Vehicle Routing in a Forestry Commissioning Operation using Ant Colony Optimisation

Edward Kent, Jason Atkin, and Rong Qu

Automated Scheduling Optimisation & Planning Group The University of Nottingham Nottingham NG8 1BB UK {eqk,jaa,rxq}@cs.nott.ac.uk

Abstract. This paper formulates a vehicle routing problem where constraints have been produced from a real world forestry commissioning dataset. In the problem, vehicles are required to fully load wood from forests and then deliver the wood to sawmills. The constraints include time windows and loading bay constraints at forests and sawmills. The loading bay constraints are examples of inter-route constraints that have not been studied in the literature as much as intra-route constraints. Inter-route constraints are constraints that cause dependencies between vehicles such that more than one vehicle is required to perform a task. Some locations have a lot of consignments at similar times, causing vehicles to queue for loading bays. The aim is to produce an optimal routing of consignments for vehicles such that the total time is minimised and there is as little queuing at forests and sawmills as possible. In this paper, the problem has been formulated into a vehicle routing problem with time windows and extra inter-route constraints. An ant colony optimisation heuristic is applied to the datasets and yields feasible solutions that appropriately use the loading bays. A number of methods of handling the inter-route constraints are also tested. It is shown that incorporating the delay times at loading bays into the ant's visibility produces solutions with the best objective values.

Keywords: Ant Colony Optimisation; Forestry Commissioning; Inter-route Constraints

1 Introduction

The problem discussed in this paper is a vehicle routing problem faced by a forestry commissioning operator in Dumfries, Scotland. The data has been provided by Optrak, a vehicle routing and consultancy company. This is a vehicle routing problem with time windows and loading bay capacity constraints.

Models of similar vehicle routing problems with time window constraints have been presented by Fisher et al [6] and by Solomon [11] and are used in this paper. The travel times in this problem are also non-euclidean, asymmetric and the triangle rule does not apply. Such conditions cause problems for many traditional heuristics such as those discussed by Solomon et al [10]. The loading bays capacity constraints are examples of inter-route constraints similar to the inter-tour resource constraints in Hempsch et al [7], which are much less studied in the literature than intra-route constraints [3]. This paper presents a number of methods to mitigate delays at the loading bays.

Smaller forestry commissioning operations have been solved using methods such as column generation (Epstein et al [4, 5]) and mixed integer linear programming models. For larger optimisation problems it is common to turn to heuristics to produce good solutions.

An ant colony optimisation (ACO) heuristic, which is a population based search that is both “robust and versatile” [2], is used to find the routing of vehicles between consignments and minimise the inter-consignment duration and violations of constraints. The heuristic can be easily adapted to accommodate a variety of different constraints, specifically the loading bay constraints in this case. It was suggested by Epstein et al [4] that solutions with periodic vehicle arrivals at loading bays may be easier to use. A variety of methods have been developed in this research for handling the loading bay constraints during the construction of solutions, such as making consignments “invisible” if they cannot be fulfilled without causing waiting time. These methods are compared and analysed in this paper.

The rest of this paper is structured as follows: Section 2 describes the problem and the loading bay constraints faced by this problem. Section 3 describes the ant colony optimisation heuristic and a number of adaptations to handle the loading bay constraints. Section 4 shows the experimental results using various adaptations to the ant colony optimisation heuristic, and discusses the consequent loading bay usage. Section 5 concludes the findings in this paper.

2 Problem Description

2.1 Routing the Forestry Commissioning Operation

The problem presented in this paper is a vehicle routing problem with time windows and additional loading bay constraints. The objective is to minimise the total time to transport logs from a set of forests to a set of sawmills. Forests have been paired with sawmills a-priori into tuples called consignments. Each consignment describes a task that needs to be fulfilled by exactly one vehicle; wood must be picked up from the forest and then driven directly to the paired sawmill. Since the start and end locations of consignments differ from each other, the driving times are asymmetric, non-euclidean and the triangle inequality does not hold, making some heuristics that exploit these characteristics potentially unsuitable for this problem.

Multiple consignments may share the same forest or the same sawmill (or both). Also, some consignments may need to be fulfilled simultaneously by different vehicles, meaning that multiple vehicles can arrive simultaneously at a forest or sawmill with a limited number of loading bays. Inter-route constraints are used to model the usage of these loading bays, as described below.

