Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

A Differential Evolution &Genetic Algorithm for Vehicle Routing Problem with Simultaneous Delivery and Pick-up and Time Windows

Cao Erbao*,. Lai Mingyong*,** . Nie Kai*,****

**College of Economics and Trade, Hunan University, Changsha410079,P.R.China(email:czp9491@163.com).*
*** Hunan Province Laboratory of Logistics Information and Simulation Technology, Changsha 410079,P.R.China*

Abstract: The vehicle routing problem with simultaneous delivery and pick-up and time windows (VRP-SDPTW) is presented from the point of strategic view that combines the logistics and reverse logistics (bidirectional logistics). A general mixed integer programming mathematic model of VRP-SDPTW is constructed, it can transform into other classical vehicle routing problems by setting different parameters. A hybrid optimization algorithm (HOA) is proposed, which is based principally on the combination of the differential evolution (DE) theory and genetic algorithm (GA). In operation process, we firstly adopted the novel decimal coding to construct the chromosome, and then the differential evolution operator is adopted as the main optimizing scheme, while such techniques of the genetic algorithm, as the novel crossover-operator mutation-operator are designed to improve the result. Therefore it is expected that advantages of both the differential evolution and the genetic algorithm are kept so that the hybrid optimum algorithm can optimize large scale highly nonlinear problems well and improve the efficiency of the optimization process, and can avoid effectively the common defects of early convergence and the diversity of population in traditional genetic algorithm. The computer simulations are used to compare the performance of the proposed method with GA and DE, and numerical results show that the performance of the proposed method is better than the other method.

1. INTRODUCTION

Reverse logistics can be defined as the reverse process of logistics. The Council of Logistics Management (CLM) defines reverse logistics as "The process of planning, implementing, and controlling the efficient, cost effective flow of raw materials, in-process inventory, finished goods and related information from the point of consumption to the point of origin for the purpose of recapturing value or proper disposal"(Stock,1992). The vehicle routing problem with simultaneous delivery and pick-up and time windows (VRP-SDPTW) is a variant of the classical vehicle routing problem (VRP) where clients require simultaneous delivery and pick-up service within the time windows. Deliveries are supplied from a single depot at the beginning of the vehicle's service, while pick-up loads are taken to the same depot at the conclusion of the service. One important characteristic of this problem is that a vehicle's load in any given route is a mix of delivery and pick-up loads, at the same time in any route the vehicle can not violate some constraints, such as: the vehicle capacity, time windows and travelling distance constraints. VRP-SDPTW as an extension for vehicle routing problem, which is a complex combinational optimization problem, and is a well-know non-polynomial hard (NP-hard) Problem. VRP-SDPTW often encountered in fact, and has broad prospects in theory and practice, for example in the soft drink industry, where empty bottles must be returned, and in the delivery to grocery stores, where reusable pallets/containers are used for the transportation of merchandise, and each customer is serviced by exactly one vehicle within its time windows. Reverse logistics is an important area in which the planning of vehicle routes takes the form of VRP-SDPTW, as companies become interested in gaining control over the whole lifecycle of their products. For example, in some countries legislation forces companies to take responsibility for their products during lifetime, especially when environmental issue are involved (as in the disposal of laser printers' cartridges).Returned goods are another example where the definition of vehicle routes may take the form of a VRP-SDPTW. Owing to difficulty of the problem itself and deficiency of attention, even now little work can be found. To the author's knowledge, VSP-SDPTW has never been previously treated by heuristics in the domestic and overseas literature.

