RAIRO-Oper. Res. 55 (2021) 2763–2768
https://doi.org/10.1051/ro/2021132

RAIRO Operations Research
www.rairo-ro.org

A NOTE ON "VARIABLE NEIGHBORHOOD SEARCH BASED ALGORITHMS FOR CROSSDOCK TRUCK ASSIGNMENT"

Shahin Gelareh*

Abstract. Some implementations of variable neighborhood search based algorithms were presented in "Daquin *et al. RAIRO: OR* **55** (2021) 2291–2323". This work is based on model in "Miao *et al. Eur. J. Oper. Res.* **192** (2009) 105–115" which has been proven to be incorrect. We reiterate and elaborate on the deficiencies in the latter and show that the authors in the former were already aware of the deficiencies in the latter and the proposed minor amendment does not overcome any of such deficiencies.

Mathematics Subject Classification. 90C10, 90C27.

Received July 17, 2021. Accepted August 8, 2021.

1. Introduction

Daquin *et al.* [1] proposed implementation of a set of Variable Neighborhood Search (VNS) techniques to solve the problem proposed in Miao *et al.* [5]. In [5], the authors proposed a mathematical formulation for truck dock assignment in a crossdock. The optimal solution of this model is then used as a reference point for measuring the quality of their proposed genetic algorithm.

In a joint work with one of the co-authors in [1], we have shown in [3] and in a more elaborated form in [2] that this model has some serious issues and fails to deliver the expected outcome. More precisely we have shown by an example that the model may deliver "*optimal*" solutions that are more than 45% away from the real optimal solution. In short, the problem description and the proposed model do not match.

In Daquin *et al.* [1], the authors introduce a simple multiplier (coefficient) in the cost function intending to correct the model itself. However, it is obvious that no modification in the objective can be a correction to the polyhedral structure and the feasible region remains untouched. In the following, we re-iterate and re-emphasize on those issues.

2. Mathematical model

According to Miao *et al.* [5] and Lim *et al.* [4]: *an over-constrained truck dock assignment problem with time window, operational time, and capacity constraint in a transshipment network through cross docks where the number of trucks exceeds the number of docks available and the capacity of the cross dock is limited, and where*

Keywords. Crossdocking, Integer programming.

Département Réseaux et Télécommunications, Université d'Artois, 62400 Béthune, France
*Corresponding author: shahin.gelareh@univ-artois.fr; shahin.gelareh@gmail.com

the objective is to minimize the operational cost of the cargo shipment and the number of unfulfilled shipments is studied.

The objective function accounts for the total cost of 1) dock operations, 2) penalties for the unfulfilled shipments.

The following parameters and variables are introduced therein:

Parameters:

N: set of trucks arriving at and/or departing from the cross dock
M: set of docks available in the cross dock
n: total number of trucks, that is $|N|$ ($|N|$ denotes cardinality of set N)
m: total number of docks, that is $|M|$
a_i: arrival time of truck i ($1 \leq i \leq n$)
d_i: departure time of truck i ($1 \leq i \leq n$)
t_{kl}: operational time for pallets from dock k to dock l ($1 \leq k, l \leq m$)
f_{ij}: number of pallets transferring from truck i to truck j ($1 \leq i, j \leq n$)
c_{kl}: operational cost per unit time from dock k to dock l ($1 \leq k, l \leq m$)
p_{ij}: penalty cost per unit cargo from truck i to truck j ($1 \leq i, j \leq n$)
C: capacity of cross dock, *i.e.* the maximum number of cargos the cross dock can hold at a time
$\hat{x}_{ij}$: 1 iff truck i departs no later than truck j arrives; 0 otherwise.

Assumption 2.1 ([5])**.** *It has been also assumed that:*
[-]

- $f_{ij} \geq 0$ *iff* $d_j \geq a_i, \forall i, j \neq i$, *otherwise* $f_{ij} = 0$ *meaning that truck* i *will transfer some cargo to truck* j *iff truck* j *departs no earlier than truck* i *arrives;*
- $a_i < d_i (1 \leq i \leq n)$ *which means for each truck, the arrival time should strictly smaller than its departure time;*
- $n > m$ *which satisfies the over-constrained condition;*
- *sort all the* a_i *and* d_i *in an increasing order, and let* t_r $(r = 1, 2, ..., 2n)$ *correspond to these* $2n$ *numbers such that* $t_1 \leq t_2 \leq \cdots \leq t_{2n}$. *Using this notation, we can easily formulate the set of capacity constraints later.*

Variables:

y_{ik}: 1 if truck i is assigned to dock k; 0, otherwise.
z_{ijkl}: 1 if truck i is assigned to dock k and truck j is assigned to dock l; 0 otherwise.

