristic algorithm is developed to solve a real-life problem. The stages of the algorithms are: determining the number of trucks required, grouping, early ordering, and routing. Significant cost savings is estimated based on real data.

1 Introduction

The Just-In-Time (JIT) philosophy originated from the work of Taiichi Ohno at Toyota Motor Company and made its way to the US about 20 years ago [1]. The JIT philosophy is now adopted by most automakers all over the world. Based on the JIT philosophy, inventory is considered a big cost contributor so that the goal is to reduce inventory levels to “zero” [2]. Therefore, parts are ordered and transported only when they are needed in the production. Based on a recent project with one of the major automakers in the US that implements a JIT system, we found out that the inbound logistics decision making process has the following procedures:

1. The production plan is determined based on dealer orders or forecasted demand, and it is derived by manufacturing needs such as line balancing.
2. The parts are ordered based on daily production needs, and the logistics group has little control on how many to order and when to order.
3. Milk-runs (routes) are determined based on average demand. Following the milk-runs trucks visit the suppliers to pick up the required parts and then come back to the manufacturing plant. Each truck has one run everyday. A supplier may be visited by one or more trucks. This is different from the typical capacitated vehicle routing problem (CVRP), in which each supplier is visited by exactly one truck. The problem is similar to the split delivery vehicle routing problem (SDVRP) [3, 4].

Because inbound logistics costs are not considered during production planning, high volatility of part consumption in the assembly line leads to frequent and small

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3982, pp. 748 – 756, 2006.

batches. The practice of assigning the suppliers to the trucks (milk-runs) based on average daily demand and keeping the same routes every day makes the problem worse. Furthermore, routes are typically determined manually, which also hurts the efficiency. Truck utilizations can be as low as 30%, while sometimes overflow happens when additional trucks are required.

In this paper we address the above mentioned problem of the mismatch between production and logistics. To eliminate this mismatch, "dynamic routing" and "early ordering" are proposed as solutions. Dynamic routing means that the routes are determined daily based on production needs. Though integrated production planning and route scheduling is implemented in some other industries [5], in the automotive industry it is difficult to fully incorporate logistics needs into production planning. The industry has implemented a manufacturing driven JIT system for such a long time that any major change requires approval from high-level management, which is usually difficult to achieve. The proposed early ordering policy will not affect the production planning process. In the automotive industry, production plans are made several days before actual production starts. Thus, the required parts are known several days before they are actually consumed in the assembly line. Early ordering policy simply allows the parts to be shipped one or two days early to save transportation cost. However, late ordering is not allowed in order not to disturb the manufacturing process. The parts that are ordered early will be stored in the inbound warehouse for one or two days and possibly increase inventory holding costs.

In Section 2, a mixed integer programming (MIP) model is proposed for the routing problem with early ordering in a JIT environment. A heuristic algorithm to solve the model is presented in Section 3. Section 4 concludes the paper and includes some cost saving estimates based on a real case.

2 A Mixed Integer Programming Model for Daily Routing with Early Ordering

In an assembly system such as the auto assembly plant described above, there are two main costs: transportation and early ordering inventory costs. The transportation cost is composed of a fixed cost for each truck, a variable cost for each mile, and a fixed cost for each stop. Among them, the fixed cost for each truck dominates the others. In the literature, most routing studies do not consider the stop issue. The automotive company that we have studied has to pay a fixed amount to the trucking companies for each stop on the routes because a stop means additional handling time and effort. The number of stops is also a constraint because a truck can not finish a route in one day if it has to make too many stops. In the standard CVRP models, since each supplier can be visited exactly once, the number of total stops is fixed. Therefore, there is no need to consider stop costs in the CVRP models. Though the routing decision influences the number of stops in an SDVRP model, stop costs and constraints are usually not addressed in the SDVRP literature [6-8].

