Proceedings of the 6th World Congress on Intelligent Control
and Automation, June 21 - 23, 2006, Dalian, China

A Multiple Vehicles Routing Problem Algorithm with Stochastic Demand*

Jianhua Fan and Xiufeng Wang
School of Information Technology and Science
Nankai University
Tianjin, 300071, China
fan109@163.com & wangxf@nankai.edu.cn

Jianhua Fan and Hongyun Ning
Department of Computer Science and Engineering
Tianjin University of Technology
Tianjin, 300191, China
ninghongyun@eyou.com

***Abstract* - A heuristic algorithm for multiple vehicles routing problem with stochastic demand is proposed and the goal is to minimize the total traveling cost. Two-phase method is adopted to deal with this problem. In the first phase, an algorithm is proposed to partition customers into clusters, and the main task of the second phase is to design an effective routing through each cluster of customers to minimize the total expected traveling cost. Both the a priori strategy and the reoptimization strategy are used to obtain the optimal routing. The experiment results indicate that this method can produce solutions of good quality and is an effective algorithm for the multiple vehicles routing problem with stochastic demand.**

Keywords—VRPSD, stochastic vehicle routing problem, reoptimization, multiple vehicles routing

I. Introduction

The Vehicle Routing Problem (VRP) has been extensively studied by operations researchers since 1959. In the classical deterministic VRP, a number of vehicles are located at a single depot and must serve a number of geographically dispersed customers. Each vehicle has a given capacity and each customer has a given demand. The objective is to minimize the total traveling cost for servicing the customers. Each vehicle starts at the depot and collects (delivers) goods from (to) a subset of the customers, fully satisfying the demand of each customer it visits, and then returns to the depot. The route that each vehicle is assigned must satisfy a number of constraints, such as vehicles capacity, time windows, driver's maximum working time and etc. The VRP is a complex combinatorial optimization problem. It can rarely be solved to optimality for sizes in excess of n=50. Many algorithms have been developed to deal with it in [1][2][3].

The classical VRP statement does not capture an important aspect of real-life distribution problems, namely that several of the problem parameters (service time, customer locations, demands, etc.) are not known with certainty. These give rise to Stochastic Vehicle Routing Problems (SVRPs). These problems have received increasing attention in recent years. Until now much less effort has been devoted to the study of SVRPs as opposed to their deterministic counterparts. This is due to the fact that when stochastic elements are introduced into this difficult combinatorial problem, it becomes in general even less tractable for analytical treatment. Several versions can be defined according to the nature of the stochastic components and to the nature of the constraints and objective function.

This problem appears in many practical situations. For example, a central bank has to collect money from several but not all of its branches every day. The capacity Q of the vehicle used may not correspond to any physical constraint but to an upper bound on the amount of money that a vehicle might carry for safety reasons. The distribution of demand at each certain branch may be different, associated with the amount of money it handles. Similarly, the distribution of packages from a post office can be modeled as another PVRP, where the probability that a certain location requires a visit is given and the capacity Q corresponds to the physical volume that a truck can carry.

In this paper, we consider the SVRP where only the customer demand is stochastic and all other parameters are deterministic, namely the Vehicle Routing Problem with Stochastic Demand (VRPSD). In the VRPSD one vehicle of finite capacity is leaving from a depot with full load and has to serve a set of customers whose exact demand is only known on arrival at the each customer location. VRPSD problems can be divided into two categories: single vehicle routing problem and multiple vehicles routing problem. A majority of literature for VRPSD study the single vehicle routing problem because of more tractability. The literatures for the multiple vehicle routing problems are seen rarely in VRPSD. The purpose of this paper is to propose a heuristic algorithm to deal with the multiple vehicles routing problems of VRPSD optimally.

The remainder of the paper is organized as follows. In section II we formally define the scope of the problem under consideration and briefly depict the main approaches of VRPSD currently. Section III gives the approximate computation of expected traveling cost in preparation for algorithm realization, whereas the heuristic algorithm of the multiple vehicles routing problem is developed in Section IV. Computational results are given in Section V and conclusions are drawn in Section VI.

II. VRPSD and Related Literature

The Vehicle Routing Problem with Stochastic Demand

* This work was supported by the National Natural Science Foundation of China (No.70572045) to Wang Xiufeng., the Education Research Council of Tianjin (No.20030618), the Science and Technology Research Council of Tianjin (No. 043600511) to Ning Hongyun.

