AN ANT SYSTEM ALGORITHM FOR A VARIETY OF VEHICLE ROUTING PROBLEMS WITH BACKHAULS [1]

S. SALHI[a] and A. C. WADE[b]

[a] *Management Mathematics Group*
School of Mathematics and Statistics
The University of Birmingham, UK
[b] *BAE Systems, CDI Division, Filton, Bristol, UK*

We introduce an ant system heuristic to solve a variety of vehicle routing problem with backhauls (VRPB). Some enhancements to the general characteristics of ant system algorithms are embedded into the search. We investigate three type of VRPB namely the classical, the fully mixed and the new version which sits between these two extremes. Computational results on test problems are reported.

keywords: metaheuristic, ant system, vehicle routing with backhauls, restricted backhauling.

1 Introduction

The vehicle routing problem with backhauls (VRPB) is an extension to the vehicle routing problem (VRP) where two different types of customer are served. The first type are linehaul customers, also known as delivery customers, who require a given quantity of product to be delivered. The second type are backhaul customers, also known as pickup customers, who require a quantity of goods to be picked up and sent back to the depot. The goods are transported to or from a single depot by a fleet of homogeneous vehicles. Methods in the literature to solve the VRPB either restrict all backhauls to be visited once all linehauls have been served, the classical VRPB, or allow mixed linehaul and backhaul customers along the routes, the mixed VRPB. Recently Wade and Salhi [2002] explored a new VRPB version where the insertion of the backhauls is permitted only after a certain position along the route. Such a flexibility is based on the user experience, the vehicle capacity, the product types, among others. In this paper we propose an ant system to solve these three type of VRPBs. The objective is to find the set of routes with the least cost where all customer demands are satisfied, each customer is visited exactly once, all routes begin and end at the depot, and finally the load on the vehicle at any point along the route must not exceed the vehicle capacity.

In the classical VRPB it is only necessary to check that the total linehaul load and the total backhaul load do not separately exceed the total vehicle capacity. For more details and references on the classical VRPB see Toth and Vigo [1996], Toth and Vigo [1997], Goetschalckx and Jacobs-Blecha [1989] and Mingozzi et al. [1999]. In the mixed VRPB it is necessary to check that the vehicle capacity is not exceeded at any point along the route.

[1] An earlier version of this paper was given at IFORS'02, Edinburgh, July, 2002.

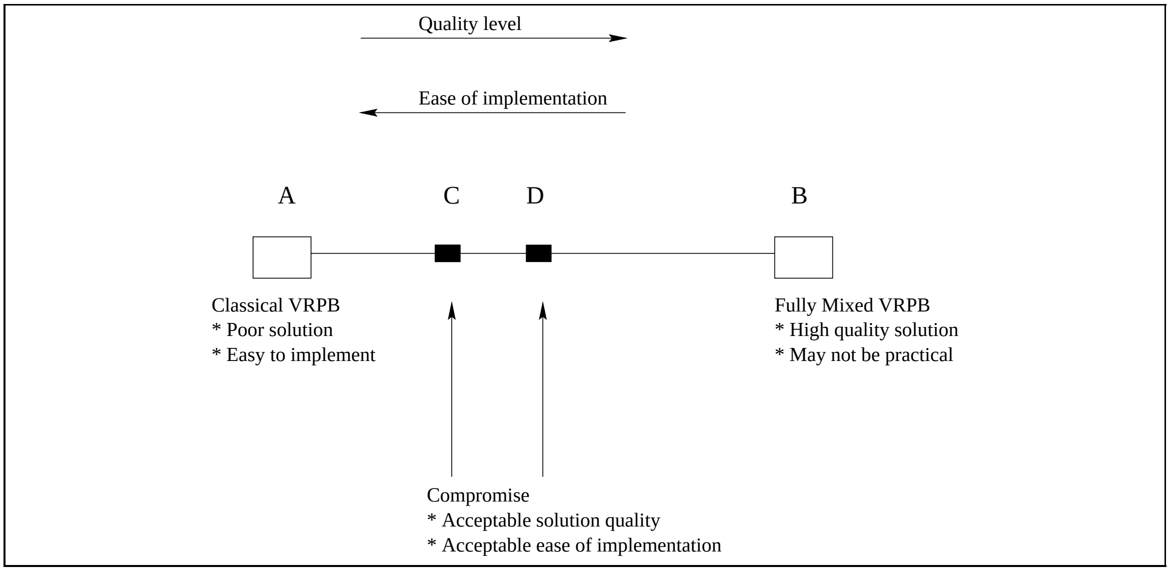

Figure 1: A Compromise VRPB (Wade and Salhi [2002])

In other words, in the mixed VRPB the vehicle load can either decrease or increase at each customer depending on whether the customer has a linehaul or a backhaul demand, respectively. The mixed VRPB has not received as much attention in the literature as its counterpart the classical VRPB. For further details and references see Deif and Bodin [1984], Casco et al. [1988], Halse [1992], and Salhi and Nagy [1999]. The improved design of vehicles, in particular vehicles that permit side loading mean that mixed routes are now a much more practical option and therefore worth investigating. However great the saving, the practical problems associated with mixed routes can still be an issue. A practical compromise between the classical VRPB and the mixed VRPB is proposed by Wade and Salhi [2002] where backhauls are allowed to be inserted based on a certain mix chosen by the user. The balance between a high quality solution and the ease of implementation, as shown in Figure 1, is an important practical issue which we believe ought to receive more research attention.

In this paper, we propose an ant system metaheuristic to solve the above three type of vehicle routing problem with backhauls (VRPB).

The paper is organised as follows; in the remaining of this section we present the fundamentals of ant system algorithms and introduce successful ant system methods used to solve the TSP and the VRP. Section 2 describes the ant system used to solve the fully mixed VRPB and the necessary modifications which are embedded into the search. The adaptation of the heuristic to the restricted VRPB, including the classical VRPB, are given in section 3. Computational results are reported in Section 4 and the conclusions and ideas for future research are summarised in Section 5.

Ant Systems

The ant system is a metaheuristic inspired by the observation of the behaviour of real life ant colonies, in particular the way in which real ants find the shortest path between food sources and their nest. While walking ants deposit a substance called pheromone on to the ground which forms a pheromone trail. Ants can detect the pheromone and choose their way according to the level of the pheromone trail. The greater the concentration of pheromone on the ground the higher the probability that an ant will choose that path. For more information see Goss et al. [1989] and Beckers et al. [1992].

Real ants, though move independently of each other, their collective behaviour is achieved via a form of indirect communication, the pheromone trail. The simulation of the real life behaviour of ants has been used to solve combinatorial optimization problems. The research community adopts an approximation to model such a complex behaviour as will be briefly described below, but any sensible approach that attempts to imitate such a behaviour will be most welcome for further research.

Ant system algorithms employ a set of agents, known as ants, who search in parallel for good solutions using a form of indirect communication. The artificial ants co-operate via the artificial pheromone level deposited on arcs which is calculated as a function of the quality of the solution found. The amount of pheromone an ant deposits is proportional to the quality of the solution generated by that ant helping direct the search towards good solutions. The artificial ants construct solutions iteratively by adding a new node to a partial solution exploiting information gained from both past performance, i.e. pheromone levels, and a greedy heuristic. The greedy heuristic, known as the visibility, is introduced in an attempt to guide the search.

The Ant System algorithm (AS), first proposed by Dorigo [1992] and Colorni et al. [1991] to solve the travelling salesman problem uses artificial pheromone trail values, τ_{ij}, associated with each arc (i, j). Initially m ants are placed on randomly selected nodes and each ant starts constructing a tour from this starting position. The tour is built by each ant successively choosing the next customer to visit probabilistically. The probability that an ant building its t^{th} tour currently situated at customer i will visit customer j next is given by a state transition rule (for instance see Dorigo [1992]). Once all ants have constructed a complete tour, i.e. all customers have been visited, then the pheromone trail levels are updated by each ant according to the global update rule (for instance see Dorigo [1992]).