Since early ordering is allowed, additional inventory holding costs are considered in the proposed model. Typically, a major component of the inventory holding cost is the cost of capital invested [1]. However, automakers and their suppliers have long-term relationships, and the payments are made periodically (e.g. weekly or biweekly).

Therefore, the inventory holding costs are mainly driven by the space occupied rather than the capital invested. The demand for one part could be several hundred pieces or more every day, and they are held in containers during shipping and handling. The number of parts in a container is called a unit load, which may have several dozen or up to hundred pieces of the same part. Therefore, the demand and the amount delivered are measured in unit loads rather than pieces. The notation and the model for the routing problem with early ordering are given below.

Parameters:

- K: the number of available trucks (k=1,2,…,K);
- T: the number of days in the planning horizon (t=1,2,…,T);
- P: the set of parts ($p \in P$);
- N: the number of suppliers (i,j=0,1,2,…,N; 0 is used for the origin);
- C: truck capacity;
- u: the inventory holding cost per unit space for early ordered parts;
- q: the fixed cost per truck per day;
- λ: the variable transportation cost per truck per mile;
- w: the cost for one stop;
- $d_{t,p}$: the demand (in unit loads) for part p on day t (with lead time);
- $c_{i,j}$: the distance from supplier i to supplier j;
- $r_{p,i}$: indicates whether or not part p is provided by supplier i (*0*: no; *1:* yes; note that each part is provided only by one supplier; $\sum_{i=1}^{N} r_{p,i} = 1$, $p \in P$);
- v_p: space required by one unit load of part p;
- S: the maximum number of stops allowed for each truck.

Decision variables:

- $o_{t,k,p}$: the unit loads of part p shipped by truck k on day t;
- $x_{t,k,i,j}$: equals 1 if truck k visits supplier j right after supplier i on day t; 0 otherwise;
- $l_{t,k,i}$: the remaining capacity of truck k after visiting supplier i on day t ($l_{0,k,t} = C$);
- $s_{t,k,i}$: equals 1 if truck k visits supplier i on day t; 0 otherwise;
- $I_{t,p}$: the inventory (in unit loads) of p ordered early on day t.

The MIP model for daily routing with early ordering:

$$\min\ u\sum_{t=1}^{T}\sum_{p\in P} v_p I_{t,p} + q\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j=1}^{N} x_{t,k,0,j} + \lambda\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{i,j=0}^{N} c_{i,j} x_{t,k,i,j} + w\sum_{t=1}^{T}\sum_{k=1}^{K}\sum_{j=1}^{N} s_{t,k,j} \tag{1.1}$$

$$s.t.\quad I_{t-1,p} - I_{t,p} + \sum_{k=1}^{K} o_{t,k,p} = d_{t,p} \qquad t = 1,...,T;\ p \in P; \tag{1.2}$$

$$-C \cdot s_{t,k,i} + \sum_{p\in P} v_p r_{p,i} o_{t,k,p} \le 0 \qquad t = 1,...,T;\ k = 1,...,K;\ i = 1,...,N; \tag{1.3}$$

$$\sum_{i=0,..,N, i\neq j} x_{t,k,i,j} - s_{t,k,j} = 0 \qquad t = 1,...,T;\ k = 1,...,K;\ j = 1,...,N; \tag{1.4}$$

$$\sum_{i=0,\dots,N,i\neq j} x_{t,k,i,j} - \sum_{i=0,\dots,N,i\neq j} x_{t,k,j,i} = 0 \qquad t=1,\dots,T;\ k=1,\dots,K;\ j=1,..,N; \quad (1.5)$$

$$Cx_{t,k,i,j} + \sum_{p\in P} v_p r_{p,i} o_{t,k,p} - l_{t,k,i} + l_{t,k,j} \leq C \qquad t=1,\dots,T;\ k=1,\dots,K;\ i,j=0,\dots,N;\ j\neq i; \quad (1.6)$$

$$\sum_{j=1}^{N} s_{t,k,j} \leq S \qquad t=1,\dots,T;\ k=1,\dots,K; \quad (1.7)$$

$l_{t,k,i}, I_{t,p}, o_{t,k,p} \geq 0;\ s_{t,k,i}, x_{t,k,i,j}$: binary ; $o_{t,k,p}$:integers.

