Chapter 13

Routing Under Uncertainty: An Application in the Scheduling of Field Service Engineers

Eleni Hadjiconstantinou
Daron Roberts

13.1 Introduction

In the classical definition of VRP, it is assumed that the associated parameters, concerning factors such as cost, customer demands, and vehicle travel times, are deterministic. This conjecture often is too simplistic in today's dynamic environment, where there exist increasing requirements on levels of productivity and service and a corresponding commitment to enlarged and more elaborate transportation systems. In parallel with the need to manage such a growing number of systems there exists an increased amount of data augmentation and volatility. Hence, when an organization does not possess enough flexibility in its labor assignments, does not possess any real-time parameter information, or cannot analyze data in an online manner, deterministic models cannot always be implemented and stochastic models need to be considered.

The *Stochastic VRP* (SVRP) differs from the VRP by the introduction of some element of variability within the system in question. Unlike its deterministic equivalent, the SVRP is ambiguously defined since it belongs to a class of a priori optimization problems (see Bertsimas, Jaillet, and Odoni [3]) for which it is impractical to consider an a posteriori approach that computes an optimal solution whenever the random variables are realized. Instead, an a priori solution attempts to obtain the best solution, over all possible problem scenarios, before the realization of any single scenario. Roberts and Hadjiconstantinou [24] evaluated the computational performance of such a solution method. The authors showed that an a priori solution for a VRP where demand is uncertain lies, on average, within 8% of the solution obtained by a reoptimization-based, a posteriori strategy.

The specific type of SVRP to be considered in this case study is the *VRP with Stochastic Service Times* (VRPSST). Roberts and Hadjiconstantinou [24] considered the factors affecting the stochastic optimum of an SVRP and concluded that given a set of fixed-recourse arrangements, route break opportunities and information disclosure patterns, a meaningful set of SVRP interpretations can be identified. Here, a new algorithm, referred to as the *Paired Tree Search Algorithm* (PTSA), is used to solve the VRPSST with variable costs of recourse. The algorithm is tested on a real-life operational problem at a utility company. The company needs to schedule its field service engineers across a range of possible maintenance jobs. Most jobs arise stochastically and have durations that are rarely pre-determined since the engineers have limited knowledge of the nature of the work required at each site. We model this stochastic scheduling and routing problem as a VRPSST and develop a solution procedure, based on the PTSA, that minimizes operating costs. Computational results for a pilot study, including an investigation into reoptimization, show significant improvements over current practice.

In sections 13.2 and 13.3, respectively, the theoretical problem is formally defined and the relevant SVRP literature is briefly reviewed. A stochastic integer formulation for the VRPSST is given in section 13.4, and the PTSA is summarized in section 13.5. In sections 13.6 and 13.7, an outline of the applied scheduling and routing problem and the key objectives proposed by management are described, and a comprehensive list of assumptions is presented. Detailed explanations of model input and output are given in sections 13.8 and 13.9, respectively, and an illustrated example is presented in section 13.10. Computational results are shown in section 13.11.

13.2 VRPSST with Variable Costs of Recourse

Let $G = (V, E)$ be a graph where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of vertices and $E = \{(v_i, v_j) : v_i, v_j \in V\}$ is a set of edges. The vertices have known and fixed locations, and every edge (v_i, v_j) has an associated nonnegative cost c_{ij} and nonnegative travel time t_{ij}. It is assumed that the graph is undirected and the matrices $C = (c_{ij})$ and $T = (t_{ij})$ satisfy the triangular inequality, i.e., (v_i, v_j) is defined only for $i < j$ and $(c_{ik} + c_{kj} \geq c_{ij}, t_{ik} + t_{kj} \geq t_{ij}$ for all $i, j, k)$. Vertex v_1 represents a depot at which a homogeneous fleet of K vehicles, each with an overall working (service and travel) time restriction of τ, is based. The remaining vertices correspond to a set of customers where each customer v_i has associated service time requirements given by discrete, independent, nonnegative random variables ξ_i with finite means μ_i and variances σ_i^2. In a first stage, a set of K vehicle routes of minimal cost are determined so that (i) each route starts and ends at the depot and (ii) each customer is visited exactly once by one vehicle. In a second stage, the first-stage routes are followed as planned but whenever τ is exceeded along a route, as a consequence of the deterministic travel times t_{ij} and the stochastic service times ξ_i, the vehicle returns to the depot and then continues along its predefined route with a replenished time allowance of τ. Because second-stage recourse costs are represented by the values of such return trips to the depot, the objective is to design a minimum expected cost set of routes such that all service time requirements are met, (i) and (ii) are satisfied, and exactly K vehicles are used.

13.3 Literature Review

The SVRP, in all its guises, has seen relatively little research in comparison with its well-known deterministic counterpart. Given the number of potential applications, this lack of research is due to the enormous complexity that the addition of a stochastic element brings to an already difficult combinatorial optimization problem. To our knowledge, the VRPSST as defined in section 13.2 has never been formulated or solved in the literature. However, the problem has very close links to two other stochastic routing problems, the *VRP with Stochastic Travel Times* (VRPST) and *VRP with Stochastic Demands* (VRPSD). Here, we briefly review both problems.

13.3.1 VRPST

The objective of the VRPST—and its one vehicle counterpart, the *Traveling Salesman Problem with Stochastic Travel Times* (TSPST)—usually involves finding an a priori solution such that the probability of completing any tour within a given deadline is maximized. In cases such as these, the VRPST is interchangeable with the multiple vehicle TSPST (m-TSPST).

Kao [15] proposed two heuristics for the TSPST, one based on dynamic programming and one based on implicit enumeration. Sniedovich [25] showed that obtaining optimal solutions using the former dynamic programming approach is reliant on the property of monotonicity, and Carraway, Morin, and Moskowitz [4] presented a generalized dynamic programming method that overcomes this problem. Lambert, Laporte, and Louveaux [16] derived a heuristic solution algorithm for the m-TSPST, based on the well-known Clarke and Wright [5] savings procedure, and found cost-effective cash-collection routes through bank branches where the amount of cash collected is limited by an insurance company and late arrival incurs a penalty relating to lost interest. Laporte, Louveaux, and Mercure [20] were the first to consider an alternative objective for the VRPST. They presented a three-index simple recourse model and a two-index recourse model for a VRPST based on finding a minimum cost a priori solution where the penalty for late arrival is proportional to the length of the delay. Using an integer L-shaped method (see Laporte and Louveaux [18]), they presented exact results for VRPSTs of up to 20 customers.

