A Set-Partitioning-Based Model for the Stochastic Vehicle Routing Problem

Clara Novoa
Texas State University

Rosemary Berger
Jeffrey Linderoth
Robert Storer
Lehigh University

Report No. 06T-008

A Set-Partitioning-Based Model for the Stochastic Vehicle Routing Problem

Clara Novoa

Department of Engineering and Technology
Texas State University
601 University Drive
San Marcos, TX 78666

cn17@txstate.edu

Rosemary Berger   Jeff Linderoth   Robert Storer

Department of Industrial and Systems Engineering
Lehigh University
200 W. Packer Ave.
Bethlehem, PA 18015

rosemary.berger@verizon.net   jtl3@lehigh.edu   rhs2@lehigh.edu

TECHNICAL REPORT 06T-008

December, 2006

Abstract

The objective of the Vehicle Routing Problem (VRP) is to construct a minimum cost set of vehicle routes that visits all customers and satisfies demands without violating the vehicle capacity constraints. The Stochastic Vehicle Routing Problem (SVRP) results when one or more elements of the VRP are modeled as random variables. In this paper, we present a set-partitioning-based modeling framework for the VRP with stochastic demands (VRPSD). The framework can be adapted easily for routing problems with randomness in other problem elements, such as random customers and random travel times. We formulate the VRPSD as a two-stage stochastic program and introduce an extended recourse strategy in which vehicles are allowed to serve additional customers from failed routes prior to returning to the depot or to serve customers from failed routes on a new route after returning to the depot. Computational experiments show that route plans generated using the new recourse function perform quite well, especially for problems with few customers per route, where cost savings of roughly 5% are possible.

Introduction

The Vehicle Routing Problem (VRP) is defined by a depot, a set of geographically dispersed customers with known demands, and a set of vehicles with fixed capacity. The objective is to find a minimum cost set of vehicle routes that visits all customers and satisfies demands without violating the vehicle capacity constraints. The VRP assumes all elements of the problem are known and deterministic. The Stochastic Vehicle Routing Problem (SVRP) results when one or more elements of the VRP are modeled as random variables to better reflect the uncertainty that exists in practice. In the literature, the set of customers to be visited, the travel times between customers, and the demands of customers all have been treated as random elements. While the focus in this paper is on the VRP with stochastic demands (VRPSD), an important feature of the proposed set-partitioning approach is its flexibility to handle randomness in other elements with little or no additional effort.

The VRPSD arises in many practical situations. Dror et al. (1985), Psaraftis (1995) and Chepuri and de Mello (2005) cite the delivery of petroleum products, industrial gases, and home heating oil. Dessouky et al. (2005) offer the example of delivering supplies to cities under a state of emergency, Yang et al. (2000) cite the delivery of items to hospitals and restaurants, Laporte et al. (1989) give the example of collecting money from banks, and Marković et al. (2005) suggest that VRPSD can be used to model the delivery and pickup of mail, packages, and recycled material from offices and industrial plants. Bertsimas (1992) also cites as a practical application a company designing routes during a strategic planning phase before exact customer demands are known.

As in previous work on the VRPSD (Bertsimas, 1992, Teodorovic and Pavkovic, 1992, Savelsbergh and Goetschalckx, 1995, Hjorring and Holt, 1999, Laporte et al., 2002), we assume that customer demands are independent and follow known probability distributions. Our solution method, which works for general joint demand distributions, does not require the assumption on independent customer demands. However, we assume independence in the problem instances used in our computational results in order to facilitate a straightforward comparison to previous results in the literature. A customer's actual demand becomes known with certainty only when a vehicle arrives at the customer's location. As a result, a planned route may "fail" when the realized demand at a particular customer exceeds the remaining vehicle capacity (supply). When a route failure occurs, a recourse action, which results in additional cost, is required to serve any unsatisfied demand.

The VRPSD often is modeled as a two-stage problem. The solution from stage one specifies a route for each vehicle. The vehicles follow the routes as planned and customer demands are revealed at each stop. When routes fail, recourse actions are implemented to serve any remaining customers. The solution from stage two specifies the actual route of each vehicle taking into account the recourse actions. The VRPSD objective is to construct a set of planned routes that minimizes the sum of the costs of the planned routes and the expected cost of the recourse actions.

Most VRPSD research assumes that when a route fails, the vehicle returns to the depot, unloads (or replenishes its supply), and then resumes the route as specified in the initial route plan (Teodorovic and Pavkovic, 1992, Hjorring and Holt, 1999, Laporte et al., 2002). We refer to this recourse policy as *traditional recourse*. The main contribution of this paper is the development of a viable computational approach for the VRPSD under a new and more realistic recourse strategy. Under the new recourse strategy, two additional options are available: (1) a vehicle that has completed its original route and has remaining capacity (supply) may serve customers from failed routes directly, without first returning to the depot and (2) a vehicle that experiences a route failure may return to the depot and then perform an "extra trip" that covers unserved customers from failed routes. We refer to this recourse policy as *extended recourse*. In this paper, the VRPSD under extended recourse is formulated as a two-stage stochastic program with integer recourse and modeled as a stochastic extension of the well-known set-partitioning formulation for the VRP. The set-partitioning approach provides significant advantages in modeling a variety of practical constraints, such as time windows and complicated route restrictions. In addition, unlike edge-based formulations, there is no need to assume that cost is proportional to distance, and nonlinear route costs can be considered. Scenarios are used to capture randomness, and effective heuristic procedures are developed to generate routes (columns) for the set-partitioning formulation.

The remainder of the paper is organized as follows. Section 1 reviews the VRPSD literature. Section 2 provides a formal description of the VRPSD and presents set-partitioning-based formulations of the problem under both traditional and extended recourse. Section 3 describes the heuristic route generation procedure and assesses the quality and efficiency of the procedure. Section 4 presents results of the computational experiments comparing the planned routes generated by the extended recourse model to the planned routes generated by the traditional recourse model and by a deterministic VRP model. Section 5 concludes the paper and provides direction for future research.

1 Literature Review

Although Tillman (1969) introduced the first practical example of a VRPSD for designing routes in a small multiple terminal delivery problem in the late 1960s, the VRPSD has received much less attention in the literature than the deterministic VRP. Most of the research on the VRPSD can be classified either as an *a priori* optimization approach or as a reoptimization approach. The static "here-and-now", or *a priori*, approach designs routes before actual demands become known and the route sequence is not changed during real-time execution. The dynamic "wait-and-see", or reoptimization, approach does not plan routes in advance but instead makes routing decisions one step at a time. Information is updated each time a vehicle arrives at a customer and observes demand; the problem is modeled as a Markov decision process in which a decision is defined as which customer to visit next and whether or not to return to the depot first. Both the traditional recourse and the proposed extended recourse strategies apply to *a priori* optimization approaches, and therefore, we restrict our attention to the *a priori* optimization literature. For a review of reoptimization approaches, see Dror et al. (1989), Dror (1993), Secomandi (2001), Novoa (2005) and Novoa and Storer (2005).

1.1 Types of Recourse

Under an *a priori* optimization scheme, recourse actions are required whenever a route failure occurs. In early work on the VRPSD, Stewart et al. (1983) propose two simple models that incorporate penalties into the objective function to account for route failures. In the first model, a fixed penalty is assessed for each failed route, while in the second model, a penalty is assessed for each unit of demand in excess of vehicle capacity. Dror and Trudeau (1986) consider an expected cost objective function where, after a route failure, all remaining customers are served by individual direct deliveries from the depot. Dror et al. (1989) introduce the *traditional recourse* strategy in which vehicles return to the depot to unload/replenish capacity when a customer demand cannot be satisfied and then return to the failure location to continue the planned route. Teodorovic and Pavkovic (1992) and Savelsbergh and Goetschalckx (1995) simplify this recourse to achieve computational efficiency by assuming at most one failure per route. Laporte et al. (2002) assume the same recourse as in Dror et al. (1989) and consider multiple route failures.

Several papers consider *proactive* recourse strategies that include returns to the depot for unloading/restocking before a failure occurs. For a single VRPSD, Bertsimas et al. (1995) and Bianchi et al. (2004) use dynamic programming to construct routes that include proactive returns to the depot that improve the expected cost of the solutions. Yang et al. (2000) also use dynamic programming to implement a proactive recourse and design routes for multiple VRPSD situations. Chepuri and de Mello (2005) employ a penalty-type recourse in which, upon failure, a route is terminated and a penalty is assessed for lost revenue and/or the cost of emergency delivery. This recourse is practical when the time frame within which demand must be met is critical but does not apply to situations in which the vehicle can return to the depot.

Ak and Erera (2006) remove the assumption of vehicles operating independent *a priori* tours and study a paired-vehicle recourse strategy. In this strategy, *a priori* vehicle routes are paired to pool the capacity of two vehicles in an attempt to reduce routing costs. One vehicle in the pair performs a route that serves a large amount of demand. If the vehicle capacity is exceeded on the route, the second vehicle in the pair, that is performing a route that serves a lower amount of demand, adds any unserved customers to the end of its route. If the second vehicle is unable to serve all additional customers on its original route, then the traditional recourse strategy is used to serve these customers. The model is solved via tabu search, and the authors report that the paired-recourse strategy can reduce expected travel cost in the range of 3% to 25%. Further, the computational times to produce these solutions are modest. The extended recourse model we propose in Section 2.2 is similar to the approach of Ak and Erera (2006) in that a strict and static assignment of vehicles to customers is not enforced.

