ACO applied to

the Pickup and Delivery problem

Profesori Indrumatori: Masterand

Badica Costin Sacerdotianu George

Sorin Ilie Viorel

Table of Contents

The Pickup and Delivery problem (Overview). . . . . . . . . . . . . . . . . . . . . . . . 1

ACO applied to the Pickup and Delivery Problem . . . . . . . . . . . . . . . . . . . . 10

Conclusions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .12

References . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13

**The Pickup and Delivery problem (Overview)**

In the General Pickup and Delivery Problem GPDP a set of routes has to be constructed in order to satisfy transportation requests.

In their paper M. W. P. Savelsbergh , M. Sol [1] took a general look at the PDP problem. A fleet of vehicles is available to operate the routes. Each vehicle has a given capacity, a start location and an end location. Each transportation request specifies the size of the load to be transported the locations where it is to be picked up, the origins, and the locations where it is to be delivered, the destinations. Each load has to be transported by one vehicle from its set of origins to its set of destinations without any transhipment at other locations.

There are three well known and extensively studied routing problems. This are special cases of the GPDP. In the Pickup and Delivery Problem (PDP) each transportation request specifies a single origin and a single destination and all vehicles depart from and return to a central depot. The Dial a Ride Problem (DARP) is a PDP in which the loads to be transported represent people. Therefore we usually speak of clients or customers instead of transportation requests and all load sizes are equal to one. The Vehicle Routing Problem (VRP) is a PDP in which either all the origins or all the destinations are located at the depot.

The GPDP is introduced in order to be able to deal with various complicating characteristics found in many practical pickup and delivery problems such as transportation requests specifying a set of origins associated with a single destination or a single origin associated with a set of destinations vehicles with different start and end locations and transportation requests evolving in real time.

The vehicle routing problem (VRP) is a combinatorial optimization and integer programming problem seeking to service a number of customers with a fleet of vehicles. Proposed by Dantzig and Ramser in 1959 [2], VRP is an important problem in the fields of transportation, distribution and logistics Often the context is that of delivering goods located at a central depot to customers who have placed orders for such goods. Implicit is the goal of minimizing the cost of distributing the goods. Many methods have been developed for searching for good solutions to the problem, but for all but the smallest problems, finding global minimum for the cost function is computationally complex.

Several variations and specializations of the vehicle routing problem exist:

* Vehicle Routing Problem with Pickup and Delivery (VRPPD): A number of goods need to be moved from certain pickup locations to other delivery locations. The goal is to find optimal routes for a fleet of vehicles to visit the pickup and drop-off locations.
* Vehicle Routing Problem with LIFO: Similar to the VRPPD, except an additional restriction is placed on the loading of the vehicles: at any delivery location, the item being delivered must be the item most recently picked up. This scheme reduces the loading and unloading times at delivery locations because there is no need to temporarily unload items other than the ones that should be dropped off.
* Vehicle Routing Problem with Time Windows (VRPTW): The delivery locations have time windows within which the deliveries (or visits) must be made.
* Capacitated Vehicle Routing Problem (with or without Time Windows): CVRP or CVRPTW. The vehicles have limited carrying capacity of the goods that must be delivered.

Several software vendors have built software products to solve the various VRP problems. Numerous articles are available for more detail on their research and results. Although VRP is related to the Job Shop Scheduling Problem, the two problems are typically solved using different techniques.

M.W.P. Savelsbergh threated the “The General Pickup and Delivery Problem” [1] as follows:

The following characteristics are taken into account.

**Transportation requests**

This is a very important characteristic of routing problems is the way in which transportation requests become available. In a static situation all requests are known at the time the routes have to be constructed. In a dynamic situation some of the requests are known at the time the routes have to be constructed and the other requests become available in real time during execution of the routes.

Consequently in a dynamic situation when a new transportation request becomes available at least one route has to be changed in order to serve this new request. Most vehicle routing problems are static whereas most pickup and delivery problems are dynamic. In practice a dynamic problem is often solved as a sequence of static problems. In its simplest form each time a new request becomes available the current set of routes is updated. This is commonly referred to as an online algorithm. However it is usually possible and beneficial to buffer incoming requests and only update the current set of routes if the buffer size exceeds a certain preset value. Another important issue concerning (buffered) online algorithms is how to incorporate information that may be known about the spatial or time distribution of future requests.