2.2 Loading Bay Constraints

Let Π_{ib} be a variable that is 1 if bay b is used by order i and 0 otherwise. Let A_i be the pickup location of consignment i and l represent the loading duration, assumed to be a constant of one hour in this problem. Let O represent the set of consignments and B represent the set of loading bays.

$$\begin{aligned}&(\Pi_{ib} + \Pi_{jb} \leq 1) \vee (A_i + l \leq A_j) \vee (A_j + l \leq A_i)\\&\forall b \in B, \forall i, j \in O, i \neq j, \text{(i and j share the same location)}\end{aligned} \tag{2.1}$$

Constraints (2.1) state that if two different consignments i and j use the same loading bay at a forest/sawmill (pickup/delivery location), then either the finish time of the first consignment must be before the start time of the second consignment or vice versa. Figure 1 shows how the pickup loading bay constraint (2.1) is violated (the shaded area) if two consignment loading bay usage times overlap. A vehicle that arrives at a busy pickup/delivery location (with no free loading bays) is allowed to wait. However, it sometimes may be preferable for a vehicle to service a different consignment first and service this consignment later, when the location becomes free again.

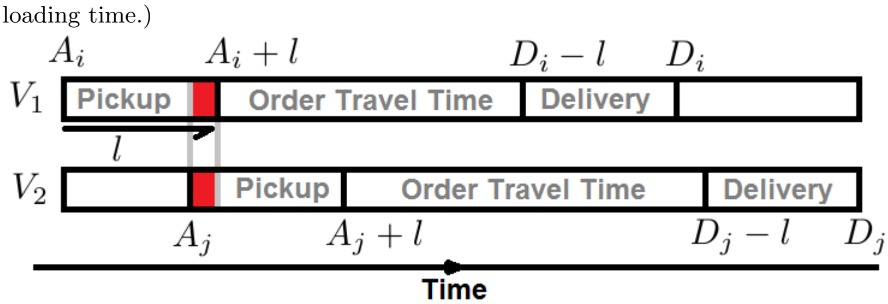
3 Algorithm Description

This section describes the ACO heuristic and a number of adaptations and implementations that handle inter-route constraints.

3.1 Ant Colony Optimisation

Ant colony optimisation (ACO) is a population based adaptive constructive heuristic [2]. It was used in Mazzeo et al [8] to build routes for a capacitated

Fig. 1. The pickup constraint is violated when there is a loading bay usage overlap (e.g when $A_i + l > A_j \wedge A_i < A_j + l$. A_i: the arrival time of a vehicle at order i, l: the loading time.)



vehicle routing problem (CVRP) (without inter-tour constraints) and obtained better results than Tabu Search in some cases. Riemann et al [9] also used an ACO heuristic in a similar way for vehicle routing problems.

ACO uses a set of constructive agents called "ants" to create paths on a graph using knowledge ("pheromones") from previous iterations. After each iteration, for every solution, the pheromones on each arc of the graph are updated based on the fitness of the solutions that used that arc. Solutions that have a better fitness will add more pheromone to the arcs it uses than solutions that have a worse fitness.

Pheromones evaporate over time at a rate of ρ to prevent the heuristic converging too early. Shorter arcs with strong pheromone will attract more ants per iteration than longer arcs with weak pheromone. When more ants traverse an arc throughout the iterations, the pheromone on the arc becomes stronger. Eventually the heuristic should identify a selection of arcs in good solutions.

In this paper, the ants in the ACO heuristic represent vehicles. Unlike the standard ACO heuristic for the travelling salesman problem (TSP) [2], more than one vehicle is needed to create a full solution for the VRP, so "ant groups" are formed that share a list of fulfilled consignments, preventing consignments from being scheduled more than once. A number of ant groups are performed in the same iteration and leave pheromones on arcs for use by later iterations of ant groups. The ACO algorithm can be found in Dorigo et al [2]. In this paper, the ACO heuristic has been further modified to handle time window constraints and loading bay constraints. Let O denote the set of consignments. Each ant in an ant group starts at the depot and a probability of $p_j, \forall j \in O$ is determined for each unassigned consignment based on a number of things: the amount of pheromone on the arc that connects the ant's current position to the consignment, the length of this arc, whether waiting time is required for a vehicle to be serviced at the forest/sawmill for the consignment and, finally, whether the time windows can be met for both the pickup and delivery parts of the consignment. Consignments that cause constraint violations when added to the ant's route can be avoided by setting the probability p_j to 0. Let Ψ represent the set of consignments that are avoided by the ant. Given that the ant is at consignment i, p_j can be calculated using function (3.1), for all $j \notin \Psi$.