A number of modifications have been proposed to the AS algorithm in order to improve the solution quality. The elitist strategy which was proposed by Dorigo et al. [1996] is based on a modified global updating rule where the arcs belonging to the global best tour received an extra amount of pheromone. This idea is then generalised by Bullnheimer et al. [1997] to include not only the global best tour but also a number of the best tours to update trail values. The ants are ranked according to their tour length and the amount of pheromone an ant adds is proportional to its rank. The Ant Colony System (ACS) proposed by Dorigo and Gambardella [1997] include three main modifications: a different transition rule based on the pseudo-random-proportional rule, a pheromone global updating rule based only on the global best solution, and a local updating rule that adjusts the pheromone level on the selected arcs. Stutzle and Hoos [1997a], Stutzle and Hoos [1997b] introduced the Max-Min ant system, known as MMAS. Their method is basically similar to AS except that pheromone trail values are restricted to the interval $[\tau_{min}, \tau_{max}]$ to avoid stagnation. The introduction of a lower limit on pheromone trail values was found to have the greatest effect on the solution quality.

Ant systems have been used successfully to solve many hard combinatorial problems such as the travelling salesman problem, see Dorigo et al. [1996], Dorigo and Gambardella [1997], Stutzle and Hoos [1997a] and Cordon et al. [2000], the quadratic assignment problem, see Maniezzo et al. [1994] and Maniezzo and Colorni [1999], the sequential ordering problem, see Gambardella and Dorigo [1997] and the vehicle routing problem see Bullnheimer et al. [1997], Bullnheimer et al. [1998] and Gambardella et al. [1999].

The use of ant systems to vehicle routing problems is briefly described below. Bullnheimer et al. [1997,1998] propose an ant system based upon their AS_{Rank} method to solve the VRP. A visibility function which incorporates the saving measure and capacity utilization was used. Improved results were obtained but at the expense of large computation time. In Bullnheimer et al. [1998] a parametric savings function is adopted for the visibility producing an improvement in solution quality as well as a reduction in computation time. Gambardella et al. [1999] solve the VRP by transforming the original problem into M-TSPs where M is the number of vehicles. Each ant builds its tour so that the vehicle capacity is not exceeded. Each solution that is generated contains M tours each corresponding to the route of a vehicle. A local search heuristic comprising edge exchanges is applied to the solutions. Gambardella et al. also solve the vehicle routing problem with time windows (VRPTW) using an ant algorithm. Two ant colonies are utilized where the first ant colony is used to minimize the number of vehicles and the second to minimize the total travel time given the number of vehicles determined by the first ant colony. The pheromone trail levels are independent for each colony but the best ants from one colony are allowed to update trail values associated with the other colony. The results reported by the ant-based methods have been shown to be competitive when compared with known methods in the literature to solve the VRP and the VRPTW.

For an overview on ant systems the book of Bonabaeu et al. [1999], the review paper by Dorigo et al. [1999] and the technical report by Dorigo and Stutzle [2000] are useful references.

2 An Ant System Heuristic for the VRPB

An algorithm based upon the Ant Colony System is adapted to solve the three types of VRPBs. Very recently Wade and Salhi [2003] proposed a number of modifications to the ACS method for the fully mixed VRPB. The basic ant algorithm can be described as follows. Initially m ants are positioned at starting customers. Each ant selects the next customer to visit according to the state transition rule which will be looked at in this section. Once an arc (i,j) has been selected by an ant then a local updating rule is applied to reduce the trail value of that arc. Each ant constructs its tour selecting one customer at a time. If no feasible customers are available to be visited due to the vehicle capacity constraint then the depot is selected by the ant. The ant then continues to build a new route repeating the process until all customers have been selected. A local search algorithm is applied to improve each solution after each ant has completed its tour. Note that exchanging a linehaul or backhaul customer in the VRPB can affect the load on the vehicle and could violate the vehicle capacity at other nodes along the route.

In the remainder of this section we shall briefly describe the modifications proposed to the Ant Colony System for the fully mixed VRPB (see Wade and Salhi [2003] for more details).

A Site-Dependent Candidate List

The purpose of the candidate list, also known as a restricted neighbourhood, is to reduce the possible number of moves that an ant must consider by ignoring far away customers which are unlikely to produce a good quality solution. It can be shown that a candidate list that is site dependent and which that takes into account the sparsity of customers in the neighbourhood of each customer is more adaptive to the characteristics of the problem. For each customer $i (i = 1, \ldots, N)$ we compute

$$C_i = \frac{\sum_{\{j:d_{ij}<D\}} d_{ij}}{|\{j : d_{ij} < D\}|}$$

where

$$D = \frac{\sum_{k=1}^{N} \tilde{d}_k}{N}$$

$\tilde{d}_k$ is the average distance of customer k to the other $N-1$ customers,
d_{ij} denotes the euclidean distance between customer site i and customer site j.
For completeness if $\{j : d_{ij} < D\} = \emptyset$, we set $C_i = 0$.
The candidate list is then found for each customer i as

$$E_i = \{j = 1...m \quad s.t. \quad d_{ij} \leq R\} \tag{1}$$

where R is the average of the C_i values.

If $|E_i| < M$ where M is a minimum threshold set to $Min\{N/4, 10\}$, then the nearest $M - |E_i|$ customers not already contained in E_i are added to E_i.

Strategic Placement of Ants

The placement of ants on the nodes is performed as followed: The neighbourhood is divided into a given number of sectors. The aim of this procedure is that the sectors should roughly correspond to the routes that would be created in a good solution. The sectors are constructed so that each sector contains an equal number of customers except the final sector which may have the remaining customers. Within each sector an ant is placed at the closest and farthest customer in relation to the depot. A given number of ants are then placed randomly on nodes remaining in each sector. For each iteration of the ant algorithm the starting position in the calculation of the sectors is rotated. This procedure should result in an even spread of ants being selected for each iteration. In other terms the number of ants, K_0: $2K \leq K_0 \leq N$ where K denotes the number of vehicle routes.

A Look Ahead-Based Visibility

The visibility is usually a greedy heuristic used within the transition rule in an attempt to guide the search. Two different visibility functions are used, the choice of which is dependant on the capacity remaining on the vehicle.

(i) If the vehicle is nearly full then a visibility that uses the insertion cost between the

current customer and the depot would be a useful selection rule in order to encourage the vehicle to travel closer to the depot towards the end of each route.
Let the unused capacity on the vehicle be (Q_V) and the average linehaul demand ($\bar{q_L}$), then the visibility is calculated as in equation (2).

$$\text{If } Q_V \leq \alpha_0 \times \bar{q_L} \text{ set } \eta_{ij} = \frac{1}{d_{ij} + d_{j0}} \tag{2}$$

where j is the customer to which the ant is considering to move, customer 0 is the depot, and α_0 is a correction factor.

(ii) At the beginning of routes, before the above cut off point is reached, a visibility that takes into account a chain of two customers is used. When considering the next customer j to be visited then the nearest customer k in relation to j is incorporated into the visibility detailed in Equations (3) and (4).

$$p_{ij}^k = \begin{cases} argmax_{l \in F_i^k} \{[\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta \cdot [\kappa_{il}]^\phi\} & \text{if } q \leq q_0 \\ P & \text{if } q > q_0 \end{cases} \tag{3}$$

where

$$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta \cdot [\kappa_{ij}]^\phi}{\sum_{l \in F_i^k} [\tau_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta \cdot [\kappa_{il}]^\phi} \tag{4}$$

$$\kappa_{ij} = \frac{1}{d_{ij} + d_{jk}}$$

$$\eta_{ij} = \frac{1}{d_{ij}}$$

The idea is to incorporate such a look ahead strategy that takes into consideration the next possible move following the selection of customer j. To reduce the computational burden no check for feasibility when considering customer k in the chain is performed. Note that modifications to the visibility that increase complexity could obviously improve solution quality but at a significant increase in computation time due to the calculation of the probability that needs to be carried out several times.

A Frequency-Based Local Trail Update

A procedure that records the number of times each arc has been selected is adopted. If an arc has been selected a greater number of times than a given percentage of the number of solutions that have been generated since the beginning of the algorithm, then the trail value on that arc is updated according to the local update rule given by Dorigo [1992] except that the parameter γ, as detailed in Equation (5) is added.

$$\tau(i,j) = (1 - \gamma \upsilon) \cdot \tau_{ij} + \upsilon \cdot \tau_0 \tag{5}$$

where υ is a pheromone decay parameter in the range $]0, 1[$, γ is the adjustment factor ($\gamma > 1$), and τ_0 refers to the initial pheronome trail level set as $\frac{N}{cost(H)}$ where $cost(H)$ refers to the cost obtained by a given heuristic (H).