1.2 Solution Approaches

Researchers have developed both heuristic and exact methods for solving the VRPSD. Because of the difficulty of the problem, a common approach is to adapt VRP heuristics to solve the stochastic version. Tillman (1969), Stewart et al. (1983), and Dror and Trudeau (1986) propose different modifications to the savings algorithm of Clarke and Wright (1964). Others develop two-phase algorithms. Savelsbergh and Goetschalckx (1995) first solve a generalized assignment problem for the VRP and then use neighborhood search methods that take into account the stochastic objective function to improve the solution. Yang et al. (2000) develop algorithms that resemble the route-first-cluster-next and the cluster-first-route-second techniques for VRPs. For a model with proactive restocking, they evaluate insertion and deletion costs using dynamic programming and obtain a lower bound on the solution cost by solving the linear programming relaxation of a set-partitioning formulation. Other papers consider metaheuristics; Teodorovic and Pavkovic (1992) build a simulated annealing-based algorithm, Gendreau et al. (1996) develop a tabu search heuristic using a proxy for the objective function in order to efficiently evaluate potential moves, and Bianchi et al. (2004) develop evolutionary, ant colony, and iterated local search algorithms for solving the single VRPSD case.

Both chance-constrained programs (CCP) and stochastic programs with recourse (SPR) can be used to explicitly model the stochastic aspect of the VRPSD. Stewart et al. (1983) describe a CCP model to identify a set of vehicle routes of minimum distance subject to the constraint that the probability of failure of any route is less than an allowable limit. They ignore the cost of recourse actions and solve the model by transforming it into an equivalent deterministic VRP, which then can be solved using standard solution techniques for the VRP. Laporte et al. (1989) develop a branch and bound algorithm to solve exactly both a CCP model and a bounded penalty model.

SPR models for the VRPSD are the most computationally challenging to solve. Most of the SPR models in the literature consider the VRPSD as a two-stage problem and all employ the traditional recourse policy of Dror et al. (1989). Although traditional recourse simplifies the model to some extent, it overestimates the recourse cost, particularly in the case of multiple vehicles. Using the integer L-shaped method proposed by Laporte and Louveaux (1993, 1998), Gendreau et al. (1995) were the first to optimally solve a VRP with stochastic customers and demands. Hjorring and Holt (1999) optimally solve a single VRPSD using the integer L-shaped method; to reduce CPU time, they introduce a tight lower bound and a general optimality cut. To optimally solve a multiple VRPSD, Laporte et al. (2002) incorporate new inequalities into the integer L-shaped method and are able to solve instances with up to 4 vehicles and 25 customers. As Dror et al. (1989) recognized, the VRPSD involves more than two stages, since every time a vehicle visits a customer and learns the demand, new decisions can be taken. Dror (1993) presents, but does not attempt to solve, a multistage stochastic integer programming model for the VRPSD. To our knowledge, ours is the first work to propose a set-partitioning-based model for VRPSD.

2 Set-Partitioning-Based Models for Stochastic Vehicle Routing

A fleet of k identical vehicles with fixed capacity Q depart from a single depot to perform only deliveries (or equivalently only pick-ups) at different customer locations. Node 0 represents the depot and $I = \{1, 2, ..., n\}$ represents the set of customer locations. Associated with each customer location i, $i = 1, 2, ..., n$, is a random variable D_i, which represents knowledge about customer i's demand at the time the routes are planned. These customer demand distributions are assumed known, but actual demand d_i for a particular customer i is revealed only when the vehicle arrives at customer i's location. No customer's demand ever exceeds the vehicle capacity and customer demands need not be assumed independent.

As a result of random demands, a planned vehicle route will fail at a particular customer location whenever the cumulative demand on the route exceeds the vehicle capacity. Furthermore, a planned route may be stopped before a route failure occurs (i.e., a proactive stop) if the probability of satisfying the next customer's demand is less than a given threshold parameter. When a route fails or is stopped proactively, a set of recourse actions is implemented to satisfy any remaining customer demands. The objective of the VRPSD is to construct a set of at most k planned vehicle routes of minimum expected cost, where the expected cost is the sum of the expected distance of the planned routes and the expected additional cost of the recourse actions.

2.1 Formulation of VRPSD with Traditional Recourse

In this section, we present a new formulation of the VRPSD with traditional recourse. Under the traditional recourse policy, when a route fails, the vehicle returns to the depot to unload/replenish and then returns to the location of the failure to continue the planned route. The objective of the problem is to construct a set of planned routes of minimum expected cost.

The new formulation presented here is a stochastic extension of a set-partitioning model for the VRP, which was originally proposed by Cullen et al. (1981) and Desrosiers et al. (1984). In the VRP model, a decision variable is associated with each feasible route. Each route is a column in the model and the objective is to select a set of columns of minimum total cost such that each row (i.e., customer) is covered by exactly one column.

To formulate the stochastic extension, let R be the set of routes. Each route $r \in R$ visits a set of customers $\mathcal{A}_r \subseteq I$, and each vehicle must visit all of the customers on its route. The cost of route r, denoted c_r, is the sum of two components: the cost of the route assuming no failure occurs and a cost $\mathcal{Q}_r$ that accounts for the extra distance traveled to and from the depot as a result of route failures. Because each vehicle follows its originally planned route, computing the expected cost $\mathcal{Q}_r$ associated with returns to the depot is a straight-forward calculation and can be done *a priori* for each route. We note, however, that unlike the deterministic VRP, the cost of the route depends on the orientation of the route. Table 1 lists the definition of sets, parameters, and variables used in the set-partitioning model under the traditional recourse strategy. The model (SP-TR) and the computation of $\mathcal{Q}_r$ follow the table.

Table 1: Definition of Entities in Set-Partitioning Model under a Traditional Recourse Strategy

Sets	
R	Set of routes.
I	Set of customers.
Parameters	
c_r	The *exact expected* cost of route $r \in R$.
a_{ir}	1 if customer $i \in I$ is on route $r \in R$, 0 otherwise.
k	Number of trucks.
Variables	
x_r	1 if route $r \in R$ is selected, 0 otherwise.

minimize

$$\sum_{r \in R} c_r x_r \quad \text{(SP-TR)}$$

subject to

$$\sum_{r \in R} a_{ir} x_r = 1 \qquad \forall i \in I \tag{1}$$

$$\sum_{r \in R} x_r \leq k \tag{2}$$

$$x_r \in \{0, 1\} \qquad \forall r \in R \tag{3}$$

The objective function seeks to minimize the total expected routing cost. Constraints (1) ensure that each customer is served by exactly one route. Constraint (2) limits the number of selected routes to at most the number of vehicles. Constraints (3) are standard binary restrictions on the decision variables.

The expected cost of a route c_r includes a cost $\mathcal{Q}_r$ that accounts for the extra distance traveled to and from the depot due to the route failures. This cost $\mathcal{Q}_r$ depends on the orientation of the route. To compute $\mathcal{Q}_r$ for a route r, without loss of generality, let $r^{\rightarrow} \stackrel{\text{def}}{=} (0, 1, 2, ..., s_r, 0)$ be a route and orientation (zero represents the depot, one represents the first customer in the sequence, and so on). Recall from the problem description that d_i represents the actual demand at customer i and Q represents the vehicle capacity. Let $\mathtt{dist}_{i,j}$ represent the distance between customers i and j. Equation (4) defines the exact expected cost of the additional trips to the depot for route $r^{\rightarrow}$ under the specified orientation and can be used for both continuous and discrete demand distributions:

$$\begin{aligned}\mathcal{Q}_{r^{\rightarrow}} \stackrel{\text{def}}{=} & \ 2\sum_{j=1}^{s_r}\sum_{l=1}^{\infty}\mathbb{P}\left(\sum_{u=1}^{j-1} d_u < lQ < \sum_{u=1}^{j} d_u\right)\mathtt{dist}_{0,j} + \\ & +\sum_{j=1}^{s_r}\sum_{l=1}^{\infty}\mathbb{P}\left(\sum_{u=1}^{j-1} d_u = lQ\right)(\mathtt{dist}_{0,j-1} + \mathtt{dist}_{0,j} - \mathtt{dist}_{j-1,j}). \end{aligned} \tag{4}$$

Dror et al. (1989), Teodorovic and Pavkovic (1992), and Laporte et al. (2002) derive equations similar to (4) assuming continuous demand distributions. The main difference between a continuous and discrete distribution is that, with a continuous demand distribution, the probability of a vehicle becoming empty at the same instant as a service completion is zero; with a discrete demand distribution, the simultaneous events are possible. In equation (4), the first probability term corresponds to having the l^{th} failure at customer j and the second probability term corresponds to having the l^{th} failure simultaneously with the service completion at customer $j-1$. Since no single customer demand exceeds the vehicle capacity, the upper limit of the summation over l can be replaced by j. Equation 5 is a compact form for equation 4:

$$\begin{aligned}\mathcal{Q}_{r^{\rightarrow}} \stackrel{\text{def}}{=} & \ 2\sum_{j=1}^{s_r}\sum_{l=1}^{j}(F^{j-1}(lQ-1) - F^{j}(lQ))\mathtt{dist}_{0,j} + \\ & +\sum_{j=1}^{s_r}\sum_{l=1}^{j}P(\sum_{u=1}^{j-1} d_u = lQ)(\mathtt{dist}_{0,j-1} + \mathtt{dist}_{0,j} - \mathtt{dist}_{j-1,j}), \end{aligned} \tag{5}$$

where $F^j(lQ) \stackrel{\text{def}}{=} \mathbb{P}\left(\sum_{u=1}^{j} d_u \leq lQ\right)$. Applying equation 5 to the reverse route $r^{\leftarrow} \stackrel{\text{def}}{=} (0, s_r, s_{r-1}, \ldots, 1, 0)$, we can obtain the expected cost $\mathcal{Q}_{r^{\leftarrow}}$. Then, the expected cost of route r is $\mathcal{Q}_r = \min(\mathcal{Q}_{r^{\rightarrow}}, \mathcal{Q}_{r^{\leftarrow}})$.

