Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

A Periodic Inventory Routing Problem at a Supermarket Chain

Vishal Gaur, Marshall L. Fisher,

OPERATIONS RESEARCH
Vol. 52, No. 6, November–December 2004, pp. 813–822
ISSN 0030-364X | EISSN 1526-5463 | 04 | 5206 | 0813

DOI 10.1287/opre.1040.0150

OR PRACTICE

A Periodic Inventory Routing Problem at a Supermarket Chain

Vishal Gaur
Leonard N. Stern School of Business, New York University, New York, New York 10012, vgaur@stern.nyu.edu

Marshall L. Fisher
The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania 19104-6340, fisher@wharton.upenn.edu

Albert Heijn, BV, a supermarket chain in the Netherlands, faces a vehicle routing and delivery scheduling problem once every three to six months. Given hourly demand forecasts for each store, travel times and distances, cost parameters, and various transportation constraints, the firm seeks to determine a weekly delivery schedule specifying the times when each store should be replenished from a central distribution center, and to determine the vehicle routes that service these requirements at minimum cost. We describe the development and implementation of a system to solve this problem at Albert Heijn. The system resulted in savings of 4% of distribution costs in its first year of implementation and is expected to yield 12%–20% savings as the firm expands its usage. It also has tactical and strategic advantages for the firm, such as in assessing the cost impact of various logistics and marketing decisions, in performance measurement, and in competing effectively through reduced lead time and increased frequency of replenishment.

Subject classifications: inventory: inventory routing; networks, matchings: application to inventory routing; transportation, vehicle routing: algorithm and implementation.
Area of review: OR Practice.
History: Received October 2001; accepted November 2003.

1. Introduction

Albert Heijn, BV is a leading supermarket chain in the Netherlands with 1,187 stores and about \$10 billion in sales. It is a subsidiary of Ahold Corporation, a large retailing conglomerate that owns many supermarket chains around the world, with about 8,500 stores and total sales of \$52.5 billion. Our data include the stores of Albert Heijn as well as its sister chain, Etos, also owned by Ahold Corporation. All figures are based on data for the year 2000. Albert Heijn faces a vehicle routing and delivery scheduling problem once every three to six months with the following main features:

1. The distribution network consists of one national distribution center (DC) and four regional DCs. Each store is assigned to a regional DC that provides all shipments to the store. Figure 1 shows the flow of information and goods through the distribution network.

2. Each store is replenished several times during a week. The time intervals between successive deliveries should not exceed a specified limit. This constraint is necessitated by the perishable nature of merchandise, small backroom space in stores, and, often, by large daily sales volume. Typically, stores receive at least one delivery each day.

3. The delivery times for a store, once determined, remain the same every week. For example, Stores A and B may be served by a truck starting at 9 A.M. on every Monday, at 10 A.M. and 4 P.M. on every Tuesday, and so on for the other days of the week. Albert Heijn calls this periodic delivery schedule the "VVM heartbeat" (VVM is short for van daag voor morgen, which translates as "today for tomorrow").

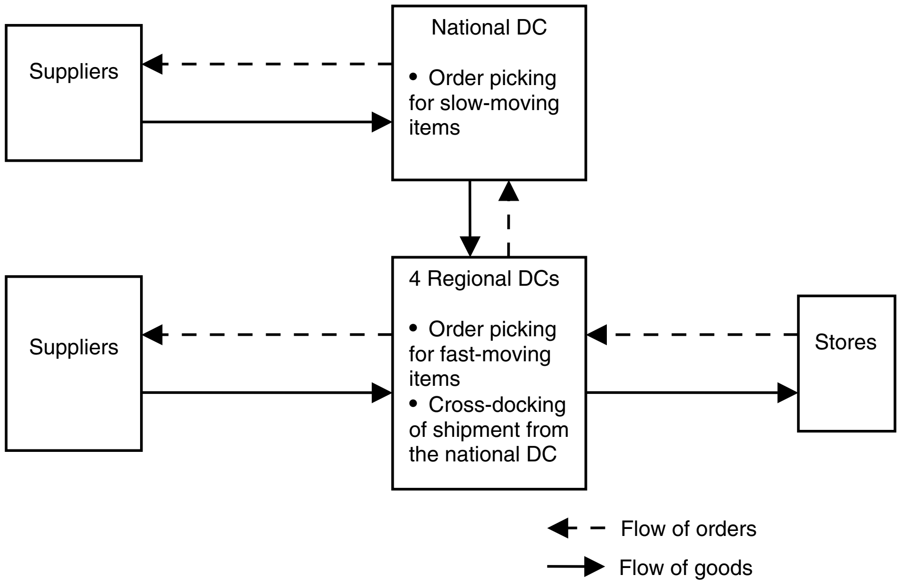
4. Demand at each store is random and varies with time. Figure 2(a) shows the hourly demand rate at one store during a week (the store is open six days during the week for 12 hours each day, and the figure shows only the opening hours). Here, demand is expressed in units of volume and aggregated across all items. Figure 2(b) shows the weekly demand rate at the same store during a year. While the hourly demand rate varies significantly during the week, the same pattern repeats every week except for a few weeks of holidays and promotions.

5. While the delivery schedule is constructed using forecasts of demand, shipments are based on actual orders received from stores. If the order from a store exceeds available truck capacity, then an extra shipment is scheduled to that store on that day.

6. Routes are assigned to a heterogeneous fleet of trucks with different fixed and variable cost parameters and different loading-unloading times. Most trucks are leased from fleet owners after the delivery schedule has been determined; however, some trucks are owned by Albert Heijn and have fleet-size restrictions.

7. The departure times of trucks from the DCs are evenly spaced so that workers at the DCs have sufficient capacity to load the trucks on time, and have a balanced

Figure 1. Distribution network of Albert Heijn.