The objective function (1.1) minimizes the sum of the inventory holding costs and the transportation costs. The first constraint set (1.2) represents the inventory evolvement over days and ensures that the production needs of all parts are satisfied. The second constraint set (1.3) ensures that only those trucks that visit supplier *i* pick up parts from supplier *i*. The third constraint set (1.4) help obtain the numbers of truck stops. The fourth constraint set (1.5) is used to keep the flow at each supplier balanced (i.e. the number of trucks arriving at a supplier equals the number of trucks leaving that supplier). The fifth constraint set (1.6) makes sure that the amount picked up by a truck does not exceed its capacity and eliminates sub-tours. The last set of constraints (1.7) is used to make sure that the number of stops by a truck does not exceed the maximum number allowed.

The proposed model combines transportation and inventory decisions. It is a variant of the SDVRP with additional constraints to address the issues of truck stops and early ordering. The standard SDVRP takes the total required space at each demand point into consideration, but this model addresses the problem at the part level. Since thousands of parts are required and hundreds of suppliers need to be visited every day, the model of the whole inbound logistics for this automaker is very large. The SDVRP itself is a well-known *NP-complete* problem [3]. Therefore, good solutions rather than optimal solutions are expected in practice for a large-scale SDVRP. The early ordering policy makes the computational burden even heavier by adding one more dimension of time into the problem. In reality, the suppliers are usually grouped by regions, and several transportation service providers are in charge of one region. Thus, the logistics problem of each region can be solved separately. However, dozens of suppliers and hundreds of parts are usually involved in a region. In a problem with 15 suppliers, 150 parts, and 7 available trucks there will be more than 3000 binary variables when only one-day early ordering is allowed. When CPLEX [9], a commercial optimization software, is used to solve the model, 800MB of memory is used up after one day of calculations while the gap between the lower and upper bounds is still more than 90%. Therefore, fast heuristics are necessary for daily decision making.

3 A Four-Stage Heuristic Algorithm

A number of constructive heuristics can be found in the literature for the CVRP or the SDVRP. Most of them are two-stage algorithms including a grouping stage and a routing stage. Typically, a bin packing problem with restrictions is used in the grouping stage. Belenguer et al. [4] develop an algorithm based on a lower bound to solve the SDVRP (this needs some more explanation). Archetti et al. [5] use dynamic programming to solve SDVRP instances where vehicles have small capacities. For the model given in Section 2, we develop a four-stage heuristic algorithm. The basic scheme is illustrated in Fig. 1.

