2006 IEEE Congress on Evolutionary Computation
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

A Multiobjective Evolutionary Algorithm for Solving Vehicle Routing Problem with Stochastic Demand

C. Y. Cheong, K. C. Tan, D. K. Liu, and J. X. Xu

Abstract— **This paper considers the routing of vehicles with limited capacity from a central depot to a set of geographically dispersed customers where actual demand is revealed only when the vehicle arrives at the customer. The solution to this vehicle routing problem with stochastic demand (VRPSD) involves the optimization of complete routing schedules with minimum travel distance, driver remuneration, and number of vehicles, subject to a number of constraints such as vehicle time window and capacity. To solve such a multiobjective combinatorial optimization problem, this paper presents a multiobjective evolutionary algorithm that incorporates two VRPSD-specific heuristics for local exploitation and a route simulation method to evaluate the fitness of solutions. A novel way of assessing the quality of solutions to the VRPSD on top of comparing their expected costs is also proposed. It is shown that the algorithm is capable of finding useful tradeoff solutions which are robust to the stochastic nature of the problem.**

I. Introduction

The Vehicle Routing Problem (VRP) is a generic name referring to a class of combinatorial optimization problems in which customers are to be served by a number of vehicles. The vehicles leave the depot, serve customers in the network, and on completion of their routes, return to the depot. Each customer is described by a certain demand. Other information includes the co-ordinates of the depot and customers, the distance between them, and the capacity of the vehicles providing the service. All these information are known in advance for the purpose of planning a set of routes which minimizes transportation cost while satisfying some side constraints. However, in many real-world applications, one or more parameters of the VRP tend to be random or stochastic in nature, giving rise to the Stochastic Vehicle Routing Problem (SVRP).

In this paper, we consider the capacity and time constrained Vehicle Routing Problem with Stochastic Demand (VRPSD), where only the customer demand is stochastic and all other parameters are known a priori. This problem appears in the delivery of home heating oil [1], trash collection, beer and soft drinks distribution, and the collection of cash from bank branches [2]. The VRPSD differs from its deterministic counterparts in that when some data are random, it is no longer possible to require that all constraints be satisfied for all realizations of the random variables [3]. The basic characteristic of the VRPSD is that the actual demand of each customer is revealed only when the vehicle reaches the customer. As such, on one hand, the vehicle routes are designed in advance by applying a particular algorithm but on the other hand, due to the uncertainty of demands of the customers, at some point along a route the capacity of the vehicle may be depleted before all demands on the route have been satisfied. Teodorović and Lucić [4], and Dror and Trudeau [5], referred to such a situation as a "route failure". In the capacity constrained VRPSD, recourse or corrective actions, e.g. making a return trip to the depot to restock, have to be designed to ensure the feasibility of solutions in case of route failure.

In the time constrained VRPSD, one possible corrective action is to apply a penalty when the duration of a route exceeds a given bound. This penalty would correspond to the overtime pay that a driver receives. As such, the situation of route failure, together with all its associated recourse policies, would generate additional transportation cost, in terms of the travel distance for the to and fro trips to the depot and the overtime pay for drivers, which are stochastic in nature. This means that the actual cost of a particular solution to the VRPSD cannot be known with certainty before the actual implementation of the solution. One of the main obstacles to solving the VRPSD is in finding an objective function which takes into consideration all these costs. It is for this reason that Laporte and Louveaux [3], and Gendreau *et al*. [6], agree that the VRPSD and the SVRP in general are inherently much more difficult to solve than their deterministic counterparts.

Many researchers have studied the VRPSD in two frameworks, namely as a Chance Constrained Program (CCP) [7] or as a Stochastic Program with Recourse (SPR). In CCPs, the problem consists of designing a set of vehicle routes for which the probability of route failure is constrained to be below a certain threshold. It was shown by Steward and Golden [8] that, under some restrictive assumptions, the problem can be reduced to a deterministic VRP and then solved using existing deterministic algorithms. Although the CCP tries to control the probability of route failure, the cost of such failures is ignored. In contrast, the SPR considers the demand distributions of customers and tries to minimize the expected transportation cost, which includes the travel cost as well as the additional cost generated by recourse policies. Gendreau *et al*. [6]

C. Y. Cheong, K. C. Tan, and J. X. Xu are with the Department of Electrical and Computer Engineering, National University of Singapore, 4, Engineering Drive 3, Singapore 117576 (email: eletankc@nus.edu.sg).

D. K. Liu is with the ARC Centre of Excellence for Autonomous Systems (CAS), Faculty of Engineering, University of Technology, Sydney, PO Box 123, Broadway, NSW 2007, Australia.

commented that SPRs are typically more difficult to solve than CCPs but their objective functions are more meaningful.