The effect is to reduce the pheromone trail value by a greater amount on arcs that have been selected many times compared to those that have only be selected by a few ants. This procedure is designed to reduce the possibility that arcs are selected repeatedly and to encourage exploration of the search space. It is also intended that customers which are chosen towards the beginning of a route by one ant will be selected later along a route by other ants so that mixing occurs on routes. This procedure is only implemented after a limited number of complete iterations of the algorithm have been performed. For instance, in our experiments we opted for three complete iterations. The aim is to allow the ants to generate routes freely according to the trail values before any influence is applied.

Combining Strategies for Global Trail Update

We use a global trail update based on the rank based version with elitist strategies as proposed by Bullnheimer et al. [1997] but with the addition that at most λ best solutions need to be within a given percentage, θ of the global best solution instead of selecting λ best solutions regardless of their relative quality to the overall best solution obtained so far. The constraint is imposed to avoid poor quality solutions being used to update the trail values which are adjusted according to Equation (6).

$$\tau_{ij}(t) = (1-\rho)\cdot\tau_{ij}(t) + \lambda\Delta\tau_{ij}^{gb} + \Delta\tau_{ij}(t) \tag{6}$$

where $\Delta\tau_{ij}^{gb}(t) = Q/L^{gb}(t)$, L^{gb} being the global best solution and Q a correction factor to normalise the data.

$\Delta\tau_{ij}(t) = \sum_{\mu=1}^{\lambda-1}\Delta\tau_{ij}^{\mu}(t)$.

$$\Delta\tau_{ij}^{\mu}(t) = \begin{cases} (\lambda-\mu)Q/L^{\mu}(t) & \text{if } (i,j)\in \text{tour of ant with rank } \mu \\ 0 & \text{otherwise} \end{cases}$$

$L^{\mu}(t)$ is the length of the μ^{th} best tour at iteration t.

Feasibility Checks

The checks need to be implemented when updating the feasible list to ensure that visiting a customer will not violate the maximum vehicle capacity at any point along the route. This is necessary for the mixed VRPB because of the fluctuating load on the vehicle. Note that in the classical VRPB it is only necessary to check that the total linehaul load and the total backhaul load on the vehicle do separately exceed the capacity of the vehicle.

The algorithm

The main steps of the proposed ant system heuristic is given in Figure 2. It consists of 3 main steps namely the initialisation step, the construction of the routes by the ant at each iteration. This relies on appropriate visibility rules, transition rules and local updating besides possible refinement of each solution once completed. In step 3 a global adjustment is performed using information of thsoe best solutions. The cycle between steps 2 and 3 is repeated until the maximum number of iterations is performed.

Step1 (Initialisation):

- Fix the number of ants to be used, K_0, and the trail values $\tau_{ij}, i, j = 1, \ldots, N$,
- Position the K_0 ants on the corresponding nodes (see subsection on strategic placement),
- Construct the candidate list $E_i, i = 1, \ldots, N$ using Equation (1), and
- Set iteration counter to 0 ($iter = 0$).

Step 2 (Route Construction):

For each ant k ($k = 1, \ldots, K_0$) do the following

Step 2a- Select the next customer to visit using a suitable state transition rule as given by Equations (2-4).

Step 2b- Update the pheromone trail for the selected arc using a local updating rule as defined by Equation (5).

Step 2c- If there are some customers not visited yet

Go to step 2a to continue building the current route or to start a new route,

Otherwise

Apply appropriate refinement procedures to improve the solution for each ant (See subsection on local search heuristics).

Step 3 (Global Updating and Termination Criterion):

Set $iter = iter + 1$,

If $iter < itermax$ update pheromone trail values $\tau_{ij}, i, j = 1, \ldots, N$ using the global updating rule as given by Equation (6), and go to step 2,

Else retain the last best solution as the final solution and stop.

Figure 2: A Skeleton for an Ant System Algorithm for the VRPB

3 The Restricted VRPB

As described in section 1, the restricted VRPB attempts to address the middle ground between the classical VRPB and the mixed VRPB. In this section we adapt the ant system algorithm described in the previous section to solve the restricted VRPB which has been tackled only once by the authors using a simple insertion type method, see Wade and Salhi [2002].

Controlling the Mix of Customers

In the insertion-type method of Wade and Salhi [2002] the mix of customers on each route was controlled by the restriction that a backhaul cannot be visited until a given restriction percentage (RP) of the total linehaul demand on that route has been delivered. This criterion is possible to implement because the routes are first created containing linehaul customers only and then backhaul customers are inserted onto these routes. This is not the case with the ant system algorithm where each ant builds a route iteratively, adding a linehaul or a backhaul customer to a partial solution. Therefore the total linehaul demand of the route is not known when a backhaul customer is considered as the next choice. There are several schemes that would help to control the mix of linehaul and backhaul customers on routes. However, in order to compare our ant system method with the results of the refined version of Wade and Salhi [2002], we address this problem by initially setting the restriction percentage in relation to the total capacity of the vehicle instead as the total linehaul demand of each route. Therefore, such a restriction will guarantee that a backhaul customer will not be visited until at least the original RP was satisfied. This restriction may initially result in the routes being less mixed than is permitted by the restriction used in the RINS heuristic which may cause a higher solution cost. However once an ant has built its tour then the total linehaul demand of each route is known, and therefore the original RP is then used when the local search heuristics are employed. The structure of the local search heuristics will be looked at in the following subsection.

Local Search Heuristics

Local search heuristics are introduced to guide the ants towards good quality solutions. It is important that feasibility checks carried out in these improvement procedures are as efficient as possible because of the large number of solutions generated.

2-Opt

The local search heuristic 2-opt was first proposed by Lin [1965]. This scheme attempts to improve each route by exchanging the two customers in that route which give the greatest saving in cost. Note that the direction of the route between two customers is reversed by the 2-opt procedure. Due to the fact that the load on the vehicle can decrease or increase at a customer stop then it is necessary to check the load on the vehicle along the segment between the two swapped customers. However for the revised VRPB it is also necessary to check that a backhaul customer is not visited until a given percentage of linehaul load has been delivered.

3-Opt

The 3-opt heuristic adopted is a modified version to the procedure first put forward by Lin [1965] and implemented by Salhi and Rand [1987]. In this algorithm not all possible exchanges of arcs are performed but a limited number only, those which are in a sequence of at least two arcs. For our restricted VRPB we also need to ensure that the linehaul constraint is satisfied by this exchange.

Shift

The heuristic shift involves two routes. A customer is removed from one route and inserted into an arc of another route. It is necessary to check the maximum load on the route receiving the customer. For the revised VRPB it is necessary to check that both routes still satisfy the linehaul restriction as the removed customer can be either a linehaul or a backhaul.

Placement of Starting Ants

In the previous section, a method which strategically place a restricted number of ants at starting nodes is used. In our restricted VRPB it is not appropriate to consider a backhaul customer as the starting position of an ant as a given percentage of linehaul load must be delivered first before a backhaul can be visited. In other words, the first customer must be a linehaul customer unless the restriction is set to zero as in the mixed VRPB.

Given such a limitation a simpler selection procedure is adopted for the restricted VRPB. We set the number of ants equal to the number of linehaul customers and we place one ant at each linehaul customer, or $\frac{N}{2}$ whichever is the smallest.

4 Computational Results

In this section we present computational results for our ant system method. Two sets of VRPB instances are used. The first set, referred to as 'class1', is proposed by Goetschalckx and Jacobs-Blecha [1989]. The problems range in size from 25 to 150 customers. The second set, known as 'class 2', is based on the VRP test problems as used by Toth and Vigo [1997]. The instances vary in size from 21 customers to 100 customers. The cost of travelling from customer i to customer j is defined as the Euclidean distance between nodes i and j. The ant algorithm was implemented in Fortran90 and the experiments executed on an Ultra Enterprise 450 (300 MHz dual processor). For each test problem the ant system was run for 200 iterations.

Results are first reported for the fully mixed VRPB ($RP = 0$). The results of the RINS algorithm developed by Wade and Salhi [2002] but with the addition of the local search heuristics given in this paper are also recorded for comparison purposes. The parameters employed in our ant system algorithm are found empirically and are given in Table (1). Where solutions using our ant algorithm are better than those published in the literature, if available, or better than the results using the improved RINS method, then they are shown in bold.