VRP-SDP is firstly proposed by Min H.(Min,1989), subsequently near 10 years, there are not correlative report until attach importance to reverse logistics, some researcher engaged in the problem (Dethloff,2001; Angelelli,2003; Tang,2002,2006). Most of the algorithms of solving the VRP-SDP are based on that of classical VRP. In recent years, most published research for the VRP-SDP has focused on the development of heuristics. Genetic algorithm (GA) is a powerful algorithm for solving engineering design and optimization problems (Gen, 2000; Baker, 2003; Christian, 2004), and have been used to tackle many combinatorial problems, including certain types of vehicle routing problem. Storn and Price (Storn, 1996) first introduced the DE algorithm in 1996. DE was successfully applied to the optimization of some well-known nonlinear, non-differentiable and non-convex functions in Storn. DE is a population based and direct stochastic search algorithm (minimizer or maximizer) whose simple, yet powerful and straightforward, features make it very attractive for numerical optimization. DE uses a rather greedy and less stochastic approach to problem solving compared to evolution algorithms. DE combines simple arithmetic operators with the classical operators of crossover, mutation and selection to evolve from a randomly generated starting population to a final solution. Recently, differential evolution algorithm have drawn great attention from researchers due to its robustness and flexibility and have been used to tackle many combinatorial problems, and its used field is fast expanding. But there are little work can be found about VRP that using differential evolution. In this paper, we integrated DE and GA to a hybrid optimization algorithm for solving the VRP-SDPTW.

We developed a mixed integer programming mathematical model for VRP-SDPTW and proposed a hybrid optimum algorithm for the problem. Computational results suggest that the hybrid optimum algorithm can achieve the optimal or near-optimum solution of VRP-SDPTW.

This paper is organized as follows: In section 2, we describe the vehicle routing problem with simultaneous delivery and pick-up service with time windows and present a hybrid integer programming mathematic model of VRP-SDPTW. In section 3, we design a hybrid optimum algorithm (HOA) to solve this problem. Then we will give a numerical experiment to reveal the effectiveness of the hybrid optimum algorithm in section 4. In section 5, we draw a conclusion.

2. FORMULATION FOR VRP-SDPTW

There are $\bar{k}$ vehicles in the depot 0, and V stands for the set of customers to be visited, where n=$|V|$ is the number of customers. The location of depot and customers are known. Each customer has a known delivery demand level d_j and a know pick-up demand level p_j, j=1,2,…,n. Delivery routes for vehicles are required to start and finish at the depot, so that all customer demands are satisfied and each customer is visited by just one vehicle. $V_0 = V \cup \{0\}$ is the set of clients plus depot (client 0); c_{ij} is the distance between i and j, and the capacity of each vehicle is Q; the decision variable x_{ijk} =1, if arc (i,j) belongs to the route operated by vehicle k, otherwise is 0 ; y_{ij} is the demand picked-up in clients routed up to node i and transported in arc (i,j); z_{ij} is the demand to be delivered to clients routed a after node i and transported in arc (i,j). t_i is the service time of customer i ; s_{ik} is the initiatory service time of vehicle k in customer i ; t_{ij} is the travel time from i to j (direct proportion to d_{ij}); the time windows of customer i is often regarded as hard and defined by an interval[a_i, b_i] , within which the service of customer i must be started..

The corresponding mixed integer programming mathematical formulation of VRP-SDPTW is given by:

$$\text{Minimize} \sum_{k=1}^{\bar{k}} \sum_{i=0}^{n} \sum_{j=0}^{n} c_{ij} x_{ijk} \tag{1}$$

$$\text{s. t.} \sum_{i=0}^{n} \sum_{k=1}^{\bar{k}} x_{ijk} = 1, j = 1, \cdots, n \tag{2}$$

$$\sum_{i=0}^{n} x_{ijk} - \sum_{i=0}^{n} x_{jik} = 0 , j = 0.1, \cdots, n;\ k = 0,1, \cdots, \bar{k} \tag{3}$$

$$\sum_{j=1}^{n} x_{0jk} \le 1,\ k = 1, \cdots, \bar{k} \tag{4}$$

$$\sum_{i=0}^{n} y_{ji} - \sum_{i=0}^{n} y_{ij} = p_{j,}\ \forall j \ne 0; \tag{5}$$

$$\sum_{i=0}^{n} z_{ij} - \sum_{i=0}^{n} z_{ji} = d_j,\ \forall j \ne 0; \tag{6}$$

$$y_{ij} + z_{ij} \le Q \sum_{k=1}^{\bar{k}} x_{ijk} , i, j = 0,1, \cdots, n \tag{7}$$