According to Yang *et al*. [9], a single, long route has the lowest expected travel distance but it may not be feasible to implement in the context of the real-world as the vehicle may take a long time to complete the route. From this finding, it is clear that the VRPSD is inherently a multiobjective optimization problem. In minimizing the travel distance of a particular solution, an algorithm for the VRPSD must also account for the feasibility of implementation of the solution in terms of the duration of the routes, i.e. both capacity and time constraints must be considered. Therefore, it is required to minimize multiple conflicting cost functions, such as the travel distance, the remuneration for drivers including overtime pay, and the number of vehicles required, concurrently, which is best solved by means of multiobjective optimization. Most of the existing literature, however, either do not consider time constraints or use single objective-based heuristic methods that incorporate penalty functions or combine the different objectives by a weighting function. Furthermore, as mentioned earlier, one of the main difficulties of solving the VRPSD is finding an objective function that is able to define properly the expected transportation cost of a solution, which includes the initial cost of travel before route failures occur as well as the additional cost generated by recourse policies. These characteristics of the VRPSD must definitely be addressed when solving the problem.

In solving the VRPSD, a multiobjective evolutionary algorithm (MOEA) [10] is proposed. The MOEA incorporates two heuristics, which are inspired by two route structures of a solution to the VRPSD identified by Dror and Trudeau [5], for local exploitation in the evolutionary search. In contrast to existing aggregating approaches, the MOEA utilizes the concepts of Pareto optimality to solve the multiobjective VRPSD. In addition, an intuitive Route Simulation Method (RSM) is proposed to address the issue of evaluating the expected costs of solutions. A procedure based on the RSM is also proposed to assess the quality of solutions on top of comparing their expected transportation costs which has been used as the main performance measure hitherto.

This paper is organized as follows: Section II gives the problem formulation of the VRPSD. Section III presents the program flow of the proposed MOEA. Section IV presents the simulation results and analysis of the proposed algorithm. Conclusions are drawn in Section V.

II. Problem Formulation

The VRPSD involves the routing of a set of identical vehicles with limited capacity from a central depot to a set of geographically dispersed customers whose demands are treated as independent random variables with known distributions and the actual demand of each customer is revealed only when the vehicle arrives at the customer.

In the capacity constrained VRPSD, route failures may occur due to the uncertainty of demands at the customers. The simple recourse policy used in this paper is as follows: when a vehicle is not able to service a customer once it arrives there due to insufficient capacity, the vehicle will unload all remaining goods at the customer, return to the depot to restock, then turn back to the customer to complete the service, and finally continue with the originally planned route. In any case if a vehicle is empty immediately after a service to a customer, it will return to the depot to restock and continue with the planned route at the customer after the last serviced customer. All these recourse actions will incur additional transportation cost, in terms of travel distance and time for the to and fro trips to the depot. The travel distance between any two points i and j, where each point can be a customer or the depot, is equal to the travel time and is denoted by c_{ij}, which is calculated using (1):

$$c_{ij} = \sqrt{(i_x - j_x)^2 + (i_y - j_y)^2} \tag{1}$$

where i_x and i_y are the x and y coordinates of the point i respectively. c_{ij} is symmetrical, i.e. $c_{ij} = c_{ji}$, and satisfies the triangular inequality, where $c_{ij} + c_{jk} \geq c_{ik}$.

In the time constrained VRPSD, there is a service time associated with each customer and the depot. The total duration of each route, including the service and travel times, should not exceed a given bound, i.e. each vehicle has a time window. This bound is calculated as the time for a vehicle to travel diagonally across the map from one corner to the other and back. This time is assumed to be 8 hours, equivalent to a driver's workday. Remuneration is such that drivers are paid $10 for each of the first 8 hours of work and $20 for every additional hour of work subsequently. This is done to penalize exceedingly long routes which may not be feasible to implement in the context of the real-world.

It is also assumed that each customer can only be serviced by one vehicle but the vehicle is allowed to service the same customer more than once. Multiple service times will be incurred if a vehicle visits a customer multiple times. Each visit to the depot for restocking will also incur a fixed service time.

III. Multiobjective Evolutionary Algorithm

A. *Variable-length Chromosome*

In the MOEA, a variable-length chromosome representation [10], shown in Fig. 1, is applied such that each chromosome encodes a complete solution, including the number of vehicles and the customers served by these vehicles. Such a representation is efficient and allows the number of vehicles to be manipulated and minimized directly for multiobjective optimization in the VRPSD.

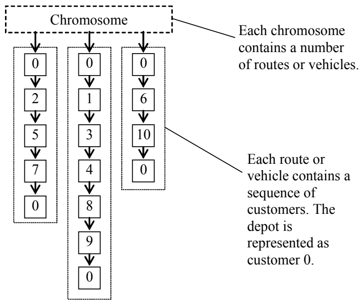


Fig. 1 Variable-length chromosome representation

B. Exploiting VRPSD Route Structure

The local search approach is vital in multiobjective evolutionary optimization in order to encourage better convergence and can contribute to the intensification of the optimization results, which is usually regarded as a complement to the evolutionary operators that mainly focus on global exploration. The two local search methods proposed here are inspired by the underlying structures of a VRPSD solution identified by Dror and Trudeau [5].