Parameter	Value	Details
α	1	Determines relative influence of trail value, $\alpha \geq 1$.
β	2	Determines relative influence of heuristic information, $\beta \geq 1$.
ϕ	2	Determines relative influence of lookahead information, $\phi \geq 1$.
γ	2	Frequency parameter for local trail update, $\gamma > 1$.
υ	0.1	Determines local pheromone trail evaporation, $0 < \upsilon < 1$.
ρ	0.4	Determines global pheromone trail evaporation, $0 < \rho < 1$.
θ	0.1	Acceptance percentage for global trail update, $0 < \theta \leq 1$.
λ	12	Maximum number of global updating ants.
α_0	0.4	Correction factor related to vehicle load ($0 < \alpha_) \leq 1$).
K_0	$\frac{N}{2}$ or N_L	Number of ants used ($N = N_L + N_B$).
$itermax$	200	Maximum number of iterations used

Table 1: Parameter Setting

The Fully Mixed VRPB ($RP = 0\%$) Results are reported for the fully mixed VRPB in Tables (2) and (3) for the Class 1 and Class 2 set of test problems respectively.

The ant heuristic employed to solve the fully mixed VRPB, uses random variables in the determination of the starting ants and in the selection process. To obtain robust solutions each problem has been run 50 times. The worst, average and best solutions are reported in the tables under the heading $Worst_a$, Av_a and $Best_a$, respectively. The total number of customers in the problem is denoted by N and the average time in seconds is given by *time*. The total number of vehicles used in the best solution produced by the ant algorithm is given under the heading K_a. Approximate results for the fully mixed VRPB are taken from Halse [1992] where available, namely for the Class 1 data set. The results are reported under the title *best*, with the total number of vehicles used given by K_b and the time in seconds t_b. For problems where results are not available from the current literature then solutions are reported under the heading H using our enhanced RINS heuristic. The number of vehicles used is given by K_H and the time in seconds t_H. According to Table (2) it can be observed that superior quality solutions are produced in over half of the problems compared to those produced by Halse [1992]. Table (3) demonstrates superior solutions to nearly all those generated by the RINS method.

The Restricted VRPB including the Classical VRPB ($RP = 100\%$) Tables (4) to (11) present solutions for the Class 1 and Class 2 set of problems for RP set to 25%, 50%, 75% and 100%. As no randomness is used for this class of VRPB, each problem is run only once.

Solutions using the ant algorithm are reported under the heading *ant* with the total number of vehicles given by K_a and the computation time in seconds denoted by *time*. Where solutions are available in the literature these are reported in the tables under the heading *best* for solution cost, K_b for the total number of vehicles and t_b in respect of the CPU time given in seconds. Exact results for RP set to 100% (classical VRPB) are taken from Mingozzi et al. [1999]. Where optimal solutions are not available to the classical VRPB then results are presented from Toth and Vigo [1997]. As there are no results available for the restricted VRPB, the solutions found by the current implementation of the RINS algorithm are reported under the heading H.

Overall Performance

It is worth reporting that the enhancement to the candidate list gave an improvement of approximately 8% for the overall set of test problems when compared to the basic ant algorithm applied to the VRPB without any modifications. Also, the amendment to the visibility, local and global updating each showed an improvement of around 2% in the overall results for a given number of iterations. The restriction to the number of ants did not show an improvement when comparing results over the same number of iterations. However, by restricting the number of ants fewer routes are created therefore more iterations can be performed for the same computational effort. A small improvement of approximately 1% was achieved when comparing results over an equal number of solutions generated. Applying all modifications to the ant system algorithm produced an improvement of 14% taken as an average over the complete set of test problems.

Table (12) shows the average improvement in cost for our ant algorithm compared to the enhanced RINS method of Wade and Salhi [2002]. Though the local search heuristics have made a significant improvement to the initial implementation of the greedy RINS method given in Wade and Salhi [2002], the proposed ant heuristic shows an overall improvement for each value of RP.

The effect of the practicality of RP

Tables (13) summarizes the effect of imposing a restriction value (RP) on the solutions when using the proposed ant system. Table (13) compares solutions for each of the RP values 0%, 25%, 50% and 75% with the classical VRPB (i.e., $RP = 100\%$) against the optimal, or best known, solutions and the ones found by the proposed ant heuristic. This table demonstrates the average overall saving achieved using our ant system when allowing a relaxation on the constraint that all backhaul customers must be visited after all linehaul customers along each route. These results illustrate that an improvement of 4% to 10% could be generated when RP is set between 75% and 50%, thus verifying the observation that significant savings could be achieved while still producing routes that can be implemented in practice, see Wade and Salhi [2002]. A similar conclusion is also drawn when based on the best published results for $RP = 100\%$.

5 Conclusions

In this paper we have presented an ant system algorithm to solve three types of vehicle routing problem with backhauls. We have analysed and modified a number of the basic attributes of ant system methods in order to solve these problems. The results shown by our ant system algorithm provide good potential for solving the mixed vehicle routing problem with backhauls and also improve on the results for the restricted VRPB found by an insertion-type heuristic enhanced by post-optimisation. The current implementation of our ant system unfortunately did not yield comparable results for those intensively studied problems such as the classical VRPB.

The following research avenues may be worth considering: for instance, a revised visibility heuristic that incorporates information concerning linehaul and backhaul load at each customer could guide the search better. More powerful route improvement heuristics could obviously improve the solutions, but such a view though useful in practice may have two drawbacks namely the performance of the ant system may be overshaddowed and the extra computational effort that may be required. From a statistical view point, it is worthwhile conducting a statistical investigation to explore the statistical significance of the sensitivity of some of the parameters as well as the effect of the refinements. One of the drawback of ant systems is the use of several control parameters some of which are obviuosly easier than other to control. An important and challenging issue is to better understand the complex behaviour of the ants and hopefully be in a position to have adaptive ways of determining some of these parameters which will make the system less sensitive and hopefully more aggressive toward generating even better solutions as real ants do.

References

R. Beckers, J. Deneubourg, and S. Goss. Trails and U-turns in the selection of the shortest path by the ant Lasius niger. *Journal of Theoretical Biology*, 159:397–415, 1992.

E. Bonabaeu, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems.* Oxford University Press, 1999.

B. Bullnheimer, R. Harlt, and C. Strauss. Applying Ant Systems to the Vehicle Routing Problem. In S. Voss et. al., editor, *Metaheuristics: Advances and Trends in Local Search Paradigms for Optimization*. Kluwer Academic Publishers, Boston, 1998.

B. Bullnheimer, R. F. Hartl, and C. Strauss. A New Rank Based Version of the Ant System: A Computational Study. Pom-03/97, Institute of Management Science, University of Vienna, Austria, 1997.

O. Casco, B. Golden, and E. Wasil. Vehicle Routing with Backhauls: Models, Algorithms and Case Studies. In B. Golden and A. Assad, editors, *Vehicle Routing: Methods and Studies*, pages 127–147. North Holland, Amsterdam, 1988.

A. Colorni, M. Dorigo, and V. Maniezzo. Distributed Optimization by Ant Colonies. In F. Varela and P. Bourgine, editors, *Proceedings of the European Conference on Atrificial Life*, pages 134–142. Elsevier Publishing, Amsterdam, 1991.

O. Cordon, I. Fernandez de Viana, F. Herrera, and L. Moreno. A New ACO Model Integrating Evolutionary Computation Concepts: The Best-Worst Ant System. In M. Dorigo, M. Middendorf, and T. Stutzle, editors, *Abstract Proceedings of ANTS2000- From Ant Colonies to Artificial Ants: A Series of International Workshops on Ant Algorithms*, pages 22–29. Université Libre de Bruxelles, Belgium, 2000.

I. Deif and L. Bodin. Extension of the Clarke and Wright Algorithm for Solving the Vehicle Routing Problem with Backhauls. In A. E. Kidder, editor, *Proceedings of the Babson Conference on Software Uses in Transportation and Logistics Managemant*, pages 75–96, 1984.

M. Dorigo. *Optimization, Learning and Natural Algorithms.* PhD thesis, Politecnico di Milano, Italy, 1992.

M. Dorigo, G. Caro, and L. Gambardell. Ant Algorithms for Discrete Optimization. *Artificial Life*, 5(3):137–172, 1999.

M. Dorigo and L. Gambardella. Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem. *IEEE Transactions on Evolutionary Computation*, 1(1):53–66, 1997.

M. Dorigo, V. Maniezzo, and A. Colorni. The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions Systems, Man, and Cybernetics- Part B*, 26 (1):29–41, 1996.

M Dorigo and T Stutzle. The Ant Colony Optimization Metaheuristic: Algorithms, Applications, and Advances. Technical Report IRIDIA-2000-32, Université Libre de Bruxelles, Belgium, 2000.