$$s_{ik} + t_i + t_{ij} - M(1 - x_{ijk}) \le s_{jk}, i, j = 0,1, \cdots; n; k = 0,1, \cdots; \bar{k} \tag{8}$$

$$a_i \le s_{ik} \le b_i,\ i, j = 0,1, \cdots; n; k = 0,1, \cdots, \bar{k} \tag{9}$$

$$\sum_{i=0}^{n} \sum_{j=0}^{n} d_{ij} x_{ijk} \le L , k = 0,1,2,..., \bar{k} \tag{10}$$

$$x_{ijk} \in \{0,1\}, y_{ij} \ge 0, z_{ij} \ge 0,\ i, j = 0,1, \cdots; n,\ k = 0,1, \cdots; \bar{k} \tag{11}$$

The objective function seeks to minimize total distance travelled. Constraints(2)ensure that each client is visited by exactly one vehicle; constraints (3)guarantee that the same vehicle arrives and departs from each client it serves; restrictions (4)define that at most $\bar{k}$ vehicles are used; restriction (5)and(6)are flow equations for pick-up and delivery demands, respectively; constraints(7)establish that pick-up and delivery demands will only be transported using arcs included in the solution; Restrictions (8) and (9) are time windows constraints; Restrictions(10)are the maximum distance constraints, L is the upper limit on the total load transported by a vehicle in any given section of the route; Finally, constraints (11) define the nature of the decision variable.

The above formulation is very universal, and can easily turn into other classical vehicle routing problems. If taking out the restraint(8) and (9), and a_i=0, $b_i = \infty$, then it transformed into VRP-SDP; if p_j=0, then it turn into VRPTW equivalent; if p_j =0and taking out the restraint(8) and (9), then it transformed into common VRP formulation; if taking out the restraint(8) and (9)in someone customer, the all former clients p_j=0, and the followed clients d_j=0, then changes into VRP-B equivalent; if for all clients only have delivery or pick-up demand(either p_j or d_j equals 0), and taking out the restraint(8) and (9), then it changes into VRP-PD equivalent; if only one vehicle can finish service, then it turns into TSP equivalent.

3. THE PROPOSED HOA FOR VRP-SDPTW

3.1 Coding and fitness function

Like in most GA for the VRP, a chromosome I(n) simply is a sequence (permutation) S of n customer nodes. The permutation representation is a simple and nature representation that encodes the identity of the element to be scheduled in each gene, and the order in which the genes appear in the chromosome string gives the schedule sequence. That means $Chrom\ (i,:) = randperm\ (n)$ and $i = \{1,2,\cdots,NP\}$ and NP is the population size, and n is the number of customers. We check the capacity constraints, time windows constraints and distance constraints at the same time from the first gene of chromosome, if do not violate the constraints, considering the next gene; if it violate the constraints in someone gene, we consider to use other vehicle from this gene starting, and repeat the above process, till all of customers were serviced. For instance, there are 10 customer nodes, a randomly generated chromosome is 8 9 2 3 4 6 1 5 7 10, and if there are only 4 vehicles available in distribution ($\bar{k}$=4), then routes: 0-8-9-0, 0-2-3-6-0, 0-4-5-0, 0-1-7-10-0 are feasible, and S stand for routes number are r=4.If the routes satisfy the vehicle capacity and time windows, then this chromosome is feasible, otherwise, it is unfeasible; If $\bar{k}$ >=r, then this chromosome is legal; otherwise, it is illegal.

In order to prevent illegal chromosome entering the next generation in great probability, a penalty function is designed. R is the total distance vehicles travelled of the corresponding chromosome based on formulation (1), let $m = r - \bar{k}$, if $r > \bar{k}$, then $m > 0$, let $R = R + M \times m$, where M is a large integer; else $m = 0$. The fitness function can be expressed as $f = 1/(R + M \times m)$.

3.2 Differential evolution theory

Differential Evolution grew out of Ken Price's attempts to solve the Chebychev Polynomial fitting Problem that had been posed to him by Rainer Storn in 1996. The basic steps are as fellows:

(1) Initiation population: We adopted an integer coding as section 3.1, the initial population is generated by random generator and the number of individual is NP, each individual is an N-dimensional solution vectors.