13.3.2 VRPSD

The few studies that have been completed on the VRPSD focused on heuristic methods. Tillman [29] developed an adapted Clarke and Wright [5] savings algorithm to account for stochastic demands, Teodorović and Pavkovic [28] presented a simulated annealing heuristic, Gendreau, Laporte, and Séguin [11] described a tabu search algorithm, and Teodorović, Krcmar-Nozić, and Pavković [27] presented a route-first, cluster-second approach. Golden and Stewart [14] were the first to apply stochastic programming to the VRPSD, and Stewart and Golden [26] presented formulations for the chance constrained case, where customers are served according to a given probability, and the penalty function case, where each customer is served with the inclusion of a possible recourse cost. Further stochastic programming formulations have been developed—see Dror and Trudeau [10], Dror, Laporte, and Trudeau [9], Laporte and Louveaux [17], Bastian and Kan [1], Dror [7], Dror, Laporte, and Louveaux [8], and Popović [21]—but these yielded no exact solutions apart from a special location-routing model presented by Laporte, Louveaux, and Mercure [19] and a special case of the probabilistic VRP with stochastic demands and deterministic customer presence by Gendreau, Laporte, and Séguin [12]. In the former paper, problems are solved to optimality for $N = |30|$, where N represents both the number of customers and possible depot sites. In the latter paper, exact solutions are given for problems of up to 70 customers; however, in such cases, the parameters are set such that the problem is in essence deterministic. More recently, Roberts and Hadjiconstantinou [23] presented a new method, on which the algorithm used in this study is based, that can successfully solve computationally difficult VRPSDs of medium size. For more information, see a detailed review by Roberts [22] and a survey of the generic SVRP, including the VRPSD, by Gendreau, Laporte, and Séguin [13].

13.4 Stochastic Integer VRPSST Formulation

Given a feasible set of routes represented by $x = [x_{ij}]$ and a set of service times arising from the random variables ξ_i for all $i = 2, \dots, n$, the VRPSST can be represented by the following two-stage stochastic program with recourse:

$$\min_x \; [f_0(x) + E_{\varepsilon\in\xi}(Q(x,\varepsilon))], \tag{13.1}$$

where $\min f_0(x)$ is the objective function of the first-stage problem and $\min E_{\varepsilon\in\xi}(Q(x,\varepsilon))$ is the objective function of the second-stage problem, i.e., $Q(x,\varepsilon)$ is the cost of recourse given that $x = [x_{ij}]$ is the first stage solution. These two stages are further defined below.

13.4.1 First-Stage Problem

The first-stage problem, $\min f_0(x)$, corresponds directly to the solution of a multiple TSP, or m-TSP, where a complete feasible set of routes is required to minimize cx. With c_{ij} and x_{ij} interpreted as c_{ji} and x_{ji} whenever $i > j$, we define integer decision variables x_{ij} as follows:

$$x_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \text{ is used in the solution and } 2 \le i < j \le n, \\ 2 & \text{if } (v_i, v_j) \text{ is used as a return trip and } i = 1,\ j > 1, \\ 0 & \text{otherwise.} \end{cases} \tag{13.2}$$

A feasible set of routes is then obtained by solving the following:

$$\min_x \; f_0(x) = cx \tag{13.3}$$

subject to

$$\sum_{j=2}^{n} x_{1j} = 2K, \tag{13.4}$$

$$\sum_{i<k} x_{ik} + \sum_{j>k} x_{kj} = 2 \qquad \forall\, v_k \in V\backslash\{v_1\}, \tag{13.5}$$

$$\sum_{i,j\in S} x_{ij} \le |S| - 1 \qquad \forall\, S \subset V \setminus \{v_1\},\ 3 \le |S| \le n-2, \tag{13.6}$$

$$0 \le x_{1j} \le 2 \qquad \forall\, v_j \in V \setminus \{v_1\}, \tag{13.7}$$

$$0 \le x_{ij} \le 1 \qquad \forall\, v_i, v_j \in V, 2 \le i < j \le n, \tag{13.8}$$

$$x_{ij} \text{ integer} \qquad \forall\, v_i, v_j \in V, 1 \le i < j \le n. \tag{13.9}$$

These first-stage deterministic constraints specify that K vehicles enter and leave the depot (13.4), that every customer receives a visit exactly once, i.e., the vertex degree constraints (13.5), and that individual routes disconnected from the depot are prohibited, i.e., the classical connectivity constraints (13.6).

13.4.2 Second-Stage Problem

The second-stage problem, min $E_{\varepsilon\in\xi}(Q(x,\varepsilon))$ in (13.1), is less well defined; however, some clarification can be sought by introducing a recursive stochastic formulation based on each a priori first-stage solution.

Consider a first-stage feasible solution characterized by the vector $x^v = [x_{ij}^v]$. Let $W(x^v) = E_{\varepsilon\in\xi}(W(x^v,\varepsilon))$ denote the expected second-stage costs and let $W^k(x^v,\varepsilon)$ denote the recourse cost of route k in x^v given the realization ε of the random variable ξ. The expected cost of K vehicle routes given a current feasible solution x^v is then simply $W(x^v) = \sum_{k=1}^{K} W^k(x^v)$, where the expected cost of any route k can be computed separately, i.e., $W^k(x^v) = E_{\varepsilon\in\xi}(W^k(x^v,\varepsilon))$.

Let p_i^l represent the probability that the lth service time ε_i^l originates from the set of realizations $\{\varepsilon_i^1,\dots,\varepsilon_i^l,\dots,\varepsilon_i^{\delta_i}\}$ of customer v_i such that $\varepsilon_i^l < \varepsilon_i^k$ for all $l < k$. It is assumed that each customer requires a service and that the maximum time requirement of any customer is always equal to or less than the overall vehicle working time restriction, i.e., $\varepsilon_i^1 > 0$ and $\varepsilon_i^{\delta_i} \le \tau$ for all $i > 1$. In addition, relabel the vertices of the kth route of x^v so that the route becomes $(v_1, v_2, \dots, v_{t_k}, v_{t_k+1} = v_1)$. By denoting g to be the remaining working time available for a vehicle on arrival at a customer v_i, the expected cost of a route k is then as follows:

$$W^k(x^v) = E_{\varepsilon\in\xi}(W^k(x^v,\varepsilon)) = \alpha_2^k(\tau),$$

where

$$\alpha_{t_k}^k(g) = 2c_{1t_k} \sum_{l|\varepsilon_{t_k}^l > g} p_{t_k}^l, \qquad 0 < g \le \tau, \text{ and} \tag{13.10}$$

$$\begin{aligned}\alpha_i^k(g) = {} & p_i^{l^*}\alpha_{i+1}^k(0) + \sum_{l|\varepsilon_i^l > g} p_i^l(\alpha_{i+1}^k(\tau - \varepsilon_i^l - t_{1i} - t_{i,i+1}) + 2c_{1i}) \\ & + \sum_{l|\varepsilon_i^l < g} p_i^l \alpha_{i+1}^k(g - \varepsilon_i^l - t_{i,i+1}) \qquad \forall\, i = 2,\dots,t_k-1,\ 0 < g \le \tau,\end{aligned} \tag{13.11}$$

and

$$p_i^{l^*} = \begin{cases} p_i^l & \text{if there exists } l = 1,\dots,\delta_i \text{ such that } \varepsilon_i^l = g, \\ 0 & \text{otherwise.}\end{cases} \tag{13.12}$$

The proof of (13.10)–(13.12) is similar to that shown by Bertsimas [2] for the VRPSD, and it follows directly from the definition of $\alpha_i^k(g)$, which represents the expected recourse cost from vertex v_i of route k given that the remaining working time available for any vehicle before entering vertex v_i is g. The three terms in (13.11) are explained as follows:

- The working time of a vehicle has been exhausted (to zero) at a given customer, but no failure occurs at this customer because of the presence of late information. The vehicle will then arrive at the next present customer along its route with zero available working time.