L. Gambardella and M. Dorigo. HAS-SOP: An Hybrid Ant System for the Sequential Ordering Problem. Technical Report 11-97,IDSIA, Lugano, Switzerland, 1997.

L. Gambardella, E. Taillard, and G. Agazzi. MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows. In D. Corne, M. Dorigo, and F. Glover, editors, *New Ideas in Optimization*, pages 63–76. McGraw Hill, London, 1999.

M. Goetschalckx and C. Jacobs-Blecha. The Vehicle Routing Problem with Backhauls. *European Journal of Operational Reasearch*, 42:39–51, 1989.

S. Goss, S. Aron, J. Deneubourg, and J. Pasteels. Self-orgainized shortcuts in the Argentine ant. *Naturwissenschaften*, 76:579–581, 1989.

K. Halse. *Modelling and Solving Complex Vehicle Routing Problems.* PhD thesis, The Technical University of Denmark, 1992.

S. Lin. Computer Solutions of the Travelling Salesman Problem. *Bell System Technical Journal*, 44:2245–2269, 1965.

V. Maniezzo, A.Colorni, and M. Dorigo. The Ant System applied to the Quadratic Assignment Problem. Technical Report IRIDIA/94-28, Université Libre de Bruxelles, Belgium, 1994.

V. Maniezzo and A. Colorni. The Ant System applied to the Quadratic Assignment Problem. *IEEE Trans. Knowledge and Data Engineering*, 1999.

A. Mingozzi, S. Giorgi, and R. Baldacci. An Exact Method for the Vehicle Routing Problem with Backhauls. *Transportation Science*, 33(3):315–329, 1999.

S. Salhi and G. Nagy. A Cluster Insertion Heuristic for the Single and Multiple Depot Vehicle Routing Problems with Backhauls. *Journal of the Operational Research Society*, 50(10):1034–1042, 1999.

S. Salhi and G. Rand. Improvements to Vehicle Routing Heuristics. *Journal of the Operational Research Society*, 38:293–295, 1987.

T. Stutzle and H. Hoos. Improvements on the Ant System: Introducing MAX-MIN Ant System. In *Proceedings of the International Conference on Artificial Neural Networks and Genetic Algorithms*, pages 245–249. Springer Verlag, Wien, 1997a.

T. Stutzle and H. Hoos. The MAX-MIN Ant System and Local Search for the Travelling Salesman Problem. In T. Baeck, Z. Michalewicz, and X. Yao, editors, *Proceedings of IEEE-ICEC-EPS'97, IEEE International Conference on Evolutionary Computation and Evolutionary Programming Conference*, pages 309–314. IEEE Press, 1997b.

P. Toth and D. Vigo. A Heuristic Algorithm for the Vehicle Routing Problem with Backhauls. In *Advanced Methods in Transportation Analysis*, pages 585–608. Springer Verlag, Berlin, 1996.

P. Toth and D. Vigo. An Exact Algorithm for the Vehicle Routing Problem with Backhauls. *Transportation Science*, 31(4):372–385, 1997.

A.C. Wade and S. Salhi. An investigation into a new class of Vehicle Routing Problem with Backhauls. *Omega: The International Journal of Management Science*, 30:479–487, 2002.

A.C. Wade and S. Salhi. An Ant System Algorithm for the Mixed Vehicle Routing Problem with Backhauls. In J.P. Sousa and M.G.Resende, editors, *MIC 2002*. 2003.

Table 2: Results for Class 1 test problems: restriction percentage 0% (Mixed VRPB)

Problem	N	CAP	$Worst_{ant}$	$Best_a$	K_a	$Average_a$	$time$	$best$	K_b	t_b
a1	25	1550	223781	**223088**	8	223374	0.35	227725	8	0.18
a2	25	2550	170849	169500	5	169797	0.31	169497	5	0.09
a3	25	4050	142270	142034	3	142126	0.34	142032	3	0.05
b1	30	1600	238512	**233001**	7	234514	0.58	233950	7	0.13
b2	30	2600	180550	**179258**	4	179760	0.76	182326	4	0.07
b3	30	4000	145702	145702	3	145702	0.71	145699	3	0.30
c1	40	1800	242023	**239192**	7	240126	1.65	242931	7	0.69
c2	40	2600	198503	**196883**	5	197287	2.08	197276	5	0.74
c3	40	4150	168150	**164891**	3	165710	1.87	167663	4	0.06
d1	38	1700	307901	**307110**	11	307383	1.56	307875	11	0.47
d3	38	2750	225439	224196	7	224598	1.28	222195	7	0.18
e1	45	2650	233165	**223774**	7	225927	2.30	222518	7	0.43
e2	45	4300	192421	190559	4	191342	3.26	190048	4	0.07
e3	45	5225	187993	**182804**	4	184621	2.42	187793	4	0.16
f1	60	3000	257191	**248333**	7	251364	5.55	254977	6	0.53
f3	60	4400	222317	217317	5	218818	6.61	215575	5	0.44
f4	60	5500	204899	**200964**	4	202280	7.58	203448	5	0.41
g1	57	2700	315959	**301235**	10	311057	7.64	304106	10	0.41
g2	57	4300	238548	235920	6	236623	4.26	235220	6	0.17
g3	57	5300	216359	214534	5	215133	9.89	213757	5	0.22
g5	57	6400	204816	203233	4	203777	5.32	202610	4	0.33
g6	57	8000	191724	**189922**	3	190572	9.88	201875	4	0.33
h1	68	4000	242430	241619	6	241850	9.94	235269	6	0.65
h2	68	5100	221003	220305	5	220532	8.57	215649	5	0.18
h3	68	6100	209368	208412	4	208693	8.80	202971	4	0.12
h5	68	7100	204834	203193	4	203647	9.21	201896	4	0.24
i1	90	3000	340621	**327168**	10	336501	22.78	329237	10	1.06
i2	90	4000	288202	**278727**	7	281864	27.78	289501	7	1.76
i3	90	5700	246914	**238626**	5	241102	24.95	244782	5	3.83
j1	94	4400	344052	**332471**	10	341272	28.02	337800	10	1.08
j2	94	5600	305217	**292698**	8	301779	32.36	298432	8	1.32
j3	94	6600	267985	**259243**	6	261264	32.15	280070	7	0.38
j4	94	8200	285208	261066	7	278866	50.55	257895	6	0.53
k1	113	4100	374500	**360954**	10	370134	57.71	361287	10	1.40
k2	113	5200	330611	323979	8	327324	57.99	320012	8	1.57
k4	113	6200	312637	298518	7	308065	58.79	296766	7	0.66
l1	150	4000	328862	416167	11	408644	159.97	412278	10	3.81
l2	150	5000	384239	**360018**	8	377962	164.42	362399	8	4.01
l4	150	6000	342899	**337620**	7	339020	160.74	341304	7	2.09
m1	125	5200	387500	**370920**	10	381006	70.79	372840	11	2.41
m3	125	6200	365982	**335486**	9	354797	80.21	336011	9	2.71
m4	125	8000	313317	310567	7	311410	83.48	305118	7	0.82
n1	150	5700	392411	**370690**	10	387926	129.97	385978	10	2.60
n3	150	6600	372671	**349516**	9	329555	151.62	352992	9	2.82
n5	150	8500	336109	323698	7	332754	166.68	319811	7	3.11

Table 3: Results for Class 2 test problems: restriction percentage 0% (Mixed VRPB)