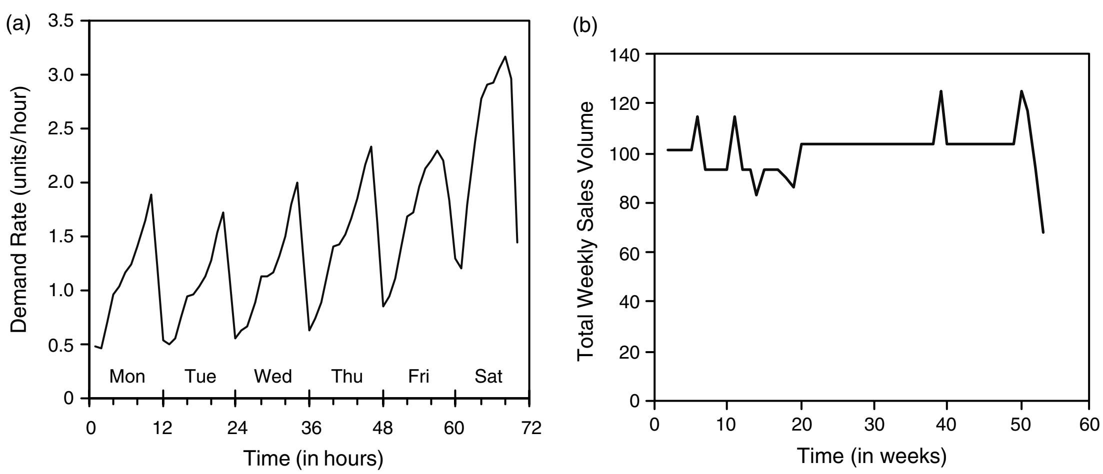
workload throughout the day. A fluctuating workload would lead to the workers having too much work in one hour and too little work the next hour, which is considered undesirable.

8. Thus, the delivery scheduling problem must incorporate not only variable transportation costs, but also randomness of demand, fixed truck rental costs, fleet-size constraints, and workload capacity at the DC.

A periodic schedule provides many advantages to Albert Heijn. It simplifies workforce scheduling at the DCs for loading trucks, and at the stores for unloading trucks and stocking shelves. Stores know in advance when to place orders, and suppliers deliver merchandise to the DCs according to the same periodic schedule. Due to this synchronization, merchandise does not have to wait in the DCs and can often be cross-docked from the suppliers' trucks to the stores' trucks. Last but not the least, Albert Heijn is able to enter into cheaper long-term contracts with truck-leasing companies for specific predetermined routes each day.

We describe the development and implementation of a system to solve the vehicle routing and delivery scheduling problem at Albert Heijn, given the above features and other complicating constraints described later in the paper. The system has three modules. The inventory routing module jointly determines the delivery times of stores and vehicle routes to service them, ignoring the variability of demand and fleet-size constraints. We refer to this problem as the periodic inventory routing problem or the periodic IRP. Using the periodic IRP as a subproblem, the truck assignment module incorporates constraints on truck fleet-size and various time restrictions such that the fixed and variable costs of transportation are minimized. Finally, the workload balancing module readjusts the departure times of routes within feasibility constraints and given time windows to

Figure 2. (a) Hourly demand rate during a week for a typical store. (b) Total weekly demand at a typical store for a one-year period.



match the target workload profile of the DC as closely as possible.

Before the development of our system, route planners at Albert Heijn used their knowledge of the geography and demand patterns of stores to assign delivery times to them. They then planned routes by solving a vehicle routing problem with time windows on a standard software package. However, this hierarchical approach can produce a worse solution than if the delivery times and routes are determined jointly. The quality of the solution to this problem can have substantial impact on the bottom line of the firm because it affects not only transportation costs, but also inventory costs, labor costs, and warehousing and distribution capacity requirements. Among these, transportation costs alone comprise about 15% of the sales at Albert Heijn, while net profit comprises about 1%–2% of sales. Thus, Albert Heijn aimed to realize significant cost savings by optimizing its replenishment schedule. We show that the implementation of our system not only gives such savings, but also provides strategic and tactical benefits to the firm.

The rest of this paper is organized as follows. Section 2 reviews the literature on inventory routing problems. Section 3 describes the mathematical formulation and our solution approach for the periodic IRP, and §4 presents the results of a computational analysis of the periodic IRP module. Section 5 describes the design and development of the entire software system. It discusses the computation of parameters, the complicating constraints, and the truck assignment and workload balancing modules. Section 6 discusses implementation issues, and §7 presents key insights from the implementation and its impact on Albert Heijn.

2. Literature Review

IRPs have been widely studied in the literature in different contexts. The models for these problems can be broadly classified into three types: the strategic IRP, the tactical or finite horizon IRP, and the infinite horizon IRP. We describe the main features of these three types of models in brief. Toth and Vigo (2002) provide recent surveys by several researchers of algorithms and applications of both vehicle routing and inventory routing problems. Also, see Bramel and Simchi-Levi (1997) and Federgruen and Simchi-Levi (1995) for earlier surveys of IRPs.

The strategic IRP deals with estimating the minimum cost (or size) vehicle fleet required to supply inventory to a set of customers when only the probability distributions of the demand rates at the customers are known. This problem arises because vehicles must be purchased or leased for long time periods. Larson (1988) and Webb and Larson (1995) study the strategic IRP in the context of acquisition of a barge fleet for the New York City Department of Environmental Protection and propose a heuristic that estimates the fleet size by dividing customers into a set of clusters.

The tactical IRP considers a supplier serving a set of customers over a finite time horizon using a fixed private truck fleet. In each period, the supplier faces a trade-off between current costs and future costs: The fewer the number of customers replenished in the current period, the lower the current period cost is, but the higher the cost is in future periods. The supplier determines which customers to visit in each period, taking into account transportation costs, target service levels, inventory levels, randomness of demand, and resource constraints. This problem is encountered in the distribution of industrial gases and in supply chains with vendor-managed inventory. Bell et al. (1983), Campbell et al. (2002), Chien et al. (1989), Dror et al. (1985), Dror and Ball (1987), and Golden et al. (1984) present different approaches to solve the tactical IRP. Bard et al. (1998) extend this problem by allowing satellite facilities where a vehicle can be refilled during a route.

