Capacitated Vehicle Routing: perturbing the landscape to fool an algorithm

Matthew J. W. Morgan
School of Computer Science
Cardiff University
Queens Buildings
5 The Parade, Roath
Cardiff, CF24 3AA
United Kingdom
M.J.W.Morgan@cs.cardiff.ac.uk

Christine L. Mumford
School of Computer Science
Cardiff University
Queens Buildings
5 The Parade, Roath
Cardiff, CF24 3AA
United Kingdom
C.L.Mumford@cs.cardiff.ac.uk

Abstract- Simple constructive heuristic techniques often product poor solutions when applied to the capacitated vehicle routing problem (CVRP). In this paper we present a hybrid genetic algorithm that lightly perturbs the customer coordinate sets, and "fools" a simple Clarke and Wright heuristic (CW) into producing much better solutions than is the case when CW is applied to the original customer locations. Furthermore, we demonstrate that the new approach compares favourably with other state-of-the-art methods when tested on a set of instances derived from the literature.

1 Introduction

The Vehicle Routing Problem (VRP) is NP-Hard and can be viewed as an amalgamation of two other well known combinatorial optimization problems: the Bin Packing Problem (BPP) and the Travelling Salesman Problem (TSP). The VRP deals with the distribution of products to a group of customers, all of whom must be serviced within a given time frame. This must be achieved using a quantity of vehicles, located at one or more depots and driven by a group of drivers using the road network infrastructure.

Essentially, the solution to this problem requires the determination of a set of routes that fulfil all given constraints and minimise overall tranportation costs. Each route must be undertaken by only one vehicle, which itself must start and return to its allocated depot.

The simplest version of the VRP is known as the Capacitated Vehicle Routing Problem (CVRP) and involves finding a set of routes for a homogenous fleet of vehicles, which must service a set of customers from a central depot. In this paper, assumptions are made that all vehicles depart from and return to the same depot and an unlimited number of vehicles are available, all of which are capable of holding an identical fixed capacity. The quantity demanded by each customer must be less than the capacity of a single vehicle and known in advance. The objective is to minimize the cost of making the customer deliveries.

The CVRP can be formally defined as a graph $G = (V, E)$ with a customer set $V = \{0, 1, ..., n\}$ (customer 0 respresenting the depot) and an edge set E. Each customer $V_i > 0$ has an associated demand $q_i > 0$ and each edge $[i, j]$ a distance $c_{ij} > 0$, with $c_{ij} = c_{ji}$ for the undirected graph. All vehicles are capable of holding capacity Q.

A plethora of algorithms have been proposed to solve the CVRP. In comparison to the simplest heuristics available, such as the Clark & Wright (CW) algorithm, many of the more successful techniques such as Tabu Search (TS), Simulated Annealing (SA) and Parallel Iterative Search methods are rather more complicated. To be truly effective, some even require the use of very powerful computers not available to the average user. Our technique is based around the CW algorithm, is able to be run on standard computer hardware and is very easy to implement.

In this paper we describe a hybridized approach, which uses the CW algorithm within a Genetic Algorithm (GA) framework to breed perturbed coordinate sets and produce high quality solutions to the CVRP.

2 Clarke and Wright

The well known algorithm of Clarke and Wright [3] is based upon the principle of savings and can be applied in a sequential or parallel form. The algorithm begins with each customer assigned to their own distinct route and then combines these routes using a savings criterion, until a solution can be obtained. Two feasible routes can be merged into a single route as long as no constraints are violated in doing so. It is often the case in problems with a limited number of vehicles, that a solution will require more vehicles than those available. This will then result in an infeasible solution.

