

VSC2014 Problem description

Swap-Body Vehicle Routing Problem (SB-VRP)

A set of *customers* is given. Each customer i have to be delivered with a given *quantity* q_i (i.e. either weight, volume, or shipping units). The delivery at each customer takes a given amount of time s_i (often the duration of the service is related to the quantity that has to be dealt with at the customer). A homogeneous fleet that consists of *trucks*, *semi-trailers* and *swap-bodies* is located at a single *depot*, D . All routes start and end at this depot. The different components and characteristics of the problem are defined with more detail in the following:

► Depot

- the place where the vehicle fleet is located
- all routes start and end at this depot

► Fleet

- A homogeneous fleet is located at one single depot
- the fleet consists of
 - trucks
 - semi-trailers
 - swap-bodies
- side loading (or unloading) of the swap-bodies is possible.
This ensures that if you visit a customer with a *train* (i.e. when the truck tows two swap-bodies) the sequence of the swap-bodies is not relevant
- the maximum load for a swap-body is Q . Only swap-bodies can be loaded, i.e. the maximum load for all trucks and semi-trailers is 0
- an unlimited number of each of the above fleet elements is available

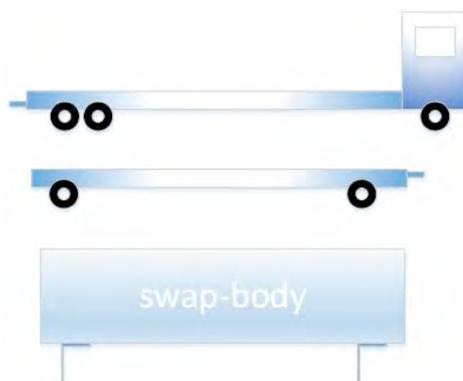


Figure 1: Truck, semi-trailer, and swap-body

- ▶ valid vehicle combinations are
 - ▶ a truck carrying one swap-body
 - ▶ a train (truck and semi-trailer) carrying two swap-bodies

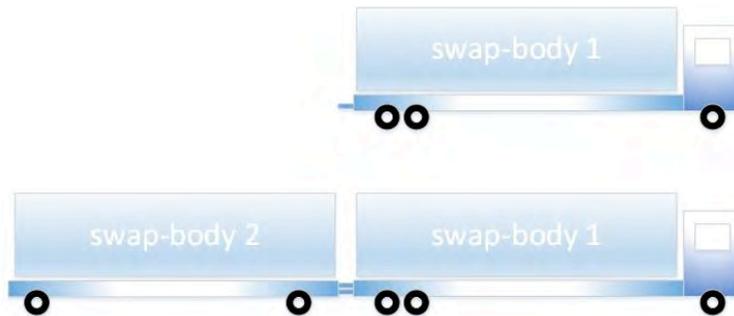


Figure 2: valid vehicle combinations (truck and train)

- ▶ en route a swap-body can be detached, left parked in specified locations, and later reattached to the truck. This allows the truck to visit customers that can be visited only by trucks with just one swap-body even if the truck departed from the depot as a train of two swap-bodies.
- ▶ en route no exchange of swap-bodies between trucks is allowed
- ▶ a truck/train has to return to the depot with exactly the same swap-bodies (one or two) it departed with
- ▶ **Customers**
 - ▶ each customer is delivered with an individual quantity q_i . Deliveries must not be split, hence each customer is visited exactly once. It is valid to load the quantity for one customer on two swap-bodies if this seems beneficial. In case, the two swap-bodies must clearly be assigned to the same truck. In order to guarantee solution feasibility the quantity of each customer does not exceed $2 \cdot Q$.
 - ▶ each customer has a flag that indicates whether it can be visited by a train or only with a truck carrying just one swap-body
 - ▶ **Swap locations**

there is a set of swap locations where different actions are allowed, swap locations can be used if the route is executed by a train. Operations allowed at a swap location are:

 - ▶ park

a truck parks the semi-trailer and continues the route with just one swap-body. The red traffic sign in figures 3-6 indicates that this swap-body is parked at the swap location.

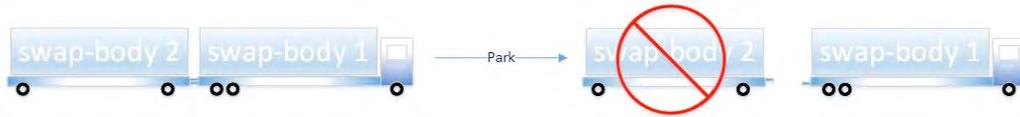


Figure 3: park

- pick up

the truck picks up the semi-trailer with the swap-body it has parked there at an earlier stage of the route



Figure 4: pickup

- swap

a truck parks the swap-body it is currently carrying and picks up the other swap-body from the semi-trailer it has parked there at an earlier stage of the route



Figure 5: swap

- exchange

a truck parks the semi-trailer, exchanges the swap-bodies between truck and semi-trailer and continues the route with this swap-body