The infinite time horizon IRP considers replenishments from a central DC to a set of customers with the objective of minimizing the sum of long-run average transportation, ordering, and inventory holding costs. The demand rates are constant, the time intervals between successive deliveries to a customer must be larger than a specified limit, and the truck fleet is homogeneous and unrestricted in size. Anily and Federgruen (1990) and Bramel and Simchi-Levi (1995) present different algorithms to solve this problem. Chan et al. (1998) derive asymptotic worst-case bounds for solutions belonging to the class of fixed partition policies and those employing the zero inventory ordering property. Gallego and Simchi-Levi (1990) and Barnes-Schuster and Bassok (1997) evaluate the long-run effectiveness of the direct shipment strategy in the infinite time horizon IRP with deterministic and stochastic demand, respectively. Kleywegt et al. (2002) consider an infinite time horizon IRP with stochastic demand and restricted fleet size.

We also note that the periodic vehicle routing problem is a close cousin of the IRP. In this problem, each customer is visited several times during the planning horizon. The problem consists of selecting a sequence of visits from a given set for each customer, and establishing vehicle routes for each day of the planning horizon (see Cordeau et al. 1997 and references cited therein).

Most applications of IRPs in the literature are in the context of supply chains with vendor-managed inventory. In contrast, our paper considers an IRP in a supermarket chain, with the supermarket, rather than the suppliers, owning the distribution network. Our application demonstrates the benefits of integrated vehicle routing and delivery scheduling in this setting. Distinguishing features of this problem are time-varying demand, periodic delivery schedule, and the minimization of both variable transportation costs and fixed rental costs for a heterogeneous restricted fleet of vehicles.

3. Periodic Inventory Routing Problem

3.1. Assumptions

In the periodic IRP, we seek to determine vehicle routes and delivery times for stores to minimize the total variable transportation costs. There is a heterogeneous fleet of trucks; however, fleet-size restrictions are ignored.

We consider solutions for the periodic IRP that belong to the class of fixed partition policies. In a fixed partition policy, the set of customers is partitioned into disjoint subsets (called regions or clusters) with each region served separately and independently from all other regions. All the stores in a region are replenished together. This policy is simple to implement because trucks have to drive the same routes every day. Moreover, this policy enables us to incorporate many types of constraints relatively easily.

Within fixed partition policies, we initially consider the case when the number of stores in a cluster is restricted to at most two. We found in our preliminary analysis that store volumes at Albert Heijn were such that very few trucks visited more than two stores. Therefore, we first compute the optimal fixed partition policy with at most two stores per cluster, and then apply a heuristic to improve the solution by constructing larger clusters. This approach is efficient because a clustering problem with at most two stores per cluster can be solved in polynomial time using weighted matching.

We assume that there is a single homogeneous product with deterministic but time-varying demand. Even though the application involves the replenishment of several items, all items are replenished from a single DC. Thus, we transform the application into a single-product problem by expressing the demand rates of all items in units of weight and volume—measured in fractions of number of containers (also called pallets), and aggregating. Section 5 discusses the assignment of buffer space in each route to manage demand uncertainty.

Because demand is time varying, we allow two types of routes within each cluster: shared routes that visit every store in the cluster, and direct shipments to individual stores in the cluster. Thus, a large store may be served by a direct shipment on a high-volume day, and a shared shipment on a low-volume day.

The objective function consists of transportation costs. Inventory holding costs are ignored because the requirement of frequent replenishment keeps inventory holding costs low.

3.2. Model

We consider a DC serving n stores, $i = 1, \ldots, n$. Let the planning horizon be divided into T hourly periods, λ_{it} be the demand rate at store i at time t, and T_0 be the maximum time allowed between successive deliveries to a store, $T_0 < T$. Let v index different truck types in the fleet, and Q_v be the capacity of trucks of type v. Let k be the index enumerating possible clusters of stores, S_k denote the set of all stores in cluster k, and R_i denote the set of all clusters containing store i.

Let c_{iv} be the cost of transportation from the DC to store i and back using truck type v, and $\bar{c}_{kv}$ be the cost of transportation from the DC to every store in cluster k and back using truck type v. We obtain $\bar{c}_{kv}$ by solving a traveling salesman problem for each cluster k for each truck type v. By the triangle inequality, $\bar{c}_{kv} \geqslant c_{iv}$ for all $i \in S_k$ and all v.

Because we divide the set of stores into disjoint clusters where each cluster is served separately, first consider the periodic IRP for a given cluster, say k. The decision variables for this problem are:

u_{iktv} = number of direct shipments to store i in cluster k at time t using truck type v, $i \in S_k$.

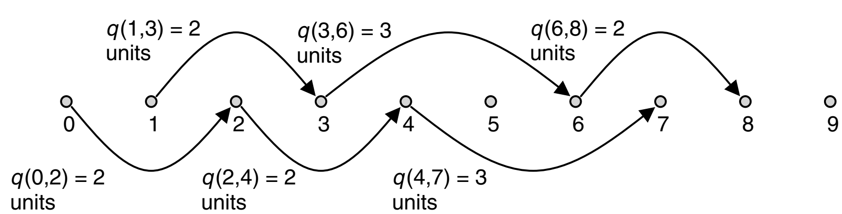
v_{ktv} = number of shared shipments to cluster k at time t using truck-type v.

q^d_{iktv}, q^s_{iktv} = quantity delivered by direct shipments and shared shipments, respectively, to store i in cluster k at time t using truck type v, and 0 otherwise, $i \in S_k$.

Because the demand is periodic, we have that $\lambda_{i,T+t} = \lambda_{it}$. Further, because the delivery schedule is periodic, we have that $u_{ik,T+t,v} = u_{iktv}$, and likewise for the remaining decision variables.