At the start of the procedure there are n routes, where each route contains exactly one customer. The overall travelling distance for a symmetrical VRP problem is $2\sum_{j=1}^{n} c_{0j}$. As any two routes (containing customers i and j respectively) are merged into a feasible single route, a new route with distance $c_{0i} + c_{ij} + c_{j0}$ is produced. The saving to the overall travelling distance by merging the two routes is :

$$\begin{aligned} s_{ij} &= 2(c_{0i} + c_{0j}) - (c_{0i} + c_{ij} + c_{j0}) \\ &= c_{0i} + c_{0j} - c_{ij} \end{aligned} \tag{1}$$

Thereafter, any two routes (each containing more than one customer) with customer i at one end of the first route and customer j at one end of the second route can be merged

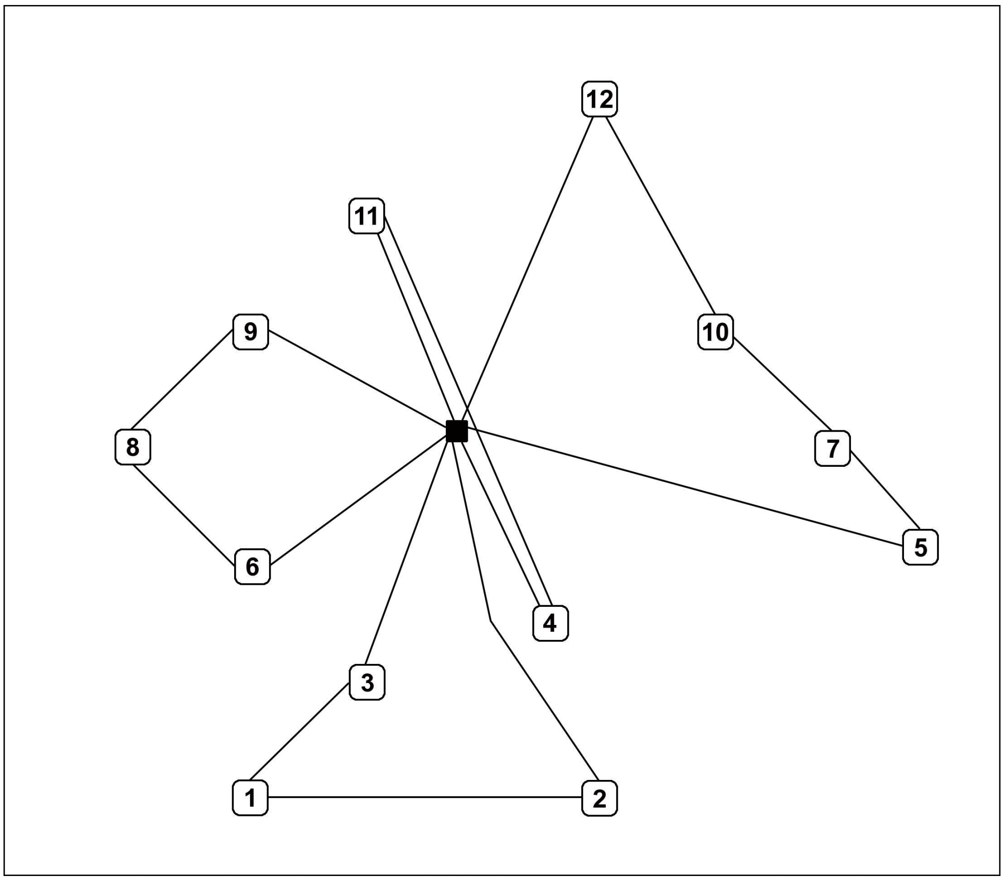

Figure 1: CW solution for a CVRP instance based upon original coordinate data.

to produce an identical saving s_{ij}. The algorithm works as follows:

Step I (Calculate the savings)

a. Calculate the saving $s_{ij} = c_{0i} + c_{0j} - c_{ij}$ for all customer pairs i and j, where $i = 1, ..., n$ and $i \neq j$.
b. Order the savings in descending order.
c. Initialise n vehicle routes $(0, i, 0)$ for $i = 1, ..., n$ and $0 = depot$.

Step II (Sequential Version)

a. Select route $(0, i, ..., j, 0)$ in turn and for each route locate the first feasible saving s_{gi} or s_{jh} which connects the currently selected route to another with edge $(g, 0)$ or $(0, h)$. Join the two routes and continue down the savings list locating any additional feasible savings for the current route.
b. When no more feasible savings are found for the current route, continue selecting the next route and repeating the feasible savings location process (step IIa) from the top of the list until no more feasible savings can be found.

The original algorithm proposed by CW has been shown to have a number of shortcomings, including a tendency to produce circumferential routes and worsening quality routes as it progresses [12]. Many generalised savings definitions have been proposed in the literature to overcome these problems, such as those by Gaskell [7] and Yellow [13].