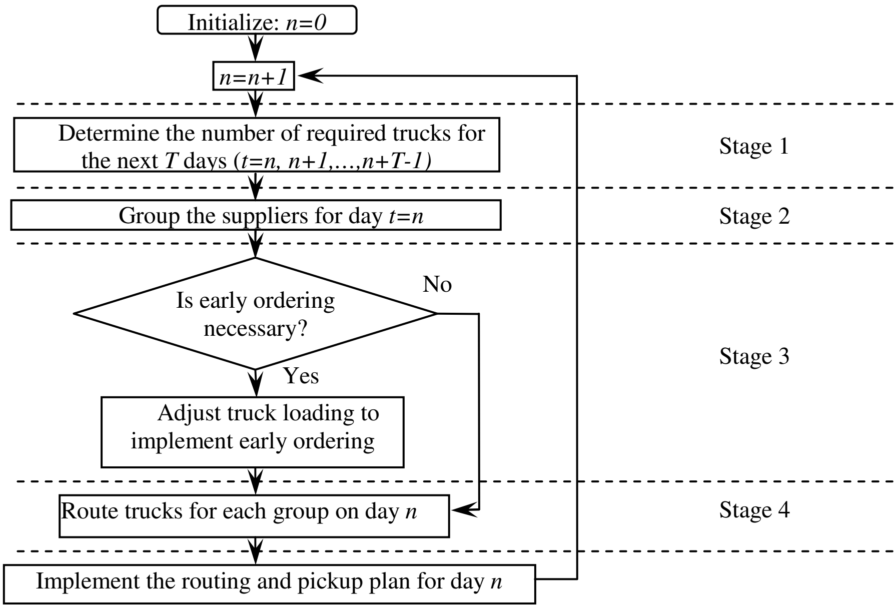

Fig. 1. The basic scheme of the four-stage heuristic

In the first stage, the number of required trucks is calculated for days $t=n$, $n+1,\ldots,$ $n+T-1$. Initially, early ordering is not considered and the number of required trucks is found by the following equation:

$$K_t = \frac{\theta_t \text{(Total required volume of day } t)}{C \text{ (truck capacity)} \times 0.9 \text{ (Allowance)}}, \tag{2}$$

where $\theta_t = \sum_{p \in P} v_p d_{t,p}$ and the allowance of 0.9 is used to account for space that may be wasted because of unit loads. The number of required trucks on day t without early ordering is $\lceil K_t \rceil$. If only one-day early ordering is allowed, early ordering is implemented if the following two conditions are satisfied:

$$\lceil K_t + K_{t+1} \rceil < \lceil K_t \rceil + \lceil K_{t+1} \rceil; \tag{3.1}$$

$$\text{and} \quad K_{t+1} - \lfloor K_{t+1} \rfloor \le UB. \tag{3.2}$$

Since the fixed cost for each truck dominates other costs, early ordering is only implemented when a truck can be saved on the next day (condition (3.1)). For example, if $K_t=2.3$ and $K_{t+1}=2.2$ then three trucks are required for both days without early ordering. If *0.2* truck space of parts can be moved from day t+1 to day t (i.e. $K_t=2.5$ and $K_{t+1}=2$), one truck can be saved on day t+1. Also, note that early ordering increases inventory holding costs. Therefore, it is implemented only when the space moved from day t+1 to day t is not more than *UB* (e.g. 30%) of the truck capacity (condition (3.2)).

In the second stage, grouping is done to determine which suppliers should be visited by each truck. There are many grouping heuristics in the literature. The basic idea is to group the nearby suppliers together without violating the truck capacity constraints. We propose the following optimization model to solve the grouping problem for each day t.

$$\min\ w\sum_{i=1}^{N}\sum_{k=1}^{K} s_{t,k,i} + \tau\sum_{i=1}^{N}\sum_{k=1}^{K}(J_{i,k}^{x+} + J_{i,k}^{x-} + J_{i,k}^{y+} + J_{i,k}^{y-}) \tag{4.1}$$

$$s.t.\quad \sum_{p\in P} v_p o_{t,k,p} \le C \qquad k=1,2,...,K; \tag{4.2}$$

$$\sum_{k=1}^{K} o_{t,k,p} = d_{t,p} \qquad p \in P; \tag{4.3}$$

$$\sum_{p\in P} v_p r_{p,i} o_{t,k,p} \le C s_{t,k,i} \qquad k=1,2,...,K;\ i=1,2,...,N; \tag{4.4}$$

$$\sum_{i=1}^{N} s_{t,k,i} \le S \qquad k=1,2,...,K; \tag{4.5}$$

$$q_k^x - b_i^x \le M(1-s_{t,k,i}) + J_{i,k}^{x+} \qquad i=1,2,...,N;\ k=1,2,...,K; \tag{4.6}$$

$$b_i^x - q_k^x \le M(1-s_{t,k,i}) + J_{i,k}^{x-} \qquad i=1,2,...,N;\ k=1,2,...,K; \tag{4.7}$$