We solve the periodic IRP for cluster k by formulating it as a set of shortest-path problems. We first illustrate this formulation with a numerical example. Example 1: Let $T = 7$, $T_0 = 3$, and consider a cluster containing a single store with unit demand in each period. We represent the periodic IRP for this cluster on a graph with $T + T_0$ nodes, as shown in Figure 3. The first T nodes correspond to the T time periods in the planning horizon, and the final T_0 nodes are constructed to ensure periodicity in the schedule. An edge, (t_1, t_2), on this graph represents successive deliveries to the cluster at times t_1 and t_2. Further, a path from node t in the set $\{0, 1, 2\}$ to node $T + t$ represents a feasible periodic delivery schedule. For example, the path $1 \to 3 \to 6 \to 8$ represents a delivery schedule with successive deliveries at $t = 1, 3$, and 6. This schedule is periodic because it starts at $t = 1$ and ends at $t = T + 1 = 8$. The schedule meets demand because it delivers two units at $t = 1$, three units at $t = 3$, and two units at $t = 6$. It also meets the delivery

Figure 3. Graph showing delivery schedules for the cluster in Example 1.



Note. Here, $T = 7$ and $T_0 = 3$. The nodes represent delivery times and the edges represent successive deliveries. A path from node $t \in \{0, 1, 2\}$ to node $7 + t$ gives a feasible delivery schedule. $q(i, j)$ denotes the quality delivered at time i when the next delivery is at time j.

frequency constraint because successive deliveries are no more than three periods apart. The path $0 \to 2 \to 4 \to 7$ represents another feasible delivery schedule with successive deliveries at $t = 0, 2$, and 4. We note that the optimal delivery schedule for the cluster is given on this graph by the lowest-cost path from node $t \in \{0, 1, 2\}$ to node $7 + t$.

In general, let G_k denote a graph with the set of nodes $N_k = \{0, \ldots, T + T_0 - 1\}$ and the set of directed edges $A_k = \{(t_1, t_2)\colon 0 \leqslant t_1 < T, t_1 < t_2 \leqslant t_1 + T_0\}$. Nodes $0, \ldots, T - 1$ correspond to possible delivery times in the current planning horizon, and nodes $T, \ldots, T + T_0 - 1$ correspond to possible delivery times in the first T_0 periods of the next planning horizon. With edge (t_1, t_2), we associate the cost of making a delivery to the cluster at time t_1 such that the next delivery occurs at time t_2. Let $c_e(t_1, t_2)$ denote the cost of edge (t_1, t_2). We have

$$c_e(t_1, t_2) = \min \sum_v \left[\bar{c}_{kv} v_{kt_1 v} + \sum_{i \in S_k} c_{iv} u_{ikt_1 v} \right] \tag{1}$$

such that

$$\sum_v (q^d_{ikt_1 v} + q^s_{ikt_1 v}) = \sum_{\tau = t_1}^{t_2 - 1} \lambda_{i\tau} \quad \text{for all } i \in S_k, \tag{2}$$

$$q^d_{ikt_1 v} \leqslant u_{ikt_1 v} Q_v \quad \text{for all } i \in S_k, v, \tag{3}$$

$$\sum_{i \in S_k} q^s_{ikt_1 v} \leqslant v_{kt_1 v} Q_v \quad \text{for all } v, \tag{4}$$

$$u_{ikt_1 v},\ v_{kt_1 v} \text{ integer} \quad \text{for all } i \in S_k, v. \tag{5}$$

Here, the objective function is equal to the sum of transportation costs of all direct shipments and shared shipments to cluster k at time t_1. Constraint (2) states that the quantity delivered to store i at time t_1 is equal to the total demand occurrence at store i between times t_1 and t_2. For simplification, we do not consider early shipments and the implied possibility of building inventory at a store in anticipation of greater routing flexibility in subsequent time periods. We also ignore demand occurrence during the time of travel in the formulation of constraint (2). Constraints (3) and (4) ensure that the number of direct shipments and the number of shared shipments, respectively, are sufficient to deliver the required quantities to cluster k at time t_1. The remaining constraint, (5), imposes integrality of $u_{ikt_1 v}$ and $v_{kt_1 v}$.

Note that (1)–(5) is a simplified vehicle routing problem in cluster k with only two types of routes. When there are at most two stores in a cluster and there is a single truck type, then the value of $c_e(t_1, t_2)$ can be written in closed form. To solve this problem for larger clusters and a heterogeneous vehicle fleet, we apply a heuristic wherein we first construct the least-cost schedule using only shared routes, and then apply single-route improvement and two-route improvement steps to replace shared shipments by cost-improving direct shipments.

Any path on G_k from node $t \in [0, T_0 - 1]$ to node $T + t$ gives a feasible periodic replenishment schedule for cluster k, with the first delivery occurring at time t and subsequent deliveries occurring at the times given by the edges in the path. The schedule is periodic because it has a length of T periods. Let $P(t) = t \to t_1 \to t_2 \to \cdots \to T + t$ denote the shortest path from node $t \in [0, T_0 - 1]$ to node $T + t$, and $C_P(t)$ denote the sum of the costs of the edges on this path. The optimal delivery schedule for cluster k can now be obtained by finding $C_P(t)$ for each node $t \in [0, T_0 - 1]$ and taking the minimum. Thus, if $f(S_k)$ denotes the cost of the optimal delivery schedule for cluster k, then we have $f(S_k) = \min\{C_P(t)\colon 0 \leqslant t < T_0\}$.

Given the values of $f(S_k)$, we formulate the periodic IRP as the following set-partitioning problem denoted **P**. Here, the decision variable x_k equals 1 if cluster k is included in the solution and 0 otherwise:

$$\min \sum_k f(S_k) x_k \tag{6}$$

such that

$$\sum_{k \in R_i} x_k = 1 \quad \text{for all } i = 1, \ldots, n, \tag{7}$$

$$x_k = 0 \text{ or } 1 \quad \text{for all } k. \tag{8}$$

Constraint (7) states that every store must be included in exactly one cluster.