Dror and Trudeau [5] showed that given that the customers' demands are independent random variables with non-negative means, the probabilities of route failures along a route is a non-decreasing sequence. To take advantage of this route structure, a local search heuristic, Shortest Path Search (SPS), is developed. Given a route, SPS builds a new route by choosing the customer that is furthest from the depot as the first customer in the route, while the customer that is nearest to the depot is chosen as the last customer. Next, the customer that is nearest to the first customer is chosen as the second customer, while the customer that is nearest to the last customer is chosen as the second last customer of the new route. This step is repeated until all the customers in the original route are re-routed. By re-routing customers in such a manner, customers that are further from the depot will be at the beginning of the route while those that are nearer to the depot will be at the end of the route. The rationale is to reduce the expected transportation cost for the recourse policy.

Another VRPSD route structure identified by Dror and Trudeau [5] is that due to the recourse policies, the expected transportation cost of a route is dependent on the direction in which the route is traversed. Subsequently, Which Directional Search (WDS) is proposed to exploit this route structure. Given a route, WDS builds a new route that runs in the opposite direction.

After each operation of SPS or WDS, the original route and the new route constructed will be evaluated and the route with the smaller sum of travel distance and driver remuneration will be retained.

C. Route Exchange Crossover

In contrast to classical one-point crossover which may produce infeasible route sequences, this paper adopts a simple route exchange crossover operator [10] that allows good sequences of routes or genes in a chromosome to be shared with other chromosomes in the evolving population. The operation of this crossover is shown in Fig. 2.

In the route exchange crossover, only the best routes of the selected chromosomes are eligible for exchange. To ensure the feasibility of chromosomes after the crossover, duplicated customers are deleted. These customers are deleted from the original routes while the newly inserted route is left intact.

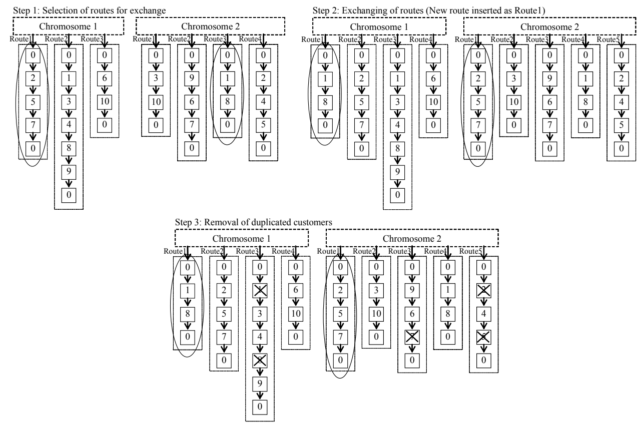


Fig. 2 Illustration of route exchange crossover

D. Multi-mode Mutation

The algorithm implements a multi-mode mutation operator [10] to complement the crossover operator in allowing a larger search space to be explored. There are four parameters associated with the multi-mode mutation, namely mutation rate, elastic rate, squeeze rate, and shuffle rate. Fig. 3 shows the operation of the multi-mode mutation.

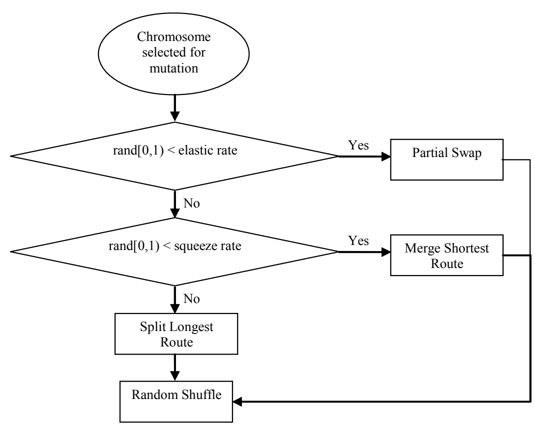


Fig. 3 Operation of multi-mode mutation

1) Partial Swap: The operation involves a number of swap moves and for each move, two routes will be randomly chosen. A segment is then randomly selected from each route and swapped to the other route. This new segment takes the place of the previous segment that has been swapped out.

2) Merge Shortest Route: This operation searches for the two routes of the chromosome with the smallest sum of travel distance and driver remuneration, and appends one route to the other.

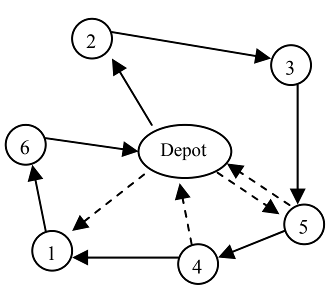
3) Split Longest Route: This operation breaks the route with the largest sum of travel distance and driver remuneration into two at a random point.

At the end of the multi-mode mutation, a random shuffling operation, where the order of customers in every route is shuffled with a probability equal to the shuffle rate, is applied.