1-4244-0332-4/06/$20.00 ©2006 IEEE

(VRPSD) is defined on a complete graph G= (V, A, D), where:

V= {0, 1, …, n} is a set of nodes (customers) with node 0 denoting the depot, and node 1, 2, …, n corresponding to the customers.

A={(i, j): i, j $\in$V, i$\neq$j} is the set of arcs joining the nodes.

D={d_{ij} : i, j $\in$V, i$\neq$j} are the traveling costs (distances) between nodes i and j.

We assume the cost matrix D is symmetric and satisfies the triangular inequality. One vehicle with capacity Q has to deliver goods to the customers according to their demands, minimizing the total expected distance traveled. All customers have stochastic demands ξ_i , i=1, 2, …, n, which is independently distributed with known probability distributions. The actual demand of each customer is only known when the vehicle arrives at the customer location. It is also assumed that ξ_i does not exceed the vehicle's capacity Q, and follows a discrete probability distribution P_{ik}= Prob (ξ_i =k), k=0, 1, 2, …, K$\leqslant$Q.

Up to now in the literature, there are two distinct strategies for VRPSD basically.

A. Reoptimization strategy:

An obvious approach to the VRPSD is to redesign the routes when the demand becomes known. Throughout the paper, we call this approach the reoptimazation strategy, which is described in [4][5]. This approach attempts to optimally solve (or near-optimally with a good heuristic) every potential instance of the original problem. Reoptimization strategy has the potential of yielding higher quality solution. There are, however, several difficulties with this approach. For example, if the combinatorial optimization problem considered is NP-hard, one might have to solve exponentially many instances of a hard problem. When the problem size is very large, it may be that such redesign of tours is not sufficiently important to justify the required effort and cost. So reoptimization strategy fits into dealing with the small size problems that can obtain the optimal solution.

Relevant solutions are discussed by Bastian and Rinnooy Kan in [6] and small-sized instances solutions can be found in [7]. Secomandi propose a rollout policy for the single vehicle routing problem with stochastic demands in [8], which is the first computationally tractable algorithm for approximately solving the problem under the reoptimization approach and can be applied to the problem with large size. Reoptimization algorithms for VRPSD are rare in the literature.

B. A priori optimization strategy

Instead of redesigning the routes every day, the other strategy is: determine a fixed a priori sequence among all the potential customers in advance and then update in a simple way this a priori solution to answer each particular instance. It includes two stages. In the first stage, a set of routes satisfying some conditions are determined. In the second stage, the first stage routes are followed as planned except that the vehicle capacity becomes exceeded, it returns to the depot to load, and deliver goods starting at the last visited customer. So the costs under consideration are:

1) Cost of traveling from one customer to another as planned.

2) Restocking cost—the cost of traveling back to the depot for restocking

3) The cost of returning to depot for restocking caused by the remaining stock in the vehicle being insufficient to satisfy demand upon arrival at a customer location.

The idea of using an a priori sequence for the solution of VRPs was first introduced in the PH. D thesis of Jaillet in [9]. Because of the feasible computation cost, there are a lot of algorithms for a priori optimization of VRPSD as in [10][11][12][13]. However, this approach needs some improvements such as Or-opt and 2-opt to enhance the solution quality, which also produce much computation.

The goal of this paper is to study the multiple vehicles routing problem, in which we use both the reoptimization strategy and the a priori optimization strategy. The multiple vehicles routing problem of this paper is treated as a two-phase method. In the first phase, clustering of customers is solved by a heuristic algorithm with the constraint of maximum demand of each route. In the second phase, in order to save computation, above all, we use the heuristic algorithms for deterministic VRP to construct an initial route sequence (a priori optimization route) through each customer cluster to minimize the total distance traveled neglecting of stochastic demand. Because the number of each cluster nodes is small, the route sequence can be rearranged by employing a computationally tractable reoptimaization approach called rollout algorithm taking into account the stochastic demand, which can deal with each route to optimality effectively. Secomandi in [8] first introduce the rollout policy into the VRPSD and obtain preferable performance, but he only concerns the single vehicle routing case. In this paper, we consider applying it into the multiple vehicles case. So we consider the computing of expected traveling cost first of all.