Problem	N	CAP	$Worst_a$	$Best_a$	K_a	Av_a	$time$	H	K_H	t_H
eil22_2	21	6000	326	326	3	326.00	0.54	325	3	0.25
eil22_3	21	6000	342	**342**	3	342.00	0.47	362	3	0.10
eil22_5	21	6000	345	**342**	3	343.13	0.50	351	4	0.11
eil23_2	22	4500	527	527	2	527.00	0.52	526	2	0.34
eil23_3	22	4500	527	**527**	2	527.00	0.53	545	2	0.17
eil23_5	22	4500	514	514	2	514.00	0.56	514	2	0.10
eil30_2	29	4500	426	**421**	2	421.63	0.94	478	2	0.74
eil30_3	29	4500	481	**481**	3	481.00	1.02	486	3	0.38
eil30_5	29	4500	479	479	3	479.00	1.11	475	3	0.22
eil33_2	32	8000	685	**681**	3	682.63	2.27	740	3	1.14
eil33_3	32	8000	688	**684**	3	684.50	1.58	726	3	0.55
eil33_5	32	8000	709	**709**	3	709.00	2.11	724	3	0.28
eil51_2	50	160	471	**466**	3	468.50	9.29	498	3	6.97
eil51_3	50	160	503	**498**	4	500.00	5.08	504	4	2.32
eil51_5	50	160	514	**501**	4	506.25	8.27	516	5	0.74
eil76a_2	75	140	698	**681**	6	688.75	18.16	705	6	29.77
eil76a_3	75	140	752	**726**	7	745.38	29.44	733	8	11.85
eil76a_5	75	140	817	**782**	9	800.50	21.83	788	9	3.97
eil76b_2	75	100	796	**777**	8	782.00	33.90	788	9	30.37
eil76b_3	75	100	874	**826**	10	865.13	11.21	833	10	11.49
eil76b_5	75	100	961	**931**	12	951.25	15.56	936	12	3.39
eil76c_2	75	180	648	**644**	5	645.88	38.88	655	5	30.72
eil76c_3	75	180	687	**663**	6	676.88	42.37	665	6	10.60
eil76c_5	75	180	733	**713**	7	725.63	17.33	719	7	3.85
eil76d_2	75	220	618	**611**	4	614.38	48.42	623	4	29.58
eil76d_3	75	220	656	**631**	5	643.13	14.16	647	5	12.93
eil76d_5	75	220	709	**669**	6	687.13	12.23	670	6	3.31
eil101_2	100	200	753	735	5	741.75	52.76	735	5	86.98
eil101_3	100	200	792	**754**	6	785.25	39.64	760	6	32.17
eil101_5	100	200	825	**789**	7	818.25	48.32	791	7	15.53

Table 4: Results for Class 1 test problems: restriction percentage 25%

Problem	N_L	N_B	CAP	ant	K_a	$time$	H	K_H	t_H
a1	20	5	1550	234718	7	0.47	227931	7	0.09
a2	20	5	2550	**170849**	5	0.57	171150	5	0.087
a3	20	5	4050	**144291**	3	0.65	152695	3	0.092
b1	20	10	1600	**239694**	7	1.01	246230	7	0.317
b2	20	10	2600	182409	4	1.40	182293	5	0.542
b3	20	10	4000	**147457**	3	1.35	149756	3	0.314
c1	20	20	1800	**262390**	7	2.23	262432	7	1.947
c2	20	20	2600	**208934**	5	3.64	212191	5	2.012
c3	20	20	4150	**176965**	3	3.56	191491	3	1.887
d1	30	8	1700	**310379**	11	2.87	312317	11	0.302
d3	30	8	2750	224196	7	2.36	224092	7	0.289
e1	30	15	2650	223774	7	3.11	220740	7	1.25
e2	30	15	4300	192220	4	5.70	190219	4	1.252
e3	30	15	5225	**183033**	4	4.46	190321	4	1.244
f1	30	30	3000	**250386**	6	10.22	257683	6	10.722
f3	30	30	4400	**220133**	4	8.92	220233	4	9.2
f4	30	30	5500	220764	4	13.25	218848	4	9.433
g1	45	12	2700	301235	10	14.53	297448	10	1.079
g2	45	12	4300	240683	6	7.85	240162	6	1.042
g3	45	12	5300	**218712**	5	13.35	219521	5	1.186
g5	45	12	6400	**198526**	4	9.80	205699	4	0.946
g6	45	12	8000	**199858**	3	18.19	203694	4	0.924
h1	45	23	4000	**242218**	6	18.32	242892	6	5.6
h2	45	23	5100	**231120**	5	11.57	232698	5	5.783
h3	45	23	6100	**214701**	4	16.74	222134	4	4.633
h5	45	23	7100	**203719**	3	17.52	213222	4	5.666
i1	45	45	3000	**342921**	10	30.75	351054	10	44.833
i2	45	45	4000	**305541**	7	51.14	305714	7	45.308
i3	45	45	5700	**266081**	5	45.98	269114	5	40.417
j1	75	19	4400	**328558**	10	49.68	338711	10	7.136
j2	75	19	5600	**297419**	8	59.57	300251	8	8.15
j3	75	19	8200	**266948**	6	65.76	274091	6	5.776
j4	75	19	6600	**267006**	7	93.16	277940	7	7.953
k1	75	38	4100	368152	10	106.24	360712	10	40.854
k2	75	38	5200	**330339**	8	110.29	330394	8	35.47
k4	75	38	6200	**299627**	7	108.34	299851	7	36.124
l1	75	75	4000	**411506**	11	283.65	415772	10	350.542
l2	75	75	5000	**367612**	8	310.87	385957	8	342.467
l4	75	75	6000	354145	7	337.65	346829	7	349.131
m1	100	25	5200	382904	10	134.63	379183	10	17.276
m3	100	25	6200	**338492**	9	147.82	346282	9	20.053
m4	100	25	8000	**318156**	7	112.70	319044	7	15.027
n1	100	50	5700	**374347**	10	230.45	381482	10	131.397
n3	100	50	6600	**350429**	9	288.36	361347	9	128.224
n5	100	50	8500	**331123**	7	225.02	333873	7	110.574

Table 5: Results for Class 2 test problems: restriction percentage 25%

Problem	N_L	N_B	CAP	ant	K_a	$time$	H	K_H	t_H
eil22_2	11	10	6000	345	3	0.65	326	3	0.22
eil22_3	14	7	6000	379	4	0.89	362	3	0.10
eil22_5	17	4	6000	364	3	1.12	351	4	0.11
eil23_2	11	11	4500	**550**	2	0.67	681	3	0.29
eil23_3	15	7	4500	**533**	2	0.99	545	2	0.15
eil23_5	18	4	4500	514	2	1.07	514	2	0.09
eil30_2	15	14	4500	**450**	2	1.30	495	2	0.63
eil30_3	20	9	4500	**493**	3	2.28	507	3	0.31
eil30_5	24	5	4500	479	3	3.04	475	3	0.20
eil33_2	16	16	8000	**692**	3	2.41	743	3	1.04
eil33_3	22	10	8000	**687**	3	6.30	733	3	0.44
eil33_5	26	6	8000	**710**	3	8.39	724	3	0.26
eil51_2	25	25	160	**482**	3	6.94	498	3	5.94
eil51_3	34	16	160	508	4	82.10	504	4	2.02
eil51_5	40	10	160	516	4	42.07	516	5	0.72
eil76a_2	38	37	140	**690**	6	442.69	707	6	28.22
eil76a_3	50	25	140	**733**	7	293.14	738	8	11.18
eil76a_5	60	15	140	810	9	440.18	788	9	3.46
eil76b_2	38	37	100	815	8	141.07	800	9	29.20
eil76b_3	50	25	100	**826**	10	31.87	833	10	10.64
eil76b_5	60	15	100	936	12	75.63	936	12	3.30
eil76c_2	38	37	180	**659**	5	154.89	662	5	27.78
eil76c_3	50	25	180	677	6	19.43	670	6	9.12
eil76c_5	60	15	180	732	7	36.81	722	7	4.14
eil76d_2	38	37	220	**620**	4	352.19	622	4	25.76
eil76d_3	50	25	220	**657**	5	125.49	662	5	10.64
eil76d_5	60	15	220	671	6	213.39	671	6	3.07
eil101_2	50	50	200	**777**	4	136.07	781	5	77.64
eil101_3	67	33	200	**772**	6	67.69	773	6	29.14
eil101_5	80	20	200	811	7	792.80	796	7	12.82