$$p_j = \frac{\tau_{ij}^{\alpha}\eta_{ij}^{\beta}}{\sum_{k \in O \setminus \Psi} \tau_{ik}^{\alpha}\eta_{ik}^{\beta}} \tag{3.1}$$

Let τ represent the amount of pheromone on the arc from the ant's current position i to the first customer in the consignment j. Let η represent the "visibility", which is typically $1/t_{ij}$ where t_{ij} is the travel time from consignment i to consignment j. Let α be the amount of influence that the pheromone has on the determination of the next consignment and let β be the amount of influence of the visibility. Using inequality (3.2), where r is a random number $r \in [0, 1)$, the decision to determine the next consignment j in the route is weighted towards "better" choices with higher values of p_j.

$$\sum_{i=0}^{j-1} p_i \leq r \leq \sum_{i=0}^{j} p_i \quad j \in O \tag{3.2}$$

3.2 Constraint Handling

ACO heuristics can be implemented differently to fit particular constraints. For example, a “Heuristic function” is used in the place of the visibility in [1] to solve a vehicle routing problem with time windows and time dependent travel times (traffic conditions). This function includes the duration of the arc as well as the waiting time required to service the customers. A similar approach can be adopted to use loading bay waiting times to influence the ant’s choice of consignment. During the construction of the route, an ant can check a loading bay to see if there is time available for both the sawmill and the forest visits for a consignment. The ant can also calculate the total duration of waiting time that will be required at the forest and the sawmill and use this in the decision making. The loading bay schedule is updated for that group each time an ant visits a particular place, to ensure that there are no loading bay conflicts and to calculate delays.

Three options for handling the loading bay constraints have been considered:

Ignoring and Repairing In this method, the loading bay usage is ignored during the ACO heuristic so infeasible solutions can be created. A repairing procedure (such as a local search heuristic) is used to re-schedule the routes after each iteration to remove loading bay conflicts. This method does not require analysis of arrival times at customers until the repairing procedure, which may reduce the runtime. However, it may not be possible to re-arrange the consignments effectively in the repair procedure, or at least without a large increase in the solution’s objective value.

Avoiding Conflicts For any consignment j, let ω_j be the waiting time, which is the shortest time before the current ant can be serviced at consignment j. The simple avoidance method will set $p_j = 0$ for all consignments j such that $\omega_j > 0$. Figure 2 shows how a loading bay usage window can be tested against a customer’s schedule. It shows an example of a customer with two loading bays. The first example (Accepting) shows that the loading bay usage (labelled insert) can be inserted into the second loading bay without any waiting time. The second example shows that the loading bay usage window cannot be directly inserted into the schedule without having to consider adding waiting time. This method avoids queuing entirely. However, for a hard dataset, queuing may be required to get to a feasible solution.

Scheduling & Penalising Waiting Times with W_1 and W_2 Let ω_j denote the waiting time for consignment j, as above. Rather than preventing the usage, an alternative approach is to penalise the delays. This can be achieved by using

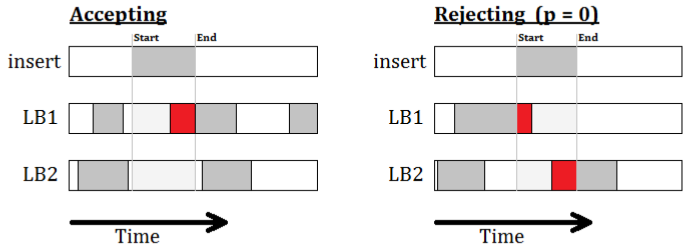


Fig. 2. A forest/sawmill j with 2 loading bays, and an example of when a visit is accepted at the loading bay and and example of when a visit is rejected (when p_j is set to 0)

a weighted visibility η_{ij} calculated by equation (3.3) where W_1 and W_2 are constants, rather than setting $\eta_{ij} = \frac{1}{t_{ij}}$ in equation (3.1).