3.3. Special Case: At Most Two Stores in a Cluster

When there are at most two stores in a cluster, then **P** is equivalent to a generalized minimum weight matching problem on a nonbipartite graph. To see this, let $G = (N, A)$ be a graph where each node corresponds to a single store (i.e., a singleton cluster) and each edge represents a cluster of two stores, i.e.,

$$N = \{1, \ldots, n\}, \quad A = \{(i, j)\colon\ i, j \in N,\ \ i \neq j, f(\{i, j\}) < \infty\}.$$

The cost of edge (i, j) is set to $f(\{i, j\})$, and a penalty of $f(\{i\})$ is associated with each node i. A matching on G is a subset of edges, $M \subseteq A$, with the property that each node in the subgraph $G(M) = (N, M)$ is met by at most one edge. The generalized minimum weight matching problem is the problem of finding a matching such that the sum of the costs of the edges in the matching and the penalties of the unmatched nodes is minimum (Christofides 1985).

Generalized minimum weight-matching-based algorithms are well known in the vehicle routing literature (see, for example, Laporte and Semet 2002). We solve the generalized minimum weight matching problem on G by converting it into a maximum weight matching problem on a graph with n vertices and no vertex penalty costs. This conversion is useful because a number of fast algorithms have been developed for the maximum weight matching problem, such as Lawler (1976), Papadimitriou and Steiglitz (1982), Galil et al. (1986), and Gabow and Tarjan (1991).

The maximum weight matching problem corresponding to the generalized minimum weight matching problem on G is defined as follows. Let $G' = (N', A')$ be a nonbipartite graph with n nodes. If $f(\{i\}) + f(\{j\}) > f(\{i, j\})$, then construct an edge (i, j) with cost $f(\{i\}) + f(\{j\}) - f(\{i, j\})$; otherwise, there is no edge connecting nodes i and j. Intuitively, the cost of an edge is the gain from combining the corresponding stores into one cluster instead of replenishing them individually.

Any matching on G' gives a solution to **P** as follows: For every unmatched vertex i, assign store i to a singleton cluster; for every edge (i, j) in the optimal matching, assign stores i and j to one cluster. Further, it can be shown that the optimal solutions to the maximum weight matching problem on G' and the generalized minimum weight matching problem on G are equivalent. Therefore, the maximum weight matching problem solves **P**.

The application of weighted matching and the shortest path problem gives us a very fast algorithm for the IRP. For example, when there is a single truck type, then a shortest path problem on each graph G_k can be solved in $O(T + T_0)^2$ time. There are $O(n^2)$ values of $f(S_k)$ to be determined, and the computation of each $f(S_k)$ requires solving T_0 shortest path problems. This takes $O(n^2 T_0 (T + T_0)^2)$ time. Converting the generalized minimum weight matching problem into a maximum weight matching problem takes $O(n^2)$ time, solving a maximum weight matching problem on a graph with n nodes takes $O(n^3)$ time, and converting the solution of the maximum weight matching problem into a solution of **P** takes $O(n)$ time. Thus, the total time complexity of the algorithm to solve the IRP with at most two stores per cluster is $O(n^2 T_0 (T + T_0)^2 + n^3)$.

3.4. The General Case for Clustering

To solve **P** for any cluster size, we construct a heuristic, the randomized sequential matching algorithm (RSMA), with two main ideas: repeated application of the generalized minimum weight matching algorithm, and randomized splitting of clusters.

In the repeated application of the generalized minimum weight matching algorithm, we begin with an initial solution where each store is a separate cluster and is served by routes independent of the other stores. In the first iteration, we obtain clusters with up to two stores. These clusters become the nodes of a new graph, and clusters of up to four stores are formed in the second iteration. Continuing in this manner, we obtain progressively larger clusters in every iteration. The algorithm terminates when no new clusters are obtained.

Repeated application of weighted matching may result in a suboptimal solution. Therefore, to improve the solution quality, we randomly split the clusters obtained and reapply weighted matching. We assign the probability of splitting a cluster as follows. For every pair of stores i and j, let T_{ij} be a tally of the number of times that stores i and j occur in the same cluster and are adjacent to each other in the optimal traveling salesman tour of the cluster. Then, we assign the probability of splitting a cluster at link (i, j) as $1/(1 + T_{ij})$. Thus, stores that occur together more frequently have a smaller chance of being separated than stores that occur together less frequently. After the split, store i and all stores preceding it on the traveling salesman tour are put in one cluster, and the stores from j onwards are put in another cluster. This probability is applied to every link in the traveling salesman tour, so that a cluster may yield many smaller clusters. After splitting, the weighted matching algorithm is reapplied and the tallies T_{ij} are updated. The algorithm terminates when no improved solution is obtained for a given number of iterations.

We tested algorithm RSMA on eight well-known problems in the vehicle routing literature, obtained from Christofides and Eilon (1969) and Fisher (1994). Because these problems do not have delivery time decisions, the shortest path formulation was not required. Instead, the cost of each cluster was obtained by computing a traveling salesman tour through the stores in that cluster using the Lin-Kernighan 3-opt heuristic (Lin and Kernighan 1973). Our algorithm matched the best feasible solution known in three of the eight cases, and was within 2.44% of the best feasible solution known in the remaining cases. See Gaur (2001) for details.

4. Computational Analysis

We illustrate the results of our implementation using three representative test problems from Albert Heijn. Each problem consists of one week of data taken from one distribution center. The number of stores to be replenished in each case is 207, and the capacity of a vehicle is 26 roll-containers. Table 1 displays the characteristics of the test problems, showing the mean, the maximum, and the minimum of store demand on each day of the week for each problem. Note that mean demand varies by the day of the week, being higher on Friday and Saturday. Also note the wide range of variation of demand across stores. The maximum daily demand on each day is larger than the vehicle capacity, showing the need for more than one delivery a day for some stores.