A depot is another important concept in routing problems. In the routing literature

the depot is usually the place where vehicles start and end their routes. Since most pickup and delivery problems are dynamic often with a long planning horizon the concept of a depot vanishes. Drivers sleep at the last location they visited or at the last location they have to visit the next day. Even for problems with a short planning horizon such as a single day where vehicles start and end at a central depota demand responsive situation leads to problems without depots. When new transportation requests become available and the current set of routes has to be updated the vehicles are spread out over the planning area.

The general pickup and delivery model is well suited for dealing with the sub problems that occur in dynamic demand responsive routing problems

Another characteristic that has to be taken into account are **time constraints.**

Apart from the vehicle capacity constraints and the intrinsic precedence constraints related to pickup and delivery side constraints related to time arise in almost every practical pickup and delivery situation. Although time constraints have become an integral part of models for vehicle routing problems for recent surveys on the vehicle routing problem with time windows that it is also described in “Desrochers Lenstra Savelsbergh[d] and Soumisnand Solomon and Desrosiers[s] “[3] they play an even more prominent role in pickup and delivery problems. Among other reasons because the most studied pickup and delivery problem is the dial a ride problem which deals with the transportation of people who specify desired pickup or delivery times.

The presence of time constraints complicates the problem considerably. If there are no time constraints finding a feasible pickup and delivery plan is trivial arbitrarily: assign transportation requests to vehicles, arbitrarily order the transportation requests assigned to a vehicle and process each transportation request separately. In the presence of time constraints the problem of finding a feasible pickup and delivery plan is N P hard. Consequently it may be difficult to construct a feasible plan especially when time constraints are restrictive. On the other hand an optimization method may benefit from the presence of time constraints since the solution space may be much smaller.

*Time constraints related to transportation requests*

This constraint denotes the time interval in which the pickup or delivery at location i must take place. Given a pickup and delivery plan and departure times of the vehicles the time windows define for each vehicle i, the arrival time Ai and the

departure time Di.

*Time constraints related to vehicles*

Vehicles are usually not available all day. Drivers have to eat and sleep and vehicles

are subjected to service plans. These constraints can be modeled as time windows for vehicles. Typically a vehicle has multiple time windows defining all the periods in which it is available.

Objective functions

A wide variety of objective functions is found in pickup and delivery problems. The most common ones are discussed in what following.

We can split these functions in two categories. Objective functions related to single vehicle pickup and delivery and to multiple vehicle. First we talk about single vehicle pickup and delivery.

*Minimize duration:* The duration of a route is the total time a vehicle needs to execute the route. Route duration includes travel times, waiting times, loading and unloading times, and break times.

*Minimize completion time*: The completion time of a route is the time that service at

the last location is completed. In case the start time of the vehicle is fixed at time zero, the completion time coincides with the route duration.

*Minimize travel time.* The travel time of a route refers to the total time spent on actual

traveling between different locations.

*Minimize route length:* The length of a route is the total distance travelled between

different locations.

*Minimize client inconvenience.* In dial a ride systems client inconvenience is measured in terms of pickup time deviation, i. e. the difference between the actual pickup time and the desired pickup time delivery time deviation, i.e. the difference between the desired delivery time and the actual delivery time and excess ride time, i.e. the difference between the realized ride time and the direct ride time. In demand responsive situations where clients request immediate service i.e. as soon as possible the difference between the time of pickup and the time of request placement may also contribute to the definition of client inconvenience. Different kinds of functions linear as well as nonlinear have been proposed to model client inconvenience.

Below are presented objective functions related to multiple vehicle pickup and delivery problems.

*Minimize the number of vehicles*: This function is almost always used in dial a ride systems combined with one of the above functions to optimize the single vehicle sub problems. Dial a ride systems are normally highly subsidized systems for the transportation of the elderly and handicapped. Therefore the objective is to minimize cost mostly together with customer inconvenience. Because drivers and vehicles are the most expensive parts in a dial a ride system minimizing the number of vehicles to serve all requests is usually the main objective.

*Maximize profit*: This function which can use all of the above function scan be used in a system where the dispatcher has the possibility of rejecting a transportation request when it is unfavourable to transport the corresponding load. Note that for example in a dial a ride system rejecting a transportation request is not allowed on this objective function should not only incorporate the costs but also the revenues

associated with the transportation of loads.

For dynamic pickup and delivery problems it is not clear what kind of objective functions should be used. In a demand responsive environment pickup and delivery

routes may be open ended. Therefore objectives such as duration completion time

and travel time have no clear meaning. Intuitively an objective function for dynamic

problems should emphasize decisions that affect the near future more than decisions

regarding the remote future. Note that if a dynamic problem is solved as a sequence of static problems the objective function for the static sub problem does not necessarily have to be equal to the objective function for the dynamic problem.