3 The Perturbation Approach

The first method to incorporate a perturbation model as a means of escaping local optima was an iterated Local Search technique for the TSP, described by Codenitti, Mazini, Margara and Resta [4, 5]. This was extended by Bradwell et al.[1] to take the primary focus away from the original set of coordinates and use a GA to breed perturbed coordinate sets, utilising simple heuristic algorithms as a solution mechanism to solve them.

3.1 The Method Of Perturbed Coordinates

The principle of this method is to lightly perturb (move) all customer coordinates, and use the resulting perturbed coordinates to produce a set of routes using a simple heuristic technique. This set of routes is then used to calculate the total distance travelled based on the original set of customer coordinates.

All coordinates are perturbed within a predefined region around each customer location.The resulting coordinate sets form the ‘chromosomes’ in the population and the CW algorithm is used to produce a solution for each of the per-

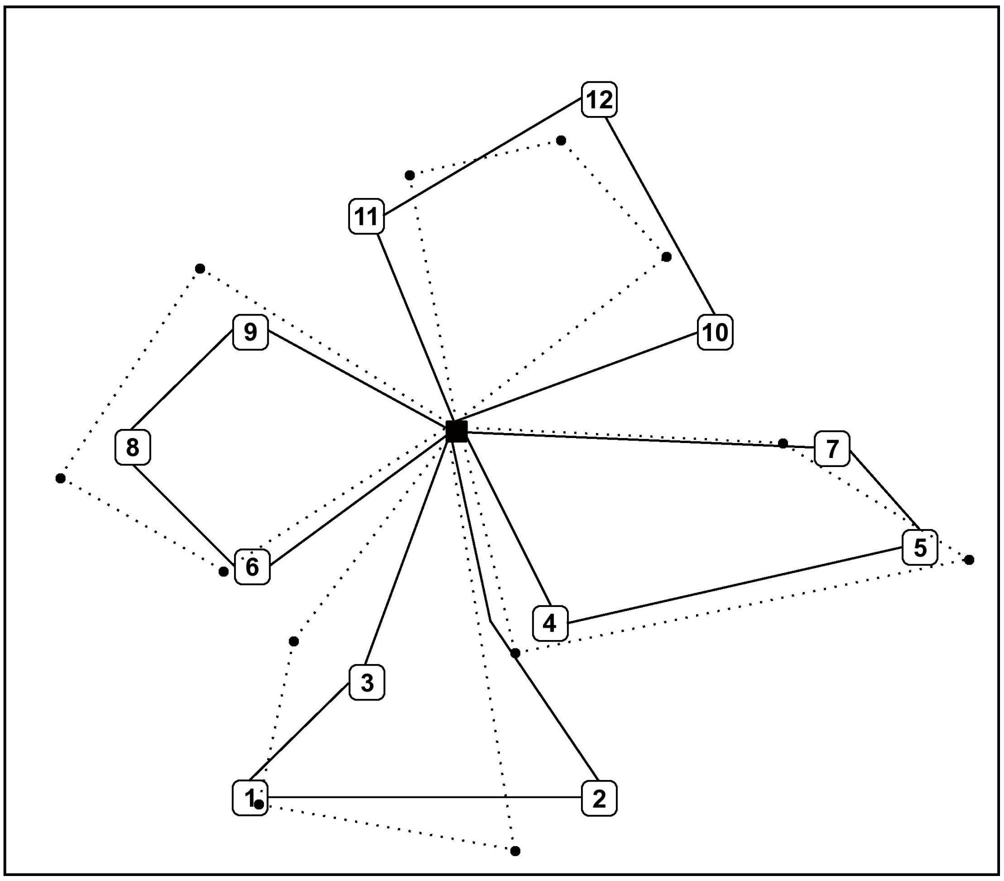

Figure 2: CW solution for CVRP instance of perturbed data (represented by the dotted lines) and the corresponding decoded solution using the original coordinate set. The final solution shows a marked improvement over Figure 1.

turbed coordinate sets in the population at any point in time. The 'true solution' distances calculated using original coordinate data then form the basis for the fitness function in the GA. See figures 1 & 2 for a visual explanation of this method.