- Whenever the remaining working time available for a vehicle on arrival at a customer becomes exceeded, then a route failure has occurred and a trip back to the depot is necessary.

- If the service time requirement of a given customer does not exceed the remaining working time available for the vehicle on arrival at this customer, then no route failure occurs and the vehicle continues along its route.

The resulting stochastic programming model is a highly complex composite program of two parts. Initially, a first-stage integer program, (13.3)–(13.9), needs to be solved to find a feasible solution structure (one of a number of feasible sets of routes) that can be implemented into the next stage of the solution method. Then, a recursive stochastic recourse formulation must be utilized to find the cost of the penalty function for the given first-stage solution and so enabling the derivation of a solution value for the entire VRPSST (13.1). Clearly, finding an optimal solution to such a complex problem in reasonable time is a difficult task. In the following section, we outline an algorithm that can be used to obtain optimal VRPSST solutions in reasonable time (while retaining a suitable limit on computer memory requirements) by providing an adequate structure for the implementation of a series of lower bounds for both the first-stage and second-stage problems.

13.5 Paired Tree Search Algorithm (PTSA)

The PTSA was developed by Roberts and Hadjiconstantinou [23] to obtain optimal solutions to the VRPSST and has its foundations in a *Stochastic Decision Tree* (SDT) approach. (For further details of this method, see Roberts [22].) An SDT is a tree that branches for each possible decision and each possible realization of the stochastic variables involved. The tree is then built up of a series of decision nodes and chance nodes where the outcome of one possible instance of the problem is obtained at each leaf of the tree. For the VRPSST, decision nodes represent the choice of an arc on a graph contributing to a vehicle route, and chance nodes correspond to the independent events generated after the customer service times have been realized. The PTSA further modifies the SDT method in the following two ways:

- Due to the possibility of recourse, a VRPSST solution refers to a set of planned routes that may not be completed in practice. To represent the first-stage problem, therefore, a search tree is linked to a SDT, i.e., a group of SDT nodes index a single node on a separate tree.

- Unless events are properly limited, the branching from SDT chance nodes can lead to dimensionality problems. In the PTSA, therefore, SDT branching occurs according to the residual working time a vehicle can have after satisfying the service time requirements of the customer in question, i.e., events equate to alternative *service time-leaving* levels. In addition, an aggregation process is established where the service time-leaving level of each chance node contributes to new nodes formed in a *rebranching* procedure. Each of the rebranched nodes then corresponds to a unique service time-leaving level, thereby limiting the discrete number of nodes retained on each level of the SDT to τ.

13.5.1 Linked Trees

The PTSA is implemented based on the use of two linked trees. An example is shown in Figure 13.1. One, a binary search decision tree (OUTER tree), relates to the first-stage deterministic problem, and the other, an SDT-based tree (INNER tree), relates to the second-stage stochastic recourse problem. The OUTER tree conforms exactly to a simple branch-and-bound method. Every branch corresponding to a possible routing segment in the VRPSST divides the feasible solution subset into two independent sets: one that refers to a customer v_i and another that refers to $\overline{v_i}$. For the example in Figure 13.1, no route constructed below node 6 can include the arc (v_2, v_3); however, it must include the arcs (v_1, v_2) and (v_2, v_4). In the INNER tree, decision nodes branch from chance nodes according to the service time-leaving level of the next customer following from the service time requirements of the given customer and the service time-leaving level of the preceding customer. Each decision node has an independent probability of occurrence in comparison with other nodes having the same parent chance node. Each node in the OUTER tree indexes at least one decision node on the INNER tree. Such pointers are shown as dotted lines in the diagram. For example, INNER tree nodes 3, 4, 8, and 9 are assigned the same OUTER tree index, i.e., $O(j) = 3$ for $j = 3, 4, 8, 9$. The algorithm adopts a nested branching scheme, and lower bounds, corresponding to both stages of the formulation, are embedded on each tree to limit the search before an optimal solution to the VRPSST can be found.

13.5.2 Lower Bounds

A lower bound can be computed at each OUTER tree node as follows. Solving the first-stage problem (13.3)–(13.9) is equivalent to finding feasible solutions of the m-TSP. A 2-perfect matching-based lower bound of the first stage problem, L^1, can be generated by relaxing the subtour connectivity constraints (13.6) and the vehicle number constraints (13.4). Moreover, with the simple addition of K artificial depots with infinite interconnecting travel costs, the necessary K vehicle routes can be obtained.

A second-stage lower bound, L^2, can be obtained by considering the recourse problem at each binary search tree node ρ in which a set of customers S previously has been served. Let $w(\rho)$ denote the customer index associated with node ρ. Consider the minimum total service time to be satisfied via return trips to the depot at ρ. Such a quantity of time depends on the service-time distributions of the remaining customers, the combined total time restriction of the remaining vehicles, and the minimum travel time required to cover the remaining customer locations. If the set of customers still needing a service at node ρ is $S' = V \backslash (S \cap \{v_1\})$, the number of vehicles available is K', the minimum travel time required to visit the customers in S' is P' (a lower bound of which can be obtained using a 2-perfect