$$\eta_{ij} = \frac{1}{W_1 t_{ij} + W_2 \omega_j} \tag{3.3}$$

For large values of W_2, waiting times can be avoided where possible since ants will be diverted, due to small values of visibility (η_{ij}). However, a strong penalty could impair the solution in a similar way to setting the probability (p_j) to 0. Consignments that cannot be scheduled without waiting times would be left until the end of the day because their corresponding probabilities have to compete with consignments that do not have waiting times. This can lead to infeasible solutions where these consignments miss their time windows.

3.3 Observing Loading Bay Usage

Although the main objective of the model is to reduce the total time (waiting and driving) the consecutive arrivals of the loading bays can be measured to give an insight into how well the loading bay capacity constraint handling techniques work. Solutions that have a large number of consecutive arrivals and no space between the loading operations may be harder to manage. Although this property is not measured in the objective value, it is possible that such solutions that have good loading bay usages could be better than those that have a lot of consecutive arrivals due to having fewer delays at loading bays. The schedule for each specific loading bay is also analysed separately. For a given loading bay schedule, clusters of loading bay usages are identified by checking for entries that are "close" together within the duration of the load/unload time (which in this case is an hour), which is considered to be far enough apart that the deliveries are independent. Clustered entries are then measured using the ratio between the loading time and the time between the entries. Figure 3 presents an example of clusters of loading bay schedule entries that are used in the calculation of the ratio. A solution that has a low average ratio means that there may be many consecutive entries in the loading bay schedules.

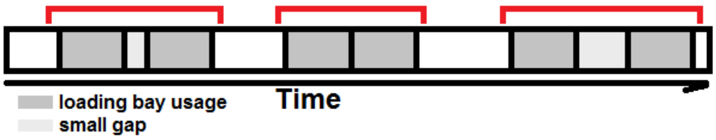


Fig. 3. A loading bay with a number of vehicle visits. The "close" visits have been clustered together, and the sum of the ratio of the time gap between the visits and the loading time is used in the calculation of the average ratio.

4 Computational Results

The six datasets which were used in this research were generated from real world data from south west Scotland. All datasets have a number of locations that are particularly busy (with many consignments in a short duration) with only one or two loading bays available. These datasets can be found at `http://www.cs.nott.ac.uk/~rxq/benchmarks.htm`. The purpose of these experiments is to analyse the different constraint handling techniques. A number of parameter settings for the penalty method are also tested, to analyse their effect on the objective values and the number of delays.

One experiment shows results without the loading bay constraints (for the purpose of comparing objective values). The other experiments use a waiting time penalty multiplier W_2 set to 0, 1 or 2. An experiment was also performed with $W_1 = 2$, to see whether better objective values can be achieved if the waiting time is not prioritised as much as the driving time.

4.1 Results

Results are given in tables 1-6 for different test datasets. Each row in each table gives the average results over ten runs of the ACO heuristic, with the same parameter settings. In each column, the parameter settings and the average values for the following properties are given: the average waiting time across all final ant groups in each run; the average objective function value for the best ant group in each run (in seconds); the average number of times there was a delay across all final ant groups; the average of the loading bay ratios across all final ant groups; the average (upper bound on the) optimality gap for the best ant group for each run. The lower bounds of each dataset were calculated in CPLEX, by assuming a single asymmetric TSP tour that goes through all consignments without time window constraints. Since CPLEX failed to find the optimal solution for any of the asymmetric TSP relaxations, the lower bound of the a-TSP was used to determine (the lower bound for) the optimality gap.

A variety of parameter settings were tried. Firstly, the number of ant groups was set to a low value (10) to view the effects of the parameter settings more quickly. ρ was set to 0.99 with $\alpha = 0.5$ and $\beta = 5$ as suggested by Dorigo et al [2] for travelling salesman problems. However, these values failed to produce good results, which is unsurprising since it is well known that different problems often require different parameter settings. After testing small changes in other parameter settings, the heuristic produced results with better objective values with $\rho = 0.9$, $\alpha = 0.7$ and $\beta = 1.5$ in preliminary tests, so these values were used for the experiments. Small changes to these parameters did not have much effect upon the objective value, but changing α to values above 1.0 or β to values below 1.0 produced worse solutions as the heuristic converged too quickly. ρ is set to a lower value because only 1000 iterations were used in order to keep the runtime low.