2.2 Formulation of VRPSD with Extended Recourse

In this section, we extend the set-partitioning formulation presented in section 2.1 to accommodate the extended recourse policy. The extended recourse strategy assumes a full service policy whereby, once a vehicle arrives at a customer location, the customer's demand is revealed before service starts and the customer will not accept a partial delivery. Under the extended recourse policy, two new recourse actions are allowed when a route fails or is stopped proactively. First, a vehicle that has completed its original route may serve customers from failed routes directly, without first returning to the depot; we call this action *a completion*. Second, a vehicle that experiences a route failure may return to the depot and then serve customers from one or more failed routes on a new route; we call this action *an extra trip*.

The VRPSD with extended recourse is formulated as a two-stage problem, which is a simplification of the situation in practice. In stage one, each vehicle follows its planned route until either it finishes serving the last customer, it fails at a customer, or it is stopped proactively. Then, we assume that all of the remaining uncertainty is revealed, and stage two begins with the implementation of a set of recourse actions that seeks to minimize the additional cost of serving the remaining customers. Although a multi-stage model would more accurately reflect the real situation, there are situations for which the two-stage approach reflects practice. For example, consider an application in which customers are served daily on regular, fixed routes. Suppose that demand is fully realized at the beginning of the day but regularity in service is favored over the potential savings achieved by re-optimizing the routes on a daily basis. Applying the two-stage model to this situation can produce a solution that balances regularity and cost savings, since the stage-two reoptimization procedure is postponed until after the planned routes have been followed to the fullest extent and have either served all customers or failed. In addition to providing a reasonable approximation to what happens in practice, the new model allows us to investigate whether using the extra capacity available on some vehicles can reduce the overall expected cost. That is, we seek to determine if modeling more realistic recourse actions than simply returning to the depot can result in planned routes that are more robust to failures.

To formulate the VRPSD with extended recourse, we again consider a set R of routes. The set partitioning approach allows for the possibility of including routes that are infeasible in expectation (that is, if all demands equal their expected values, the total demand over the route exceeds vehicle capacity). For the extended recourse model, allowing infeasible routes is problematic. First, in the motivating case where a set of "regular routes" that are followed over multiple days is desired, it makes little practical sense to allow infeasible routes. Second, the extended recourse model assumes that all uncertainty is revealed at the end of phase one. In order to minimize the impact of this somewhat unrealistic assumption, we explicitly force all routes to be feasible in expectation in this model. By mandating feasible routes, the number of unserved customers at the end of phase one will be small, thus the impact of revealing all uncertainty will be minimized. For the SP-TR model based on traditional recourse, allowing routes that are infeasible in expectation does not cause problems as it does in the set partitioning with extended recourse (SP-ER) case. In traditional recourse, the original route is always followed and thus implicitly sticks to the assumption that demand is revealed only upon arrival at the customer. Allowing infeasible routes may (or may not) be advantageous in this case since the final result may be lower total expected cost. For this reason we do allow infeasible routes when comparing the quality of the solution produced by SP-TR model to previous work (Section 4.1).

Let $\mathcal{A}_r \subseteq I$ be the planned set of customers to be visited on route r. To incorporate demand uncertainty, the model is augmented with a set S of scenarios where each scenario represents a realization of the random demands. For each planned route $r \in R$ and scenario $s \in S$, there is a set of customers $\mathcal{T}_{rs} \subseteq \mathcal{A}_r$ that is served during stage one. The model is further augmented with completions and extra trips. A completion is associated with a route r and a scenario s; it is an additional column in the model that represents additional customers that could be served after a vehicle completes route r given the realization of demands in scenario s. In the model, each pair (r,s) of route $r \in R$ and scenario $s \in S$ may have a set of completions $\mathcal{C}_{rs}$, and each completion $c \in \mathcal{C}_{rs}$ visits a set of customers $\mathcal{W}_c \subseteq I$. An extra trip is an additional column in the model that represents a new route that a vehicle could serve after it returns to the depot to unload/replenish given the realization of demands in scenario s. In the model, E_s is the set of extra trips associated with scenario s.

Table 2 lists the definition of sets, parameters, and variables used in the two-stage stochastic integer model with the new extended recourse strategy. The model (SP-ER) follows the table.

Table 2: Definition of Entities in Set-Partitioning Model under the Extended Recourse Strategy

Sets	
R	Set of routes.
S	Set of scenarios.
E_s	Set of extra trips in scenario $s \in S$.
I	Set of customers.
C_{rs}	Set of completions for route r in scenario s.
C_s	Set of completions for scenario $s \in S$. ($C_s = \cup_{r \in R} C_{rs}$)
Parameters	
a_{ir}	1 if customer $i \in I$ is on planned route $r \in R$, 0 otherwise.
t_{irs}	1 if customer $i \in I$ is served on route $r \in R$ in scenario $s \in S$, 0 otherwise.
w_{ics}	1 if customer $i \in I$ is served by completion $c \in C_s$ in scenario $s \in S$, 0 otherwise.
v_{ies}	1 if customer $i \in I$ is served by extra trip $e \in E_s$ in scenario $s \in S$, 0 otherwise.
c_r	The *expected* cost of planned route $r \in R$.
d_{cs}	Cost of completion $c \in C_s$ in scenario $s \in S$.
f_{es}	Cost of extra trip $e \in E_s$ in scenario $s \in S$.
p_s	Probability of scenario $s \in S$.
k	Number of trucks.
Variables	
x_r	1 if planned route $r \in R$ is selected, 0 otherwise.
y_{cs}	1 if completion $c \in C_s$ is used in scenario $s \in S$, 0 otherwise.
z_{es}	1 if extra trip $e \in E_s$ is used in scenario $s \in S$, 0 otherwise.

minimize

$$\sum_{r \in R} c_r x_r + \sum_{s \in S} p_s \left(\sum_{c \in C_s} d_{cs} y_{cs} + \sum_{e \in E_s} f_{es} z_{es} \right) \quad \text{(SP-ER)}$$

subject to

$$\sum_{r \in R} a_{ir} x_r = 1 \quad \forall i \in I \tag{6}$$

$$\sum_{r \in R} t_{irs} x_r + \sum_{c \in C_s} w_{ics} y_{cs} + \sum_{e \in E_s} v_{ies} z_{es} = 1 \quad \forall i \in I, \forall s \in S \tag{7}$$

$$x_r - \sum_{c \in C_{rs}} y_{cs} \geq 0 \quad \forall r \in R, \forall s \in S \tag{8}$$

$$\sum_{r \in R} x_r \leq k \tag{9}$$

$$\sum_{e \in E_s} z_{es} \leq k \quad \forall s \in S \tag{10}$$

$$x_r \in \{0,1\} \quad \forall r \in R \tag{11}$$

$$y_{cs} \in \{0,1\} \quad \forall s \in S, \forall c \in C_s \tag{12}$$

$$z_{es} \in \{0,1\} \quad \forall s \in S, \forall e \in E_s \tag{13}$$

The objective function seeks to minimize the expected cost of the planned routes plus the expected cost of the completions and extra trips. Constraints 6 ensure that each customer is on a planned route. Constraints 7 require each customer to be served in each scenario. Constraints 8 limit a selected route to doing at most one completion. Constraint 9 limits the number of selected routes to at most the number of vehicles. Constraints 10 limit the number of extra trips in each scenario to at most the number of vehicles. Constraints 11, 12 and 13 are standard binary restrictions on the decision variables.

In the model (SP-ER), the cost c_r of route $r \in R$ is the expected cost of the route with respect to the set of defined demand scenarios. That is, to calculate the expected cost of a route, we sum over all of the scenarios the product of the cost of the route to the point of completion or failure and the probability of the scenario. Since the model involves only two stages, we assume that the routes are planned at time zero and all of the uncertainty is revealed at the end of stage one, which is the point when all vehicles have completed serving the last customer or have failed. As a result, we only need to consider feasible completions and extra trips associated with the scenarios and we can easily compute their associated cost. To compute the cost d_{cs} of a completion c associated with a route in scenario s, recall that a vehicle with remaining capacity/supply can serve additional customers directly prior to returning to the depot. The cost of such a completion is the cost of starting from the last customer on the planned route, serving a set of customers and then returning to the depot *minus* the cost of traveling from the last customer on the planned route to the depot. Subtracting the cost of this last leg of the planned route is necessary since it is included in the cost of the planned route.