E. Route Simulation Method (RSM)

Fig. 4 shows a route sequence, Depot → 2 → 3 → 5 → 4 → 1 → 6 → Depot, with a set of generated demands. The solid arrows indicate the route that the vehicle will take if this were a deterministic VRP. However, due to the recourse policies in the event of a route failure in the VRPSD, the actual route taken by the vehicle cannot be known with certainty before the route is actually implemented. The RSM simulates the implementation of the route by generating a set of demands for all the customers based on their demand distributions, treating these demands as if they were the real demands revealed when a vehicle first arrives at the customer. For this particular example, it is assumed that the vehicle capacity is 15 and each arrow indicates a unit of distance. The vehicle first leaves the depot and arrives at customer 2. It is able to satisfy its demand with a remaining capacity of 9. The vehicle then travels to customer 3 and satisfies its demand. The capacity of the vehicle is 7 when it reaches customer 5. The vehicle then finds that it is unable to satisfy the demand of customer 5, so it unloads all remaining goods and makes a return trip to the depot to restock. This recourse is indicated by the dashed arrows between the depot and customer 5. The vehicle then unloads two units of goods and leaves customer 5 for customer 4 with a capacity of 13. After serving customer 4, the vehicle is empty and returns to the depot to restock. Since the demand of customer 4 has been satisfied, the vehicle travels to customer 1 from the depot. The vehicle then satisfies the demands of customers 1 and 6 and returns to the depot. From this simulation, we are able to obtain the total distance traveled by the vehicle (10 units for this example) and the remuneration for the driver for a particular realization of the set of customer demands.

Due to the stochastic nature of the cost considered, we need to repeat the above operation N times for every route of a particular solution using different sets of demands randomly generated based on the demand distributions of the customers and then taking the average to obtain the expected transportation cost of the solution.



Customer	Real Demand
1	5
2	6
3	2
4	13
5	9
6	5

Fig. 4 Example to show the operation of the RSM

In this paper, three different RSM settings are studied. Generate Every Generation (GEG) refers to the setting where the N demand sets used by the RSM are refreshed every generation. Generate Every M (GEM) refers to the setting where the N demand sets are only refreshed at the end of every M generations. Lastly, Alternate Every M (AEM) refers to the setting where for M generations, the RSM uses the N randomly generated demand sets and for the next M generations, the RSM uses the mean values of the customers' demand distributions to simulate the implementations of the routes. The N demand sets are then refreshed for use over the next M generations and the process repeats.

F. Pareto Ranking

As mentioned in the introduction, the VRPSD is a multiobjective optimization problem where a number of objectives such as the travel distance, the remuneration for drivers, and the number of vehicles required, need to be minimized concurrently. In contrast to single objective optimization, the solution to a multiobjective optimization problem exists in the form of alternate tradeoffs known as the Pareto optimal set. Each objective component of any non-dominated solution in the Pareto optimal set can only be improved by degrading at least one of its other objective components. In the total absence of information regarding the preference of objectives, a ranking scheme based upon the Pareto optimality is regarded as an appropriate approach to represent the fitness of each solution in an evolutionary algorithm for multiobjective optimization.

Thus, the role of multiobjective optimization in the VRPSD is to discover such a set of Pareto-optimal solutions from which the decision maker can select an optimal solution based on the current situation. The Pareto fitness ranking scheme [11] for evolutionary multiobjective optimization is adopted here to assign the relative strength of solutions. The ranking approach assigns the same smallest rank to all non-dominated solutions, while the dominated ones are inversely ranked according to the number of solutions dominating them. In Fig. 5, a hypothetical solution (the black dot) is plotted in the objective domain at coordinates (1, 1, 1). Each solution will define a rectangular box encompassing the origin as shown in the figure. Another solution will dominate this solution if and only if it is within or on the box defined by the first solution but not equal to the first solution.

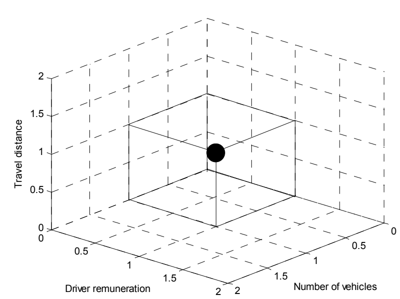


Fig. 5 Example to show how a solution is dominated by another solution

G. MOEA Flowchart

The algorithmic flow of the MOEA is shown in Fig. 6. At the start of the algorithm, a database of customers' information is built. The information consists of the coordinates, the mean and variance of the demand distribution, and the service time of each customer. Other information includes the capacity and time window of a vehicle, and the coordinates and service time of the depot.

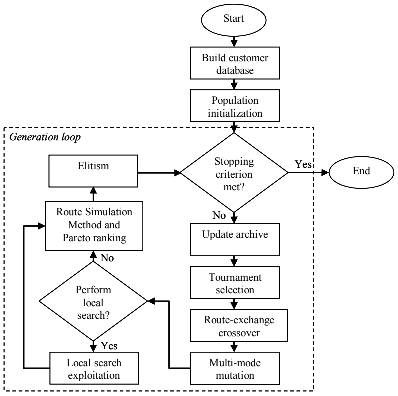


Fig. 6 Flowchart of MOEA

1) Initialization: The first chromosome is built such that the sum of the mean values of the customer demands on each route does not exceed the vehicle capacity. Furthermore, the sum of the travel and service times on each route must not exceed the vehicle time window. The number of vehicles required in this first chromosome is then taken as the maximum number of vehicles that each of the remaining chromosomes can use. For each chromosome, the number of vehicles is randomly picked from the feasible range. The routes are then built such that each route has approximately the same number of customers. This procedure is done so that the initial population has a wide range of chromosomes with different number of vehicles to start with.