Table 1. 300 Consignments, 40 Vehicles, 79 Points

Expt.	W_1	W_2	Waiting Time	Objective	Delay	Ratios	Gap %
1	off		N/A	5.464E6	N/A	N/A	15.01
2	avoid		N/A	5.512E6	N/A	0.46	15.74
3	1.0	0.0	5.463E5	5.493E6	9.51	0.4	15.45
4	1.0	1.0	5.412E5	5.48E6	9.49	0.4	15.26
5	1.0	2.0	5.451E5	5.485E6	9.5	0.4	15.33
6	2.0	1.0	5.38E5	5.485E6	9.59	0.4	15.32

Table 2. 350 Consignments, 40 Vehicles, 84 Points

Expt.	W_1	W_2	Waiting Time	Objective	Delays	Ratio	Gap %
1	off		N/A	6.297E6	N/A	N/A	16.47
2	avoid		N/A	N/A	N/A	N/A	N/A
3	1.0	0.0	6.656E5	6.431E6	26.6	0.36	18.2
4	1.0	1.0	6.697E5	6.43E6	26.74	0.36	18.19
5	1.0	2.0	6.691E5	6.431E6	26.66	0.36	18.21
6	2.0	1.0	6.649E5	6.435E6	26.72	0.36	18.25

Table 3. 400 Consignments, 40 Vehicles, 98 Points

Expt.	W_1	W_2	Waiting Time	Objective	Delays	Ratio	Gap %
1	off		N/A	7.016E6	N/A	N/A	12.58
2	avoid		N/A	7.09E6	N/A	0.46	13.49
3	1.0	0.0	5.752E5	7.072E6	19.6	0.38	13.27
4	1.0	1.0	5.8E5	7.079E6	19.8	0.38	13.36
5	1.0	2.0	5.717E5	7.082E6	19.7	0.38	13.39
6	2.0	1.0	5.771E5	7.076E6	19.72	0.38	13.32

Table 4. 420 Consignments, 40 Vehicles, 93 Points

Expt.	W_1	W_2	Waiting Time	Objective	Delays	Ratio	Gap %
1	off		N/A	6.961E6	N/A	N/A	13.33
2	avoid		N/A	7.036E6	N/A	0.45	14.25
3	1.0	0.0	4.305E5	7.026E6	24.56	0.38	14.13
4	1.0	1.0	4.271E5	7.033E6	24.66	0.38	14.21
5	1.0	2.0	4.367E5	7.031E6	24.74	0.38	14.19
6	2.0	1.0	4.308E5	7.036E6	24.75	0.38	14.24

Table 5. 420 Consignments, 40 Vehicles, 95 Points

Expt.	W_1	W_2	Waiting Time	Objective	Delays	Ratio	Gap %
1	off		N/A	7.249E6	N/A	N/A	12.25
2	avoid		N/A	7.371E6	N/A	0.45	13.69
3	1.0	0.0	4.395E5	7.343E6	29.7	0.37	13.37
4	1.0	1.0	4.36E5	7.332E6	29.59	0.37	13.24
5	1.0	2.0	4.458E5	7.328E6	29.72	0.37	13.2
6	2.0	1.0	4.405E5	7.345E6	29.8	0.37	13.4

Table 6. 420 Consignments, 40 Vehicles, 95 Points

Expt.	W_1	W_2	Waiting Time	Objective	Delays	Ratio	Gap %
1	off		N/A	7.573E6	N/A	N/A	12.34
2	avoid		N/A	N/A	N/A	N/A	N/A
3	1.0	0.0	5.262E5	7.692E6	33.8	0.38	13.7
4	1.0	1.0	5.165E5	7.699E6	33.91	0.37	13.78
5	1.0	2.0	5.209E5	7.685E6	33.85	0.38	13.62
6	2.0	1.0	5.249E5	7.692E6	33.91	0.37	13.7

4.2 Discussion

The "avoid queuing" method failed to produce any feasible solutions for datasets 2 and 6. The time windows could not be met for these datasets because ants avoid consignments that require queuing, so these consignments were assigned later in the route and the time windows were missed. There may exist solutions where vehicles travel times cause arrivals to be outside of each others loading bay usage times. However, the ant colony algorithm could not find any of these solutions for datasets 2 and 6.