3.2 The Perturbation Formula

All x and y coordinates in the initial population are perturbed using the following formula:

$$\begin{aligned} x' &= (int)(x + (r - 0.5) * f * X) \\ y' &= (int)(y + (r - 0.5) * f * Y) \end{aligned} \quad (2)$$

where:

r = random number $0 \leq r \leq 1$

f = perturbation factor

X = $|X_{max} - X_{min}|$, range of X coordinate values

Y = $|Y_{max} - Y_{min}|$, range of Y coordinate values

The same formula is further applied for all mutations throughout the algorithm. The accentuated perturbation achieved from these mutations, together with a crossover operator, provides the mechanism to fool the CW algorithm into producing superior solutions.

4 A New Hybridized Approach

The new approach combines a GA, a method of perturbing customer coordinates and a CW algorithm (see Figure 3).

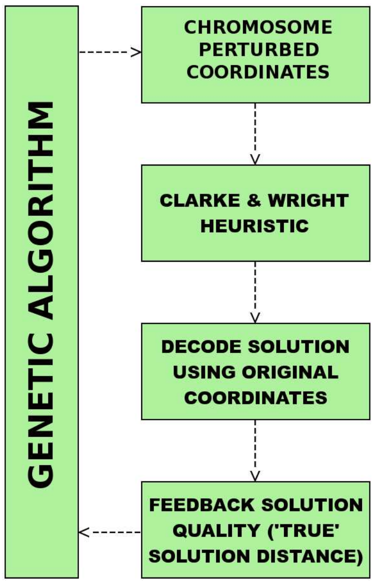
GENETIC ALGORITHM
CHROMOSOME PERTURBED COORDINATES
CLARKE & WRIGHT HEURISTIC
DECODE SOLUTION USING ORIGINAL COORDINATES
FEEDBACK SOLUTION QUALITY ('TRUE' SOLUTION DISTANCE)

Figure 3: Outline structure of the new hybridized approach.

The GA is responsible for evolving new offspring from the population of chromosomes. Each member of the population contains a perturbed coordinate set of customer coordinate pairs from which solutions are then attained using the CW algorithm. A decoding procedure is then used to recalculate the solution using the original customer coordinates. The following sections describe the approach in detail.

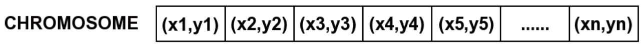

Figure 4: Chromosome representation.

4.1 Population Structure and Initialisation

Chromosomes are encoded as an ordered list of perturbed x and y coordinate pairs (see Figure 4). By using perturbed coordinate sets and employing the CW algorithm as the solution mechanism, the need for a complex representation is negated.

The process for generating the initial population is based upon the techniques described by Bradwell, Williams and Valenzuela [1] for the TSP, where city coordinates are perturbed within a rectangular region around each city. For the CVRP, we use a preset rectangular region around each pair of customer coordinates, sizing the region based upon the results of initial experimentation (see section 6.1).

The initial population consists of p chromsomes, where p equals the chosen size of the population. For each chromosome, customers are selected in order and their x and y coordinates are randomly perturbed using equation 2 described in the section 3.2. This process is then repeated for each required chromosome. An outline of the new approach in the form of pseudocode is detailed in Figure 5.

Initial experimentation suggested a population size of 100 to be sufficient to maintain solution quality and allow efficient running times. This population size is used for all results presented within this paper. However, indications are that solution quality can be maintained with substantially smaller populations. Further work will need to be undertaken to fully substantiate this.

4.2 Crossover

A number of experimental runs were carried out to evaluate the effect of using different crossover operators. The three operators chosen for this purpose were single point (SPX), uniform (UX) and 2-Point (2PX). 30 replicate runs of the GA were made for a range of problems instances from 50 to 150 customers. As expected, all the crossover operators provided different rates of convergance, however it was UX that consistently provided better quality solutions.