III. The Expected Traveling Cost Computation

In order to evaluate a particular route, we need an effective method to compute the expected distance traveled by the vehicle (that is the objective function). In this paper, we use the following computation. Let s= (0, 1, …, n) be an a priori optimization routing. After the service completion at customer j, suppose the vehicle has a remaining load q, and let $f_j(q)$ denote the total expected cost from node j onward. With this notation, the expected cost of the a priori tour is $f_0(Q)$. If S_j represents the set of all possible loads that a vehicle can have after service completion at customer j, then $f_j(q)$ for q $\in S_j$ satisfies the dynamic programming recursion,

$$f_j(q) = \min\{f_j^{\,p}(q), f_j^{\,r}(q)\} \tag{1}$$

$$\begin{aligned} f_j^{\,p}(q) &= d_{j,j+1} + \sum_{k:k\leq q} f_{j+1}(q-k)p_{j+1,k} \\ &+ \sum_{k:k\succ q}[2d_{j+1,0} + f_{j+1}(q+Q-k)]p_{j+1,k} \end{aligned} \tag{2}$$

$$f_j^{\,r}(q) = d_{j,0} + d_{0,j+1} + \sum_{k=1}^{K} f_{j+1}(Q-k)p_{j+1,k} \tag{3}$$

With the boundary condition

$$f_n(q) = d_{n,0} \qquad q \in S_n \tag{4}$$

In (2)-(3), $f_j^{\,p}(q)$ represents the expected cost corresponding to going directly to the next customer, whereas $f_j^{\,r}(q)$ represents the expected cost in case preventive restocking is chosen. The above dynamic programming process is needed to determine the optimal policy recursively. As shown by Yang et al in [12], given the a priori tour, for each customer j there is a load threshold h_j such that, if the residual load after serving j is greater than or equal to h_j, then it is better to proceed to the next planned customer, otherwise it is better to go back to the depot for preventive restocking.

IV. MULTIPLE VEHICLES ROUTING

In the vehicle routing problems, a single route will always be optimal even when the total customer demand exceeds vehicle capacity due to the recourse policy of restocking the vehicle, which is proved by W. Yang in [12]. But in practical applications, the idea of multiple vehicles routing is usually motivated by the drivers' need to be acquainted with their area and customer base. In addition to that, the drivers' work time and each vehicle driving mileages is other considering factor. The purpose of multiple vehicles routing is to design a set of routes to minimize the total expected traveling cost under some constraints. In the context of our algorithm, we assume that the maximum demand of each route is less than or equal to two vehicle capacities, say C, so that the number of routing failures is at most one. In this section, we first partition customers into clusters (the first phase), then construct an optimal routing through each cluster to minimize the expected traveling cost using rollout algorithm (the second phase).

A. Clustering of Customers

Motivated by the cone covering method proposed by Fisher and Jaikumar in [14], we propose the radiation approach described as follows: draw a radial outward from depot as the initial location, and move from initial radial clockwise until the total expected demand covered by the radial area is less than or equal to C, but adding the next closest customer would violate that property. This is the first customer cluster. Iteratively take the step above from predecessor until all customers are covered. The set of customer clusters is formed, named R= $\{r_1, r_2, \ldots, r_l\}$. The procedure of clustering of customers is depicted in Fig.1.

B. Routing through Clusters

Unlike the deterministic VRP problem, in the context of the VRP with stochastic demands the computation time increases exponentially due to the introduction of a more complicated and computationally intensive cost function. We adopt two-phase method to implement the routing through cluster that does not require prohibitive computation time. In the first phase, a deterministic heuristic VRP algorithm is applied to construct an initial set of routes that minimizes the total distance traveled with respect to the mean customer demands. There are many solutions for it, such as saving algorithm by Clark-Wright, Tabu search, GA, SA and so on.

In the second phase, a computationally tractable heuristic algorithm for computing a reoptimization-type routing policy for the single vehicle case of VRPSD is applied to improve the current set of routes with respect to the objective function. This is accomplished by employing a rollout algorithm to route the vehicle through the customers concurrently with service.