Since extra trips start from the depot, the cost f_{es} of an extra trip e associated with scenario s is simply the cost of starting at the depot, serving a set of customers and returning to the depot.

One of the advantages of the set-partitioning-based model is that it can be adapted to model vehicle routing problems with other types of randomness. For example, to include random travel times in the model, the set of customers actually served in a scenario is determined by a realization of random demands and random travel times. To model the case in which "call-in" customers appear randomly in the instance, we modify the constraints 7 to be active over the set I_s of active customers in scenario s.

3 Route Generation

The main disadvantage of the set-partitioning-based models for the VRPSD is that they contain an exponential number of route variables (x_r) and, in the extended recourse model, an exponential number of constraints 8. For instances of practical size, enumerating all of the feasible routes and solving the resulting integer program is unlikely to be computationally tractable. Instead, only a subset of the feasible routes is included in the model. The possibility of implicitly considering all routes via column generation is an avenue of future research. The objective of the route generation procedure is to generate a set of routes that is rich enough to produce good solutions yet small enough for the integer program to be solved in reasonable time. In this section, we describe briefly how the routes are generated, and we illustrate empirically that good solutions are achieved using the routes we generate. For complete details of the route generation procedure, see Novoa (2005).

3.1 Generating the Planned Routes

For a given instance, the route generation procedure generates a set R of routes that is comprised of three subsets of routes:

- all routes containing a single customer
- a selected set of routes containing between $\underline{s}$ and $\overline{s}$ customers
- a small number of routes containing between two and $\underline{s} - 1$ customers

The parameters $\underline{s}$ and $\overline{s}$ are functions of the demand distribution. Let $\bar{d}_i$ denote the mean demand for customer i and let d_{min_i} (d_{max_i}) denote the minimum (maximum) possible demand value for customer i. Then, values for these parameters are

$$\underline{s} = \left\lfloor \frac{1}{2} \left(\frac{nQ}{\sum_{i \in I} \bar{d}_i} + \frac{nQ}{\sum_{i \in I} d_{max_i}} \right) \right\rfloor$$

and

$$\overline{s} = \left\lfloor \frac{1}{2} \left(\frac{nQ}{\sum_{i \in I} \bar{d}_i} + \frac{nQ}{\sum_{i \in I} d_{min_i}} \right) \right\rfloor .$$

The $\underline{s}$ ($\overline{s}$) value is the average of two average values: the average number of customers on a route when all customers demand the mean value and the average number of customers on a route when all customers demand the maximum (minimum) possible demand value. The total number of routes in the second and third subsets are input parameters to the route generation procedure. Within each subset, the total number of routes is distributed evenly across the different route sizes.

The construction heuristic to generate routes containing between $\underline{s}$ and $\overline{s}$ customers incorporates ideas from the ant colony algorithms described in Dorigo and Gambardella (1997) and Gagné et al. (2001). Each route begins with a randomly selected customer. Subsequent customers are selected based on a desirability value that incorporates the distance between the current customer i and a candidate customer l, the frequency with which arc (i, l) has appeared in routes with the same number of customers and a random component to add diversity. Each route generated by the construction heuristic is tested for further improvement by performing a two-arc interchange heuristic. Furthermore, any route identical to a previously generated one is discarded.

3.2 Generating Completions and Extra Trips

For the extended recourse model, completions and extra trips must be generated for each route and demand scenario. While the cost savings associated with the extended recourse policy are likely to increase with the complexity of the completions and extra trips, we first focus on relatively simple extended recourse actions. For each scenario s and each route r, the set C_{rs} of completions consists of all possible single-customer completions. That is, we calculate the final position of the vehicle at the end of phase one and its remaining capacity. If, given its current capacity, the vehicle can serve an unserved customer whose demand is now known, then a completion column is created and included in the set C_{rs}. In our experiments, C_{rs} includes all feasible single customer completions. For the set E_s of extra trips associated with a scenario s, two alternatives were considered. In the first alternative, all routes including between one and $\overline{s}_{extra}$ customers were generated, where $\overline{s}_{extra}$, an input parameter to the route generation procedure, is set nearly equal to the estimated maximum number of unserved customers in an instance. For each set of customers, the optimal route sequence was computed, and only feasible routes for the scenario were included in the model. In the second alternative, the set of extra trips for a scenario is identical to the original set of planned routes, except that any infeasible column with respect to demand is removed. Clearly, there is a trade-off between the computational time to generate the extra trips and the degradation in the solution quality.

3.3 Performance of the Route Generation Heuristic

In this section, we present the results of computational experiments for the deterministic VRP, which are intended to demonstrate that the route generation procedure produces a good set of routes. For instances of the deterministic VRP, we compare the solution produced by our set-partitioning model to the solution produced by the SYMPHONY branch-and-cut implementation for solving deterministic VRP's as described in Ralphs et al. (2003). For the comparison, a modified version of the traditional recourse model (SP-TR) is used; the cost c_r is the distance of the route, not the expected distance, and the constraint 2 on the number of vehicles is removed.

The test instances consist of two sets of 60 randomly generated problems. Each set of 60 instances results from considering six problem sizes ($n = \{5, 8, 20, 30, 40, 60\}$), two vehicle capacities (see Table 3), and five randomized replicates ($r1, r2, .., r5$), which are generated by changing the random seeds used to select customer demand distributions and locations. The customer locations are generated uniformly in the unit square and the depot location is fixed at $(0, 0)$. The demand for each customer is generated by first choosing a distribution from Table 4 at random and then randomly generating the demand observation from the chosen distribution. The second set of 60 instances is distinguished from the first set primarily by increased variance of customer demand.

Table 3: Vehicle Capacity Levels Studied

	Set 1		Set 2	
n	Q_1	Q_2	Q_1	Q_2
5	9	5	9	9
8	14	9	13	10
20	91	58	60	45
30	137	87	90	68
40	183	116	120	90
60	274	175	180	135

Table 4: Demand Distributions

	Instances in Set 1		Instances in Set 2	
n	Demand Distribution	Variance	Demand Distribution	Variance
5,8	U[1,3] U[2,4] U[3,5]	0.33	U[1,7] U[2,8] U[3,9]	3.00
20,30,40,60	U[1,5] U[6,10] U[11,15]	1.33	U[1,7] U[6,12] U[11,17]	3.00

The objective function value of the set-partitioning solution was compared to the optimal solution found by SYMPHONY for 106 of the 120 instances and to the best integer solution found by SYMPHONY for the remaining 14 instances. Overall, the average percent increase in cost of the set-partitioning solutions over the SYMPHONY solutions was 2.9% for instances in set 1 and 2.3% for instances in set 2. For instances with 30 customers or less, the heuristic performed very well, with the average percent increase in cost of 1.2% for set 1 and 0.8% for set 2. For instances with more than 30 customers, the heuristic provided acceptably good solutions, with an average cost increase of 6.5% for set 1 and 5.4% for set 2. Based on the results of these experiments, we concluded that the heuristic route generation procedure provides a rich enough set of routes for the VRPSD. Note that any further improvements to the route generation scheme will increase the effectiveness of our set-partitioning-based approach to solving the VRPSD.

4 Numerical Results

In this section, two numerical studies quantify the effectiveness of a set-partitioning approach to the VRPSD. In the first study, we show that the set-partitioning model (SP-TR) for the VRPSD under the traditional recourse strategy allows us to quickly and easily obtain solutions of better quality than those appearing in the open literature. In the second study, we quantify the gains in solution quality that can be expected by allowing more complex recourse actions, such as those in the extended recourse strategy, than by simply returning to the depot. Computational experiments in Section 4.1 were run on an Intel Pentium IV CPU, running at 2.4GHz, and equipped with 512MB of RAM. Computational experiments in Section 4.2 were run on a Beowulf cluster with Intel Pentium III processors running at 1.1GHz and equipped with 512MB of RAM.

4.1 Set-Partitioning Models for Traditional Recourse

In the first study, we compare the solutions obtained by the SP-TR model to solutions reported in the literature by Stewart et al. (1983), Dror and Trudeau (1986), and Savelsbergh and Goetschalckx (1995) for a modified version of the 75-customer VRP problem from Christofides and Eilon (1969). This instance is the only one used by all of the authors. The instance was originally a deterministic VRP, and it first appeared in its stochastic version in Dror and Trudeau (1986).