2.1. Mathematical model

We call this problem CROSS-DOCK:

[CROSS-DOCK]

$$\min \sum_{k=1}^{m} \sum_{l=1}^{m} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{kl} t_{kl} z_{ijkl} + \sum_{i=1}^{n} \left(\sum_{j=1}^{n} p_{ij} f_{ij} \left(1 - \sum_{k=1}^{m} \sum_{l=1}^{m} z_{ijkl} \right) \right) \tag{2.1}$$

s.t.

$$\sum_{k=1}^{m} y_{ik} \leq 1, \qquad \forall i \tag{2.2}$$

$$z_{ijkl} \leq y_{ik}, \qquad \forall i, j, k, l : j \neq i \tag{2.3}$$

$$z_{ijkl} \leq y_{jl}, \qquad \forall i, j, k, l : j \neq i \tag{2.4}$$

$$y_{ik} + y_{jl} - 1 \leq z_{ijkl} \qquad \forall i,j,k,l : j \neq i \tag{2.5}$$

$$\hat{x}_{ij} + \hat{x}_{ji} \geq z_{ijkk} \qquad \forall i,j,k,l : j \neq i \tag{2.6}$$

$$\sum_{k=1}^{m}\sum_{l=1}^{m}\sum_{j}^{n}\sum_{i\in\{i:a_i\leq t_r\}} f_{ij}z_{ijkl} - \sum_{k=1}^{m}\sum_{l=1}^{m}\sum_{i}^{n}\sum_{j\in\{j:d_j\leq t_r\}} f_{ij}z_{ijkl} \leq C \qquad \forall r \in \{1,2,\ldots,2n\} \tag{2.7}$$

$$f_{ij}z_{ijkl}(d_j - a_i - t_{kl}) \geq 0 \qquad \forall i,j,k,l : j \neq i \tag{2.8}$$

$$y_{ik} \in \{0,1\}, z_{ijkl} \in \{0,1\} \tag{2.9}$$

The first part in (2.1) accounts for the cost of transport, and the second part calculates the penalty for the unfulfilled transports. Equation (2.2) ensures that a truck can be assigned to no more than one dock. Consider constraints (2.8) before the others. Equation (2.8) indicates that, if truck i uses dock k and truck j uses dock l then the variable z_{ijkl} *may* take 1, only if the arrival of i plus the transfer time from k to l is not later than the departure of j. That means, there is a sense of *direction of flow* associated to each variable z_{ijkl} which is not covered in the definition of this variable in [5]. Therefore, to this end, we associate such flow direction to the variable z_{ijkl}.

Equations (2.3) and (2.4) indicate that for a pallet transfer between truck i and truck j, i can use dock k and j can use dock l, if i is allocated to k and j is allocated to l. Constraints (2.5) imply that if the truck i is assigned to dock k and truck j to dock l, a bidirectional transfer between the two trucks – from i to j represented by z_{ijkl} and from j to i represented by z_{jilk} – must take place. If the transfer does not take place, then *not* both trucks can be docked. Moreover, if transfer from i to j does not take place, from j to i must not take place either. *I.e.* $z_{ijkl} = 1$ iff $z_{jilk} = 1$ and $z_{ijkl} = 0$ iff $z_{jilk} = 0$. Constraints (2.6) ensure that truck i and truck j can use the same dock for realizing the transfer of pallets from i to j, only if their time windows do not intersect – *i.e.*, i leaves no later than j arrives. Constraints (2.7) guarantee that at every event time (arrival and/or departure of a truck), the capacity of cross dock is respected.

For a given i, j, k, l: $j \neq i$, the corresponding constraint in (2.8) exists only if f_{ij} and $(d_j - a_i - t_{kl})$ are nonzero, otherwise f_{ij} or $(d_j - a_i - t_{kl})$ would void the constraint. As a consequence, this would be z_{jilk} which influences the value that z_{ijkl} must take – as the left-hand side of (2.5) is the same for z_{ijkl} and z_{jilk}.

3. Issues in the model

The model possesses several issues some of which we could identify are listed in the sequel:

3.1. Constraints (2.5) eliminate correct solutions of the problem

Case 1: Given $i, j \neq i$ such that $f_{ij} = 0$, according to Note 2.1, if j would have already left before i arrives then $f_{ij} = 0$ and i would not be able to deliver to j. But, if $f_{ji} \neq 0$ and a sufficient capacity on the cross dock would have been available, truck j might have already dropped off its cargo on the cross dock buffer – before it departs. This cargo can be later on transferred to i, once i arrives. Having $f_{ij} = 0$, voids the corresponding constraint in (2.8) and this constraint does not make any decision on z_{ijkl}. If for some k, l, $z_{jilk} = 1$, as it is possible, then constraint (2.5) forces z_{ijkl} to take 1, too, for a zero-size load of pallets from i to j.