2) Evaluation: After the initial evolving population is formed, all the chromosomes are evaluated based on the RSM and ranked according to their Pareto's dominance in the population. Following the ranking process, an archive population is updated. The archive population has the same size as the evolving population and is used to store all the best solutions found during the search. The archive population updating process consists of a few steps. The evolving population is first appended to the archive population. All repeated chromosomes, in terms of the objective domain, are deleted. Pareto ranking is then performed on the remaining chromosomes in the population. The larger ranked (weaker) chromosomes are then deleted such that the size of the archive population remains the same as before the updating process. The evolving population remains intact during the updating process.

3) Genetic operations: The binary tournament selection scheme is then performed. All the chromosomes in the evolving population are randomly grouped into pairs and from each pair, the chromosome with the smaller rank is selected for reproduction. This procedure is performed twice to preserve the original population size. The genetic operators consist of the route-exchange crossover and the multi-mode mutation. To further improve the internal routings of customers, SPS and WDS are applied to the evolving and archive populations every 50 generations for better local exploitation in the evolutionary search.

4) Elitism: A simple elitism mechanism is employed in the MOEA for faster convergence and better routing solutions. The elitism strategy involves randomly picking a number of good chromosomes (3% of the population size) from the pool of chromosomes in the archive population belonging to the best three Pareto ranks. The chosen chromosomes then replace the worst ranked chromosomes in the evolving population.

This is one complete generation of the MOEA and the evolution process iterates until the computing budget [12] is exhausted. One unit of the computing budget is defined as one run of the RSM on a particular solution using a particular demand set.

IV. Simulation Results

The MOEA was programmed in C++ and simulations were performed on an Intel Pentium 4 2.8 GHz computer. Table I shows the parameter settings chosen after some preliminary experiments.

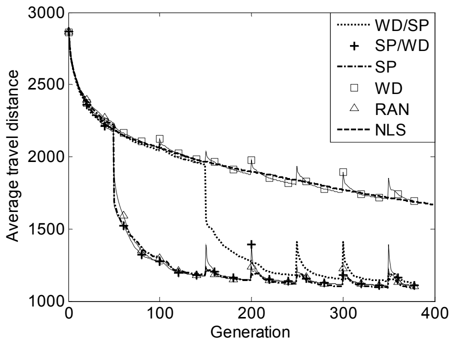
TABLE I
Parameter Settings for Simulation Study

Parameter	Value
Population size	500
Crossover rate	0.7
Mutation rate	0.4
Elastic rate	0.5
Squeeze rate	0.5
Shuffle rate	0.3
Computing budget	2000000

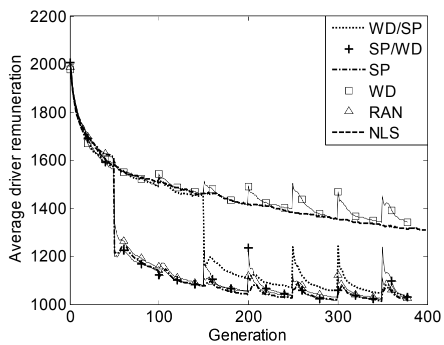
Bianchi *et al*. [13] highlighted that there is no commonly used benchmark for the VRPSD in the literature. As such, many authors generated their own test problems. Yang *et al*. [9], and Teodorović and Lucić [4], randomly generated the locations of the depot and the customers. The characteristics of each customer's demand, mean and variance, are also randomly generated. Dror and Trudeau [5] adapted a test problem for the deterministic VRP for the VRPSD. As Dror and Trudeau [5] is the only reference that provided the actual test problem used, for comparison purposes, all simulations in this paper were performed using that test problem.

A. Performance of Local Search Operators

This section demonstrates the effectiveness of local exploitation in the MOEA and also analyzes the effectiveness of various settings in which the local search heuristics are incorporated with the MOEA. Simulations were conducted using six different settings. Three of the settings include MOEA with no local search (NLS), with only WDS (WD), and with only SPS (SP). WD/SP is a setting which involves the application of WDS for the first two local exploitations and alternates between the two local search heuristics every 100 generations. On the other hand, SP/WD starts with SPS and alternates every 100 generations. The final setting is RAN where each chromosome will have equal chance of being operated by either SPS or WDS on all of its routes. The simulations were conducted using the GEG setting with *N* set to 10.