The objective function for the static sub problem may reflect some knowledge or anticipation of future requests.

Solution approaches

The class of pickup and delivery problems has been divided into *static* and *dynamic* problems since their characteristics as well as their solution approaches differ considerably. Within either of these classes we distinguish single vehicle and multiple vehicle problems. Obviously in the single vehicle PDP all transportation requests are handled by the same vehicle whereas in the multiple vehicle PDP the transportation requests have to be divided over the set of vehicles.

Assigning transportation requests to vehicles in the PDP is much more difficult than assigning transportation requests to vehicles in the VRP. In the VRP all the origins of transportation requests are located at the depot. Therefore transportation requests with geographically close destinations are likely to be served by the same vehicle.

In the PDP geographically close destinations may have origins that are geographically far apart and we cannot conclude that they are likely to be served by the same vehicle. For each of the resulting subclasses one or more papers are discussed. The level of detail depends on the originality viability and importance of the described solution approach. A performance evaluation of the solution methods is only provided if such information is present in the corresponding paper. Furthermore we do not cover iterative improvement methods in any detail. For these the interested reader is referred to Kindervater and Savelsbergh [4] .

1. **The static pickup and delivery problem**
   1. The static singlevehicle pickup and delivery problem

The next section covers the static single vehicle pickup and delivery problem.

The static single vehicle pickup and delivery problem is probably the most studied variant of the PDP. First of all because the dial a ride problem belongs to this class but also because it appears as a sub problem in multiple vehicle pickup and delivery problems. We discuss solution approaches for problems with and without time windows separately.

1. The static 1 - PDP without time windows

*Optimization*

Psaraftis [5] in considers immediate request dial a ride problems. In these problems every client requesting service wishes to be served as soon as possible. The objective is to minimize a weighted combination of the time needed to serve all clients and the total degree of dissatisfaction clients experience until their delivery. Dissatisfaction is assumed to be a linear function of the time each client waits to be picked up and of the time he spends riding in the vehicle until his delivery.

*Approximation*

Stein [6] presents a probabilistic analysis of a simple approximation algorithm for the single vehicle dial a ride problem without capacity constraints. The algorithm constructs a TSP tour through all origins and a TSP tour through all the destinations and then concatenates them.

1. The static 1-PDP with time windows

*Optimization*

Psaraftis [7] modifies the dynamic programming algorithm discussed above to solve the static single vehicle dial a ride problems with time windows. The major difference between the new and the original algorithm is the use of forward instead of backward recursion. Time windows are handled by the elimination of time infeasible states.

*Approximation*

Sexton and Bodin [8][9] consider the dial a ride problem with desired delivery times specified by the clients. The objective is to minimize client inconvenience which is

defined as a weighted combination of delivery time deviation and excess ride time

The presented solution approach applies Benders decomposition to a mixed non

linear programming formulation which separates the routing and scheduling component.

* 1. The static multiple vehicle pickup and delivery problem

1. The static mPDP without time windows

*Approximation*

Cullen Jarvis and Ratliff [10] propose an interactive approach for the multiple vehicle

Dial a ride problem with a homogeneous fleet i.e. equal vehicle capacities. The problem is decomposed into a clustering part and a chaining part. Both parts are solved in an interactive setting i.e. man and machine cooperate to obtain high quality solutions. The algorithmic approach in both parts is based on set partitioning and column generation.

1. The static mPDP with time windows

Optimization

Dumas Desrosiers and Soumis [11] present a set partitioning formulation for the static pickup and delivery problem with time windows and a column generation scheme to solve it to optimality. The approach is very robust in the sense that it can be adapted easily to handle different objective functions and variants with multiple depots and an inhomogeneous fleet of vehicles.

Approximation

Dumas Desrosiers and Soumis [12] develop an approximation algorithm for the dial a ride problem based on their optimization algorithm discussed above. The basic idea is to create route segments for small groups of clients called mini-clusters. A mini cluster is a segment of a route starting and ending with an empty vehicle. Each mini-cluster is then treated as a transportation request that entirely fills a vehicle. The optimization algorithm is now applied to this set of transportation requests. This reduces the number of rows in the set partitioning matrix. The sub problem is now much easier to solve because each transportation request corresponds to a full truck load. In their approach, Cullen Jarvis and Ratli ff [9] clusters cannot be identified with rows because they do not partition the set of clients. Mini clusters are constructed simply by taking a known pickup and delivery plan constructed using any existing algorithm and cutting it into pieces such that each piece starts and ends with an empty vehicle. In this setup the approach can best be viewed as an improvement method.