4.3 Mutation

A simple mutation procedure is used which consists of randomly selecting m (mutation rate), customers from the offspring chromosome, and further perturbing the selected customers coordinates using the perturbation formula. A set of experiments using a range of perturbation factor values for

```
// initialise population of chromosomes
while $i < p$
{
    for( int $j = 0$; $j <$ noCustomer; $j$++)
    {
        chromosome[$i$][$j$] = perturb(Coords)
    }
    // evaluate fitness of chromosome using C&W heuristic
    distance[$i$] = evaluateCW(chromosome[$i$])
}
// Test termination criteria
while not done do
{
    // Selection
    $P1$ = getChromosome($randomNum$)
    $P2$ = getChromosome($randomNum$)
    // Crossover
    $O1$ = cross($P1, P2$)
    // Mutation
    $O2$ = mutate($O1$)
    // Evaluate mutated chromosome
    $routeSolution$ = evaluateCW(chromosome[$O2$])
    // Decode solution using original coordinate set
    $routeDistance$ = decode(routeSolution)
    // Write back to population if better than weaker parent
    if( $routeDistance <$ distance[$P1$] || distance[$P2$] )
    {
        $distance[weaker] = routeDistance$
    }
}
```

Figure 5: Pseudocode for new hybridized approach

mutation and varying mutation rates were carried out. Each experiment consisted of 15 replicate runs of the GA with different values of f and different rates of mutation on 50, 100 and 200 customer problem instances.

The experimentation established that a high mutation rate of around 10% of total customers and a perturbation factor of double the rate used for the initial population is preferred. This has the effect of exaggerating the movement of the mutated customers and further fools the CW heuristic into producing even better results.

4.4 Solution Mechanism/Decoding Procedure

Following selection, crossover and mutation, the resulting offspring chromosome contains a perturbed coordinate set which is decoded using a CW algorithm. This template solution is stored in a separate array encoded as a list of customers from each route with a delimiter between each route (see Figure 6).

Using the template solution, a decoding sequence is applied to produce a 'true solution' to the CVRP. The first route in the list is extracted and a depot node inserted at the beginning and end of the route. The distance from the depot through each of the customers in the route and back to the depot is calculated using the original set of coordinates. This procedure is then repeated for the remaining routes until the 'true' distances of all routes in the list has been calculated. The overall solution to the CVRP is derived from

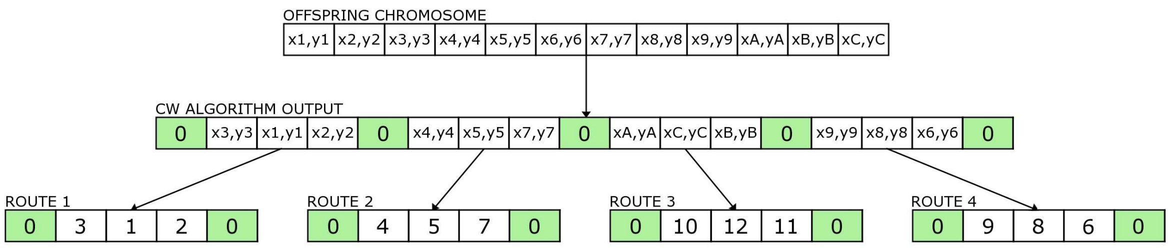
OFFSPRING CHROMOSOME
x1,y1 x2,y2 x3,y3 x4,y4 x5,y5 x6,y6 x7,y7 x8,y8 x9,y9 xA,yA xB,yB xC,yC
CW ALGORITHM OUTPUT
0 x3,y3 x1,y1 x2,y2 0 x4,y4 x5,y5 x7,y7 0 xA,yA xC,yC xB,yB 0 x9,y9 x8,y8 x6,y6 0
ROUTE 1
0 3 1 2 0
ROUTE 2
0 4 5 7 0
ROUTE 3
0 10 12 11 0
ROUTE 4
0 9 8 6 0

Figure 6: Outline of solution and decoding process.

the sum of all calculated route distances.

The overall distance calculated from the offspring is then compared to the solution distance of the parent chromosome used to create it. If the distance is better than the weaker parent, that chromosome is replaced in the population with the generated offsping. Offspring weaker than their parents are discarded.

5 Experimental Results

In this section we present the results obtained from a set of experiments to evaluate the performance of the hybrid algorithm presented in this paper. The algorithm is implemented in Java and the experiments carried out on a computer with in Intel Pentium 4, 2.8GHz processor and using the GNU/Linux Operating System. Experiments are designed to evaluate the relative performance of our algorithm against other metaheuristic approaches.