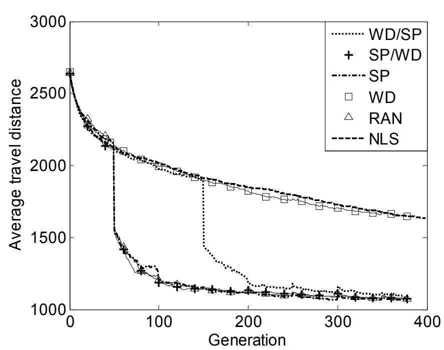


(a)

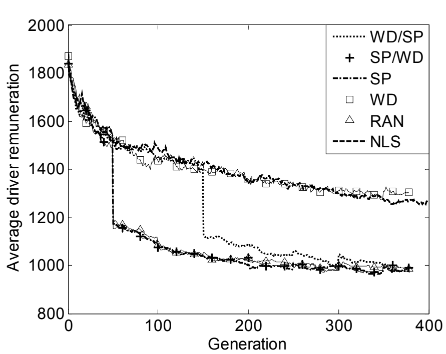


(b)

Fig. 7(a) Average travel distance and (b) average driver remuneration of archive populations of different local search settings



(a)



(b)

Fig. 8(a) Average travel distance and (b) average driver remuneration of non-dominated solutions of different local search settings

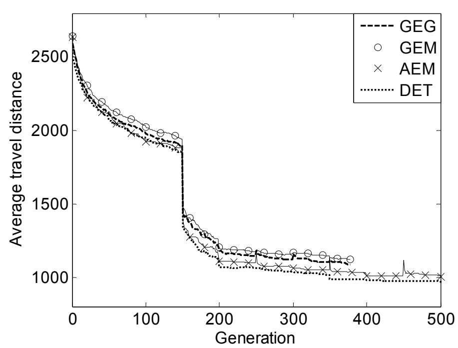
The convergence traces of the travel distance and the driver remuneration for the six settings are plotted in Fig. 7(a)-(b) and Fig. 8(a)-(b). Fig. 7(a)-(b) shows the convergence of the respective costs, averaged over all the solutions in the archive population, over the generations. Fig. 8(a)-(b) shows the same costs averaged over all the non-dominated solutions in the archive population. The plots show the effectiveness of local exploitation in the MOEA as the five settings which use local search perform better than NLS. The effectiveness of SPS is evident since the four settings, namely WD/SP, SP/WD, SP, and RAN, which make use of the local search operator, are able to find solutions with travel distance and driver remuneration significantly lower than those found by WD and NLS. SPS is able to speed up convergence as it causes sharp dips in the respective costs of the solutions found whenever it is performed. The performances of WD/SP, SP/WD, SP, and RAN are comparable and the setting WD/SP is selected as the default setting for any further analysis.

It is observed in Fig. 7(a)-(b) that there are distinctive spikes in the convergence traces which coincide with the occurrences of local search. This is despite the fact that during local search, a new route is constructed and compared with the original route and the better route is retained. This happens because in comparing the new and original routes during local search, the solutions in the archive population are re-evaluated by the RSM. This re-evaluation acts to complement the RSM and is important in the stochastic problem where the costs of solutions are sensitive to the demand sets that are used by the RSM. For a particular solution, the fitness evaluated using the RSM can be very different depending on the demand sets used. As such, it is essential that a solution to the VRPSD be robust to the stochastic nature of the problem and its fitness should not differ too much with each evaluation by the RSM. The re-evaluation of all the solutions in the archive population during local search ensures that only robust solutions stay non-dominated. The effect of this can be seen in Fig. 8(a)-(b) which considers only non-dominated solutions in the archive population. The spikes in these convergence traces during local search are significantly smaller, if not negligible.

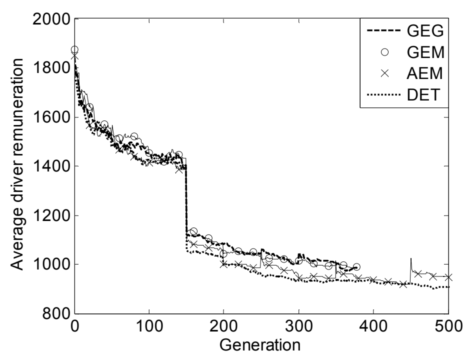
B. Comparison with a Deterministic Approach

In the absence of a stochastic procedure to deal with stochastic demands, one can generate the routes using a deterministic vehicle routing algorithm by treating the expected demand at each customer as its deterministic demand [9]. The attraction of this deterministic approach is its relative simplicity and familiarity to practitioners. The MOEA can in fact be modified into a deterministic vehicle routing algorithm (DET) by solely using the mean demand set in the RSM. However, what makes the MOEA different from a deterministic vehicle routing algorithm is the RSM's ability to operate on demand sets which are randomly generated based on the demand distributions. This section will show that the RSM's ability to operate on randomly generated demand sets can lead to solutions which are more robust to the stochastic nature of the problem compared to the deterministic approach and that the expected transportation costs of such solutions are good estimates of the true performance of the solutions. In addition, a RSM-based procedure is proposed to assess the quality of solutions on top of comparing their expected costs.

Simulations were conducted for GEG, GEM, AEM, and DET, with N and M both set to 10. The convergence traces of the travel distance and the driver remuneration for the four settings are plotted in Fig. 9(a)-(b). Due to the nature with which the RSM is run in AEM and DET, i.e. the RSM is run only once, instead of N times, when evaluating solutions using the mean demand set, these two settings took more than 500 generations to complete. However, by the 500th generation, these two settings have converged and the plots in Fig. 9(a)-(b) show only the convergence traces up to the 500th generation. By comparing the convergence traces of the four settings, it appears that DET is able to churn out the best solutions since both the average travel distance and the average driver remuneration are the lowest among the four settings at the termination of the algorithm.