(2) Mutation operation: The chromosome of offspring generate by parent gene difference, mutation is an operation that adds a vector differential to a population vector of individuals, according to the following equation: $v = Chrom(c,:) + F * [Chrom(a,:) - Chrom(b,:)]$, (12);

where a, b, c are generated randomly and mutually different, they decide three chromosomes together; the scaling factor F is a constant from [0,2].Because we adopted an integer code, and each chromosome represents a sequence of the customers, each gene stands for a customer, when the offspring gene oversteps the range, we must consider an auxiliary operator. For the largest gene of offspring gives the largest customer ordinal number N , the second evaluated as $N-1$, the rest may be deduced by analogy, we can prove that this operator equates to an affine transform, for example, the offspring chromosome is [-7,1,-5,2,-3, 0, 3, 5], the number of the customers is 8, we obtain the offspring chromosome that is [1, 5, 2, 6, 3, 4, 7, 8].

(3)Crossover operation: Following the mutation operation, crossover operation is applied to the population. Crossover operation is employed to generate a trial vector by replacing certain parameters of the target vector by the corresponding parameters of a randomly generated donor vector based on the following equation

$$trial(j)^{G+1} = \begin{cases} v(j)^{G+1}, & rand(j) \le CR \quad or \quad j = randn(i) \\ Chrom(i,j)^{G}, rand(j) > CR & and \quad j \ne randn(i) \end{cases} \quad (13)$$

Where G is the number of current iteration, $CR \in [0,1]$ is the crossover probability factor. In order to improve the population's diversity and the ability of breaking away from the local optimum, we present a new self-adapting differential evolution algorithm, the key factor is the crossover probability (CR) is time varying, it changes from small to large with iteration number based on the following equation

$$CR = CR_{\min} + G * \frac{CR_{\max} - CR_{\min}}{MAXGEN}, \quad (14)$$

where $CR_{\min}$ is the proposed minimum crossover probability, and $CR_{\max}$ is the maximum crossover probability, G is the number of current iteration, $MAXGEN$ is the number of maximum iteration. In the early stage of evolution, the crossover probability is smaller, which can improve the global searching capability; in the later stage of evolution, the crossover probability is larger, which can improve the local searching capability.

(4) Estimation and Selection operation: Selection is the procedure by which better-than-average solutions are determined for recombination to generate new offspring. Above-average individuals, which contain good schema, have an above-average chance of passing on their schema to the next generation. In the simple selection scheme, the parent genotypes are assigned a number of offspring based on their ratio of fitness values with the aggregate fitness of the parent pool. The parent is replaced by its offspring if the fitness of the offspring is better than that of its parent.

Contrarily, the parent is retained in the next generation if the fitness of the offspring is worst than that of its parent, according to the following equation:

$$Chrom(i,:)^{G+1} = \begin{cases} trial^{G+1}, f(Chrom(i,:)^{G+1}) < f(trial^{G}) \\ Chrom(i,:)^{G}, otherwise \end{cases}$$

(15)

Usually, the performance of a DE algorithm depends on three variables: the population size NP, the scaling factor F and the crossover probability CR .

3.3 Improving GA operations

GA are probabilistic search optimizing algorithms that were inspired by the process of natural evolution and the principles of "survival of the fittest" (Holland, 1975). A GA starts with an initial random population of feasible solutions; when applied to scheduling, each individual in the population corresponds to one possible solution of the scheduling problem. The algorithm iteratively generates new candidate pool of solutions from presently available solutions and replaces some or all of the existing members of the current solution pool with the newly created feasible solutions. The quality of the solution pool tends to improve with the passage of time as a consequence of the genetic operators. A GA uses both genetic and evolutionary operations. The genetic operations mimic the process of heredity of genes to create new offspring in each generation. The evolutionary operation mimics the process of Darwinian evolution to create populations from generation to generation. The genetic operators are crossover operators and mutation operators; the evolutionary operator is the selection operator.