Two independent route generation experiments were done. Their solutions, denoted S1 and S2, from solving the SP-TR model are compared to the solutions obtained by Stewart and Golden ("St-Go"), Dror and Trudeau ("Dr-Tr"), and Savelsbergh and Goetschalckx ("Sa-Go"). In the first experiment, a set of 17,789 routes is included and the resulting minimum cost feasible solution is one with 10 routes. In the second experiment, a set of 27,714 routes is included and the resulting minimum cost feasible solution is one with 9 routes. Note that routes infeasible in expectation were allowed. Typical route sizes ranged from 5 to 13 customers. Table 5 reports the number of vehicles in the solution (k), the expected filling coefficient $f \stackrel{\text{def}}{=} \frac{\sum_{i=1}^{n} \mathbb{E}[d_i]}{kQ}$, the total route distance without accounting for stochastic demand, and the expected cost. The value of f is the ratio of expected demand to total vehicle capacity. Values of f close to one imply that there is a relatively high probability of route failure. Table 5 also includes the mean (μ) and variance (σ^2) of the demand and the probability of failure ($P(d > Q)$) for the fullest and emptiest route in each solution. From the table, we can see that both S1 and S2 have lower expected costs than the previously reported solutions. The Savelsbergh and Goetschalckx (1995) solution, however, is the most appropriate benchmark, since it uses a traditional recourse policy assuming at most one failure per route. Tables 6 and 7 present the detailed route sequence for solutions S1 and S2. Comparing S2 to the Savelsbergh and Goetschalckx solution, it has two routes, instead of one, with expected demand larger than the vehicle capacity, which results in a high probability of failure. However, the directions of the routes are such that failures occur near the depot and their cost impact is low. On the other hand, disregarding any possible fixed cost for the routes, S1 has lower expected cost and lower probabilities of route failure than Savelsbergh and Goetschalckx (1995) and Dror and Trudeau (1986) solutions.

Due to differences in computing technology, it is not possible to compare solution times across the various methods. However, the computational times to generate the routes and solve the SP-TR instances are quite reasonable. For the solutions S1 and S2, Table 8 shows the number of unique routes generated, the CPU time to generate the routes and the CPU time to solve the model with CPLEX v9.1. The time to generate the routes also includes the time to generate the input data file for AMPL. Table 8 also reports the number of simplex iterations and branch and bound nodes (B&B Nodes) required.

Table 5: Our Route Generation Procedure vs. Previous Results

					Fullest route			Emptiest route		
Author	k	f	Distance	E[cost]	μ	σ^2	$P(d > Q)$	μ	σ^2	$P(d > Q)$
St-Go	11	0.78	1016.34	1019.98	145	99.95	0.067	30	3.38	0.000
Dr-Tr	10	0.85	856.16	884.91	166	167.31	0.679	53	5.63	0.000
SP-TR S1	10	0.85	841.45	851.55	147	48.07	0.031	118	77.73	0.000
Sa-Go	9	0.95	805.42	855.55	164	145.75	0.629	139	97.11	0.016
SP-TR S2	9	0.95	819.92	853.67	199	59.12	1.000	135	155.98	0.023

The results of this experiment suggest that the set-partitioning-based model with the route generation heuristic is able to generate solutions for the VRPSD under traditional recourse that

Table 6: Our Solution S1 for the Eilon 75 Customer Problem

Route	Mean	Variance	$P(d > Q)$	Distance	E[cost]	Route Sequence
1	130	204.41	0.0179	74.38	74.89	0, 30, 74, 21, 61, 28, 2
2	118	77.73	0.0000	86.91	86.91	0, 58, 10, 31, 9, 39, 72
3	147	48.07	0.0304	47.41	47.84	0, 4, 67, 26, 12, 40, 17
4	144	140.72	0.0887	77.23	79.81	0, 38, 65, 66, 11, 53, 7
5	142	70.51	0.0160	81.30	81.62	0, 35, 14, 59, 19, 8, 46, 34
6	120	48.37	0.0000	80.34	80.34	0, 45, 29, 15, 57, 54, 13, 27, 52
7	134	111.07	0.0068	109.81	110.10	0, 62, 22, 64, 42, 43, 41, 56, 23
8	142	112.06	0.0445	65.58	65.84	0, 51, 16, 63, 1, 73, 33, 6, 68, 75
9	144	145.45	0.0923	115.16	118.80	0, 3, 44, 32, 50, 25, 55, 18, 24, 49
10	143	105.81	0.0492	103.33	105.40	0, 5, 37, 20, 70, 60, 71, 69, 36, 47, 48
Total	1364	1064.20		841.45	851.55	

Table 7: Our Solution S2 for the Eilon 75 Customer Problem

Route	Mean	Variance	$P(d > Q)$	Distance	E[cost]	Route Sequence
1	141	108.42	0.0340	100.07	101.40	0, 38, 65, 66, 11, 59, 35
2	148	204.72	0.2008	79.22	85.01	0, 30, 74, 21, 61, 28, 62, 2
3	140	58.80	0.0046	84.31	84.36	0, 26, 58, 10, 31, 72, 12, 40
4	135	155.98	0.0227	95.55	96.36	0, 73, 22, 64, 42, 41, 43, 1, 33
5	146	130.78	0.1104	99.36	103.90	0, 44, 32, 50, 18, 55, 25, 9, 39
6	175	106.18	0.9273	64.27	75.50	0, 67, 7, 53, 14, 19, 8, 46, 34
7	137	134.38	0.0236	96.36	96.74	0, 17, 3, 24, 49, 56, 23, 63, 16, 51
8	143	105.81	0.0492	103.33	105.40	0, 5, 37, 20, 70, 60, 71, 69, 36, 47, 48
9	199	59.12	1.0000	97.45	105.00	0, 52, 27, 13, 54, 57, 15, 29, 45, 4, 75, 68,6
Total	1364	1064.19		819.92	853.67	

Table 8: No. of Routes, Running Times, and No. of Iterations in S1 and S2

			CPU seconds				
		Routes	Create	Solve		MIP Simplex	B&B
	k	Generated	Routes	Model	Total	Iterations	Nodes
S1	10	17789	22.52	180.81	203.33	11280	920
S2	9	27714	54.39	771.29	825.68	36576	2883

are competitive with those in the literature. Given that the computational times necessary to generate a diverse set of routes and solve the corresponding integer programming model are reasonable, it indicates that the approach is quite promising for the VRPSD.

4.2 Set-Partitioning Models for Extended Recourse

In the second study, we seek to quantify the benefits of modeling more realistic recourse actions than simply returning to the depot when there is a failure. To do so, we compare the *implementation* cost of the route plans generated by three models:

1. the stochastic set-partitioning-based model with extended recourse (SP-ER),
2. the stochastic set-partitioning-based model with traditional recourse (SP-TR),
3. a deterministic VRP model that employs the expected customer demands.

The implementation cost of a route plan estimates the actual cost of the recourse actions that would be used in practice. In practice, dispatchers employ various strategies for covering failed routes after a failure occurs or if it appears likely that a failure will occur. In this study, we assume that dispatchers rely on optimization to determine a "best" course of action when routes fail. Therefore, to emulate dispatcher behavior, we solve a deterministic optimization model, called the *restoration model*, to select the best set of completions and extra trips for a given set of unserved customers, their realized demands, and the locations and remaining capacities of the vehicles. The restoration model SP-RM(x,s) is parameterized by a set of planned routes x and a demand scenario s. The actual cost of planned route r under demand scenario s is denoted as $c_r(s)$. After all the planned routes have been followed until they fail or reach the last customer, the vehicles have some remaining capacity and a set of customers remains unserved. The objective of the restoration model is to select a minimum cost set of completions and extra trips that satisfies the demands of the unserved customers and returns the vehicles to the depot. Table 9 lists the definition of sets, parameters, and variables for the restoration model SP-RM(x, s) and is followed by the model.

Table 9: Definition of Entities in the Restoration Model

Sets	
E	Set of extra trips.
$U(x,s)$	Set of unserved customers on route set x in scenario s.
$R(x)$	Set of vehicle routes taken by route plan x.
C_l	Set of completions for route l, $l \in R(x)$.
C	Total set of completions. $C \stackrel{\text{def}}{=} \cup_{l \in R(x)} C_l$
Parameters	
w_{ic}	1 if customer $i \in U(x,s)$ is covered by completion $c \in C$, 0 otherwise.
v_{ie}	1 if customer $i \in U(x,s)$ is covered on extra trip $e \in E$, 0 otherwise.
d_c	Cost of completion $c \in C$.
f_e	Cost of extra route $e \in E$.
k	Number of trucks.
Variables	
y_c	1 if completion $c \in C$ is selected, 0 otherwise.
z_e	1 if extra trip $e \in E$ is selected, 0 otherwise

$$z^*_{\text{SP-RM}}(x,s) \stackrel{\text{def}}{=} \min \sum_{c\in C} d_c y_c + \sum_{e\in E} f_e z_e \quad (\text{SP-RM}(x,s))$$

subject to

$$\sum_{c\in C} w_{ic} y_c + \sum_{e\in E} v_{ie} z_e = 1 \qquad \forall i \in U(x,s) \tag{14}$$

$$\sum_{c\in C_l} y_c = 1 \qquad \forall l \in R(x) \tag{15}$$

$$\sum_{e\in E} z_e \leq k \tag{16}$$

$$y_c \in \{0,1\} \qquad \forall c \in C \tag{17}$$

$$z_e \in \{0,1\} \qquad \forall e \in E \tag{18}$$

The objective function seeks to minimize the cost of the selected completions and extra trips. Constraints 14 require all unserved customers to be covered. Constraints 15 require each vehicle to do exactly one completion. To guarantee that all vehicles finish at the depot, we include as a possible completion a column that sends the vehicle directly to the depot. Constraint 16 limits the number of extra trips to at most the number of vehicles available. Constraints 17 and 18 are standard binary restrictions on the decision variables. The restoration model used to generate a plan to recover from disruptions explicitly uses completions and extra trips to emulate how we imagine intelligent dispatchers would actually behave. Of course, this may bias the results of the experiment in favor of a method that can account for such intelligent recourse in the initial planning stage. However, a fundamental goal of our research is to show that it is indeed possible to produces routes amenable to more intelligent and realistic recourse. One goal of our testing is aimed at verifying that this hypothesis is true and at quantifying the advantage that results.

