HOM project instructions 2018/19

Public transportation vehicle parking

Problem description

These instructions describe the problem of vehicle parking in depots of a public transportation company. Vehicles are placed in parking lanes, with respect to the departure time: Each vehicle needs to be able to leave the depot in the morning easily and on time. At first, this problem looks like a bin-packing, as the decision is made where to place a vehicle knowing the length of each vehicle and the length of each lane. However, numerous operational constraints make the problem much more complex. For example:

- Given vehicles must be placed on lanes with specific equipment: tramways can be parked exclusively on lanes with rails, trolleybuses exclusively on lanes with electrical lines, etc.
- A vehicle cannot leave a lane if another vehicle is in front of it. The time of departure of a vehicle must be prior to the departure of the next vehicle in line. Also, in certain cases, vehicles from one lane may block vehicles from another lane, i.e., there are cases in which all of the vehicles from one lane must departure prior to the departure of any vehicles from another lane.
- The departure time of a vehicle must be precisely respected.
- Certain vehicles may have a specific schedule. For example, some vehicles may be used during rush hours or they may have a special maintenance schedule.
- etc.

The following text describes the format of a problem instance (input file), formulates the objectives and constraints, and describes your project task related to this problem.

Problem instance

Problem instance defines the number of vehicles to be placed on lanes, the number of lanes, and the characteristics of both, which are related to the constraints of the problem. The data provided in the instance file is defined as follows:

number of vehicles	total number of vehicles that need to be placed on the lanes of a depot
number of lanes	total number of parking lanes
vehicle length	length of each vehicle
vehicle series	series of each vehicle (related to the type of a vehicle, e.g., tramway, bus, ...)
constraints related to lane equipment	feasibility of parking a vehicle on a lane; binary matrix in which rows represent vehicles and columns represent lanes; 1 if a vehicle can be placed on a lane, 0 otherwise
lane length	length of each lane
departure time	time at which a vehicle must leave the depot
schedule type	type of schedule (related to the schedule specifics of certain vehicles)
blocked lanes	list of blocked lanes; the first number in each row represents a blocking lane, and is followed by the list of lanes blocked by a blocking lane

```
49
29

42 42 42 53 53 42 53 53 53 42 53 53 53 42 53 53 53 42 53 42 42 53 53 42 ...

2 2 2 1 1 2 1 1 1 2 1 1 1 2 1 1 1 2 1 2 2 1 1 2 2 1 2 2 2 1 2 2 1 2 2 2 ...

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
...
[etc.]
...
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

```
74 83 90 97 96 133 132 132 134 134 134 128 85 88 124 120 124 107 107 107 ...

288 333 345 360 376 393 396 399 406 410 425 438 381 419 397 255 257 262 ...

6 6 6 6 6 6 6 6 6 6 6 6 1 1 4 6 6 6 6 6 6 6 6 6 6 6 6 6 6 0 0 0 0 0 0 0 ...