We compare five different solutions for each test problem. Solution A is computed by the route planners at Albert Heijn, using their previous system. We obtain solutions B and C by using delivery times identical to Albert Heijn and solving a vehicle routing problem with time windows separately for each day of the week. In B, at most two deliveries are allowed per route, and thus we apply weighted matching to compute the optimal solution. In C, there is no restriction on the number of deliveries per route, and thus we apply algorithm RSMA. Solutions D and E are obtained by optimizing both delivery times and vehicle routes. In D, at most two stores are allowed per cluster, while in E there is no restriction on cluster size.

Table 1. Characteristics of IRP test problems from Albert Heijn.

	Characteristics of daily store demand (expressed in roll containers)								
	Problem 1			Problem 2			Problem 3		
	Mean	Max	Min	Mean	Max	Min	Mean	Max	Min
Monday (Day 1)	10.57	34	3	10.5	35	3	10.62	45	3
Tuesday (2)	9.80	45	3	9.66	25	3	9.67	30	3
Wednesday (3)	10.27	42	2	10.49	52	2	10.31	39	2
Thursday (4)	13.71	40	3	14.19	58	3	13.73	50	3
Friday (5)	15.86	43	3	16.25	60	3	16.25	56	3
Saturday (6)	16.36	68	1	17.12	79	3	16.53	80	3

Notes. Number of stores $=207$; vehicle capacity $=26$ roll containers.

Solutions A, B, and C serve as benchmarks to evaluate the savings to Albert Heijn from using our system. Comparison of A with B and C identifies the gain to Albert Heijn from optimizing the vehicle routes with existing delivery times. Comparison of A with D and E identifies the gain to Albert Heijn from jointly optimizing vehicle routes and delivery times. Finally, the comparison of B and C with D and E, respectively, gives the incremental gain from inventory routing.

Table 2 reports summary statistics for these solutions: the total cost for the week, the distance travelled, the total time required for transportation and loading-unloading, the total number of routes driven, and the average number of deliveries per route. The table also shows the % savings in cost, distance, and time for each solution relative to A. We observe that D gives cost savings of 13.8% relative to A and 5.6% relative to B, and E gives cost savings of 17.0% relative to A and 6.0% relative to C. In addition, the total number of routes in B, C, D, and E decline by 20.3%–28.4% relative to A.

Note that the average number of deliveries per route decreased when going from solutions B and C to solutions D and E, respectively. This is so because the number of direct shipments increased when delivery times were set optimally. Also note that the number of deliveries per route in solutions C and E is just marginally larger than in solutions B and D. This is so because there are relatively few clusters with more than two stores. The advantage of a large cluster is that it pools delivery volume across several stores, and thus increases the capacity utilization of trucks. The disadvantage of a large cluster is that it increases the length of the shared route, thus increasing the distance driven per unit shipment, and the transportation cost. Due to this tradeoff, large clusters were formed mainly in cities where stores are located close together.

The above results demonstrate the potential cost savings to Albert Heijn from our system. While these savings are based on the inventory routing module, additional savings were obtained in truck assignment and workload balancing.

Table 2. Computational results for the IRP test problems from Albert Heijn.

Problem	Solution type	Cost	Distance (kilometers)	Time (minutes)	# of routes	# of deliveries per route	% change relative to solution A: Cost	Distance (kilometers)	Time (minutes)
1	A	242,751.4	84,328.7	230,585	985	1.42			
	B	221,732.8	72,632.1	213,784	773	1.81	8.7	13.9	7.3
	C	215,414.6	69,367.3	208,553	697	2.01	11.3	17.7	9.6
	D	208,323.0	64,012.0	203,899	763	1.74	14.2	24.1	11.6
	E	200,463.7	59,500.6	197,716	704	1.91	17.4	29.4	14.3
2	A	247,501.2	85,740.9	235,268	988	1.45			
	B	227,687.8	74,611.2	219,505	796	1.81	8.0	13.0	6.7
	C	220,400.9	70,826.4	213,486	712	2.02	10.9	17.4	9.3
	D	214,054.0	66,042.8	209,314	784	1.73	13.5	23.0	11.0
	E	205,939.2	61,566.7	202,799	727	1.87	16.8	28.2	13.8
3	A	243,146.8	84,225.2	231,134	976	1.44			
	B	223,889.2	73,542.8	215,716	782	1.80	7.9	12.7	6.7
	C	217,094.6	70,035.5	210,088	702	2.00	10.7	16.8	9.1
	D	209,983.2	64,502.5	205,538	767	1.75	13.6	23.4	11.1
	E	202,440.1	60,666.9	199,248	708	1.89	16.7	28.0	13.8

Notes. A = Albert Heijn's solution; B = Vehicle routing problem with time windows with at most two deliveries per route and delivery times identical to Albert Heijn; C = Vehicle routing problem with time windows with no restriction on the number of deliveries per route and delivery times identical to Albert Heijn; D = Fixed partition strategy with at most two stores per cluster and optimal delivery times; E = Fixed partition strategy with no restriction on the number of stores per cluster and optimal delivery times.

5. Application Design

The team to develop the vehicle route scheduling (VRS) system is comprised of the authors, the retail R&D division of Albert Heijn, the logistics manager and route planners from one pilot distribution center, and a computer software firm to program the user interface. Important aspects of the three modules of the system are as below.

Inventory Routing Module. Seven data files are used as inputs in this module: a store directory, hourly demand forecasts (mean and standard deviation) at each store, truck parameters, matrices of travel times and distances between pairs of zip codes, estimated traffic delays between pairs of zip codes at different times of the days, delivery time windows for all stores, and store-vehicle feasibility constraints.

The standard deviations of demand forecasts are used to set buffer space on each route. We set the buffer space on a route as the product of a desired service level and the standard deviation of demand for the period covered by the shipment. Occasionally, an order exceeds the forecast by more than the buffer space, causing a spillover. Spillovers are handled on a daily basis by scheduling additional routes. Because both spillovers and buffer space are expensive, we determined the optimal service level through an intensive simulation exercise trading off the expected cost of spillovers against the cost of added buffer space to the routes.