To estimate the cost of a route plan $\hat{x}$ generated by one of the models, we use the following procedure. First, we randomly generate 1000 demand scenarios $s_1, s_2, \ldots, s_{1000}$. For each scenario s_j, $j = 1, \ldots, 1000$, the routes in $\hat{x}$ are followed until they are complete, fail, or are stopped. This yields a first stage cost of $c_r(s_j)^T\hat{x}$ and provides the necessary input to a restoration model SP-RM$(\hat{x}, s_j)$. This input includes the location and remaining capacity of each vehicle and the demands of customers not yet served. Next, SP-RM$(\hat{x}, s_j)$ is solved, and the estimated cost of the route plan $\hat{x}$ under demand scenario s_j is calculated as

$$c_r(s_j)^T\hat{x} + z^*_{\text{SP-RM}}(\hat{x}, s_j).$$

For a route plan $\hat{x}$, the *actual* implementation cost is then estimated as the average value over the scenarios,

$$\hat{\alpha}(\hat{x}) = 0.001 \sum_{j=1}^{1000} \left(c_r(s_j)^T\hat{x} + z^*_{\text{SP-RM}}(\hat{x}, s_j) \right). \tag{19}$$

The simulation procedure was run on a random subset of the 120 instances from test sets 1 and 2 described in Section 3.3. Recall that the instances in test set 2 tend to contain fewer customers on feasible routes and thus are more amenable to set-partitioning-based approaches. For each problem instance i in the reduced test sets $\mathcal{I}$, the three models VRP, SP-TR and SP-ER were solved, resulting in three sets of planned routes, denoted x^i_{VRP}, $x^i_{\text{SP-TR}}$, and $x^i_{\text{SP-ER}}$, respectively. In creating the set-partitioning-based models SP-TR and SP-ER, 3000 routes and 30 demand scenarios were used for instances from test set 1, while 6000 routes and 60 scenarios were used for instances from test set 2. These instances and sizes were chosen such that all instances could be solved within a time limit of eight hours for the test set 1 instances or 16 hours for the test set 2 instances. In all cases, only columns feasible in expectation were included, as discussed previously.

The simulation procedure for estimating implementation cost requires significant computation since 1000 restoration models must be solved for each route plan. Thus, if all three methods generate the exact same route plan, i.e. $x^i_{\text{VRP}} = x^i_{\text{SP-TR}} = x^i_{\text{SP-ER}}$, there is no need to estimate the implementation cost for instance i since it will be the same for each plan. Of the 63 instances in our reduced test set 1, 23 resulted in different route plans for each of the three methods. Of the 19 instances in the reduced test set 2, 13 resulted in different route plans. Our computational results are based solely on these instances and should be interpreted accordingly.

The results of the simulation experiment are nicely summarized using performance profiles, introduced by Dolan and Moré (2002). To that end, we let $\hat{\alpha}(x^i_m)$ be the actual implementation cost (19) for route plan x^i_m generated by "method" m on instance i, where the method m is one of VRP, SP-ER, or SP-TR. A performance profile is a measure of the relative quality of each of these methods. The ratio

$$\chi_{mi} \stackrel{\text{def}}{=} \frac{\hat{\alpha}(x^i_m)}{\min\{\hat{\alpha}(x^i_{\text{VRP}}), \hat{\alpha}(x^i_{\text{SP-TR}}), \hat{\alpha}(x^i_{\text{SP-ER}})\}}$$

is the factor by which method m is worse than the best method for solving instance i. The quantity

$$\rho_m(\tau) = |\{i \in \mathcal{I} \mid \chi_{mi} \leq \tau\}| / |\mathcal{I}|$$

is the fraction of instances for which the route plan produced by method $m \in$ { VRP, SP-ER, or SP-TR} was within a factor of τ of the best of the three. The performance profile for routing method m is a graph of $\rho_m(\tau)$. In general, the higher the graph of a solver, the better the relative performance. Figure 1 shows a performance profile of the three routing methods on the instances from test set 1, and Figure 2 shows a performance profile of the three routing methods on the instances from test set 2. In both cases, the routes produced by solving the model SP-ER seem to be of the highest quality when considering the actual cost of implementing the routes. For instances in the first set, SP-ER produced the lowest estimated cost for 15 of the 23 instances, SP-TR produced the lowest for 7 of 23 cases, and the deterministic VRP produced the lowest for 4 of 23 cases. (There were three ties). For instances in the second set, SP-ER produced the lowest estimated cost for 12 of the 13 instances, SP-TR produced the lowest for none of the 13, and the deterministic VRP produced the lowest for 1 of the 13. Table 10 compares the average percent difference in actual routing cost $\hat{\alpha}(\cdot)$ for the different route plans over the test sets of instances. An analysis of variance of the estimated actual routing costs shown in Table 10 reveals that the differences are highly statistically significant.

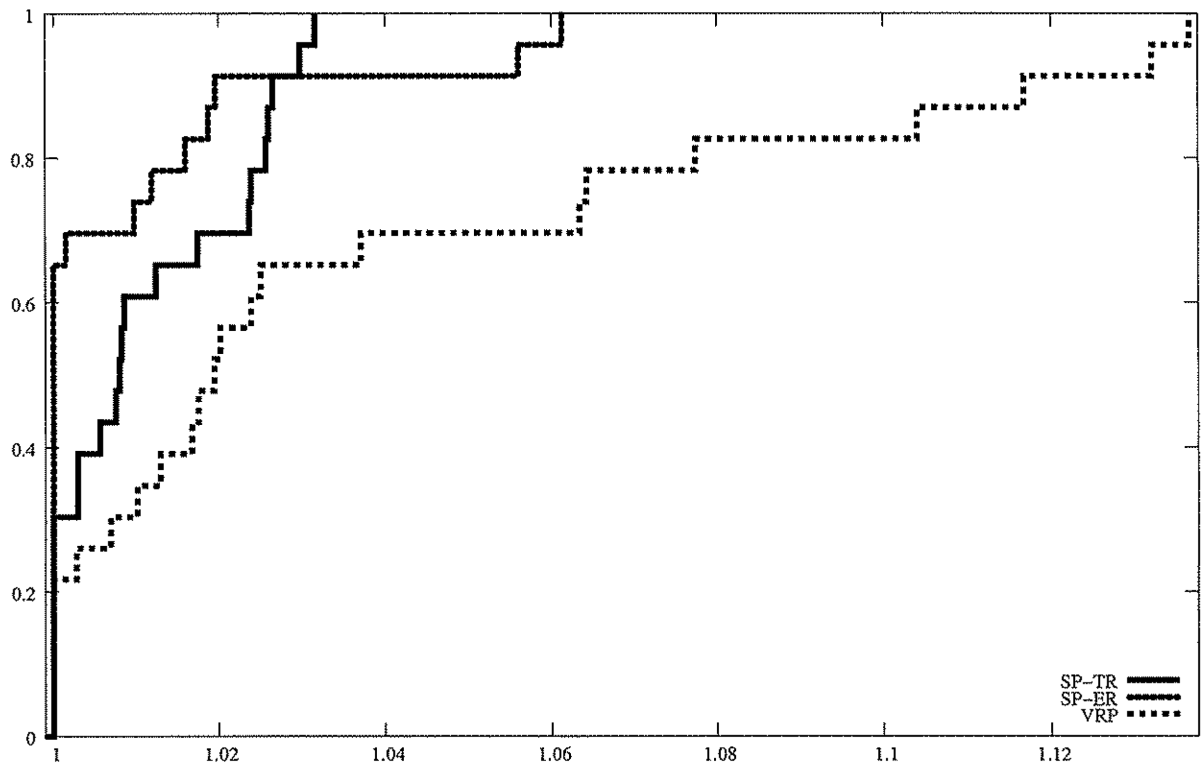

Figure 1: Performance Profile of Route Plan Actual Costs, Test Set #1

Table 10: Average Relative % Change in Cost

	Test Set #1	Test Set #2
% $x_{\text{SP-ER}}$ vs. x_{VRP}	-3.06%	-2.19%
% $x_{\text{SP-ER}}$ vs. $x_{\text{SP-TR}}$	-0.33%	-4.94%
% $x_{\text{SP-TR}}$ vs. x_{VRP}	-2.73%	2.74%

For the models SP-TR and SP-ER, Table 11 reports the average time in seconds required to generate the candidate routes, and Table 12 reports the average time in seconds required for