For other datasets, this approach produced feasible solutions because the time windows were lenient enough, or the loading bays were more plentiful. However, the objective values were worse than the other loading bay constraint handling methods. There are no delays for these solutions that are caused by loading bays because vehicles do not drive to consignments that have no loading bays available at the time of the vehicle's arrival. This causes the vehicles to drive to consignments that are further away and thus, routes are longer in these solutions. However, the loading bay ratio was the best in these solutions, meaning that the loading bays are less busy. Figure 4 shows an example of two loading bay schedules; the first example shows vehicles that arrive at similar times, and so the ratio of time between the loading bay usage and the total loading time is small because the loading bay usage is consecutive. The second has vehicles that arrive outside of each other's loading bay usage times, thus there are gaps between the entries and so the ratio is larger. The "ignoring queuing" method produces better

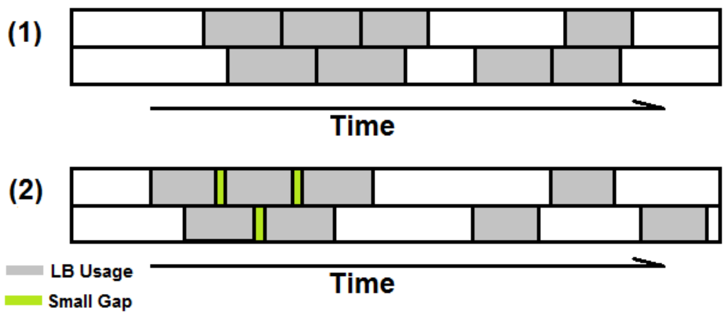


Fig. 4. Possible Effects of W_2 or of Avoiding Queuing. In (1), $W_2 = 0$ and thus the vehicles arrive in similar times and have to wait for the loading bay to be free. In (2), $W_2 = 2$ or the ants avoid queuing. The vehicles arrive slightly further apart, meaning there is no queuing.

objective values because the loading bay constraints are relaxed, so the heuristic does not add waiting time to the entries at busy periods. The ratio of the loading time and gaps between the loading times is not measurable because entries are able to overlap. Of course, this makes the solutions infeasible in practice.

Considering W_1 and W_2 The objective values and loading bay ratios of the solutions obtained when different parameter settings of W_1 and W_2 are used are similar. For this reason, a number of Mann-Whitney U tests were performed on the results of the experiments on each dataset to test the difference in the results. Specifically, for two given sets of data, a percentage is given for the number of entries in the set that are larger than entries in the other. A percentage of $U = 100\%$ means that all entries in the first set are larger than those in the second set.

The objectives appear to vary with the different parameter settings. For example, comparing $W_2 = 0$ and $W_2 = 2$ gives $U = 30\%$ in dataset 3 and $U = 83\%$ in dataset 3. This means that objective values for $W_2 = 0$ were generally smaller than the objective values when $W_2 = 2$ for dataset 3, but for dataset 5 they were generally larger. However, over all tests over all datasets with settings $W_2 = 0$ and $W_2 = 2$, five of these datasets had larger objective values when $W_2 = 0$ because $U > 50\%$. Similarly, four out of six datasets had a result of $U > 50\%$ for tests between $W_2 = 0$ and $W_2 = 1$. This implies that penalising waiting times can potentially aid the heuristic to find good solutions more so than setting $W_2 = 0$. Ignoring waiting time by setting $W_2 = 0$ means that the heuristic is able to accept solutions that have large waiting times, worsening the objective values. For the comparisons between $W_1 = 1$ and $W_1 = 2$, four out of six datasets showed that the objective values were larger for $W_1 = 2$. These four datasets were also the same four datasets where the objective values were larger when $W_2 = 0$ for the tests between $W_2 = 1$ and $W_2 = 0$. Thus, the behaviour of the objective values is similar when setting $W_1 = 2$ or $W_2 = 0$.