Figure 6: exchange

- the state diagram describes valid transitions

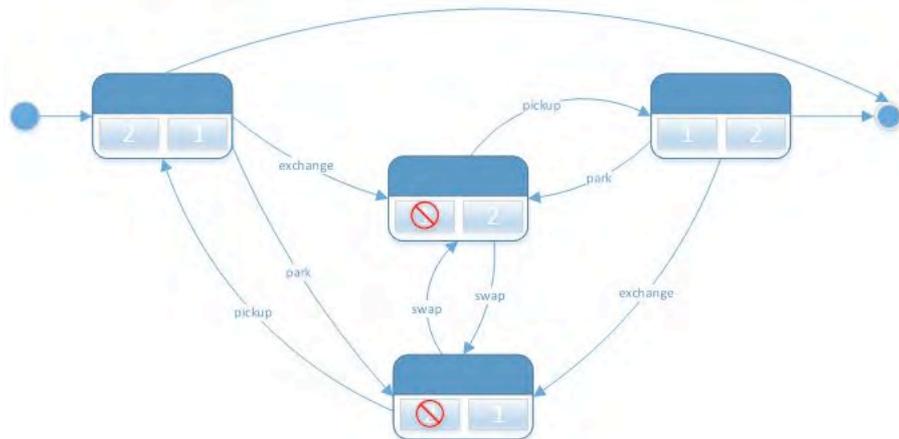


Figure 7: valid action transitions at swap locations

- the number of swap location visits per route is not limited
- we define a specific handling time per action (parking takes probably less time than swapping)
- **Distances and Driving times**
 - distances and driving times between all relevant locations are given in a matrix. The values are based on shortest path calculations in digital road networks.
 - distance and driving time matrices are asymmetric (i.e. distance and time between two locations may be different in the two possible directions).
- **Constraints**
 - we only allow round trips (routes start and end at the depot D)
 - we restrict the maximum duration of all routes
 - the loading capacity of all swap-bodies must not be exceeded
- **Cost structure and Objective Function**

The goal is to minimize the total costs of the plan as a sum of the following cost components:

- **Truck**
fixed cost if this truck is used in the solution and
variable costs (cost / km and cost / h)
- **semi-trailer**
fixed cost if this semi-trailer is used in the solution and
variable costs (cost / km)
- **swap-bodies**
no costs are related to swap-bodies
- with this definition the driver costs are approximated quite satisfactorily for practical purposes

Test instances

We are planning to provide three different test instances for the participants. Instances will be provided as plain text files including all instance data according to a specified format.

- ▶ small instance
 - approx. 50 customers
 - approx. 15 to 20 swap locations
 - approx. 15 customers where no train is allowed for service
- ▶ medium size
 - approx. 200 customers
 - approx. 50 swap locations
 - approx.. 20 customers where no train is allowed for service
- ▶ large
 - approx. 500 customers
 - approx.. 100 swap locations
 - approx. 50 customers where no train is allowed for service

For each instance, two additional variants will be considered in the evaluation of the proposed methods:

- ▶ variant 1: all customers can be visited by a train
- ▶ variant 2: no customer can be visited by a train

Furthermore we will define an additional set of “private” instances which shall be used by the jury to determine the winner of the competition.

Data files

All data are given in text files, where the first line gives the description and the following lines give data separated by a semicolon ;

Locations.txt

- ▶ Depots, Swap Locations and Customers data are available through a separate text file called **Locations** where for each customer is reported:
 - ▶ location type (DEPOT, SWAP_LOCATION or CUSTOMER)
 - ▶ a descriptive name, the post code and the city (for descriptive purposes only)
 - ▶ X and Y coordinates of the geographic point of the location: The pair of X,Y coordinates of the location will be used as identifier to determine the distance and time of the trip to other locations in the distance and time tables.
 - ▶ quantity to be delivered in load units, for customers only
 - ▶ possibility of being visited by a train (1=possible, 0=impossible), for customers only
 - ▶ possibility of performing a swap (1=yes, 0=no), for swap locations only

- ▶ service time in minutes, for customers only

Vehicles.txt

- ▶ For each equipment type (TRUCK, SEMI_TRAILER and SWAP_BODY) is reported
 - ▶ type name
 - ▶ capacity in load units
 - ▶ distance cost in monetary units per Km
 - ▶ time cost in monetary units per hour
 - ▶ fixed cost in monetary units per use

SwapActions.txt

- ▶ For swap action type (PARK, SWAP, EXCHANGE and PICKUP) is reported
 - ▶ type name
 - ▶ fixed duration in minutes

DistanceTimesCoordinates.txt

- ▶ It is a list of X and Y coordinates of the geographic points corresponding to locations.

DistanceTimesData.txt

- ▶ For each point corresponding to location (DEPOT, SWAP_LOCATION and CUSTOMER) IN THE SAME ORDER AS THAT REPORTED in DistanceTimesCoordinates.txt file:
 - ▶ The distance d_{ij} and driving period p_{ij} between the start-location i and each destination-location j for $j = 1, \dots, n$ are stored in separate rows separated by a semicolon (;).

For example: A distance table that consists of the distances and driving periods between 3 locations is stored as follows:

$d_{11}; p_{11}; d_{12}; p_{12}; d_{13}; p_{13}$
 $d_{21}; p_{21}; d_{22}; p_{22}; d_{23}; p_{23}$
 $d_{31}; p_{31}; d_{32}; p_{32}; d_{33}; p_{33}$

- ▶ All distances in this matrix are in meters and times are in seconds

A zip file including a sample instance together with a solution will be distributed together with this announcement.

Feasibility tool

We will provide a simple tool for the participants of the contest (and for us of course) to check the feasibility and the objective value of the obtained solutions. Probably this will be a simple console application for Windows.