1. **The dynamic pickup and delivery problem**

As in most combinatorial optimization problems dynamic aspects of the pickup and

delivery problem are not very well studied

2.1) The dynamic single vehicle pickup and delivery problem

Psaraftis [5] extends the dynamic programming algorithm described in Section for

the static immediate request dial a ride problem to the dynamic case. Indefinite deferment of customers i.e. continuously reassigning service of a customer to the last position in the pickup and delivery sequence is prevented with a special priority constraint.

The times at which requests for service are received define a natural order among the clients. In general the position that a particular customer holds in the sequence of pickups will not be the same as his position in this natural order. The difference between these two positions defines a pickup position shift. A delivery position shift can be similarly defined as the difference between the position in the sequence of deliveries and the position in the natural order. A priority constraint bounding the two position shifts from above prevents indefinite deferment.

The dynamic problem is solved as a sequence of static problems. Each time a new request for service is received, a slightly modified instance of the static problem is solved to update the current route. Obviously, all clients that have already been picked up and delivered can be discarded and the new client has to be incorporated. The starting location of the vehicle and the origins of the clients that have been picked up but not yet delivered have to be set to the location of the vehicle at the time of the update.

The dynamic multiple vehicle pickup and delivery problem

In Psaraftis [5] paper it is presented an algorithm for the dynamic multiple vehicle problem in which the vehicles are in fact ships In this case the capacity of the ports also has to be considered in order to avoid waiting times when loads are to be picked up or delivered.

Since here discussed various characteristics of pickup and delivery problems.

Most reallife pickup and delivery problems that we are aware of are demand responsive. Currently very little is known about online algorithms for dynamic pickup and delivery problems. Although the single vehicle pickup and delivery problem is NP hard it can be solved very efficiently with dynamic programming as long as the number of transportation requests is relatively smallwhich is the case in many practical situations. Therefore the main problem in solving multiplevehicle pickup and delivery problems is the assignment of transportation requests to vehicles. Consequently pickup and delivery problems as well as many other routing and scheduling problems seem to be well suited for solution approaches based on set partitioning with column generation. Although this approach has already been explored successfully it is likely to remain an active research area for the next decade.

In vehicle routing problems customers that are geographically close to each other are likely to be served by the same vehicle. A concept similar to that of geographical closeness in vehicle routing problems does not exist for pickup and delivery problems. Although several alternatives have been proposed none of them has turned out to be entirely satisfactory. The development of a useful closeness concept seems crucial for progress in approximation algorithm.

Many interesting questions arise in the context of interactive planning systems for pickup and delivery problems. Representing solutions for instance is nontrivial an optimal solution to a single vehicle pickup and delivery problem may look very bad when drawn on a mapeven without time windows and within finite vehicle capacity.

**ACO applied to the Pickup and Delivery Problem**

K. Doerner R.F. Hartl M. Reimann [13] proposed in their paper an ACO algorithm to optimize the total costs associated with the pickup and delivery of full truckloads under time window constraints in a hub network. They perform a thorough technical analysis of the ACO by comparing different pheromone decoding schemes, different visibility information and various population sizes. Furthermore we propose

a post-optimization technique to improve the solutions. Their results show that appropriate data structures signiﬁcantly improve the solution quality.

Considering a problem, where customers place orders with a logistics service provider, requiring shipments between two locations. In general, due to small shipment sizes, loads are not transported directly but via distribution centers. Thus, shipments occur between the pickup location of an order and the closest distribution center, between distribution centers and between a distribution center and

the delivery location of an order. Furthermore, each customer demands speciﬁed times for the collection and the delivery of the order. The objective of the service provider is to minimize total costs associated with the satisfaction of all demands.

The distribution process described above basically consists of three stages. The ﬁrst stage, the transportation of goods from the customers’ locations to the distribution centers is usually done on vehicle round-trips using small trucks. The same applies to the third stage, the transportation between distribution centers and customer delivery locations. Such problems have been treated very recently, e.g. in Irnich (2000) [14]. In this paper is discussed the second stage, i.e. the transportation processes between the distribution centers. At the distribution centers orders, requiring drop-off at the same destination, are consolidated to full truckloads. Therefore, trucks moving between any two distribution centers are fully loaded and thus go directly from their source to their destination. Clearly the time windows at the customers lead to corresponding time windows at the two distribution centers associated with an order.