3.3.1 Modifications in crossover

Crossover is the principal mechanism by which a GA arranges for good schema present on different chromosomes to aggregate on a single individual. New candidate solutions are generated by combining features of selected parent solutions. In genetic algorithm, crossover plays an important role in exchanging information among chromosomes. It leads to an effective combination of partial solutions in other chromosomes and speeds up the search procedure. We try a novel order crossover operator (NOX) in this paper.

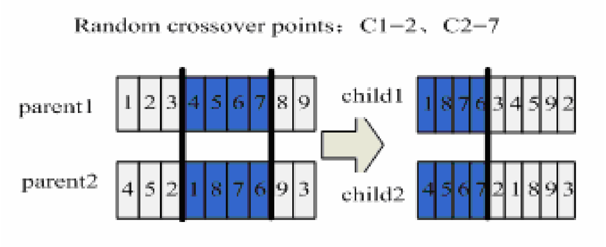

Figure 4.1 Example of the novel order crossover operator.

The NOX operator may be the most useful crossover operator for operating the permutation. It chooses first two crossover points and exchanges the subsection of the chromosome between the two points, and then it fills up in front of the chromosomes. An example of crossover is given in Figure 4.1. The NOX operator can generate the child that different from their parent even if the parent are uniform, which better than other crossover operators.

3.3.2 Improved mutation operator

Mutation operators act upon a single chromosome and produce a new genotype by making a random change to values of one or more of the genes or swapping the value of two or more genes. The principal use of mutation is to reintroduce genetic diversity to avoid getting trapped in local optima. The frequency of mutation is often kept very low to avoid disruption of good solutions. The objective of the mutation is to disrupt the current chromosome slightly by inserting a new gene. In this research we use swapping mutation and inversion operator together as a modified mutation operator.

(1) Swapping mutation

Selects two positions randomly and then swaps the genes on these positions. As shown in Figure 3, from the parent we select two position, i1=2 and i2=6, and the values on position 2 and 6 will exchange form the parent to produce a child. An example of swapping mutation operator is given in Figure 4. 2.

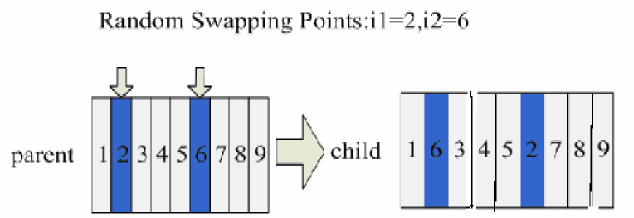

Figure 4.2 Example of the swapping operator.

(2) Inversion operator

Finding out two cutting points within a chromosome randomly and then inverts the substring between these two positions and produce a child. For instance, we select two positions i1=3 and i2=7 in chromosome parents,

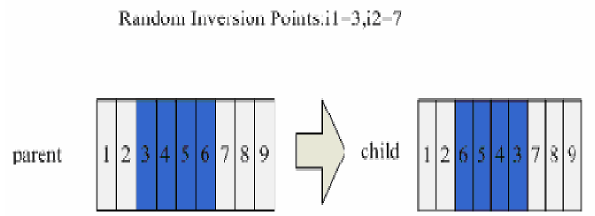

Figure 4.3 Example of the inversion operator.

then invert the substring between position 3 and 7 to produce the child that shorter the total distance of parent. An example of inversion operator is given in Figure 4.3.

3.4 The hybrid optimum algorithm

As we know, there are some complementarities of differential evolution algorithm and genetic algorithm, so we integrated the basic theory of differential evolution and the basic framework of genetic algorithm to design a hybrid optimum algorithm, which can fully make use of their advantage and overcome their disadvantage. We adopted the improving operators of genetic algorithm as an assistant algorithm in order to increase the diversity of population. The computational process of hybrid optimum algorithm for solving VRP-SDPTW is stated using a flowchart as shown in Figure 5.1