5 4
8 7
11 10
12 13 14
16 15
19 18
21 22
23 22
25 24
```

Problem formulation

The problem defines the following **constrains**:

1. A vehicle is assigned to exactly one lane.
2. A lane is occupied by exactly one series - "vehicle series" in the instance (e.g., only tramways or only trolleybuses).
3. A vehicle can be placed only on a lane with the necessary equipment - "constraints related to lane equipment" (e.g., tramways can be parked on lanes with rails, buses on any lane).
4. The sum of vehicle lengths on a lane cannot exceed the lane capacity, while including the distance of 0.5 between vehicles in a lane.
5. A vehicle is assigned to exactly one position (number of a vehicle in a lane).
6. A position is occupied by exactly one vehicle.
7. The departure time of any vehicle must be prior to the vehicle following it - "departure time" in the instance.
8. The departure of all the vehicles in a blocking lane must be prior to the departure of any vehicle in blocked lanes - "blocked lanes".

This problem has **two global objectives**, both of which can be decomposed into three objectives.

First global objective is to minimize the number of different vehicle series in adjacent lanes, the number of occupied lanes, and remaining space on used lanes. The first global objective ignores the temporal component and can be formulated as:

$$min \;\; p_1 f_1 + p_2 f_2 + p_3 f_3$$

where the description of functions f_1, f_2 and f_3, and the values of weight factors p_1, p_2 and p_3 are given in table below. (**Note**: The table lists values of $\frac{1}{p_i}$.)

obj.	value	weight factor
f_1	Number of different series in adjacent lanes. (One of the constraints is that all the vehicles in a lane are the same series.) E.g., if there are 5 lanes and the series of parked vehicles in these lanes are [1 1 1 2 2], the value of the objective function will be 1, and if the series of vehicles in lanes are [1 2 1 2 1], then the value of the function will be 4.	$\frac{1}{p_1}$ = total number of lanes - 1
f_2	Number of used lanes.	$\frac{1}{p_2}$ = total number of lanes
f_3	Sum of remaining capacity on used lanes. Note: between each 2 vehicles there is a distance of 0.5, which is considered as used capacity.	$\frac{1}{p_3}$ = sum of capacities of all lanes - sum of lengths of all vehicles

Second global objective is to maximize the number of vehicles with the same schedule type on each lane, the number of vehicles with the same schedule type between successive lanes, and to maximize best practices between any two departures. The second global objective is focused on scheduling departures from the depot, and can be formulated as:

$$max \;\; r_1 g_1 + r_2 g_2 + r_3 g_3$$

where the description of functions g_1, g_2 and g_3, and the values of weight factors r_1, r_2 and r_3 are given in table below. (**Note**: The table lists values of $\frac{1}{r_i}$.)

obj.	value	weight factor
g_1	Number of adjacent vehicle pairs in a lane with the same schedule type. E.g., if there are 5 vehicles in a lane with schedule types [1 1 1 2 2], the function value will be 3, and if the schedule types are [1 2 2 3 4], then the function value will be 1. Sum up the values for all lanes.	$\frac{1}{r_1}$ = total number of vehicles - number of used lanes
g_2	Number of adjacent lanes for which the last vehicle in a lane is of same schedule type as the first vehicle in next lane.	$\frac{1}{r_2}$ = total number of lanes - 1
g_3	Sum of rewards for time intervals between departures, for all adjacent vehicle pairs, in all lanes. For example, if there are 3 lanes with 2, 3, and 4 vehicles parked respectively, then we sum up the reward/penal for the time interval between one pair of vehicles in the first lane, two pairs of vehicles in the second lane, and 3 pairs of vehicles in the third lane. The reward/penal (n) is calculated depending on the time interval (vr) between departures of a pair of vehicles, according to: $$n = \begin{cases} 15 & 10 \leqslant vr \leqslant 20 \\ 10 & vr > 20 \\ -4 * (10 - vr) & vr < 10 \end{cases}$$ All time intervals longer than 10 minutes are rewarded, while the reward is even higher for ideal intervals from 10 to 20 minutes. Intervals shorter that 10 minutes are penalized.	$\frac{1}{r_3}$ = 15 * number of evaluated pairs (adjacent vehicle pairs)

Project task

1. Design and implement a heuristic algorithm to solve the given problem.
2. Execute your algorithm for given instances of the problem.
3. Save **3 solutions for each instance**: solution obtained after 1 minute of algorithm execution, 5 minutes of algorithm execution and without time constraints.

4. For each solution, save the value of both objective functions, and the number of iterations in which you evaluated the objective function until obtaining the saved solution.

5. Create a report that describes your implemented heuristic algorithm. The report should include the following:

   - A description of the problem.
   - A description of the implemented heuristic algorithm (solution representation, objective/fitness function, way of construction of an initial solution, iteration size and termination criterion, heuristic specific design elements, etc.).
   - Pseudocode of the implemented algorithm.
   - Analysis of results and discussion (e.g., influence of certain parameters of the heuristic algorithm on the quality of solutions, execution time of the algorithm)
   - Conclusion (e.g., discussion of potential further improvements of the implemented algorithm).

Files with obtained solutions must:

- be named “res-time-instance.txt”, where “time” equals “1m”, “5m” or “un” (unlimited), and “instance” equals “i1”, “i2”, etc.
- contain information about the placement of vehicles in the public transportation depot, in the following format:

```
5
7
12 31
23 11
1 6 17
...
```

The rows in the solution file represent lanes, and the values in each row represent identifiers of parked vehicles. For example, vehicle 5 is parked in the first lane, vehicle 7 in the second lane, vehicles 12 and 31 (in that order) in third lane, etc. Vehicle identifiers must be in line with the order in which they are defined in the instance file. Assume that the vehicle identifiers are integers from the interval [1 .. $total\_number\_of\_vehicles$].