Table 1: Summary of cvrp dataset instances.

Prob.	Size	Type	Problem Source
1	50	Uniform	Christofides et al. (1979)
2	75	Uniform	Christofides et al. (1979)
3	100	Uniform	Christofides et al. (1979)
4	150	Uniform	Christofides et al. (1979)
5	199	Uniform	Christofides et al. (1979)
6	120	Cluster	Christofides et al. (1979)
7	75a	Cluster	Rochat et al. (1995)
8	75b	Cluster	Rochat et al. (1995)
9	75c	Cluster	Rochat et al. (1995)
10	75d	Cluster	Rochat et al. (1995)
11	100a	Cluster	Rochat et al. (1995)
12	100b	Cluster	Rochat et al. (1995)
13	100c	Cluster	Rochat et al. (1995)
14	100d	Cluster	Rochat et al. (1995)

The datasets chosen as the basis for the experiments are those presented in [2, 10] and shown in the Table 1. The initial instances correspond to problems 1-5 and 11. These have been re-numbered 1-6 respectively for the purposes of this paper. Problems 1-5 are all problem instances of randomly generated uniform points. In contrast problems 6-14 consist of customers which are generated in clusters, to imitate real world problems.

5.1 Scaling the Perturbation Zone

The size of the perturbation region around each customer location confines the movement of each customer to that area. But just how far should the customer coordinates be allowed to be perturbed? It is feasible that the area of the optimal perturbation region varies according to the number of customers n and the overall rectangular region R, comprising all customers within a problem instance.

For each problem we carry out 8 sets of 15 experimental runs for different f settings, in order to analyse the effect of using different perturbation rates. All runs are made with a population size of 100, UX crossover, a mutation rate of 10% of the number of customers in a given problem, and halted after 1500 generations.

Although a value of $f = 0.05$ appears to give marginally better results than other settings overall, the results from these pilot studies are far from clear cut, and indicate that the approach is remarkably robust across a wide range of different values of f, between 0.01 and 0.2. Clearly, more work needs to be done in future to clarify the situation. For the time being, however, we have extracted the best overall result for each set of runs and used it as a basis for comparison in Tables 2 and 3. The full set of results have been included in the Appendix.

Table 2: Comparison of CW algorithm with new hybridized approach.

		Standard	Hybridized Approach	f
Prob.	Size	CW	Best Result	Value
1	50	585.00	524.61	0.05
2	75	900.00	838.60	0.05
3	100	886.00	831.25	0.08
4	150	1204.00	1046.72	0.02
5	199	1504.00	1319.11	0.03
6	120	1079.00	1042.11	0.03
7	75a	1645.50	1618.36	0.05
8	75b	1356.56	1344.62	0.05
9	75c	1334.84	1291.01	0.05
10	75d	1428.53	1365.42	0.05
11	100a	2166.04	2054.62	0.1
12	100b	2034.31	1940.36	0.05
13	100c	1434.89	1406.20	0.05
14	100d	1682.25	1591.32	0.05

5.2 Comparisons with Other Methods

The best results achieved using the new hybridized approach in comparison to those achieved using the standard CW algorithm are shown in Table 2. It can be clearly seen that using perturbed coordinates in conjunction with the CW algorithm allow vastly superior solutions to be attained than is possible with the standard CW algorithm.

The best overall results are further compared to results achieved with other metaheuristic techniques applied to the same data instances in Table 3. Most notably, the new hybridized approach produces 2 new best-known solutions for problems 75b and 100b. For all other problem instances, solutions either equal or lie within approximately 1% of the other metaheuristics. Considering the ceiling of 1500 generations applied to each run, it would seem feasible that the initial results achieved from this preliminary study may be further improved by running the algorithm for more generations. In some cases, further investigation will need to be undertaken to establish this with any certainty.

Table 3: Results of hybridized approach.