C. Rollout Algorithm

Rollout algorithms were first proposed for the approximate solution of discrete optimization problems by Bertsekas and Tsitsklis in [15], suggesting that if an already good policy for the problem at hand is known (a heuristic policy), then it can be "rolled out" when computing controls so that the resulting decisions are improved ones over those belonging to the base policy. Rollout algorithm is a class of suboptimal solution methods inspired by the policy iteration methodology of Dynamic Programming (DP for short) and the approximate policy iteration methodology of neuro-dynamic programming (NDP for short). One may view a rollout algorithm as a single step of the classical policy iteration method, starting from some given easily implemental policy. Algorithms of this type have been sporadically proposed in several DP application contexts. The rollout algorithm has been applied to combinatorial optimization problems in [16], stochastic scheduling problems in [17], shortest path, assignment, matching, etc. The problem is described as follows.

Given a finite set U of feasible solutions and a cost function g(u). Each solution u has the form u= $(u_1, u_2,\ldots, u_N)$ with N components. Our purpose is to find a solution $u \in U$ that minimizes g(u). The soul of the rollout algorithm is to redefine the component sequence v= $(v_1, v_2,\ldots, v_N)$ on the basis of the initial solution u=$(u_1, u_2,\ldots, u_N)$ to minimize the g (u). We can view the problem as sequential decision problem, whereby the components $u_1, u_2,\ldots, u_N$ are selected one-at-a-time. When the first n components of solution are confirmed, we say that it is in the nth stage. The initial state is

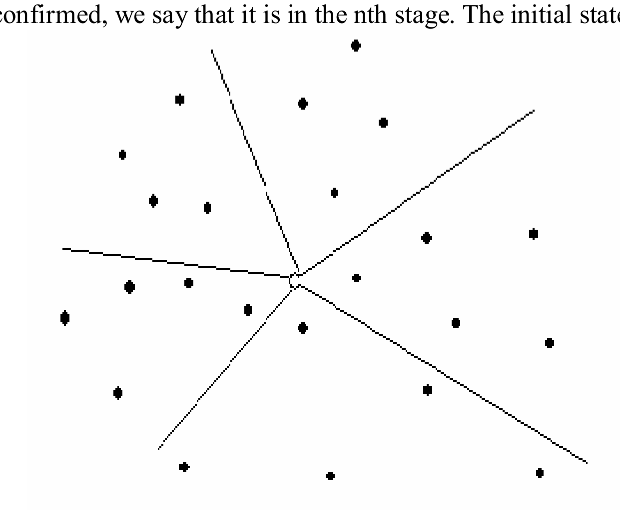

Fig. 1 Clustering of customers

a dummy state. From this state, the first component v_1 of v belong to the set

$U_1=\{u_1' \mid$ there exists a solution form $(u_1', u_2', \ldots, u_N') \in U\}$

More generally, the nth component v_n belong to the set

$U_n=\{u_n' \mid$ there exists a solution form $(v_1, v_2, \ldots, v_{n-1}, u_n', \ldots, u_N') \in U\}$

Let $J(v_1, v_2, \ldots, v_n)$ be the optimal cost starting from the state$(v_1,v_2,\ldots, v_n)$, the selection of v_n lies on the following algorithm:

$$v_n = \arg\min_{u_n \in U_n} J(v_1, v_2, \ldots, u_n), n = 1, \ldots, N$$

By a sequence of N single component minimizations we can obtain an optimal solution v. The key point is that the J (…) should be efficiently computable.

D. *A Rollout Policy for VRPSD*

Given a priori tour T=(0, 1, … , n, 0) through the customers. The T can be constructed by deterministic VRP approaches that are relatively simple and can save much computation time. Assume that the vehicle visits customers according to T by returning to the depot when route failures occurs. We define by $T^{\&}_m$ the cyclic tour algorithm that starts at location m, follows T cyclically and terminates when the next location in the nodes sequence has been visited (already fully served customers are assumed to be skipped).