The primary goal at this stage is to minimize total costs associated with the satisfaction of all truckload movements. These costs consist of two cost factors. The ﬁrst one is associated with the ﬂeet size required to perform the deliveries, while the second one covers the costs associated with the actual movements, loaded as well as empty, by the utilized ﬂeet.

The algorithms described below solve this problem subject to the following assumptions:

1. All orders are known in advance.

2. All orders are consolidated to full truckloads.

3. Time windows for each order have to be respected strictly.

4. A tour must not exceed a given time-span.

5. Each truck is assigned to a speciﬁc depot, where it has to return to after each tour.

An exact solution procedure for a similar problem can be found also in Desrosiers et al. (1988), where a situation with full truckloads is considered. However, they do not deal with time window constraints. Apart from that, their algorithm can only solve very small problems. As our aim is to solve problems of larger real world applications we chose to develop an appropriate heuristic.

They talk about a graph-based model formulation where J = {1,….n} denote the set of orders, and D = {1,….m} denote the set of distribution centers, which are the home depots of the trucks. Then, the problem considered in this paper can be represented by a weighted directed graph G = {V, A, d}, where V = { . . ., . . . } is a set of vertices. The vertices to denote the orders 1 to n (we will refer to this subset of V as Vo), the vertices to denote the depots 1 to m (we will refer to this subset of V as ). They also defined a set of arcs between nodes. An arc () represents the empty vehicle movement required between orders i and j, and is weighted with the non-negative distance .

Figure below shows the graph for a problem with 4 orders and 2 depots.

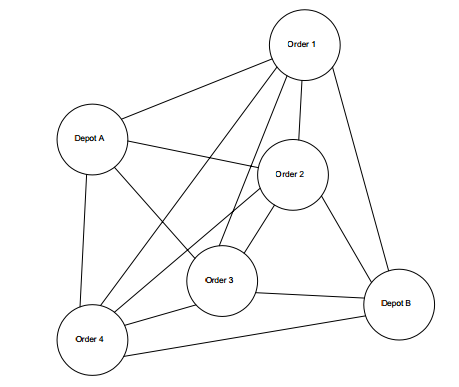


Figure 1: Graphical representation (2 depots, 4 orders)

Based on this graph our optimization problem can be stated as follows: Subject to the constraint, that all order nodes are visited exactly once and each cycle contains exactly one depot node the problem of ﬁnding the minimal ﬂeet size corresponds to ﬁnding the minimum number of cycles in the graph, while the problem of minimizing total vehicle movements corresponds to ﬁnding a number of cycles, such that the total length of all cycles is minimal.

The approach followed in this paper was to solve these objectives simultaneously rather than sequentially.

Conclusions

Several types of PDP where threated and it was presented a technical analysis of an ACO algorithm applied to a special case of the pickup and delivery problem with time window constraints. Their objective function was to minimize total costs, i.e. to optimize both the ﬂeet size and empty vehicle movements.

References

1. The general pickup and delivery problem (1995), M. W. P. Savelsbergh , M. Sol

1. Dantzig, G.B.; Ramser, J.H. (1959). "The Truck Dispatching Problem".
2. M Desrochers, JK Lenstra, MWP Savelsbergh and F Soumis Vehicle routing with time windows optimization and approximation” B.L. Golden and A.A. Assad
3. G.A.P. Kindervater and M.W.P. Savelsbergh “Local search in physical distribution management”
4. H Psaraftis “A dynamic programming solution to the single vehicle many to many immediate request dial a ride problem.
5. DM Stein ‑ An asymptotic probabilistic analysis of a routing problem.
6. H Psaraftis ‑ An exact algorithm for the single vehicle many to many dial a ride problem with time windows.
7. T Sexton and L Bodin ‑Optimizing single vehicle many to many operations with desired delivery times I Scheduling.
8. T Sexton and L Bodin ‑Optimizing single vehicle many to many operations with desired delivery times II Routing.
9. FH CullenJJ Jarvis and HD Ratliff – “Set partitioning based heuristics for interactive routing”
10. Y Dumas, J Desrosiers and F Soumis“The pickup and delivery problem with

time windows”

1. Y Dumas, J Desrosiers and F Soumis- “Large scale multivehicle dialaride systems”
2. K. Doerner, R.F. Hartl, M. Reimann “Ant Colony Optimization applied to the Pickup and Delivery Problem “
3. S. Irnich – “A Ynified Modeling and Solution Framework for Vehicle Routing andLocal Search-base Metaheuristics”