Prob.	Size	Hybridized Approach Best Result	Avg Time	Best Solution Value
1	50	**524.61**	1m 20s	**524.61 [11, 8, 9]**
2	75	838.60	2m 15s	**835.26 [11]**
3	100	831.25	5m	**826.14 [11]**
4	150	1046.72	12m	**1028.42 [11]**
5	199	1319.11	15m	**1298.79 [11]**
6	120	**1042.11**	8m	**1042.11 [11, 8, 9]**
7	75a	**1618.36**	2m 15s	**1618.36 [10]**
8	75b	**1344.62**	2m 15s	1344.64 [10]
9	75c	**1291.01**	2m 15s	**1291.01 [10]**
10	75d	**1365.42**	2m 15s	**1365.42 [10]**
11	100a	2054.62	5m	**2041.34 [6]**
12	100b	**1940.36**	5m	1940.61 [10]
13	100c	**1406.20**	5m	**1406.20 [6]**
14	100d	1591.32	5m	**1581.25 [6]**

6 Conclusions and Future Work

Within this paper we present a preliminary study which outlines the use of a new hybridized algorithm for solving the CVRP, competitive with current state-of-the-art methods. The algorithm exploits a GA which breeds perturbed customer coordinate sets and uses a simple CW algorithm as a solution mechanism. Although a GA was chosen for this initial investigation, it could be easily substituted for an alternative optimization method such as SA or TS and would provide an interesting point for further investigation.

In this preliminary study experimental runs are limited to 1500 generations. Our main objective is to improve on the standard CW algorithm and verify the potential of using perturbed coordinate sets in conjunction with other algorithmic techniques. Clearly this technique has managed to lift the solution quality above that of the standard CW algorithm. In fact, the algorithm has provided 2 best-known solutions and matched other state-of-the-art algorithms for many of the other problem instances.

We have shown that it is possible to produce high quality solutions to the CVRP using a very simple hybridized algorithm. Harnessing a basic GA as a container to breed perturbed customer coordinate sets has clearly succeeded in fooling the CW algorithm into generating far superior quality solutions when compared to those attained using standard coordinate sets. Indeed, our method generally lifts the solution quality by about 8%.

In comparison to some of the best published results achieved for these datasets using alternative approaches, most notably the TS [8], SA [9] and a Parallel Iterative Search Method [11], our algorithm performs formidably. Solutions from our approach are within about 1% of these more complicated implementations and we believe that the relative quality attained in conjunction with simplicity of our approach allows great scope for the use of this method on both small and large problem instances. Although the validity of our method needs to be properly established through formal comparative studies with other state-of-the-art methods.

Work is currently underway to incorporate alternative construction based heuristics for solving the CVRP into the GA framework. This work will include an empirical study of the strengths and weaknesses of different heuristics when applied to perturbed coordinate sets, including a full statistical analysis of dataset solutions and a thorough evaluation of the effect of different crossover operators. A further detailed investigation into the related size of the perturbation zone and problem size, in conjunction with an analysis of the effect of using different shaped perturbation zones, will then be carried out. The algorithm will finally be extended to solve the Vehicle Routing Problem with Time Windows (VRPTW).

Bibliography

[1] R. Bradwell, L. P. Williams, and C. L. Valenzuela. Breeding perturbed city coordinates and fooling travelling salesman heuristic algorithms. In *International Conference on Artificial Neural Networks and Genetic Algorithms (ICANNGA97)*. Norwich, 1997.

[2] N. Christofides, A. Mingozzi, and P. Toth. *Combinatorial Optimization (N. Christofides, A. Mingozzi, P. Toth and C. Sandi, eds.)*. Wiley, Chichester, 1979.

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

[4] B. Codettini, G. Manzini, L. Margara, and G. Resta. Global strategies for augmenting the efficieny of TSP heuristics. In *Proc. 3rd Workshop on Algorithms and Data Structures*, pages 253–264. Lecture Notes in Computer Science, Vol 709, Springer-Verlag, Berlin, 1993.

[5] B. Codettini, G. Manzini, L. Margara, and G. Resta. Perturbation: an efficient technique for solution of

very large instances of the euclidean TSP. *INFORMS Journal on Computing*, 8(2):125–133, 1996.

[6] L. M. Gambardella, ’E. D. Taillard and G. Agazzi MACS-VRPTW: A multiple any colony system for vehicle routing problems with time windows. *Technical Report IDSIA-09-99*, IDSIA, Lugano, Switzerland, 1999.

[7] T. J. Gaskell. Bases for vehicle fleet scheduling. *Operational Research Quarterly*, 18:281–295, 1967.