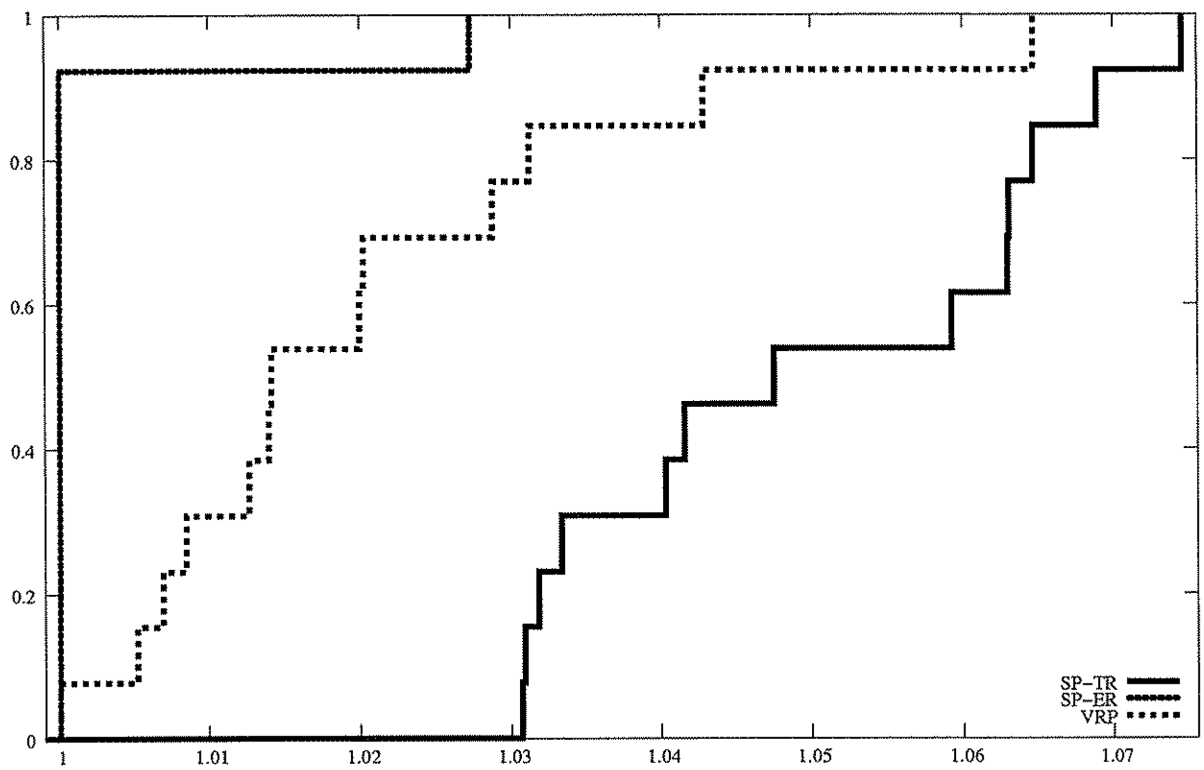

Figure 2: Performance Profile of Route Plan Actual Costs, Test Set #2

CPLEX v9.1 to solve instances of the models on a Intel Pentium III CPU, running at 1133MHz. For the extended recourse model, the time to generate candidate routes is negligible compared to the time required to solve an instance. However, since the *a priori* route plan can be computed several hours ahead of route execution, the solution time may not be a significant disadvantage for SP-ER. Further, the model SP-ER has significant structure that can be exploited by a specialized decomposition algorithm.

Table 11: Avg. Time (CPU Seconds) to Generate Columns and Parameters for the SP Models

	Test Set 1		Test Set 2	
n	SP-ER	SP-TR	SP-ER	SP-TR
5	0.13	0.06	N/A	N/A
8	0.28	0.10	0.26	0.17
20	4.98	0.70	1.26	0.36
30	19.03	2.02	4.59	0.47
40	38.07	4.32	9.34	0.64

Table 12: Avg. Time (CPU Seconds) to Solve the SP Models

	Test Set 1		Test Set 2	
n	$x_{\text{SP-ER}}$	$x_{\text{SP-TR}}$	$x_{\text{SP-ER}}$	$x_{\text{SP-TR}}$
5	0.15	0.06	N/A	N/A
8	3.27	0.06	1.66	0.01
20	10827.79	0.40	239.05	0.01
30	8096.82	1.61	16154.06	0.01
40	5569.34	1.56	32631.01	0.01

4.3 Sample Average Approximation

In most practical problems involving uncertainty, the total number of scenarios ($|S|$) is far too large to consider using them all. The SVRP is no exception, and for reasons of tractability, the model SP-ER contains a very small subset of the scenarios. In this section, we quantify the impact of considering only a limited number of scenarios by computing statistically valid bounds on the optimal objective function value of the model SP-ER for two of our test instances. A number of other authors also have suggested recently that using a limited number of scenarios via a *sample average* or *sample-path* approach can be an extremely effective technique for solving two-stage stochastic problems with a discrete structure (Shapiro and Homem-de-Mello, 2000, Kleywegt et al., 2001, Verweij et al., 2003, Linderoth et al., 2006).

In the sample-average method for solving SP-ER, instead of including all scenarios S, a random subset $S' \subset S$ with $|S'| = N$ is used. The optimal solution value of SP-ER is denoted as $z^*_{\text{SP-ER}}$, and the optimal solution value of its sampled approximation, using a random sample size of N, is denoted as $z^*_{\text{SP-ER}}(N)$. The value $z^*_{\text{SP-ER}}(N)$ is a random variable, as it depends on the random subset S'. However, it is well-known (e.g., Mak et al. (1999)) that the expected value of the random variable $z^*_{\text{SP-ER}}(N)$ provides a biased estimate of $z^*_{\text{SP-ER}}$, i.e.:

$$\mathbb{E}[z^*_{\text{SP-ER}}(N)] \leq z^*_{\text{SP-ER}}.$$

In order to estimate $\mathbb{E}[z^*_{\text{SP-ER}}(N)]$, M independent samples $S'_1, S'_2, \dots S'_{|M|}$, each of size N, are taken. For each sample S'_j, the approximating problem SP-ER is solved to obtain an optimal objective value ℓ_j. The average value of the ℓ_j provides an unbiased estimate of $\mathbb{E}[z^*_{\text{SP-ER}}(N)]$ and thus a statistical lower bound on the true optimal objective value $z^*_{\text{SP-ER}}$. Using the values ℓ_j, a confidence interval on the estimate of $\mathbb{E}[z^*_{\text{SP-ER}}(N)]$ also can be constructed as in Mak et al. (1999).

A statistical upper bound on $z^*_{\text{SP-ER}}$ can be obtained by estimating the *actual* routing cost for any fixed, feasible set of routes $\hat{x}$. Computing the routing cost for the solution $\hat{x}$ is equivalent to solving the problem SP-ER with the additional constraint $x = \hat{x}$, so the routing cost $\alpha(\hat{x})$ can be computed by solving the restoration model SP-RM for each scenario, namely

$$\alpha(\hat{x}) = \sum_{r \in R} c_r \hat{x}_r + \sum_{s \in S} p_s z^*_{\text{SP-RM}}(\hat{x}, s). \tag{20}$$

In Equation (20), $\alpha(\hat{x})$ is the actual cost of implementing routes $\hat{x}$ and $z^*_{\text{SP-RM}}(\hat{x}, s)$ is the optimal solution value of the restoration model SP-RM$(\hat{x}, s)$. The value $\alpha(\hat{x})$ provides an upper bound on $z^*_{\text{SP-ER}}$, but $|S|$ is too large to consider computing $\alpha(\hat{x})$ exactly; instead, the upper bound is estimated by considering only a random subset of the scenarios $T \subset S$. The value $\alpha(\hat{x})$ is then estimated as

$$\hat{\alpha}(\hat{x}) = \sum_{r \in R} c_r \hat{x}_r + \sum_{s \in T} |T|^{-1} z^*_{\text{SP-RM}}(\hat{x}, s).$$

For each of the M solutions $x^*_1, x^*_2, \dots x^*_{|M|}$ to the sampled approximations used in obtaining a lower bound, the value $\hat{\alpha}(\hat{x}^*_j), j = 1, \dots, |M|$, is computed. The average value of the $\hat{\alpha}(\hat{x}^*_j)$ provides a statistical upper bound on $z^*_{\text{SP-ER}}$. Further, a confidence interval on the upper bound can be computed as in Freimer et al. (2005).

Table 13 shows the statistical lower and upper bounds for $z^*_{\text{SP-ER}}$ for two randomly selected instances from our test set. To obtain these estimates, sample average instances of size $N = 60$ scenarios were used, and either $M = 29$ or $M = 16$ total sample average instances were solved. In computing the upper bounds, samples of size $|T| = 501$ were used to compute

$\hat{\alpha}(\hat{x}_j^*)$ for each solution x_j^*. The bounds in Table 13 indicate that the routes obtained from the model SP-ER are of very high quality, even when using an extremely small number of samples ($N = 60$). The solution values are within 2% of the true optimal solution.