Table 6: Results for Class 1 test problems: restriction percentage 50%

Problem	N_L	N_B	CAP	ant	K_a	$time$	H	K_H	t_H
a1	20	5	1550	233767	7	0.50	227931	7	0.09
a2	20	5	2550	**175425**	5	0.59	176228	5	0.08
a3	20	5	4050	**154311**	3	0.62	159120	3	0.08
b1	20	10	1600	**243359**	7	1.05	246230	7	0.30
b2	20	10	2600	**190702**	4	1.35	198027	5	0.39
b3	20	10	4000	157694	3	1.43	152084	3	0.29
c1	20	20	1800	**256506**	6	2.29	266849	7	1.79
c2	20	20	2600	**211292**	5	3.76	213399	5	1.81
c3	20	20	4150	**182818**	3	3.43	195734	3	1.51
d1	30	8	1700	312761	11	2.98	312317	11	0.28
d3	30	8	2750	226106	7	2.49	224577	7	0.25
e1	30	15	2650	**225683**	7	3.19	226435	7	1.07
e2	30	15	4300	**192501**	4	5.50	194208	4	0.98
e3	30	15	5225	**191407**	4	4.71	192882	4	0.98
f1	30	30	3000	267358	6	10.57	258462	6	9.49
f3	30	30	4400	**231779**	4	10.20	231782	4	7.66
f4	30	30	5500	228933	4	13.71	227399	4	7.60
g1	45	12	2700	301307	10	16.61	297448	10	0.95
g2	45	12	4300	244887	6	8.12	244757	6	0.87
g3	45	12	5300	**221854**	5	15.26	222804	5	0.78
g5	45	12	6400	**213423**	4	10.06	214685	4	0.74
g6	45	12	8000	**211750**	3	19.22	212712	4	0.76
h1	45	23	4000	**250208**	6	17.68	250267	6	4.50
h2	45	23	5100	**232272**	5	13.02	236552	5	5.46
h3	45	23	6100	**215285**	4	16.15	232949	4	4.15
h5	45	23	7100	**223885**	4	19.71	225942	4	4.60
i1	45	45	3000	**347900**	10	31.56	350635	10	40.11
i2	45	45	4000	**301392**	7	49.35	312835	7	37.84
i3	45	45	5700	**256506**	5	51.73	275071	5	35.17
j1	75	19	4400	**330861**	10	50.99	339942	10	6.32
j2	75	19	5600	**301818**	8	62.94	303037	8	8.08
j3	75	19	8200	274659	6	75.15	274292	6	4.15
j4	75	19	6600	**276689**	7	89.89	279921	7	6.88
k1	75	38	4100	**368152**	10	109.03	368570	10	35.13
k2	75	38	5200	**338820**	8	116.53	348097	8	29.49
k4	75	38	6200	**312161**	7	121.88	321633	7	32.98
l1	75	75	4000	**418861**	10	299.70	420864	10	309.78
l2	75	75	5000	**375764**	8	355.28	385628	8	291.91
l4	75	75	6000	**358591**	7	346.54	371059	7	285.52
m1	100	25	5200	**391753**	10	129.91	395559	10	16.98
m3	100	25	6200	**349206**	9	166.29	355641	9	11.34
m4	100	25	8000	**330492**	7	128.80	336607	7	20.40
n1	100	50	5700	**381811**	10	259.26	386936	10	107.52
n3	100	50	6600	**365755**	9	304.69	375467	9	105.83
n5	100	50	8500	**353839**	7	217.12	356891	7	87.31

Table 7: Results for Class 2 test problems: restriction percentage 50%

Problem	N_L	N_B	CAP	ant	K_a	$time$	H	K_H	t_H
eil22_2	11	10	6000	345	3	0.62	326	3	0.19
eil22_3	14	7	6000	371	3	0.93	362	3	0.09
eil22_5	17	4	6000	364	3	1.10	351	4	0.09
eil23_2	11	11	4500	**557**	2	0.70	630	2	0.23
eil23_3	15	7	4500	**541**	2	0.88	550	2	0.11
eil23_5	18	4	4500	**540**	2	1.02	606	2	0.15
eil30_2	15	14	4500	**491**	2	1.27	510	2	0.51
eil30_3	20	9	4500	**504**	3	2.39	521	3	0.27
eil30_5	24	5	4500	489	3	2.90	486	3	0.15
eil33_2	16	16	8000	**703**	3	2.14	761	3	0.75
eil33_3	22	10	8000	**704**	3	6.19	745	3	0.34
eil33_5	26	6	8000	726	3	7.45	725	3	0.19
eil51_2	25	25	160	**525**	3	7.27	532	3	4.98
eil51_3	34	16	160	**520**	4	86.75	522	4	1.59
eil51_5	40	10	160	523	4	40.16	523	5	0.62
eil76a_2	38	37	140	**710**	6	434.79	723	6	21.95
eil76a_3	50	25	140	749	7	306.77	733	8	8.17
eil76a_5	60	15	140	821	9	420.17	788	9	2.49
eil76b_2	38	37	100	**828**	8	125.40	843	9	25.84
eil76b_3	50	25	100	839	10	31.30	835	10	8.53
eil76b_5	60	15	100	948	12	79.14	937	12	2.63
eil76c_2	38	37	180	711	5	163.66	668	5	21.55
eil76c_3	50	25	180	**694**	6	18.54	705	6	7.45
eil76c_5	60	15	180	**732**	7	32.72	733	7	2.64
eil76d_2	38	37	220	**641**	4	345.90	657	4	20.51
eil76d_3	50	25	220	689	5	132.59	666	5	8.30
eil76d_5	60	15	220	691	6	209.58	684	6	2.47
eil101_2	50	50	200	**780**	4	129.89	785	5	62.44
eil101_3	67	33	200	**785**	6	71.53	790	6	21.66
eil101_5	80	20	200	837	7	778.64	824	7	12.00

Table 8: Results for Class 1 test problems: restriction percentage 75%

Problem	N_L	N_B	CAP	ant	K_a	$time$	H	K_H	t_H
a1	20	5	1550	233718	7	0.56	229480	7	0.08
a2	20	5	2550	180487	5	0.59	175848	5	0.07
a3	20	5	4050	**155925**	3	0.60	162912	3	0.06
b1	20	10	1600	**245157**	7	0.92	246230	7	0.27
b2	20	10	2600	**196607**	4	1.31	200989	5	0.38
b3	20	10	4000	**168090**	3	1.60	169946	3	0.28
c1	20	20	1800	**258899**	7	2.03	281079	7	1.71
c2	20	20	2600	**220397**	5	3.74	227519	5	1.51
c3	20	20	4150	**188901**	3	3.78	198672	3	1.26
d1	30	8	1700	316975	11	3.40	315762	11	0.25
d3	30	8	2750	**233106**	7	2.14	237954	7	0.22
e1	30	15	2650	240302	7	3.68	232930	7	0.89
e2	30	15	4300	**204770**	4	5.32	208460	4	0.75
e3	30	15	5225	211122	4	4.74	210038	4	0.75
f1	30	30	3000	**270946**	6	10.52	272609	6	7.30
f3	30	30	4400	256480	5	10.57	254609	4	6.45
f4	30	30	5500	240234	4	12.37	240185	4	6.16
g1	45	12	2700	317290	10	13.21	303590	10	0.81
g2	45	12	4300	248721	6	8.08	243545	6	0.85
g3	45	12	5300	**226156**	5	12.46	233003	5	0.63
g5	45	12	6400	**216606**	4	8.91	231325	4	0.60
g6	45	12	8000	**213588**	3	21.54	225815	4	1.04
h1	45	23	4000	**266352**	6	20.87	278390	6	3.63
h2	45	23	5100	**253038**	5	11.91	257153	5	3.95
h3	45	23	6100	**242317**	4	21.45	262536	4	3.14
h5	45	23	7100	**247188**	3	20.74	253407	4	3.48
i1	45	45	3000	**361486**	10	27.96	372187	10	36.46
i2	45	45	4000	**341343**	7	47.73	347940	7	30.49
i3	45	45	5700	**301409**	5	41.80	306855	5	30.03
j1	75	19	4400	346971	10	51.14	346769	10	4.65
j2	75	19	5600	**310411**	8	70.55	316271	8	3.41
j3	75	19	8200	**285035**	6	61.38	285247	6	4.40
j4	75	19	6600	**286477**	7	84.03	291027	7	4.81
k1	75	38	4100	**389065**	10	96.58	392254	10	28.81
k2	75	38	5200	**379447**	8	130.61	386051	8	30.96
k4	75	38	6200	**343206**	7	111.53	347661	7	25.56
l1	75	75	4000	**455179**	10	264.74	456325	10	263.67
l2	75	75	5000	**410936**	8	368.14	413293	8	246.37
l4	75	75	6000	385022	7	306.95	384375	7	237.10
m1	100	25	5200	408521	10	138.59	404860	10	19.20
m3	100	25	6200	368747	9	166.81	365178	9	10.41
m4	100	25	8000	**354171**	7	133.46	364931	7	10.87
n1	100	50	5700	**419812**	10	215.09	421556	10	86.59
n3	100	50	6600	**390794**	9	296.85	398811	9	75.98
n5	100	50	8500	**376537**	7	266.47	390228	7	61.30