A number of Mann-Whitney U tests showed that the loading bay ratios were better when $W_2 = 0$. This is because, when the waiting time at loading bays is penalised, consignments that have no loading bays available are avoided until the end of the day. Many vehicles then arrive at similar times at the end of the day causing queuing at the loading bays. The Mann-Whitney U test results for the loading bay ratios also coincided with the Mann-Whitney U tests for the waiting times; the waiting time is worse when the loading bay ratios are small.

5 Conclusion

In this paper, a forestry commissioning routing problem was presented based on real world problem datasets. The problem is a Vehicle Routing Problem with time windows and inter-route constraints. These inter-route constraints consist of loading bay capacity limitations at pickup and delivery points, meaning that only a limited number of vehicles are able to be serviced simultaneously. The forestry commissioning routing problem was explained and the loading bay constraints were shown. These constraints contained information to ensure that loading bays were used properly.

An Ant Colony Optimisation heuristic was used and a number of problem-specific modifications to the heuristic were tested. These modifications were created to handle the (inter-route) loading bay constraints to avoid loading bay queues, ignore the inter-route constraints, or penalise waiting times. Results showed that, for less constrained problems, queuing can be avoided, but only at the cost of increased objective function values. Penalising the waiting time by setting $W_2 = 1$ or $W_2 = 2$ in the visibility function was found to produce solutions with better objective values and having no cost for waiting time. Setting $W_2 = 0$ could result in solutions with long waiting times. Similarly, using a large penalty for travel times ($W_1 = 2$), was also found to decrease solution value, for the same datasets for which having no delay cost did so. The best objective values were attained for the parameter settings $W_1 = 1$ and $W_2 = 1$ or $W_2 = 2$.

The simple penalisation method for handling the loading bay constraints that are present in this model can also be adopted in other heuristics. The waiting times can be calculated and then included into the objective function of any heuristic with a penalty value. This method may also work well in other problems that have inter-route constraints and is worth further investigation.

References

1. Donati, A.V., Montemanni, R., Casagrande, N., Rizzoli, A.E., Gambardella, L.M.: Time dependent vehicle routing problem with a multi ant colony system. European Journal of Operational Research 185(3), 1174–1191 (2008)
2. Dorigo, M., Maniezzo, V., Colorni, A.: Ant System: Optimization by a Colony of Cooperating Agents. IEEE Transactions on Systems, Management and Cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Management and Cybernetics Society 26(1), 29–41 (Jan 1996)
3. Drexl, M.: Synchronization in Vehicle Routing - A Survey of VRPs with Multiple Synchronization Constraints. Transportation Science 46(3), 1–58 (2011)
4. Epstein, R., Morales, R., Seron, J., Verso, P.T.R.A.: A Truck Scheduling System Improves Efficiency in the Forest Industries. Institute for Operations Research and the Management Sciences 1996(26), 1–12 (1996)
5. Epstein, R., Sero, J., Weintraub, A.: Use of OR Systems in the Chilean Forest Industries. Interfaces 29(1), 7–29 (1999)
6. Fisher, M.L., Jörnsten, K.O., Madsen, O.B.G.: Vehicle Routing with Time Windows: Two Optimization Algorithms. Operations Research 45(3), 488–492 (May 1997)
7. Hempsch, C., Irnich, S.: Vehicle routing problems with inter-tour resource constraints. In: The Vehicle Routing Problem: Latest Advances and New Challenges, pp. 421–444. Springer (2008)
8. Mazzeo, S., Loiseau, I.: An Ant Colony Algorithm for the Capacitated Vehicle Routing Problem. Electronic Notes in Discrete Mathematics 18, 181–186 (Dec 2004)
9. Reimann, M.: D-Ants: Savings Based Ants Divide and Conquer the Vehicle Routing Problem. Computers & Operations Research 31(4), 563–591 (Apr 2004)
10. Solomon, M.M.: Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. Operations Research 35(2), 254–265 (1987)
11. Solomon, M.M., Desrosiers, J.: Time window constrained routing and scheduling problems. Transportation science 22(1), 1–13 (1988)