$$q_k^y - b_i^y \le M(1-s_{t,k,i}) + J_{i,k}^{y+} \qquad i=1,2,...,N;\ k=1,2,...,K; \tag{4.8}$$

$$b_i^y \text{-} q_k^y \le M(1-s_{t,k,i}) + J_{i,k}^{y-} \qquad i=1,2,...,N;\ k=1,2,...,K; \tag{4.9}$$

$$s_{t,k,i} : binary;\quad o_{t,k,p} \ge 0,\ \text{integer};\ q_k^x, q_k^y, J_{i,k}^{x+}, J_{i,k}^{x-}, J_{i,k}^{y+}, J_{i,k}^{y-} \ge 0\,.$$

where

b_i^x, b_i^y :	the (x, y) coordinators of supplier i;
M:	a large number to facilitate modeling;
τ:	cost per unit distance;
q_k^x, q_k^y :	the latitude and longitude of the virtual center for truck k;
$J_{i,k}^{x+}, J_{i,k}^{x-}, J_{i,k}^{y+}, J_{i,k}^{y-}$:	the rectangular distance between supplier i and virtual center of truck k.

If truck k serves a set of suppliers (I_k) in the grouping model, its virtual center is defined as (q_k^x, q_k^y) that minimizes $\sum_{i\in I_k}\left(\left|b_i^x - q_k^x\right| + \left|b_i^y - q_k^y\right|\right)$. The objective function (4.1) minimizes the total costs including the stop cost and the distance cost approximated by the sum of rectangular distances between the virtual center of the trucks and the suppliers served by those trucks. The first constraint set (4.2) forces the load of each truck to be less than or equal to the capacity. Constraint set (4.3) has demand satisfied for all parts requested by the assembly line. Constraint set (4.4) ensures that truck k stops at supplier i if parts are to be picked by truck k from that supplier. Constraint set (4.5) makes sure that the number of stops by a truck does not exceed the maximum number allowed. The last four constraint sets (4.6-4.9) are used to obtain the rectangular distances from the suppliers to the virtual centers of the trucks. Though the model looks cumbersome, the number of variables and constraints are significantly less than those of the original model (1.1-1.7). A problem with 15 suppliers, 150 parts and 7 available trucks has about 105 binary variables and 750 constraints. CPLEX can yield a solution to such a relatively small problem in 5 minutes with less than 1% gap on average.

In the third stage, how to implement the early ordering policy is determined. Let $L_{k,t}$ be the remaining capacity of truck k on day t. The following algorithm given in Fig. 2 is developed for implementing one-day early ordering. Assume e_t is the total space occupied by the parts ordered early on day t ($e_t = 0.9C(K_{t+1} - \lfloor K_{t+1} \rfloor)$.

Step 0. Initialize k ($k = 1$)

Step 1. Move all parts provided by one supplier i, served by truck k and satisfying both of the following conditions:

- they are needed on both day t and $t+1$:

$$\sum_{p \in P} r_{p,i} o_{t,k,p} \geq 0 \quad \text{and} \quad \sum_{p \in P} r_{p,i} d_{t+1,p} \geq 0\,;$$

- the total day $t+1$ volume from the supplier i does not exceed the remaining capacity: $\sum_{p \in P} r_{p,i} d_{t+1,p} \leq C - L_k$;

If a movement happens,

- update the utilized capacity L_k of truck k;
- update the total early ordered volume.

Step 3. If the total early ordered volume reaches e_i, go to end.

Step 4. If $k < K$: $k=k+1$ and go to step 1.

Step 5. Move the parts satisfying the following conditions for truck $k=1,\ldots,K$ until the total early ordered volume reaches e_i:

- It is needed on both day t and $t+1$: $o_{t,k,p} \geq 0$ and $d_{t+1,p} \geq 0$
- Day $t+1$ volume of the same part does not exceed the remaining capacity of the truck: $d_{t+1,p} \leq C - L_k$

Step 6. If the total early ordered volume is still smaller than e_i, arbitrarily move the parts needed on both days until the early ordered volume reaches e_i.