Table 9: Results for Class 2 test problems: restriction percentage 75%

Problem	N_L	N_B	CAP	ant	K_a	$time$	H	K_H	t_H
eil22_2	11	10	6000	350	3	0.59	339	3	0.17
eil22_3	14	7	6000	386	3	0.88	380	3	0.08
eil22_5	17	4	6000	**364**	3	0.95	366	4	0.11
eil23_2	11	11	4500	**608**	2	0.64	691	2	0.22
eil23_3	15	7	4500	**623**	2	0.89	648	2	0.11
eil23_5	18	4	4500	617	2	1.05	615	2	0.08
eil30_2	15	14	4500	**495**	2	1.18	527	2	0.48
eil30_3	20	9	4500	**532**	3	2.20	538	3	0.23
eil30_5	24	5	4500	489	3	2.97	486	3	0.19
eil33_2	16	16	8000	**725**	3	1.99	781	3	0.71
eil33_3	22	10	8000	**725**	3	5.59	753	3	0.31
eil33_5	26	6	8000	748	3	4.70	735	3	0.14
eil51_2	25	25	160	**558**	3	10.15	565	3	3.79
eil51_3	34	16	160	**546**	4	67.05	595	4	1.44
eil51_5	40	10	160	**571**	4	27.79	575	5	0.60
eil76a_2	38	37	140	**745**	6	257.04	752	6	18.65
eil76a_3	50	25	140	**758**	7	315.62	767	8	7.09
eil76a_5	60	15	140	821	9	451.05	821	9	1.67
eil76b_2	38	37	100	840	8	134.94	840	9	23.99
eil76b_3	50	25	100	**850**	10	58.27	853	10	8.28
eil76b_5	60	15	100	955	12	221.86	953	12	2.64
eil76c_2	38	37	180	**739**	5	137.37	741	5	19.40
eil76c_3	50	25	180	**747**	6	18.65	751	6	5.49
eil76c_5	60	15	180	750	7	45.81	740	7	2.58
eil76d_2	38	37	220	703	4	206.79	701	4	16.86
eil76d_3	50	25	220	**721**	5	188.49	739	5	6.03
eil76d_5	60	15	220	710	6	391.54	705	6	1.71
eil101_2	50	50	200	**863**	4	263.39	870	5	53.79
eil101_3	67	33	200	854	6	189.82	840	6	18.14
eil101_5	80	20	200	**840**	7	8210.41	850	7	10.33

Table 10: Results for Class 1 test problems: restriction percentage 100% (Classical VRPB)

Problem	N_L	N_B	CAP	ant	K_a	$time$	$best$	K_b	t_b
a1	20	5	1550	233581	7	0.35	229886	8	5
a2	20	5	2550	181176	5	0.54	180119	5	4
a3	20	5	4050	158673	3	0.62	155796	3	12
b1	20	10	1600	250554	7	0.76	239080	7	14
b2	20	10	2600	199222	4	1.32	198048	5	40
b3	20	10	4000	173628	3	1.29	169372	3	4
c1	20	20	1800	268959	7	2.12	249448	7	17
c2	20	20	2600	222888	5	2.73	215020	5	18
c3	20	20	4150	201457	3	3.35	195366	3	25
d1	30	8	1700	320672	11	2.74	316709	11	13
d3	30	8	2750	240232	7	1.77	239479	7	51
e1	30	15	2650	242412	7	2.96	238880	7	12
e2	30	15	4300	218234	4	4.27	212263	4	41
e3	30	15	5225	212613	4	4.25	206659	4	64
f1	30	30	3000	282819	6	9.62	263173	6	2049
f3	30	30	4400	260850	4	6.98	241120	5	76
f4	30	30	5500	247399	3	12.65	233862	4	173
g1	45	12	2700	309244	10	13.86	306959	10	3556
g2	45	12	4300	250256	6	10.43	245441	6	229
g3	45	12	5300	233820	5	12.57	229507	5	967
g5	45	12	6400	224007	4	9.36	221730	5	157
g6	45	12	8000	217296	3	13.64	213457	4	103
h1	45	23	4000	269062	6	17.47	268933	6	454
h2	45	23	5100	262355	5	13.32	253365	5	221
h3	45	23	6100	255386	4	12.09	247449	4	177
h5	45	23	7100	259045	3	13.14	246121	4	277
i1	45	45	3000	380549	9	29.33	353021	10	20225
i2	45	45	4000	351957	7	48.82	309943	7	6395
i3	45	45	5700	324937	5	50.67	294833	5	18045
j1	75	19	4400	356644	10	37.26	335006	10	1640
j2	75	19	5600	335696	8	66.02	315644	8	218
j3	75	19	8200	298647	6	61.92	282447	6	363
j4	75	19	6600	322778	7	69.87	300548	7	260
k1	75	38	4100	424297	10	101.31	394637	10	*
k2	75	38	5200	385798	8	91.23	362360	8	2618
k4	75	38	6200	371624	7	103.42	358308	7	265
l1	75	75	4000	457071	10	270.48	447422	10	20225
l2	75	75	5000	415840	8	233.15	411435	7	6395
l4	75	75	6000	427619	7	289.85	395911	7	8642
m1	100	25	5200	413035	10	126.76	409635	10	1640
m3	100	25	6200	405542	9	141.10	410689	8	218
m4	100	25	8000	369066	7	84.52	352740	7	260
n1	100	50	5700	441995	10	219.75	430054	10	*
n3	100	50	6600	419819	9	216.27	411543	9	3812
n5	100	50	8500	423963	7	214.79	412704	7	265

Table 11: Results for Class 2 test problems: restriction percentage 100% (Classical VRPB)

Problem	N_L	N_B	CAP	ant	K_a	$time$	$best$	K_b	t_b
eil22_2	11	10	6000	374	3	0.49	371	3	0.19
eil22_3	14	7	6000	390	3	0.66	366	3	0.07
eil22_5	17	4	6000	380	3	0.80	375	3	0.03
eil23_2	11	11	4500	701	2	0.60	682	2	0.21
eil23_3	15	7	4500	650	2	0.83	649	2	0.07
eil23_5	18	4	4500	642	2	1.00	623	2	0.02
eil30_2	15	14	4500	515	2	1.16	501	2	0.43
eil30_3	20	9	4500	555	3	1.63	537	3	0.15
eil30_5	24	5	4500	528	3	2.10	514	3	0.04
eil33_2	16	16	8000	754	3	1.55	738	3	0.66
eil33_3	22	10	8000	773	3	2.57	750	3	0.20
eil33_5	26	6	8000	763	3	3.32	736	3	0.06
eil51_2	25	25	160	575	3	8.83	559	3	4.09
eil51_3	34	16	160	566	4	14.50	548	4	1.06
eil51_5	40	10	160	588	4	14.43	565	4	0.30
eil76a_2	38	37	140	787	6	53.06	739	6	23.05
eil76a_3	50	25	140	823	7	58.64	768	7	6.48
eil76a_5	60	15	140	848	8	105.77	781	8	1.37
eil76b_2	38	37	100	878	8	31.91	801	8	28.89
eil76b_3	50	25	100	910	10	23.51	873	10	7.28
eil76b_5	60	15	100	981	12	109.29	919	12	1.62
eil76c_2	38	37	180	758	5	55.72	713	5	22.38
eil76c_3	50	25	180	780	6	29.50	734	6	5.47
eil76c_5	60	15	180	788	7	27.07	733	7	1.23
eil76d_2	38	37	220	757	4	56.36	690	4	20.87
eil76d_3	50	25	220	744	5	33.90	715	5	5.16
eil76d_5	60	15	220	740	6	72.17	694	6	1.12
eil101_2	50	50	200	872	4	76.39	843	4	59.90
eil101_3	67	33	200	901	6	241.87	846	6	14.87
eil101_5	80	20	200	910	7	240.06	908	7	3.13

RP	0	25	50	75	100
Class 1	3	1	1	1	3
Class 2	2	1	1	1	3

Table 12: Average % Improvement in Cost: Ant Against the Enhanced RINS Heuristic

	Ants's results wrt RP=100%					Ants's results wrt optimal for RP=100%				
RP	0	25	50	75	100	0	25	50	75	100
Class 1	14	12	10	4	-	10	8	6	0	-4
Class 2	13	11	9	5	-	9	7	5	0	-5

Table 13: Average % Improvement in Cost: Effect of varying RP