The truck parameters for each truck type include fixed and variable costs, loading-unloading times, fleet size, constraints on number of hours worked during the day, and both upper and lower bounds on the waiting time between successive routes assigned to a truck. Loading-unloading times and other variable costs are incorporated in the IRP module. Fixed costs and constraints on fleet size, number of hours worked, and waiting time are implemented in the truck assignment module described below.

The cost of each route must include driver pay, tolls, and vehicle-related costs such as depreciation, fuel, and maintenance. While some of these costs are directly related to mileage, others do not depend on mileage but on total elapsed time, including time spent on traffic delays, parking, and loading and unloading the trucks. Therefore, the system computes the cost of each route dynamically as a function of the set of stores to be visited, the time of the day, delivery quantities, and various truck-specific parameters. For this, data on shortest travel times and distances between pairs of zip codes, and data on estimated traffic delays between pairs of zip codes at different times of the day, are purchased from a vendor periodically and uploaded to the system.

We note that travel times and distances are approximate because they are computed using an average of the cost per kilometer and cost per hour across the entire fleet. They do not incorporate differences between truck types and traffic delays. The specification of traffic delays can also lead to pre-emption. For example, if the normal travel time from Point A to Point B were 12 minutes, and the route planner estimated a traffic delay of 10 minutes from 8:00 A.M. to 8:29 A.M., but 0 minutes from 8:30 A.M. to 8:59 A.M., then a truck starting at 8:29 A.M. would arrive at 8:51 A.M. and a truck starting at 8:30 A.M. would arrive at 8:42 A.M.! Rules were encoded to avoid such pre-emption in the computation of travel times.

Other complicating constraints added to the algorithm include delivery time windows, store-specific loading and unloading times, and truck-store feasibility constraints. The set-partitioning and shortest-path-based approach to solve the IRP enabled us to incorporate these constraints relatively quickly in the shortest path subproblem of the formulation.

Truck Assignment Module. This module takes the delivery times and routes obtained from the IRP as input. It assigns routes to trucks for each day such that the sum of fixed leasing costs, waiting costs, and driving costs is minimized, and various time constraints are satisfied. Departure times of routes may be changed within a given tolerance. Further, routes may be reassigned from one truck type to another.

Because fleet size is constrained for some truck types, we embed the IRP module and the truck assignment module in a Lagrangian relaxation-based heuristic. The heuristic associates penalties with constraints on the number of hours available on each truck type on each day. In each iteration, the penalties are updated and the IRP module and the truck assignment module are rerun to generate feasible solutions.

Workload Balancing Module. Each DC has a target workload profile specified for each 15 minute interval of the day. For example, the loading-unloading capacity, expressed in number of pallets, during 9:15 A.M. to 9:30 A.M. may be three times the loading capacity during 2 A.M. to 2:15 A.M. The objective of this module is to perturb the schedule such that the workload at the DC for loading and unloading of trucks has minimum deviation from the targeted profile. Given a set of routes for each day and constraints on the lengths of time windows within which each route is feasible, this module applies a series of heuristics to balance the workload at the DC. The heuristics are complicated by the fact that reassigning a route to a different time can change the duration and the cost of the route due to traffic jam delays.

The VRS system uses a modular architecture so that it can be modified over time for different distribution networks in the firm. The analytical modules are programmed in C++ language, the data files are organized in a Microsoft Access database, and the user interface is programmed in Visual Basic. The user interface performs many functions, including data entry, data integrity verification, report generation, sensitivity analysis, scenario management, and execution of all analytical modules of the system.

6. Implementation

While the development of the VRS system was sponsored by Albert Heijn's retail R&D division, the approval for its implementation depended on the distribution logistics division, which had a separate managerial line of reporting. Therefore, the system had to succeed on many counts to get buy-in from the distribution logistics division. These included not just demonstration of cost savings, but also support from route planners and a gradual phasing of the implementation.

Route planners were involved in system design and testing from the beginning. Prior to our implementation, they used to meet many of the requirements manually, including workload balancing, minimization of fleet size, and traffic delays. Therefore, they provided valuable inputs in system specification and the design of heuristics. Interestingly, the route planners would often try to manually improve the solutions obtained from our system. This enabled us to distinguish between "hard" and "soft" constraints. Soft constraints were parameterized to add flexibility, often at the cost of larger input files.

The system was tested on a large number of simulated and real-life test problems, and was run in parallel with the existing system at Albert Heijn for a period of six months before it was accepted for implementation. It was implemented in three phases, introducing small organizational changes in each phase. At first, it was used with existing delivery times at all stores only to optimize vehicle routes and truck assignment. In the second phase, delivery times were allowed to vary, but the size of each cluster was restricted to at most two stores. In this phase, store managers were convinced of the benefits of setting delivery times optimally rather than arbitrarily. Finally, variable delivery times and unrestricted cluster sizes were allowed.

7. Impact

Albert Heijn originally implemented the VRS system to manage store deliveries in the Netherlands. It was used once every three months to produce a detailed weekly delivery schedule for each store. It was also used every day as a vehicle routing system to manage exceptions, and schedule home deliveries for Albert Heijn's Web-based grocery business. The company realized savings of about 4% of its transportation cost in the first year of implementation, and expects total savings of 12%–20% of its transportation cost in the future as it allows a larger number of stores to change their delivery times from historical values to the ones recommended by the algorithm.

Since the original implementation, the functionality of the system has been enhanced considerably. The algorithm has been generalized to include both deliveries and pickups. Thus, it now schedules shipments from suppliers to the DCs and from the central DC to the regional DCs as well. A complex route in the system could start from a DC, make deliveries to some stores, pick up merchandise from a supplier, make deliveries to other DCs, and finally return to the starting DC. The system has been rolled out at all DCs in the Netherlands and in other supermarket chains in Europe owned by Ahold Corporation, supporting about $16.6 billion in sales.