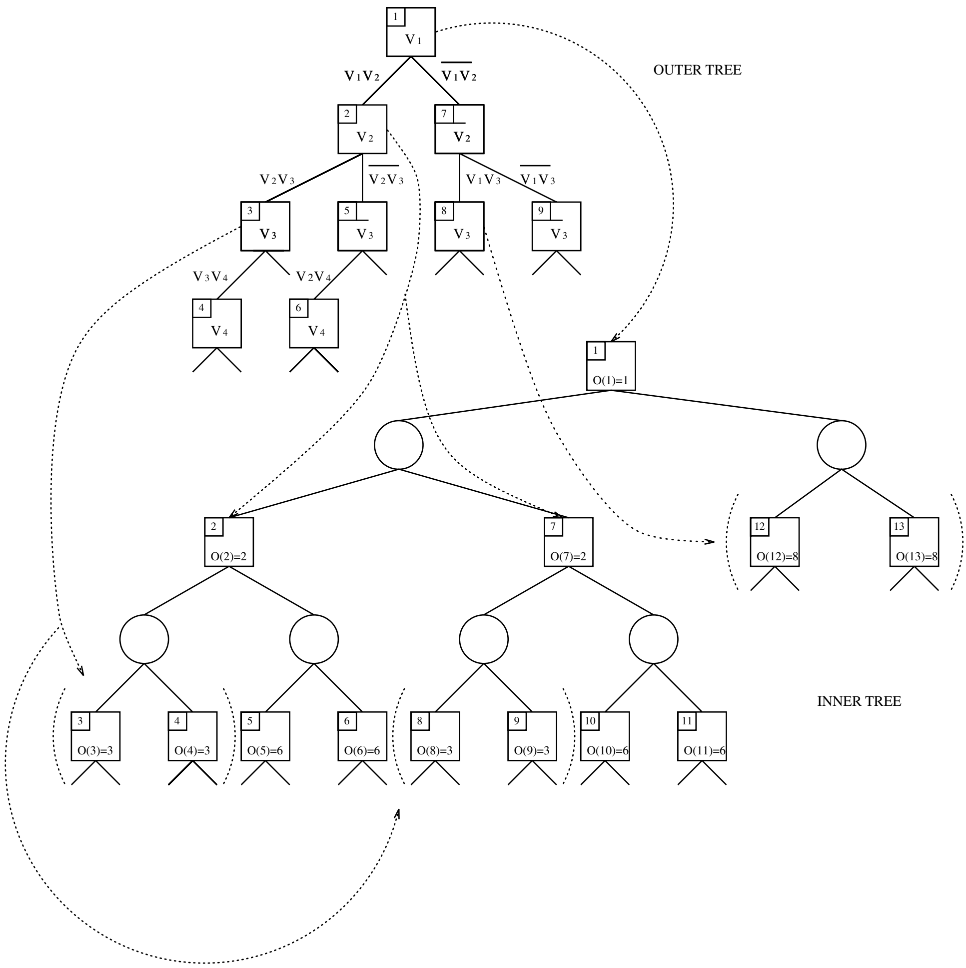

Figure 13.1. *Linked trees in the PTSA.*

matching approach), and each service time set, $\{\varepsilon_i^1, \dots, \varepsilon_i^{\delta_i}\}$, is ordered in ascending size, then the minimum remaining working time $g(\rho)$ to be satisfied through recourse is given by

$$g(\rho) = \begin{cases} P' + \sum_{v_k \in S'} (\varepsilon_k^1) - \tau(K'+1) + 1 & \text{if } w(\rho) \neq 1, \\ P' + \sum_{v_k \in S'} (\varepsilon_k^1) - \tau(K'+1) & \text{if } w(\rho) = 1. \end{cases} \tag{13.13}$$

The proof of (13.13) is straightforward and can be given as follows. Two cases are examined:

1. If the current vehicle is situated at the depot, i.e., $w(\rho) = 1$, then its overall working time restriction is given by τ. In addition, there are K' identical vehicles, so the total working time of the vehicles remaining to serve the customers in S' is given by $\tau + \tau K' = \tau(K'+1)$.

2. If the current vehicle is situated at a customer, i.e., $w(\rho) \neq 1$, then its overall working time restriction is less than τ; the maximum working time now available is $\tau - 1$. Since, in addition, there are K' vehicles (each with a working time limit τ) remaining to serve the customers in S', the total working time available is given by $\tau + \tau K' - 1 = \tau(K' + 1) - 1$. Hence, the minimum remaining working time $g(p)$ to be satisfied through recourse is given by (13.13).

If $g(\rho)$ is greater than zero, then a route failure will definitely occur irrespective of how the remaining customers are routed. Indeed, the minimum number of route failures, $f(\rho)$, that must occur while serving the remaining customers is given by $f(\rho) = \lceil g(\rho)/\tau \rceil$, where $\lceil * \rceil$ represents the smallest integer not less than $*$. Now, given that c_0 represents the remaining least-cost single trip to the depot, i.e., $c_0 = \min_{v_k}[c_{1k} \mid v_k \in S']$, the lower bound L^2 is given by

$$L^2(\rho) = \begin{cases} 2c_0 f(\rho) + z_2^\lambda & \text{if } g(\rho) > 0, \\ z_2^\lambda & \text{otherwise,} \end{cases} \tag{13.14}$$

where z_2^λ is the total recourse cost for all OUTER tree nodes on the leaf from the root node of the tree up to and including node λ, the parent node of ρ.

13.5.3 Computational Implementation

In this section, a brief description of the complete algorithm, including the lower bounds, is presented. Let z^* denote a simple upper bound obtained at the root node of the OUTER tree in a heuristic fashion. At each OUTER tree decision node ρ representing customer $w(\rho)$ with a parent node λ the current partial route k is extended using arc $(v_{w(\lambda)}, v_{w(\rho)})$. If an infeasible first-stage solution to a corresponding 2-perfect matching problem is found, then backtracking occurs; otherwise, the search continues and lower bounds, L^1 and L^2, on the first- and second-stage problems, respectively, are computed on the OUTER tree. If $(L^1 + L^2)$ is greater than the best incumbent feasible VRPSST solution value, z^*, then node ρ is fathomed; otherwise, the search is transferred to the INNER tree. The set of INNER tree nodes, Λ^λ, that previously was developed and used to index λ are located and full branching occurs on the INNER tree from all nodes in Λ^λ to generate a series of new nodes linked to ρ, Λ^ρ, with unique load-leaving levels. The recourse cost z_2^ρ is updated accordingly.

Once branching is completed, the search is transferred to the OUTER tree. The current solution value is computed by $z^\rho = z_1^\rho + z_2^\rho$, where $z_1^\rho = L^1$. If a feasible first-stage set of nodes is found at node ρ and $z^\rho < z^*$, then z^* is updated accordingly by $z^* = z^\rho$.

13.6 Applied Maintenance Scheduling Problem

The PTSA was used to solve a real-life operational problem at a utility company, which has been modeled as a VRPSST. The company has a large number of major assets, including depots, work sites, buildings and machinery, and employs Field Service Engineers (FSEs) to maintain all these assets. FSEs are home-based and work independently in a set geographical region. An average day for an FSE involves 8 hours and 15 minutes of work, and overtime is paid for work completed over this allotted time. Typically, a number of jobs (usually fewer than 10) are completed per day at a number of alternative sites (usually fewer than 5). Accordingly, an FSE may complete up to 30 jobs per week at up to 20 locations.

13.6.1 Maintenance Scheduling System in Practice

Jobs are assigned to an FSE in a variety of ways; however, each job has a basic form of prioritization and, for all but the most reactive jobs, requires some form of localized scheduling and routing. Currently, before deciding which jobs to complete each day, an FSE considers a variety of factors, including

- *priority*, which is a known upper limit of time before which a job must be completed,
- *site location*, which is known and fixed,
- *travel time*, which is estimated based on FSE knowledge of the geographical area and local traffic systems and so forth, and
- *service time*, which is the length of time taken to complete a job and which is estimated according to incomplete knowledge and FSE experience.

In this study, FSE jobs are defined according to their associated priority and belong to one of the following three categories: (i) reactive (R)—emergency call-outs with a priority given in terms of hours; (ii) preplanned (P)—regular jobs with a priority given in terms of months; and (iii) unplanned (U)—irregular jobs that require some form of local prioritization usually given in terms of days or weeks. The characteristics of these *job types*, obtained from a database storing information for 8 months of FSE work, are shown in Table 13.1. Specifically, column 2 shows the proportion of jobs that were classified under a particular job type during this period (Total Number), column 3 displays the proportion of total time spent completing jobs of a particular job type (Total Time), column 4 highlights the average duration of time taken to complete individual jobs of a particular job type (Average Service Time), and column 5 displays approximations of the upper limits of priority per job type that accord with FSE efficiency targets.

