

# Pickup & Delivery Problem with Multiple Time Windows

Preben Bucher-Johannessen

October 12, 2018

## Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Problem Formulation</b>	<b>1</b>
<b>3</b>	<b>Solution in AMPL</b>	<b>3</b>
<b>4</b>	<b>References</b>	<b>4</b>

# 1 Introduction

TODO..

## 2 Problem Formulation

Our problem is typically for a vehicle manufacturing company that has a certain set of factories where the vehicles are produced, each with a set of demanded parts for production. The parts, or orders (typically named transport orders but we refer to orders here), can be delivered from a set of suppliers. At a given point in time (could be delivery for one day or a week) we consider a planning problem with a demand for orders that should be satisfied with the given suppliers using vehicles provided by different logistics companies.

The vehicles have different capacities, cost structures, incompatibilities and start at the first pickup location at the first pickup time (ie. costs to get from start to first pickup are not considered neither is time). The cost structure can be multidimensional taking into account total distance and volume in different formats (variable costs, fix costs etc).

When a delivery is assigned to a vehicle, the vehicle must load the delivery from the supplier, and deliver the delivery at the factory dock. The docks in each factory can differ according to which order is being delivered and there is a limit to how many docks each vehicle can visit at each factory.

Each delivery/pickup can have several time windows, lapsing sometimes over several days. If a car arrives before a time window it has to wait.

The mathematical formulation of the problem will now follow where the set of vehicles used is denoted by  $V$  and capacity of each vehicle  $v \in V$  is denoted by  $K_v$ . We let  $n$  be the number of orders in the problem, so if  $i$  is a specific pickup-node then  $i + n$  corresponds to the delivery node for the same order. Here it follows that for each vehicle we have a set  $O_v$  of orders that vehicle  $v \in V$  can transport based on the incompatibilities. The set of pickup Nodes (supplier docks) we denote using  $N^P$  and each delivery node (Factory dock) is denoted by  $N^D$ . Each Factory,  $f \in F$ , also has a set of Nodes belonging to the same factory which we denote  $N_f$ . The delivery nodes that belong to one factory can then be represented as  $N_f^D = N_f \subset N^D$ . Each vehicle has a set of Nodes it can travel to (corresponding to orders  $O_v$ ) represented by  $N_v$ . This set also includes an origin node,  $o(v)$  and a destination node  $d(v)$  corresponding to the first pickup and the last delivery of vehicle  $v$ . The set of Edges that each vehicle can traverse is represented by  $E_v$ . The factory docking limit is denoted by  $D_f$ . Each vehicle has a given

amount of docks per factory it wants to deliver to represented by  $d_f^v$ .

Each node has a set of  $h$  time windows represented by  $T_i^h$  which contains tuples of  $[T_{ik}, \overline{T_{ik}}]$  where  $k \in \{0, 1, \dots, h\}$ . each vehicle has a current time based on which node its currently at, denoted by  $t_i^v$  and a current load, denoted by  $l_i^v$ , where  $i \in n$  and  $v \in V$ . The distance from node  $i$  to node  $j$  is denoted by  $d_{ij}$ . The cost of vehicle  $v$  driving from node  $i$  to  $j$  is denoted by  $C_{ij}^{vq(v)w(v)}$  and the corresponding time between the nodes  $T_{ij}^v$ .  $l_i^v$  is the load of vehicle  $v$  leaving node  $i$  and  $x_{ij}^v$  is a binary variable indicating if vehicle  $v$  is travelling between  $i$  and  $j$  node. The cost of not transporting an order will be the same for each order (should be set relatively high to avoid dummy transports) so we denote this with  $C^N$  and each order,  $i$ , not transported is represented by a binary variable  $y_i$ .

$$\min \sum_{v \in V} \sum_{(i,j) \in E_v} C_{ij}^{vq(v)w(v)} x_{ij}^v + \sum_{i \in N^P} C^N y_i \quad (1)$$

subject to:

$$\sum_{v \in V} \sum_{j \in N_v} x_{ij}^v + y_i \geq 1, \quad i \in N^P \quad (2)$$

$$\sum_{j \in N_v} x_{ij}^v - \sum_{j \in N_v} x_{ji}^v = 0, \quad v \in V, i \in N_v \setminus \{o(v), d(v)\} \quad (3)$$

$$\sum_{j \in N_v} x_{o(v)j}^v = 1, \quad v \in V \quad (4)$$

$$\sum_{j \in N_v} x_{jd(v)}^v = 1, \quad v \in V \quad (5)$$

$$\sum_{j \in N_v} x_{ij}^v - \sum_{j \in N_v} x_{(i+n)j}^v = 0, \quad v \in V, i \in N_v^P \quad (6)$$

$$0 \leq l_i^v \leq K_v, \quad v \in V, i \in N_v^P \quad (7)$$

$$0 \leq d_{if}^v \leq D_f, \quad v \in V, i \in N_v^D, f \in F \quad (8)$$

,

$$t_{iv} + T_{ij}^v - t_j^v \leq (\overline{T_{ik}} + T_{ij}^v)(1 - x_{ij}^v), \quad v \in V, (i, j) \in A_v \quad (9)$$

$$t_i^v + T_{i(i+n)}^v - t_{(i+n)}^v \leq 0, \quad v \in V, i \in N_v^P \quad (10)$$

,

$$\underline{T}_{ik} \leq t_i^v \leq \overline{T}_{ik}, \quad v \in V, i \in N_v, \quad (11)$$

$$y_i \in \{0, 1\} \quad i \in N_v^P \quad (12)$$

$$x_{ij}^v \in \{0, 1\} \quad v \in V, (i, j) \in A_v \quad (13)$$

The objective function (1) sums up to the cost of the spot cars and the aim is to minimize these costs.

List of Variables:

$v \in V$  - Vehicles

$n$  - number of orders

$i$  - orders

$K_v$  - Vehicle Capacity

$w(v)$  - Max load for Vehicle  $v$

$q(v)$  - Max travel distance for Vehicle  $v$

$O_v$  - Set of orders  $i$  that vehicle  $v$  can transport

$N^P$  - pickup nodes

$N^D$  - delivery Nodes

$f \in F$  - Factory

$D_f$  - Factory docking limit

$d_{if}^v$  - docking amount at node  $i$  per vehicle  $v$  and factory  $f$

$C_{ij}^{vqw}$  - Cost of transport from  $i$  to  $j$  with vehicle  $v$  given the vehicles  $q$  and  $w$

$T_i^h$  - Node  $i$  has a sets of  $h$  time windows in tuples  $[T_{ik}, \overline{T}_{ik}]$  where  $k \in \{0, 1, \dots, h\}$

$T_{ij}^v$  - Travel time from  $i$  to  $j$  using  $v$   $E_v$  - Edges that vehicle  $v$  can traverse

$l_i^v$  - Load at node  $i$  with vehicle  $v$

$x_{ij}^v$  - binary variable indicating travel from  $i$  to  $j$  with  $v$

$y_i$  - binary indicating order not picked up

$C^N$  - cost of not transporting an order

### 3 Solution in AMPL

TODO..

## 4 References

Lu 2012

<https://www.sciencedirect.com/science/article/pii/S0305054813002165>

<https://www.sciencedirect.com/science/article/pii/S0305048304001550>

<https://link.springer.com/article/10.1007/s10732-005-5432-5>

<https://pubsonline.informs.org/doi/abs/10.1287/trsc.1050.0135>