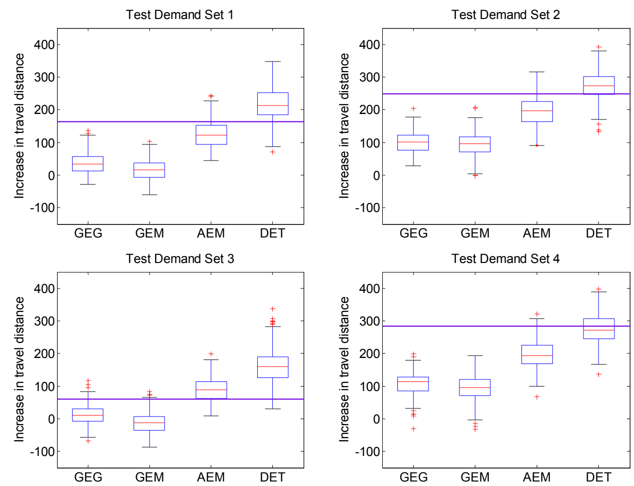
(a)



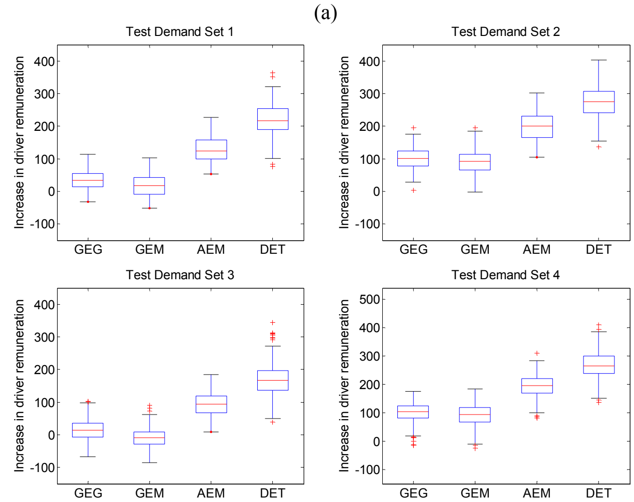
(b)

Fig. 9(a) Average travel distance and (b) average driver remuneration of non-dominated solutions of GEG, GEM, AEM, and DET

During the decision making process, the logistic manager will look at the expected transportation costs of all the candidate solutions and choose the solution that best suits the company's logistic condition, in terms of the available vehicle fleet size and the company's priorities. Thus, it is important for the expected cost of each solution to give a good estimate of the true performance of the solution. As such, it is necessary to compare the results to the VRPSD based on this aspect on top of comparing their expected costs.



(a)



(b)

Fig. 10 Increase in (a) travel distance and (b) driver remuneration after implementing non-dominated solutions of GEG, GEM, AEM, and DET

To perform such a comparison, a test demand set is randomly generated based on the customers' demand distributions. This test demand set will represent the real demands that the vehicles of a particular solution would experience when the solution is implemented. The RSM is then operated, using the test demand set, on the non-dominated solutions of the four settings, GEG, GEM, AEM, and DET, to simulate the actual costs of implementing the solutions. To ensure that the results are not biased towards any test demand set, the same procedure is repeated for three other randomly generated test demand sets. The increase in travel distance and driver remuneration, from the expected values, after simulating the implemention on the four test demand sets are found for every non-dominated solution of the four settings. The results of this comparison are represented in box plots in Fig. 10(a)-(b). Each box plot represents the distribution of the deviations between the respective actual and expected costs, where the horizontal line within the box encodes the median, and the upper and lower ends of the box are the upper and lower quartiles respectively. The two horizontal lines beyond the box give an indication of the spread of the data. A plus sign outside the box represents an outlier. It can be seen from Fig. 10(a)-(b) that the expected costs of solutions obtained by GEG and GEM deviate less from the corresponding actual costs for all the test demand sets. AEM and DET produce solutions that have expected costs that are poor estimates of the actual costs. The spreads of their deviations are also larger compared to those of GEG and GEM which will result in poorer predictability in the deviations.

A test was also conducted to compare the robustness of the solution found by Dror and Trudeau [5] with those found by the four settings. The solution of Dror and Trudeau [5] was implemented using the simple recourse policy described in Section II. The increases in travel distances for the four test demand sets are obtained and plotted as four horizontal lines in the respective box plots in Fig. 10(a) since Dror and Trudeau [5] only considered the single objective of travel distance. From Fig. 10(a), it can be seen that the solution of Dror and Trudeau [5] is not as robust as those found by GEG and GEM. For test demand sets 2 and 4, the increases in distances after implementing the solution are comparable with those found by DET. This is despite the fact that Dror and Trudeau [5] used a worst case recourse policy where in case of a route failure, all the remaining customers in the route are served through individual deliveries.