13.6.2 Stochastic Problem Setting

In any stochastic environment, there exists a specific information state that refers to the amount of information available to the decision makers at the time of decision making as opposed to the time when full information becomes available. In this study, the decision makers are the engineers, decision making refers to local scheduling and routing, and full information occurs with hindsight after a job is completed. The presence of a stochastic

Table 13.1. *Job characteristics for an engineer.*

Job type	Total number	Total time	Average service time	Priority
Preplanned	59%	23%	1 hour	3 months
Unplanned	11%	19%	5 hours	2 weeks
Reactive	30%	58%	6 hours	1 hour

information state is highlighted by the facts that (i) service times are deemed stochastic as opposed to fixed, and (ii) there exist inconsistencies in the way jobs are reported, e.g., a qualitative study with FSEs revealed an estimated ratio of 20:70:10 in the Total Number of P, U, and R jobs in contrast to the actual ratio of 59:11:30 (see Table 13.1). In practice, there also exist a variety of reoptimization methods, i.e., operational systems, that can be utilized in such a problem environment.

Table 13.1 shows that more than 50% of an FSE's work time is spent doing reactive jobs. Such jobs, however, total only 30% of all jobs completed. Indeed, as P and U jobs are large in number and have shorter durations which are stochastic in nature they are seen as schedulable. Conversely, reactive jobs are seen as uncontrollable and reducible only by improved engineering techniques and preventative maintenance, i.e., an increased number of planned maintenance jobs should decrease the overall number of emergency cases.

To summarize, two uncertainties are present in the FSE scheduling and routing system. First, certain maintenance jobs completed by FSEs can arise in a probabilistic manner, and, second, the time required to complete individual jobs is unknown, i.e., the occurrence of FSE jobs can be stochastic and the completion times of FSE jobs is stochastic. Consequently, the problem of determining optimal schedules is very complex. To simplify the approach, consider a finite period of time within which a series of P, U, and R jobs have to be completed by an FSE. (Note that the geographical boundary of such jobs will be specified by the site locations themselves.) The basic routing and scheduling problem can then be described as follows: If an FSE has a list of U and P jobs to complete within a finite planning horizon (e.g., a 5-day working week), how should those jobs be scheduled to minimize overall cost, taking into account both reactive call-outs (R jobs) and the uncertain nature of job service times?

13.7 Modeling the Applied Problem as a VRPSST

The authors were asked to examine the possible restructuring and refinement of the existing FSE scheduling and routing system with a view to reducing costs or improving productivity and the level of service associated with maintenance operations at the utility company. These issues were addressed by developing a VRPSST-based optimization model of the basic FSE scheduling and routing problem given above and validating the model using historical information. More specifically, the optimization model can be used to identify optimal schedules of P and U jobs for a given FSE and, therefore, can be used to recommend the most efficient daily routes, taking into account stochastic service times (and reactive call-outs). The relative performance of the model then can be evaluated by analyzing existing FSE schedules; i.e., model output can be used to predict schedules based on historical information and a comparison can then be made between results obtained manually and results that could have been obtained with the use of the model. Finally, it is possible to investigate the impact of using the model at two different stages of implementation. Such analysis provides a measure of the comparative efficiency of the current manual system against that of the model at two stages of practical use primarily concerned with reactive call-out recognition.

To model the FSE problem as a VRPSST, each "vehicle" corresponds to a "day" in a given planning horizon. The time restriction, τ, then equates to the normal hours of each working day. In addition, VRPSST customers correspond to jobs that require a service, and, as before, each job has an assigned geographical location, each route starts and ends at a fixed point (the depot), and each job is serviced on one day only, i.e., each customer is visited exactly once by one vehicle. The time matrix represents the travel times between customer sites, and the cost matrix may represent travel time, travel cost, or travel distance between customer sites. The objective of the VRPSST described in this context is then to design a minimum expected cost set of routes given that recourse costs (represented by return trips back to the depot) are interpreted as the cost incurred to return to a location to complete a particular job on a day outside the planning horizon.

13.8 Model Input

The implementation of the optimization model requires the availability of input data in the format required by the model. The primary operations involved in scheduling an FSE include the prioritization of jobs to be scheduled, the classification of individual job times from a host of job characteristics, and the inclusion of geographical site locations. The two main inputs required are discussed below.

13.8.1 Job Locations and the Road Network

The results presented in this case study are based on real distances calculated between any two site locations using a real Road Network System (RNS). This system covers the pilot study region of the utility company and, for this reason, 10 figure OS references were obtained for all pilot study–based sites and FSE home locations. The road network contains more than 4000 road segments (arcs) and about 1600 intersections of road segments (nodes identified by grid references).

The RNS stores not only the length of the links between road junctions but also the type of road that makes up a link. With such information, it becomes possible to adjust for differing vehicle speeds on different types of road, e.g., motorway (60 mph), A road (40 mph), and B road (20 mph), and hence to calculate accurate vehicle travel times between any two locations. Note that the route that gives the minimum vehicle travel time between any two sites may well be different from the minimum mileage route between the same two sites. In this study, vehicle routes are planned on the basis of a minimum vehicle travel cost between sites, which represents an equally weighted combination of both factors. Shortest routes between any two site locations on the road network and associated path information are determined using a shortest path algorithm; see Dijkstra [6].

13.8.2 Service Times

A standard mathematical distribution is required to describe the service time of a particular job and act as input into the VRPSST maintenance model. For our purposes, an FSE job is not defined by precise engineering detail but by what an FSE predicts a particular job to entail since, for all but the most trivial of jobs, the precise specifics of a job are unknown until the problem is diagnosed on site. By examining service time distributions for the same job types using data for individual FSEs, log-normal distributions with differing means and standard deviations were found to fit with adequate statistical confidence. Therefore, when

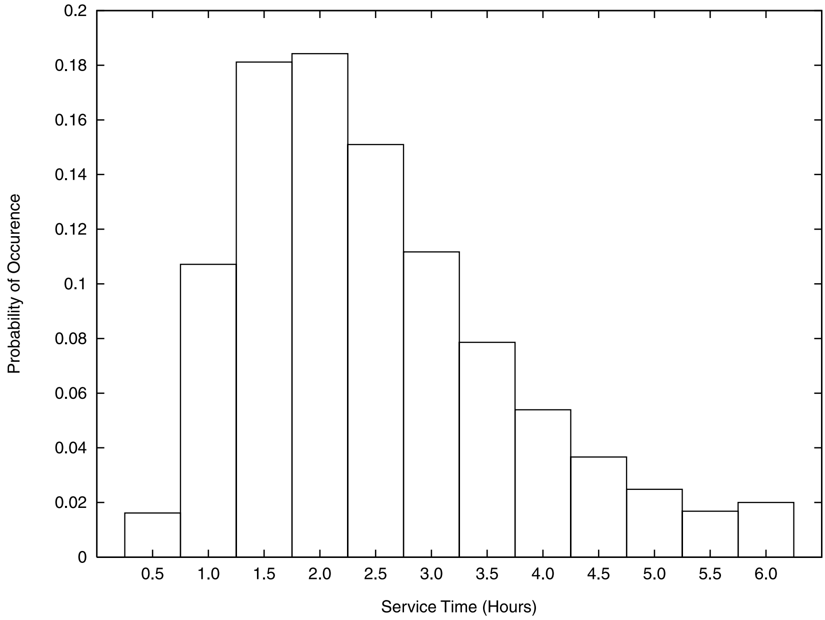

Figure 13.2. *Example FSE service time distribution input.*

a service time distribution is required, the FSE's mean service time, and standard deviation, for a given job type contribute to a discretized log-normal distribution that can be entered into the model. An example of such an input distribution, where there exist 12 discrete service time possibilities, is shown in Figure 13.2. Notice that for modeling purposes, a limit of 6 hours is maintained, i.e., the probability that a time above 6 hours is realized contributes to a summed discrete probability of occurrence corresponding to exactly 6 hours.