Fig. 2. The Algorithm to Determine the Early Ordering Policy

The algorithm that determines the early ordering policy first tries to reduce the number of stops on day *t+1* without increasing the number of stops on day *t*. The second priority is to reduce the number of handlings of the same parts.

The fourth stage deals with the routing problem for each truck which is a standard Travel Sales Problem (TSP). Though the TSP is an *NP-hard* problem, its optimal solution can be obtained in seconds by CPLEX in this case because each truck usually has at most five stops.

The concept of rolling horizon is used for the overall algorithm. If only one-day early ordering is allowed ($T=2$), the first step is implemented for two days ($t=n$ and $n+1$). Grouping and routing models are only solved for the first day ($t=n$). On the next day, the second day's demand information will be updated, and the early ordered parts will be deducted from it. The four-stage algorithm will be implemented after updating $n=n+1$ with the new information of the third day's demand.

4 Implementation and Conclusion

The proposed four-stage algorithm is implemented on a real inbound logistics problem faced by a major automotive company in the US. The region under study has 15 suppliers and 158 parts, and usually 4 to 7 trucks are required every day. Only one-day early ordering is allowed because of information availability and inventory concerns. One month's worth of real data is used, and the result is obtained in 10 minutes. The average truck utilization is improved from 40% to 80% while its variability over trucks also becomes much smaller. The total cost savings is about 20% including 24% savings on the number of used trucks, 17% savings on the total number of stops, and 15% savings on the total traveled distance. Early ordering does not happen frequently in that about 2% of the parts (in space) are ordered early. Of the total 20% savings, about 4% is contributed by the early ordering policy.

This paper presents a mixed integer programming model and a heuristic algorithm to improve the inbound logistics for Just-In-Time manufacturing systems. A dynamic routing policy and an early ordering policy are proposed to reduce the total cost. The proposed model and the algorithm are tested using real-life data obtained from an auto manufacturer. The recommended policies and the proposed heuristic algorithm works well in the sense that the computational speed is high while the quality of the solution is much better than that of the solution currently used by the auto manufacturer.

In this paper, only a constructive heuristic is discussed without any improvement phase. A future research extension is to develop more sophisticated improvement heuristics to obtain better solutions to the problem.

References

1. Askin, R., Golderg, J.: Design and Analysis of Lean Manufacturing. John Wiley & Sons, New York (2002).
2. Monden, Y.: Toyota production system - 3rd edition: An Integrated Approach to Just-In-Time. Institute of Industrial Engineering, Norcross, GA (1993).
3. Dror, M., Trudeau, P.: Savings by Split Delivery Routing. Transportation Science 23 (1989) 141-145.

4. Dror, M., Laporte, G., Trudeau P.: Vehicle Routing with Split Delivery. Discrete Applied Mathematics 50 (1994) 239-254.
5. Bredström, D., Rönnqvist, M.: Integrated Production Planning and Route Scheduling in Pulp Mill Industry. Proceedings of the 35th Hawaii International Conference on System Sciences (2002).
6. Belenguer, J.M., Martinez, M.C., Mota, E.: Lower Bound for the Split Delivery Vehicle Routing Problem. Operations Research 48 (2000) 801-810.
7. Archetti, C., Mansini, R., Speranza, M.G.: The Split Delivery Vehicle Routing Problem with Small Capacity. Technical Report, Department of Quantitative Methods, University of Brescia (2001).
8. Lee, C.G., Epelman, M., White, C.C.: A Shortest Path Approach to the Multiple-vehicle Routing with Split Picks-up. To appear in Transportation Research - Part B (2005).
9. ILOG, ILOG CPLEX 9.0 User's Manual, 2003.