Through the implementation at Albert Heijn, we found that the value of planning delivery times and vehicle routes simultaneously extends beyond optimizing the logistics process. An important role of the system is conducting sensitivity analysis for many kinds of operational factors—for example, measuring the cost impact of changes in forecast accuracy, delivery frequency, truck types, length of delivery time windows, etc. It is also used in the performance analysis of stores with respect to their ordering decisions. Orders from some stores have much higher variability than others because of poor execution. This increases both the buffer space required in the trucks delivering to these stores and the frequency of route spillovers. With our system, the company is able to measure the cost impact of order variability and communicate more effectively with store managers to improve ordering discipline.

Finally, Albert Heijn benefits strategically by integrating inventory control and vehicle routing decisions. By improving its logistics, the firm is able to increase its frequency of replenishment and reduce lead time. Thus, it is able to provide more fresh products and greater variety within the same shelf space. These factors enable Albert Heijn to differentiate itself from its competitors in the supermarket industry in spite of tough price competition and branded commodity products.

Acknowledgments

The authors gratefully acknowledge Frank Jansen and Robert van-Lunteren of Ahold Corporation for their intellectual contribution and support for the project. They also thank Noel Watson for assistance in the implementation of their system.

References

Anily, S., A. Federgruen. 1990. One warehouse multiple retailer systems with vehicle routing. *Management Sci.* **36** 92–114.

Bard, J. F., L. Huang, P. Jaillet, M. Dror. 1998. A decomposition approach to the inventory routing problem with satellite facilities. *Transportation Sci.* **32** 189–203.

Barnes-Schuster, D., Y. Bassok. 1997. Direct shipping and the dynamic single-depot/multi-retailer inventory system. *Eur. J. Oper. Res.* **101** 509–518.

Bell, W. J., L. M. Dalberto, M. L. Fisher, A. J. Greenfield, R. Jaikumar, P. Kedia, Robert G. Mack, Paul J. Prutzman. 1983. Improving the distribution of industrial gases with an on-line computerized routing and scheduling optimizer. *Interfaces* **13** 4–23.

Bramel, J., D. Simchi-Levi. 1995. A location based heuristic for general routing problems. *Oper. Res.* **43** 649–660.

Bramel, J., D. Simchi-Levi. 1997. *The Logic of Logistics*. Springer Series in Operations Research, Springer, New York.

Campbell, A., L. Clarke, M. W. P. Savelsbergh. 2002. Inventory routing in practice. P. Toth, D. Vigo, eds. *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, PA, 309–330.

Chan, L. M. A., A. Federgruen, D. Simchi-Levi. 1998. Probabilistic analyses and practical algorithms for inventory-routing models. *Oper. Res.* **46** 96–106.

Chien, T. W., A. Balakrishnan, R. T. Wong. 1989. An integrated inventory allocation and vehicle routing problem. *Transportation Sci.* **23** 67–76.

Christofides, N. 1985. Vehicle routing. E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, D. B. Shmoys, eds. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Wiley, New York, 431–448.

Christofides, N., S. Eilon. 1969. An algorithm for the vehicle dispatching problems. *Oper. Res.* **20** 309–318.

Cordeau, J., M. Gendreau, G. Laporte. 1997. A tabu search heuristic for period and multi-depot vehicle routing problems. *Networks* **30** 105–119.

Dror, M., M. Ball. 1987. Inventory/routing: Reduction from an annual to a short-period problem. *Naval Res. Logist.* **34** 891–905.

Dror, M., M. Ball, B. Golden. 1985. A computational comparison of algorithms for the inventory routing problem. *Ann. Oper. Res.* **4** 3–23.

Federgruen, A., D. Simchi-Levi. 1995. Analytical analysis of vehicle routing and inventory management problems. M. O. Ball, T. L. Magnanti, C. L. Monma, G. L. Nemhauser, eds. *Network Routing. Handbooks in OR and MS*, Vol. 8. North-Holland, Amsterdam, The Netherlands, 297–373.

Fisher, M. L. 1994. Optimal solution of vehicle routing problems using minimum k-trees. *Oper. Res.* **42** 626–642.

Gabow, H. N., R. E. Tarjan. 1991. Faster scaling algorithms for general graph matching problems. *J. Assoc. Comput. Machinery* **38** 815–853.

Galil, Z., S. Micali, H. N. Gabow. 1986. An $O(EV \log V)$ algorithm for finding a maximal weighted matching in general graphs. *SIAM J. Comput.* **15** 120–130.

Gallego, G., D. Simchi-Levi. 1990. On the effectiveness of direct shipping strategy for the one-warehouse multi-retailer R-systems. *Management Sci.* **36** 240–243.

Gaur, V. 2001. Essays in retail operations management. Unpublished Ph.D. dissertation, The Wharton School, University of Pennsylvania, Philadelphia, PA.

Golden, B., A. Assad, R. Dahl. 1984. Analysis of a large scale vehicle routing problem with an inventory component. *Large Scale Systems* **7** 181–190.

Kleywegt, A., V. Nori, M. Savelsbergh. 2002. The stochastic inventory routing problem with direct deliveries. *Transportation Sci.* **36** 94–118.

Laporte, G., F. Semet. 2002. Classical heuristics for the capacitated VRP. P. Toth, D. Vigo, eds. *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, PA, 109–128.

Larson, R. 1988. Transporting sludge to the 106-mile site: An inventory/routing model for fleet size and logistics system design. *Transportation Sci.* **22** 186–198.

Lawler, E. L. 1976. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston, New York.

Lin, S., B. Kernighan. 1973. An effective heuristic algorithm for the traveling salesman problem. *Oper. Res.* **21** 498–516.

Papadimitriou, C. H., K. Steiglitz. 1982. *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, Englewood Cliffs, NJ.

Toth, P., D. Vigo. 2002. *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, PA.

Webb, I., R. Larson. 1995. Period and phase of customer replenishment: A new approach to the strategic inventory/routing problem. *Eur. J. Oper. Res.* **85** 132–148.