13.9 Model Output: Computational Considerations

The model was coded in FORTRAN and run on a Silicon Graphics Workstation Indigo R4000 (100MHz). The evaluation of the computational performance of the model is based on 12 scenarios (each scenario corresponding to one week's data) from one month's historical data for three FSEs based in the pilot study region.

13.9.1 Framework for the Analysis of Results

The actual input for each scenario, obtained from historical information, displays when and where each job was completed, what its priority was, and how long each job took to be completed (in hours) for a given FSE's working week. The following information can be obtained from such a weekly input: (i) a list of all P and U jobs to be scheduled on Monday morning, (ii) the actual service time per day (with and without reactive call-outs), (iii) the actual travel time per day (with and without reactive call-outs), (iv) the actual distance traveled per day (with and without reactive call-outs), and (v) the actual overtime per day (with and without reactive call-outs), i.e., the time beyond 8 hours and 15 minutes.

The manual sequence of jobs completed (scheduled) in practice by a given FSE for each day of the week is referred to as the *manual schedule*. The corresponding optimal sequence of P and U jobs, obtained by the optimization model at the beginning of the planning week, is referred to as the *optimal schedule*. Days in the optimal schedule are not ordered over the planning period, and, for that reason, a secondary ordering process is established based on the number of jobs per day and the expected service time per day.

When the actual job completion times obtained from the manual data for a particular scenario are entered into the optimal schedule, then the latter becomes the *actual-optimal schedule*, which can be compared to the existing manual schedule. The following assumptions have been used in developing actual-optimal schedules based on the optimal model output:

- All P and U jobs completed in the manual schedule are available for scheduling at the beginning of that week.

- There exists a fixed limit of overtime allowed per day within the actual-optimal schedule that corresponds to the average amount of overtime used in the historical data, i.e., a job in an actual-optimal schedule will create overtime only when the total time used in that day, plus the expected time of the new job, is less than the working day plus the fixed amount of overtime allowed on average per day in the manual schedule.

- If an FSE does not have time to complete a P or U job, even allowing for overtime, then the job will be completed at the end of any permitted subsequent daily schedule. Conversely, if an FSE has some spare time at the end of a day, then the last scheduled job in the week will be completed.

- If a reactive call-out occurs in the historical data, then the reactive job is completed in the actual-optimal schedule and the current P or U job is abandoned to be completed later in the week (see section 13.9.2).

By finding the optimal schedule at the beginning of each scenario and using the above framework to compare the manual schedule with the actual-optimal schedule, it is possible to find estimates for the reduction in total travel time, distance traveled, and overtime that can be obtained by using the optimization model instead of the manual method. In addition, by investigating the impact of implementing the model in two stages, described below, it is possible to consider the issue of reoptimization and the cost of going online.

13.9.2 Reoptimization

The model initially schedules P and U jobs allowing for stochastic service times. Reactive call-outs are not included as they do not exist at the time of scheduling. Indeed, one possible manifestation of the model is to exclude the contribution of reactive call-outs entirely and to consider only jobs that can be scheduled; this would correspond to an FSE system in which reactive call-outs are never encountered and would therefore partially invalidate any associated results. Two different levels of inclusion of reactive call-outs, which correspond to two different stages of model implementation, are therefore included:

- *Simple inclusion*. Jobs occur as scheduled; however, when reactive call-outs arise they are implemented just as they occur in reality, and when they end the original schedule is continued. No secondary optimization is completed once the original schedule has been interrupted and so the method is similar to the manual system of dealing with reactive call-outs. (This system corresponds to one run of the optimization model at the beginning of the planning horizon.)

- *Reoptimization*. Jobs occur as scheduled; however, whenever reactive call-outs arise, and have been completed, reoptimization occurs. (This system corresponds to the running of the model within an online system by, for example, continually rerunning the model following the completion of a day that has included a reactive call-out.)

13.10 Example Scenario

Table 13.2 displays data for a week of FSE work in the pilot study region. Five reactive jobs occur during the week: one on Monday morning, one in the middle of the day on Monday,

Table 13.2. *Actual data for the example scenario.*

Day	Job type	Service time (hrs)	Index
Monday	R	1.50	-
	U	2.25	3
	R	2.50	-
	U	2.00	4
Tuesday	R	3.25	-
	U	1.50	5
Wednesday	U	5.00	2
	U	3.25	6
Thursday	R	4.00	-
	U	1.50	8
	P	0.25	7a
	P	0.25	7b
	U	2.25	12
Friday	P	0.50	11a
	P	0.25	9a
	U	2.25	10
	P	0.25	11b
	P	0.25	11c
	P	0.25	9b
	P	0.50	9c
	U	2.00	13
	U	1.00	14
	R	1.00	-

Table 13.3. *Input data for the example scenario.*

Job type	Index (*combined jobs)	Probability Distribution μ_i (hrs)	σ_i (hrs)	δ_i
U	2	3.04	1.80	34
U	3	3.04	1.80	34
U	4	3.04	1.80	34
U	5	3.04	1.80	34
U	6	3.04	1.80	34
P	7*	0.61	0.46	17
U	8	3.04	1.80	34
P	9*	0.92	0.69	24
U	10	3.04	1.80	34
P	11*	0.92	0.69	24
U	12	3.04	1.80	34
U	13	3.04	1.80	34
U	14	3.04	1.80	34

one on Tuesday, one on Thursday morning, and one on Friday afternoon. Table 13.3 displays the inputs to the VRPSST model. The index column is simply used to identify inputted jobs and allows for a combination of P jobs when total expected service time is small. For example, indices 7a and 7b are used to identify two small jobs that are combined to generate a single service time distribution that should be input to the model and represented as job 7. The expected service times (μ_i), the standard deviations (σ_i), and the number of discrete service time points (δ_i) describing each log-normal distribution of a given job are also shown.