Table II summarizes the importance of this analysis. We have previously commented that DET produces the "best" solutions among the four settings. This is reflected in Table II as DET produces solutions with the lowest expected travel distance and driver remuneration. However, DET also produces solutions with the largest increase in travel distance and driver remuneration considering test demand set 1 as shown in Fig. 10(a)-(b). Using the multiplicative aggregation method [14] of actual travel distance and driver remuneration, Table II takes all these factors into consideration and shows that GEG has the lowest multiplicative aggregate of 1.14, distinctly lower than the values for AEM and DET. From this analysis, it is evident that DET is prone to giving the logistics manager inaccurate information. In order for the logistic manager to correctly select a solution, it is important that the expected costs of solutions give good approximations of the actual costs. This shows that the stochastic nature of the VRPSD cannot be neglected and the expected cost is not a good indicator of solution quality. It is clear that the RSM is a robust technique to evaluate the expected cost of a solution, which was previously considered in the literature as one of the main difficulties to solving the VRPSD.

TABLE II
COMPARISON WITH A DETERMINISTIC APPROACH

	Ex. Dist.	Increase Dist.	Actual Dist.	Ex. Rem.	Increase Rem.	Actual Rem.	Mul. Agg. ($x10^6$)
GEG	1086.59	33.85	1120.44	985.95	32.87	1018.83	1.14
GEM	1120.52	16.535	1137.06	990.33	18.64	1008.97	1.15
AEM	1002.08	122.35	1124.43	947.04	125.04	1072.08	1.21
DET	970.17	213.4	1183.57	909.59	217.35	1126.93	1.33

V. CONCLUSIONS

This paper is among the first to integrate multiobjective optimization paradigm in solving the VRPSD. The proposed MOEA optimizes all the objectives concurrently. The effectiveness of the two proposed VRPSD-specific local search heuristics and the various settings in which local exploitation is incorporated with the MOEA has been studied. Simulations have been performed to show that the solutions obtained by the MOEA, equipped with the RSM, are robust to the stochastic nature of the problem. The expected costs of such solutions are good approximations of the actual costs of implementing the solutions, thus providing the logistic manager with accurate information based on which decision will be made.

REFERENCES

[1] M. Dror, M. O. Ball, and B. L. Golden, "Computational comparison of algorithms for inventory routing", *Annals of Operations Research*, vol. 4, pp. 3 – 23, 1985.

[2] V. Lambert, G. Laporte, and F. V. Louveaux, "Designing Collection Routes through Bank Branches", *Computers and Operations Research*, vol. 20, pp. 783 – 791, 1993.

[3] G. Laporte and F. V. Louveaux, "Solving stochastic routing Problems with the Integer L-shaped Method", *Fleet Management and Logistics*, T. G. Crainic and G. Laporte (eds.), Kluwer Academic Publishers, Boston, pp. 159 – 167, 1998.

[4] D. Teodorović and P. Lucić, "Intelligent vehicle routing system", in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, Dearborn, MI, USA, 2000, pp. 482 – 487, 2000.

[5] M. Dror and P. Trudeau, "Stochastic vehicle routing with modified savings algorithm", *European Journal of Operational Research*, vol. 23, no. 2, pp. 228 – 235, 1986.

[6] M. Gendreau, G. Laporte, and R. Séguin, "Stochastic vehicle routing", *European Journal of Operational Research*, vol. 88, no. 1, pp. 3 – 12, 1996.

[7] A. Charnes and W. Cooper, "Deterministic equivalents for optimizing and satisfying under chance constraints", *Operations Research*, vol. 11, pp. 18 – 39, 1963.

[8] W. R. Stewart and B. L. Golden, "Stochastic vehicle routing: A comprehensive approach", *European Journal of Operational Research*, vol. 14, no. 4, pp. 371 – 385, 1983.

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

[10] K. C. Tan, T. H. Lee, Y. H. Chew, and L. H. Lee, "A multiobjective evolutionary algorithm for solving vehicle routing problem with time windows", in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Washington DC, USA, 2003, vol. 1, pp. 361 – 366, 2003.

[11] C. M. Fonseca, "Multiobjective genetic algorithms with application to control engineering problems", Dept. Automatic Control and Systems Eng., University of Sheffield, Sheffield, UK, Ph.D. Thesis, 1995.

[12] L. H. Lee and E. P. Chew, "A simulation study on sampling and selecting under fixed computing budget", in *Proceedings of the 2003 Winter Simulation Conference*, New Orleans, LA, USA, 2003.

[13] L. Bianchi, M. Birattari, M. Chiarandini, M. Manfrin, M. Mastrolilli, L. Paquete, O. Rossi-Doria, and T. Schiavinotto, "Metaheuristics for the vehicle routing problem with stochastic demands", in *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature*, Birmingham, UK, vol. 3242 of Lecture Notes in Computer Science, pp. 450 – 460, 2004.

[14] D. Van Veldhuizen and G. B. Lamont, "Multiobjective evolutionary algorithm research: a history and analysis," *Technical Report TR-98-03*, Department of Electrical and Computer Engineering, Air Force Institute of Technology, Ohio, 1998.