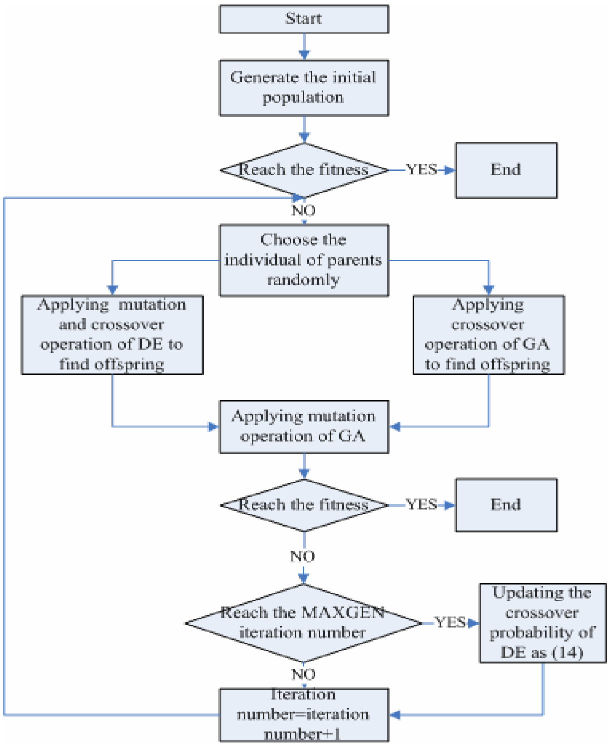

Figure 5.1.Main calculation procedures of the proposed HOA

4. COMPUTATIONAL EXPERIMENTS

The algorithm described in the previous section is coded in MATLAB language and applied to the 8-customer vehicle routing problem with simultaneous delivery and pick-up. There are 3 vehicles in the depot, capacity of each vehicle is 8 tons, delivery and pick-up demands of the 8 customers are listed in Table 1, the distances matrix is listed in (Li Jun, 1999). Parameters for the proposed hybrid optimum algorithm are as follows: NP is 40, $MAXGEN$ is 500, F is 0.5, $CR_{\min}$ =0.3, $CR_{\max}$ =0.9; in genetic operator, probability of crossover operation is 0.9, probability of mutation is 0.1; and the maximum distance L is 400 kilometers.10 independent trials are carried out to evaluate the average performance. The results of simulations are presented in Table 2, and the best solution obtained is showed in Figure 5.3.

Table 1 The delivery and pick-up demands of customers.

Customer i	1	2	3	4	5	6	7	8
Delivery demands	2	1.5	4.5	3	1.5	4	2.5	3
Pick-up demands	3	1	2	2	3	4	1.5	3
Service time	1	0.5	1	1	1	1.5	1	0.8
Time windows	[6,7]	[5,7]	[1,3]	[4,7]	[3,5]	[2,5]	[4,6]	[1.5,4]

Table 2 The optimal computational results of iterating 500.

Vehicles	Distance (Km)	Percentage of reaching optimal solution	Computational time (second)	Routes
3	795	100%	13.0150	0-8-7-2-0;0-6-4-0; 0-3-5-1-0

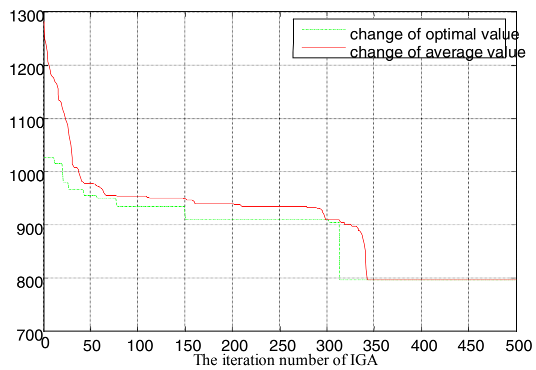

Figure 5.3. The results of iterating 500.