When no reactive call-outs are considered, the output of the optimization model is the optimal schedule shown in Table 13.4. This list would have been used to schedule the FSE in an implemented system. Notice that there is a 5.1% difference between the expected service time and the actual service time (ST) over the whole week. Using the list of assumptions given in section 13.9.1, this optimal schedule can now be used to obtain the actual-optimal schedule shown in Table 13.5. The corresponding manual schedule is also shown in Table 13.5. Notice that the actual-optimal schedule differs from the optimal schedule because, since extra time was available on Thursday, jobs 5 and 11 could be added to Thursday's schedule. Contrasting the actual-optimal schedule with the manual schedule, the following can be noted: (i) the 5-day original schedule becomes a 4-day schedule in the optimized case, (ii) the spread of jobs is more even in the optimized schedule and, hence, overtime (OT) is cut from 3.17 hours to a total of 0.08 hours (a reduction of 97.5%),

Table 13.4. *The optimal schedule for the example scenario.*

	Sequence of jobs	Expected service time (Hours)
M	H-7-9-10-8-H	6.40
T	H-4-14-6-H	7.30
W	H-2-13-H	4.87
T	H-12-3–H	4.87
F	H-11-5-H	3.35
		26.79

Table 13.5. *Schedules for the example scenario with no reactive call-outs.*

			ST (hrs)	TT (hrs)	OT (hrs)	TD (km)
Manual schedule	M	H-3-4-H	4.25	1.33	0.00	80
	T	H-5-H	1.50	0.67	0.00	40
	W	H-2-6-H	8.25	1.00	1.00	80
	T	H-8-7a-7b-12-H	4.25	1.00	0.00	60
	F	H-11a-9a-10-11bc-9bc-13-14-H	7.25	3.17	2.17	200
			25.50	7.17	3.17	460
Actual-optimal schedule	M	H-7-9-10-8-H	5.25	1.33	0.00	110
	T	H-4-14-6-H	6.25	1.33	0.00	80
	W	H-2-13-H	7.00	0.83	0.00	60
	T	H-12-3-11-5-H	7.00	1.33	0.08	70
			25.50	4.83	0.08	320

(iii) travel time (TT) decreases dramatically in the optimized schedule from 7.17 hours to 4.83 hours (a reduction of 32.6%), and (iv) travel distance (TD) decreases in the optimized schedule from 460 to 320 kilometers (a reduction of 30.4%).

Employing simple inclusion in the example scenario results in a 5.9% reduction in travel time, an 11% reduction in overtime, and a 5% reduction in distance traveled; profiles of the use of FSE time in this case, for the manual and actual-optimal schedules, are shown in Figure 13.3. Table 13.6 displays both schedules if reoptimization is employed in the

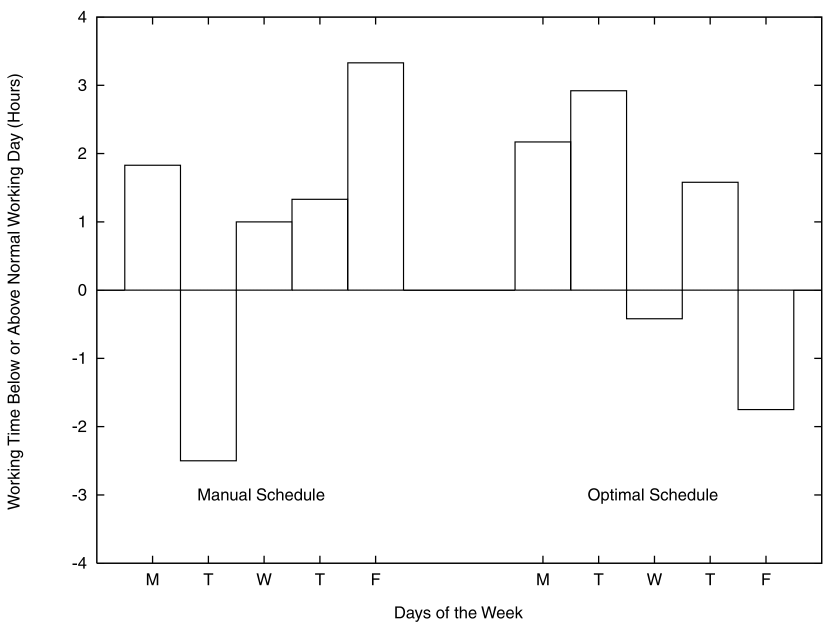

Figure 13.3. *A time profile of manual and actual-optimal schedules.*

Table 13.6. *Schedules for the example scenario with reoptimization.*

			ST (hrs)	TT (hrs)	OT (hrs)	TD (km)
Manual schedule	M	H-R-3-R-4-H	8.25	1.83	1.83	120
	T	H-R-5-H	4.75	1.00	0.00	70
	W	H-2-6-H	8.25	1.00	1.00	80
	T	H-R-8-7a-7b-12-H	8.25	1.33	1.33	70
	F	H-11a-9a-10-11bc-9bc-13-14-R-H	8.25	3.33	3.33	200
			37.75	8.50	7.50	540
Actual-optimal schedule	M	H-R-7-9-R-10-H	7.75	2.67	2.17	190
	T	H-R-4-14-6-H	9.50	1.67	2.92	100
	W	H-3-5-11-13-H	6.75	1.33	0.00	80
	T	H-R-2-H	9.00	1.00	1.75	70
	F	H-8-12-R-H	4.71	1.00	0.00	70
			37.75	7.67	6.83	510

scenario. The three reruns of the VRPSST model, which occur due to reactive call-outs on Monday, Tuesday, and Thursday, alter the nature of the original optimal schedule and result in a 9.8% reduction in travel time, a reduction in overtime of 8.9%, and a reduction in distance traveled of 5.6%. Notice that, in this particular scenario, although travel time decreases under reoptimization, the percentage reduction in overtime is slightly less than in the case of simple inclusion.

13.11 Overall Computational Results

The results for all scenarios are shown in Tables 13.7–13.9. These tables display the percentage reduction in travel time, overtime, and distance traveled achieved when the optimization

Table 13.7. *Results: reactive call-outs ignored.*

Scenario	TT (% reduction)	OT (% reduction)	TD (% reduction)
1	21.4	9.8	20.3
2	0.0	26.4	5.2
3	15.3	25.0	18.0
4	6.5	17.9	14.5
5	7.1	0.0	19.2
6	6.5	81.0	9.7
7	20.6	38.6	20.0
8	12.5	0.0	18.8
9	10.7	80.0	12.5
10	25.0	0.0	28.3
11	0.0	0.0	23.1
12	32.6	97.4	30.4
Average	13.6	32.5	18.2

Table 13.8. *Results: reactive call-outs—simple inclusion.*

Scenario	TT (% reduction)	OT (% reduction)	TD (% reduction)
2	0.0	0.0	3.4
3	10.9	22.9	14.9
5	2.9	14.8	6.7
6	16.7	26.2	11.1
7	21.4	30.7	21.1
8	14.7	0.0	18.8
9	5.9	44.0	7.7
10	8.3	38.8	15.7
11	0.0	2.8	8.3
12	5.9	11.1	5.6
Average	8.3	20.1	11.0

model is used for each scenario in the case of no reactive call-outs, reactive call-outs with simple inclusion, and reactive call-outs with reoptimization. Computation times required to obtain the optimal schedule for each scenario, together with their associated VRPSST problem sizes, are shown in Table 13.10.