$$T^{\&}_m = (0, m, m+1, \ldots, n, 1, \ldots, m-1, 0)$$

The expected length of $T^{\&}_m$ at location m can be interpreted as the cost-to-go at location m following the $T^{\&}_m$, i.e., the expected distance to termination given that the system is at location m at present. The expected cost-to-go at location m can be computed using (1)-(3) recursively, until the last location having not been visited. So the first customer m_1 to be visited can be obtained by the following minimization:

$$m_1 = \arg\min_{m \in \{1,\ldots,n\}} f_{0 \otimes T^{\&}_m}(Q) \tag{5}$$

$$T^{\&}_m = (0, m, m+1, \ldots, n, 1, \ldots, m-1, 0)$$

$f_{0\otimes T^{\&}_m}(Q)$ is the expected cost of route $T^{\&}_m$. The demand of customer m_1 is fully served and capacity is updated. The vehicle moves to location m_1 and need to compute which location to arrive at next. Generally, assume that current location is m, and let N (m) the set of nodes that still need to be visited. Then the next location to be visited is computed as follows. Let

$$\begin{aligned} J^0(m) = \min_{j \in N(m)} \{ & d_{m,j} + \sum_{k:k \le q} f_j(q-k) p_{j,k} \\ & + \sum_{k:k \succ q} [2d_{j,0} + f_j(q+Q-k)] p_{j,k} \} \end{aligned} \tag{6}$$

$$J^1(m) = \min_{j \in N(m)} \{ d_{m,0} + d_{0,j} + \sum_{k=1}^{K} f_j(Q-k) p_{j,k} \} \tag{7}$$

$J^0(m)$ in (6) above-mentioned is the cost-to-go corresponding to moving the vehicle directly to customer location j from m and subjecting to the nodes visiting sequence in $T^{\&}_m$. While $J^1(m)$ in (7) is the cost-to-go of that the vehicle visit depot before actually reaching location j and following the nodes sequence in $T^{\&}_m$. This means that vehicle reach the new node j with full capacity. When $J^0(m) < J^1(m)$, the vehicle move to the next location j directly; otherwise to the depot before reaching j. The process is repeated until reaching the last node having not been visited. After the last node having been visited, the vehicle returns to the depot.

V. Computational Experiments

To our knowledge, no benchmark instances exist for this class of problem in this paper. We adopt the method of Yang in [12] that all the problems are randomly generated in this study. The customer locations are uniformly generated in a 100×100 square with the depot located at coordinates (50, 50).

Without loss of generality, the cost of travel is $1 per unit distance so that the distances are equal to the traveling cost, and the vehicle capacity Q is assumed to be 1. So the demands of each customer are defined as a fraction of the vehicle capacity and the demand distributions follow a discrete distribution with (0, 0.1, 0.2, …, 0.9, 1).

To prove effectiveness of the algorithm in this paper, we compare it with the other two approaches.

A. *Comparison with A Priori Strategy for Each Routeing*

A priori strategy for each route is to construct a priori sequence for each cluster customers instead of reoptimizing. Here we use the well-known Clarke-Wright savings algorithm to construct a priori route under stochastic demand. We test problems ranging in size from 50 to 200 customers for both heuristic algorithms. Table I summarize the computational results. The last column is the total saving cost percent of algorithm in this paper compared with the a priori sequence. From table I we can conclude that our algorithm of reoptimization on the basis of a priori outperform the simple a priori algorithm. The total traveling cost can get much saving.

B. *Comparison with A Deterministic Approach*

Another alternative method is to use deterministic counterpart to deal with stochastic demands in the absence of an effective stochastic procedure. In this situation, a route can be constructed by treating the expected demand at each customer as its deterministic demand. When the vehicle runs out of the load, it returns to the depot for replenishment.

The main advantage of this method is its simplicity to computation. Compared with the stochastic instance, it can save much time. So in the situation of importance of rapid solution excess the solution quality, this method is a relative good alternative. But the result of this deterministic algorithm is dissatisfying compared with the algorithm with stochastic factor. Table II indicates that the algorithm in this paper produce significant cost savings. The larger the number of nodes, the greater the saving obtained by our algorithm. Therefore the deterministic algorithm is only applicable in the small size problems.

VI. CONCLUSION

This paper presents a heuristic algorithm for multiple vehicles routing problem with stochastic demand. We use the two-phase method to solve this complicated combinatorial optimal problem. In the first phase, we propose an effective heuristic algorithm for clustering of customers. In the second phase, we use deterministic VRP algorithm to construct an initial rouging for each cluster, and redefine the customers sequence by an improved reoptimization-type policy called rollout algorithm. The experiment results indicate encouraging performance. One limitation of the present approach is that it does not take into account other constraints of routing, such as driver's work time and each vehicle driving mileages, which are the topics for further research.