Table 13: Bounds on the Optimal Solution Value $z^*_{\text{SP-ER}}$

Instance Name	N	M	Lower Bound	95% C.I	Upper Bound	95% C.I
set1-20r3-16	60	29	16.16	(16.0541, 16.2659)	16.4879	(16.4871, 16.4885)
set2-30r4-18	60	16	30.40	(30.3208, 30.4792)	30.8874	(30.8868, 30.8880)

5 Conclusions and Future Research

The main contributions of this paper are the development of set-partitioning-based models for the VRPSD and the development of a computational approach for solving the VRPSD under a new and more realistic recourse strategy. The set-partitioning-based models can be adapted easily for routing problems with randomness in other problem elements, such as random customers and random travel times. The main disadvantage of the set-partitioning-based models is the need to explicitly generate a large set of routes to obtain good solutions. For the VRPSD models studied here, we describe an effective heuristic method for generating routes and demonstrate empirically that the heuristic produces competitive solutions. For the VRPSD under a traditional recourse strategy in which, upon failure, the vehicle returns to the depot, unloads/replenishes and then returns to the point of failure, we compared the results of our approach to those previously published in the literature. For the Christofides and Eilon (1969) 75-customer instance, our model required just 3.4 and 13.76 minutes to produce two solutions with lower expected costs, 0.4% and 0.22%, than the best published results (Savelsbergh and Goetschalckx, 1995).

To better represent the range of options available to dispatchers when they are designing recovery routes, we introduce the extended recourse strategy; the extended recourse strategy allows a vehicle to serve additional customers at the end of its route before returning to the depot, called *a completion*, or to serve customers on a new route after returning to the depot, called *an extra trip*. A stochastic extension of the set-partitioning model allows us to easily model this new recourse strategy. We use a simulation procedure to quantify the benefits of modeling these more realistic recourse actions. An auxiliary optimization problem is solved to determine the "actual" recourse actions that would be implemented. Computational tests show that if vehicles have small capacities (i.e., about 1-3 customers per route), the new recourse policy performs close to 5% better than the traditional one as a result of the few unsatisfied customers per route being served by completions. We expect that this new recourse strategy also will provide lower cost solutions in cases where there is a significant fixed cost each time a vehicle goes to the depot and then resumes its route. If the vehicles have larger capacities and hence a large number of unsatisfied customers per route, the new recourse policy does not significantly outperform the traditional strategy. However, in either case, the stochastic model produces up to 3.06% lower cost solutions than those produced by a deterministic model in which it is assumed that the demand is equal to the mean.

For the size of the problems studied in this paper, the route generation heuristic provided good solutions and instances of the model can be solved relatively quickly. As the size of the problems grows, however, the large set of columns required to obtain good solutions will make the explicit approach computationally intractable. A topic for future research is to investigate column generation and branch-and-price approaches for solving the set-partitioning-based models proposed here. Another area to investigate is exploiting the decomposable structure of the set-partitioning-based model for the extended recourse, which would allow us to solve the recourse problem independently and in parallel for each scenario. A final topic is to investigate the quality of the routes if the integrality restrictions on the recourse variables are removed.

Acknowledgement

This work was supported in part by the US Department of Energy under grant DE-FG02-05ER25694 and by the National Science Foundation under grant CMMI-0522796. Computational resources at Lehigh are provided in part by equipment purchased by the NSF through the IGERT Grant DGE-9972780.

References

Ak, A., A. Erera. 2006. A paired-vehicle recourse strategy for the vehicle routing problem with stochastic demands. *Transportation Science,* To appear.

Bertsimas, D. J. 1992. A vehicle routing problem with stochastic demand. *Operations Research* **40**(3) 574–585.

Bertsimas, D. J., P. Chervi, M. Peterson. 1995. Computational approaches to stochastic vehicle routing problems. *Transportation Science* **29**(4) 342–352.

Bianchi, L., M. Birattari, M. Chiarandini, M. Manfrin, M. Mastrolilli, L. Paquete, O. Rossi-Doria, T. Schiavinotto. 2004. Metaheuristics for the vehicle routing problem with stochastic demands. *Parallel Problem Solving from Nature - PPSN VIII, LNCS*, vol. 3242. Springer-Verlag, Birmingham, UK, 450–459.

Chepuri, K., T. Homem de Mello. 2005. Solving the vehicle routing problem with stochastic demands using the cross entropy method. *Annals of Operations Research* **134** 153–181.

Christofides, N., S. Eilon. 1969. An algorithm for the vehicle dispatching problem. *Operational Research Quarterly* 309–318.

Clarke, G., J.W. Wright. 1964. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research* **12** 568–581.

Cullen, F. H., J. J. Jarvis, H. D. Ratliff. 1981. Set partitioning based heuristic for interactive routing. *Networks* **11** 125–144.

Desrosiers, J., F. Soumis, M. Desrochers. 1984. Routing with time windows by column generation. *Networks* **14** 545–565.

Dessouky, M. M., F. Ordónez, H. Jia, Z. Shen. 2005. *Patient Flow: Reducing Delay Management in Health Care Systems*, chap. 15. 1st ed. Springer.

Dolan, E., J. Moré. 2002. Benchmarking optimization software with performance profiles. *Mathematical Programming* **91** 201–213.

Dorigo, M., L.M. Gambardella. 1997. Ant colonies for the traveling salesman problem. *Bio Systems* **43** 73–81.

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

Dror, M., M. Ball, B. Golden. 1985. A computational comparison of algorithms for the inventory routing problem. *Annals of Operations Research* **6** 3–23.

Dror, M., G. Laporte, P. Trudeau. 1989. Vehicle routing with stochastic demands: Properties and solution frameworks. *Transportation Science* **23**(3) 166–176.

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

Freimer, M., D. Thomas, J. Linderoth. 2005. Reducing bias in stochastic linear programming with sampling methods. Technical Report 05T-002, Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, Pennsylvania.

Gagné, C., W.L. Price, M. Gravel. 2001. Scheduling a single machine with sequence dependent setup time using ant colony optimization. Tech. rep., Université Laval, Quebec, Canada.

Gendreau, M., G. Laporte, R. Séguin. 1995. An exact algorithm for the vehicle routing problem with stochastic demands and customers. *Transportation Science* **29**(2) 143–155.

Gendreau, M., G. Laporte, R. Séguin. 1996. A tabu search heuristic for the vehicle routing problem with stochastic demands and customers. *Operations Research* **44**(3) 469–477.

Hjorring, C., J. Holt. 1999. New optimality cuts for a single-vehicle stochastic routing problem. *Annals of Operations Research* **86** 569–584.

Kleywegt, A., A. Shapiro, T. Homem-de-Mello. 2001. The sample average approximation method for stochastic discrete optimization. *SIAM Journal on Optimization* **2** 479–502.

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

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

Laporte, G., F. V. Louveaux. 1998. *in Fleet Management and Logistics*. 1st ed. Kluwer, Boston, MA, 159–167.

Laporte, G., F. V. Louveaux, L. Van Hamme. 2002. An integer L-Shaped algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research* **50**(3) 415–423.

Linderoth, J. T., A. Shapiro, S. J. Wright. 2006. The empirical behavior of sampling methods for stochastic programming. *Annals of Operations Research* **142** 219–245.

Mak, W. K., D. P. Morton, R. K. Wood. 1999. Monte carlo bounding techniques for determining solution quality in stochastic programs. *Operations Research Letters* **24** 47–56.

Marković, H., I. Ćavar, T. Carić. 2005. Using data mining to forecast uncertain demands in stochastic vehicle routing problem. Presented at 13th International Symposium on Elecronics in Transport (ISEP), Slovenia, 2005.

Novoa, C. 2005. Static and dynamic approaches for solving the vehicle routing problem with stochastic demands. Ph.d. dissertation, Lehigh University, Industrial and Systems Engineering Department, Bethlehem, PA.

Novoa, C., R. Storer. 2005. The vehicle routing problem with stochastic demands combining static and dynamic approaches. *Proceedings of the Institute of Industrial Engineers (IIE) Annual Conference, Orlando, Fl.* Paper 1525, Pages 1–6.

Psaraftis, H. 1995. Dynamic vehicle routing: Status and prospects. *Annals of Operations Research* **61** 143–164.

Ralphs, T.K., L. Kopman, W.R. Pulleyblank, L.E. Trotter Jr. 2003. *On the Capacitated Vehicle Routing Problem*, vol. 94. Springer-Verlag, Heidelberg, Germany,, 343–359.

Savelsbergh, M., M. Goetschalckx. 1995. A comparison of the efficiency of fixed versus variable vehicle routes. *Journal of Business Logistics* **16** 163–188.

Secomandi, N. 2001. A rollout policy for the vehicle routing problem with stochastic demands. *Operations Research* **49**(5) 796–802.

Shapiro, A., T. Homem-de-Mello. 2000. On the rate of convergence of optimal solutions of Monte Carlo approximations of stochastic programs. *SIAM Journal on Optimization* **11** 70–86.

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

Teodorovic, D., G. Pavkovic. 1992. A simulated annealing technique approach to the vehicle routing problem in the case of stochastic demand. *Transportation Planning and Technology* **16** 261–273.

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

Verweij, B., S. Ahmed, A. Kleywegt, G. Nemhauser, A. Shapiro. 2003. The sample average approximation method applied to stochastic routing problems: A computational study. *Computational Optimization and Applications* **24** 289–333.

Yang, W.-H., K. Mathur, R. H. Ballou. 2000. Stochastic vehicle routing problem with restocking. *Transportation Science* **34**(1) 99–112.