If reactive call-outs are ignored (see Table 13.7), the average results over all scenarios indicate a substantial reduction in travel time (14%) and distance traveled (18%), together with a dramatic reduction in overtime (33%). Table 13.8 displays the results when reactive call-outs are implemented using simple inclusion. No results are available for scenarios 1 and 4 as no reactive call-outs occurred during these 2 weeks. Notice that, although the optimal schedule clearly outperforms the manual schedule, the improvements over current practice are slightly less than in the previous case (8%, 11%, and 20%, respectively). This decrease is because reactive call-outs occur at random points of the week and no secondary optimization is completed once the original schedule has been interrupted. Once reoptimization is implemented only minor increases in the reductions beyond the simple inclusion

Table 13.9. *Results: reactive call-outs—reoptimization.*

Scenario	TT (% reduction)	OT (% reduction)	TD (% reduction)
2	0.0	0.0	3.4
3	10.9	22.9	14.9
5	8.8	7.4	6.7
6	22.2	23.1	22.2
7	21.4	30.7	23.7
8	14.7	0.0	18.8
9	5.9	30.8	7.7
10	8.3	38.8	15.7
11	0.0	41.7	4.2
12	9.8	8.9	5.6
Average	9.6	19.6	11.7

Table 13.10. *Problem data and computation times for each scenario.*

Scenario	Jobs ($n-1$)	Days (K)	Computational times (hours)
1	13	5	8.41
2	14	6	11.16
3	10	5	8.87
4	16	6	13.58
5	11	5	11.64
6	10	5	8.46
7	10	5	8.73
8*	6	4	0.00
9	9	4	4.99
10	11	4	3.70
11	13	5	7.58
12	13	5	8.38
Average	11.33	4.92	7.96

*Optimal solution found at the root node of the OUTER tree.

case are achieved. Clearly, these results indicate that reoptimization may be unnecessary in certain practical cases where large amounts of unscheduled reactive call-outs disrupt the optimal routing system. Nevertheless, results were completed only up to the end of the planning horizon, not on a rolling weekly reoptimization basis, and, therefore, depending on costs and efficiency targets, reoptimization may still be comparatively important to management.

13.12 Conclusion

The modeling approach presented in this study involved the construction of a VRPSST-based model and the application of a new solution method to identify optimal schedules of jobs and to recommend the most efficient routes for the FSEs taking into account stochastic service times and reactive call-outs. The performance of the model was evaluated by analyzing existing FSE schedules and investigating the impact on FSE performance of using the model at two different stages of implementation, one of which is a simulated on-line system.

The results of the optimization model show that improvements of approximately 8% and 11% in total travel time and distance traveled, respectively, can be achieved, for a given FSE, when stochastic service times and reactive call-outs are included in the FSE's weekly schedule. In addition, the average reduction in overtime in such cases is approximately 20%.

Bibliography

[1] C. Bastian and A. Kan. The stochastic vehicle routing problem revisited. *European Journal of Operational Research*, 56:407–412, 1992.

[2] D. Bertsimas. A vehicle routing problem with stochastic demand. *Operations Research*, 40:574–585, 1992.

[3] D. Bertsimas, P. Jaillet, and A. Odoni. A priori optimisation. *Operations Research*, 38:1019–1033, 1990.

[4] R.L. Carraway, T.L. Morin, and H. Moskowitz. Generalized dynamic programming for stochastic combinatorial optimisation. *Operations Research*, 37:819–829, 1989.

[5] G. Clarke and J.V. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12:568–581, 1964.

[6] E.W. Dijkstra. A note on two problems in connection with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[7] M. Dror. Modelling vehicle routing with uncertain demands as a stochastic program: Properties of the corresponding solution. *European Journal of Operational Research*, 64:432–441, 1993.

[8] M. Dror, G. Laporte, and F. Louveaux. Vehicle routing with stochastic demands and restricted failures. *ZOR—Methods and Models of Operations Research*, 37:273–283, 1993.

[9] M. Dror, G. Laporte, and P. Trudeau. Vehicle routing with stochastic demands: Properties and solution frameworks. *Transportation Science*, 23:166–176, 1989.

[10] M. Dror and P. Trudeau. Stochastic vehicle routing with modified savings algorithm. *European Journal of Operational Research*, 23:228–235, 1986.

[11] M. Gendreau, G. Laporte, and R. Séguin. A tabu search heuristic for the vehicle routing problem with stochastic demands and customers. Working paper, CRT, Montreal University, Canada, 1994.

[12] M. Gendreau, G. Laporte, and R. Séguin. An exact algorithm for the vehicle routing problem with stochastic demands and customers. *Transportation Science*, 29:143–155, 1995.

[13] M. Gendreau, G. Laporte, and R. Séguin. Stochastic vehicle routing. *European Journal of Operational Research*, 88:3–12, 1996.

[14] B.L. Golden and W.R. Stewart Jr. Vehicle routing with probabilistic demands. *Computer Science and Statistics: Tenth Annual Symposium on the Interface*, 503:252–259, 1978.

[15] E.P.C. Kao. A preference order dynamic program for a stochastic travelling salesman problem. *Operations Research*, 26:1033–1045, 1978.

[16] V. Lambert, G. Laporte, and F. Louveaux. Designing collection routes through bank branches. *Computers and Operations Research*, 20:783–791, 1993.

[17] G. Laporte and F. Louveaux. Formulations and bounds for the stochastic capacitated vehicle routing with uncertain supplies. In J.J. Gabszewicz, J.F. Richard, and L.A. Wolsey, editors, *Economic Decision Making: Games, Econometrics and Approximate Algorithms*, North-Holland, Amsterdam, 1990, pp. 443–455.

[18] G. Laporte and F. Louveaux. The integer L-shaped method for stochastic integer programs with complete recourse. *Operations Research Letters*, 13:133–142, 1993.

[19] G. Laporte, F. Louveaux, and H. Mercure. Models and exact solutions for a class of stochastic location-routing problems. *European Journal of Operational Research*, 39:71–78, 1989.

[20] G. Laporte, F. Louveaux, and H. Mercure. The vehicle routing problem with stochastic travel times. *Transportation Science*, 26:161–170, 1992.

[21] J. Popović. Vehicle routing in the case of uncertain demand: A Bayesian approach. *Transportation Planning and Technology*, 19:19–29, 1995.

[22] D. Roberts. Algorithms for stochastic vehicle routing problems. Ph.D. thesis, Imperial College, University of London, 1998.

[23] D. Roberts and E. Hadjiconstantinou. Algorithmic developments in stochastic vehicle routing. In P. Kischka, H. Lorenz, U. Derigs, W. Domschke, P. Kleinschmidt, and R. Moehring, editors, *Operations Research Proceedings* 1997, Springer-Verlag, Berlin, 1998, pp. 156–161.

[24] D. Roberts and E. Hadjiconstantinou. A computational approach to the vehicle routing problem with stochastic demands. In P. Borne, M. Ksouri, and A. El Kamel, editors, *Computational Engineering in Systems Applications*, IEEE, 1998, pp. 139–144.

[25] E. Sniedovich. Analysis of a preference order travelling salesman problem. *Operations Research*, 29:1234–1237, 1981.

[26] W.R. Stewart Jr. and B.L. Golden. Stochastic vehicle routing: A comprehensive approach. *European Journal of Operational Research*, 14:371–385, 1983.

[27] D. Teodorović, E. Krcmar-Nozić, and G. Pavković. The mixed fleet stochastic routing problem. *Transportation Planning and Technology*, 19:31–43, 1995.

[28] D. Teodorović and G. Pavković. A simulated annealing technique approach to the vrp in the case of stochastic demand. *Transportation Planning and Technology*, 16:261–273, 1992.

[29] F. Tillman. The multiple terminal delivery problem with probabilistic demands. *Transportation Science*, 3:192–204, 1969.