ACKNOWLEDGMENT

This work was supported by the National Nature Science Foundation of China (No.70572045) , the Education Research Council of Tianjin(No.20030618) and Science and Technology Research Council of Tianjin (No.043600511).This support is gratefully acknowledged. Thanks are also due to my tutor for his valuable suggestion and many helpful comments offered for improving the accuracy and presentation of the research problem.

TABLE I
COMPARISON WITH
A PRIORI STRATEGY FOR EACH ROUTE

No	Number of nodes	Cost of this algorithm	Cost of a priori strategy	Saving cost (%)
1	50	2506	2692	6.9
2	75	3720	4026	7.6
3	100	4876	5204	6.3
4	125	6152	6716	8.4
5	150	7409	8062	8.1
6	175	8365	9142	8.5
7	200	9613	10610	9.4

TABLE II
COMPARISON WITH A DETERMINISTIC METHOD

No	Number of nodes	Cost of this algorithm	Cost of deterministic algorithm	Saving cost (%)
1	50	2506	2911	13.9
2	75	3720	4366	14.8
3	100	4876	5784	15.7
4	125	6152	7246	15.1
5	150	7409	8747	15.3
6	175	8365	10030	16.6
7	200	9613	11971	19.7

REFERENCES

[1] L. D. Bodin, B.L.Golden, A.Assad, et al, "Routing and scheduling of vehicles and crews," *Computer Operation Research,* vol.16, pp. 69-211, 1983.

[2] B. L. golden and A.A.Assad, *Vehicle Routing: Methods and Studies*, North-Holland, Amsterdam, 1988.

[3] G. Laporte, "The vehicle routing problem: an overview of exact and approximate algorithms," *European Journal of Operational Research*, vol.59, pp.345-358, 1992.

[4] M. Dror, G. Laporte and P. Trudeau, "Vehicle routing with stochastic demands: properties and solution frameworks," Transportation Science, vol.23, pp.166-176, 1989.

[5] H. N. Psaraftis, "Dynamic vehicle routing," *Ann. Oper. Res*, vol.61, pp.143-164 , 1995.

[6] C. Bastian and A. H. G. Rinooy Kan, "The stochastic vehicle routing problem revisited," *European Journal of Operational Research*, vol.56, pp.407-412, 1992.

[7] N. Secomandi, *Exact and heuristic dynamic programming algorithms for the vehicle routing problem with stochastic demand*, PhD dissertation, Department of Decision and Information Sciences, University of Houston, Houston, TX (1998).

[8] N. Secomandi, "A rollout policy for the vehicle routing problem with stochastic demands," *Operations Research*, vol. 49, pp. 796-802, 2001.

[9] P. Jaillet, *Probabilistic Traveling Salesman Problems*, Ph. D Thesis, Technical Report, No. 185, Operations Research Center Massachusetts Institute of Technology, Cambridge, Mass, 1985.

[10] D. Haugland and C. Sin, *Designing delivery districts for the vehicle routing problem with stochastic demands*, Technical Report, No. 5020, Norway, 2005.

[11]L. Bianchi and M. Birattari, *Research on the vehicle routing problem with stochastic demand*, Technical Report, No. IDSIA-07-04, IDSIA, March 2004.

[12]W. Yang, K. Mathur and R. H. Ballou, "Stochastic vehicle routing problem with restocking," *Transportation Science*, vol. 34, pp. 99-112, 2000.

[13]L Bianchi and M. Birattari, *Hybrid metaheuristics for the vehicle routing Problem with stochastic demands*, Technical Report, No. IDSIA-06-05, Dalle Molle Institute for Artificial Intelligence, Switzerlan, 2005.

[14]M. L. Fisher and J. Jaikumar, "A generalized assignment heuristic for vehicle routing," *Networks*, vol.11, pp.109-124, 1981.

[15]G. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic programming*, Belmont, MA: Athena Scientific, 1996.

[16]J. N. Bertsekas and C. Tsitsiklis, "Rollout algorithms for combinatorial optimization," *Journal of Heuristics*, vol.3, pp.245-262, 1997.

[17]D. Bertsekas and A. Castanon, "Rollout algorithms stochastic scheduling problems," *Journal of Heuristics*, vol.5, pp.89-108, 1999.