[8] M. Gendreau, A. Hertz, and G. Laporte. A tabu search heuristic for the vehicle routing problem. *Management Science*, 40(10):1276–1290, 1994.

[9] I. H. Osman. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. *Ann. Operations Research*, 41:421–451, 1993.

[10] Y. Rochat and É. D. Taillard. Probabilistic diversification and intensification in local search for vehicle routing. *Journal of Heuristics*, 1:147–167, 1995.

[11] É. D. Taillard. Parallel iterative search methods for vehicle routing problems. *Networks*, 23:661–673, 1993.

[12] P. Toth and D. Vigo. *The Vehicle Routing Problem (P. Toth and D. Vigo, eds.), SIAM monographs on discrete mathematics and applications*. Society for Industrial and Applied Mathematics, 2002.

[13] P. Yellow. A computational modification to the savings method vehicle scheduling. *Operational Research Quarterly*, 21:281–283, 1970.

Appendix

		f = 0.01		f = 0.02		f = 0.03		f = 0.04	
Prob.	Size	Best	Avg	Best	Avg	Best	Avg	Best	Avg
1	50	524.93	526.74	525.93	526.82	524.93	525.72	524.61	524.97
2	75	838.60	840.16	838.60	839.72	839.10	843.67	846.40	852.16
3	100	836.38	837.14	836.89	837.38	839.87	840.24	836.85	838.32
4	150	1058.48	1059.03	1046.72	1052.27	1058.19	1059.43	1064.54	1066.92
5	199	1335.82	1340.13	1332.68	1336.25	1319.11	1326.48	1345.16	1349.57
6	120	1045.62	1045.97	1046.35	1046.32	1042.11	1043.87	1046.62	1047.44
7	75	1622.24	1622.24	1620.70	1621.63	1618.36	1619.56	1618.36	1620.26
8	75	1344.62	1344.63	1344.62	1344.66	1344.62	1344.62	1344.62	1344.62
9	75	1291.01	1291.01	1291.01	1291.01	1291.01	1291.01	1291.01	1291.01
10	75	1389.32	1393.00	1389.32	1390.43	1386.70	1387.57	1383.99	1386.87
11	100	2072.64	2074.83	2073.06	2075.03	2067.57	2071.38	2072.43	2074.89
12	100	1948.56	1951.43	1940.97	1945.68	1942.81	1945.15	1940.70	1943.55
13	100	1406.24	1406.31	1406.20	1406.21	1406.20	1407.06	1406.24	1406.81
14	100	1598.36	1599.06	1598.36	1599.72	1598.38	1600.32	1601.30	1603.75

		f = 0.05		f = 0.08		f = 0.1		f = 0.2	
Prob.	Size	Best	Avg	Best	Avg	Best	Avg	Best	Avg
1	50	524.61	527.96	525.13	528.63	531.52	539.86	532.99	538.27
2	75	838.60	841.37	839.88	843.21	845.75	848.36	854.14	861.94
3	100	840.11	842.28	831.25	832.95	835.17	836.01	880.81	852.42
4	150	1060.75	1064.77	1065.68	1066.45	1072.47	1073.69	1096.78	1116.39
5	199	1359.09	1364.92	1355.00	1359.18	1357.94	1366.84	1449.38	1453.64
6	120	1045.13	1046.83	1046.33	1049.73	1044.91	1046.72	1061.21	1067.46
7	75	1618.36	1618.36	1618.36	1618.36	1618.36	1618.49	1629.12	1631.76
8	75	1344.62	1344.62	1345.01	1345.34	1345.92	1346.23	1360.58	1362.66
9	75	1291.01	1291.01	1291.01	1291.01	1291.01	1291.01	1298.62	1301.92
10	75	1365.42	1369.76	1366.48	1368.48	1370.89	1381.56	1404.83	1404.15
11	100	2062.90	2063.14	2062.38	2067.81	2054.62	2059.13	2108.82	2116.31
12	100	1940.36	1940.67	1941.26	1942.20	1947.56	1949.39	1976.10	1988.58
13	100	1406.20	1406.80	1411.38	1413.82	1419.73	1422.37	1462.49	1468.34
14	100	1591.32	1597.64	1599.53	1601.97	1594.42	1597.34	1637.94	1658.71