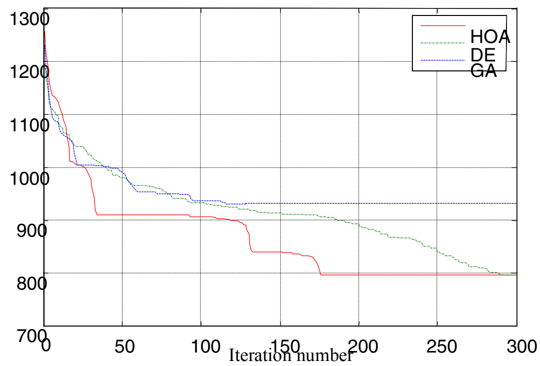

Figure 6.1 .The comparison of HOA with GA and DE

The program is coded using MATLAB language and simulations are performed on a personal computer with 3.06MHz Pentium 4 processor and 1G of RAM, the vehicle that needs is 3 and the runtime is very short. The finally results as follows: the total distance is 795km, needing 3 vehicles finish the services, the first route is 0—3—5—1—0,the delivery and pick-up demand both are 8 tons and achieve full loads, the route distance is 215km and need 7.3 hours finish the service; the second route is : 0—8—7—2—0,the delivery demand is 7 tons, full loads ratio is 87.5%, the pick-up demand is 5.5 tons, the route distance is 315km and need 8.6 hours finish the service; the last route is 0—6—4—0, corresponding delivery demand is 7 tons, full loads ratio is 87.5%, pick-up demand is 6 tons, the route distance is 265km and need 7.8 hours finish the service. According to the tables, the results obtained by our HOA are robust, the distance and time approximate balance of the three route.

We also compared the performance of the proposed HOA with pure genetic algorithm and pure differential evolution, Figure 6.1 described the convergence character and searching capability for VRP-SDPTW, obviously the proposed HOA is better than the others.

5. CONCLUSIONS

In this paper, we described the vehicle routing problem with simultaneous delivery and pick-up in the context of reverse logistic and obtained a mix integer programming mathematic model. Then we presented a hybrid optimum algorithm for the vehicle routing problem with simultaneous delivery and pick-up, decimal permutation encoding was used to represent solution and penalty function was designed to eliminate illegal solutions; the self-adapting differential evolution crossover operation, improving genetic operators were used as core factors to prevent premature convergence and accelerate searching procedure. We observed and compared the performance of the proposed algorithm with pure genetic algorithm and pure differential evolution algorithm. The computational results illustrate that the performance of the proposed HOA is better than the other methods.

REFERENCES

Angelelli, E. Mansini, R. (2003). A branch-and-price algorithm for a simultaneous pick-up and delivery problem, *Working Paper, Article presented at the EURO/INFORMS Meeting.*

Baker Barrie M, Ayechew M. A.(2003) A Genetic Algorithm for the Vehicle Routing Problem, *Computers & Operations Research,* Vol. 30, pp.787-800.

Christian Prins. (2004) A Simple and Effective Evolutionary Algorithm for the Vehicle Routing Problem, *Computers & Operations Research,* Vol.31, pp.1985-2002.

Dethloff, J.(2001).Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up, *OR Spektrum.23*, 79-96.

Gen, M. Cheng, R.(2000).Genetic Algorithms &Engineering Optimization, *New York: Wiley.*

Holland, J. (1975). Adaptation in Natural and Artificial Systems. Ann Arbor, *MI:University of Michigan Press.*

Li Jun. (1999). Generalized assignment heuristics for vehicle scheduling, *System Engineering Theory & Practice,* Vol.17 NO.1, pp. 27 – 33. (in Chinese)

Min H. (1989). The multiple vehicle routing problem with simultaneous delivery and pickup points, *Transportation Research A.* Vol.23, pp.377-386.

Stock J R. (1992). Reverse Logistics, *Council of Logistics Management, Oak Brook , IL.*

Storn R. (1996). Differential evolution design of an IIR-filter, *Proceedings IEEE Conference Evolutionary Computation,Nagoya,* Japan: IEEE, pp.268-273

Tang FA, Galvao RD. (2002). Vehicle routing problems with simultaneous pick-up and delivery service, J*ournal of the Operational Research Society of India (OPSEARCH),* Vol.39, pp.19-33.

Tang FA, Galvao RD. (2006). A tabu search algorithm for the vehicle routing problems with simultaneous pick-up and delivery service, *Computer & Operation Research,* Vol.33, pp.595-619.