When $f_{ij} = 0$ by Note 2.1 and $z_{ijkl} = 1$ (due to equality with z_{jilk}), we have $p_{ij}f_{ij}\left(1 - \sum_{k=1}^{m}\sum_{l=1}^{m} z_{ijkl}\right)$ equal to zero because no penalty is going to be paid for a flow that does not exist, *i.e.* $f_{ij} = 0$. However, still the first term in the objective function (*i.e.* for the same i, j, $\sum_{k=1}^{m}\sum_{l=1}^{m} c_{kl}t_{kl}z_{ijkl}$) is contributing in the objective values by forcing to pay transport cost for pallets, which do not exist. That means, for $f_{ij} = 0$ we still have to pay for the transfer cost.

This is a direct consequence of tying up the destiny of z_{ijkl} and z_{jilk} to each other using (2.5).

Case 2: Given $i, j, k, l : j \neq i$, suppose that $(d_j - a_i) \geq 0$ and $(d_j - a_i - t_{kl}) \leq 0$ such that j does not leave before i arrives, but there is no sufficient time to make the transfer between dock k and dock l. However, if j

has left something on the buffer before it leaves, i which arrives later can still take it and z_{jilk} can take 1. This does not fall into the category of Note 2.1 and f_{ij} is not necessarily 0 (might be strictly positive). Therefore, corresponding constraint in (2.8) exists and forces $z_{ijkl} = 0$. On the other hand, Equation (2.5) forces z_{jilk} to take 0 and avoids it to take 1.

In such a case, having $z_{jilk} = 1$ in a feasible solution of the real-life problem is possible (assuming a sufficient capacity), but it is eliminated by this model.

Therefore, forcing z_{jilk} and z_{ijkl} to take the same values (*i.e.* having (2.5) in the model) is *incorrect* and those constraints imposing such conditions must be removed from CROSS-DOCK.

3.2. Constraints (2.6)

Because a variable z_{ijkl} implies a sense of direction of flow, we make the following re-definition:

z_{ijkl}: 1 if pallets of truck i which is assigned to dock k are transferred to the truck j which is assigned to dock l; 0 otherwise.

There are two different cases allowing two distinct trucks i and j use the same dock k: 1) truck i drops its pallets and leaves dock k before j arrives at dock k, 2) truck j drops its pallets and leaves dock k before truck i arrives at dock k. Therefore, z_{ijkk} only depends on whether i leaves before j arrives or not. Whether j departs before i arrives or not is *not* directly related to z_{ijkk}. Because if j depart before i arrives (*i.e.* $\hat{x}_{ij} = 0$ and $\hat{x}_{ji} = 1$), still there is no possibility of transferring from i to j, but the constraint turns to $z_{ijkk} \leq \hat{x}_{ij} + \hat{x}_{ji} = 1$ which is of no effect unless it causes numerical deficiencies. While $z_{ijkk} \leq \hat{x}_{ij} = 0$, clearly sets the variable to the correct value. Consequently, constraints (2.6) should be replaced by:

$$\hat{x}_{ij} \geq z_{ijkk} \qquad \forall i, j \neq i, k, l \tag{3.1}$$

4. Example

Let $m = 2$ be the number of docks and $n = 4$ be the number of trucks. Moreover, let: $c = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$, $t = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}$, $p = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}$, $f = \begin{pmatrix} 0 & 2 & 2 & 2 \\ 2 & 0 & 2 & 2 \\ 2 & 2 & 0 & 2 \\ 0 & 2 & 2 & 0 \end{pmatrix}$, $a = \begin{pmatrix} 15.42 \\ 15.50 \\ 17.00 \\ 16.52 \end{pmatrix}$ and $d = \begin{pmatrix} 16.41 \\ 16.41 \\ 18.00 \\ 17.57 \end{pmatrix}$.

The optimal solution, s^*, of this problem using the model CROSS-DOCK has an objective value 16 and all the variables take 0. We have $\hat{x}_{13} = 1 \hat{x}_{14} = 1, \hat{x}_{23} = 1, \hat{x}_{24} = 1$.

Here, the objective only accounts for the penalties because no pallet transfer takes place.

The *claim* is that this model does not correctly represent the problem. In other words, there are other feasible solutions with better objective values to the problem which are eliminated by this model (in particular, constraints (2.5)).

We remove constraint (2.5) and rectify (2.6) and call the model R-CROSS-DOCK:
[R-CROSS-DOCK]

$$\min \sum_{k=1}^{m} \sum_{l=1}^{m} \sum_{i=1}^{n} \sum_{j=1}^{n} c_{kl} t_{kl} z_{ijk} + \sum_{i=1}^{n} \left(\sum_{j=1}^{n} p_{ij} f_{ij} \left(1 - \sum_{k=1}^{m} \sum_{l=1}^{m} z_{ijkl} \right) \right) \tag{4.1}$$

s.t.

$$\sum_{k=1}^{m} y_{ik} \leq 1, \qquad \forall i \tag{4.2}$$

$$z_{ijkl} \leq y_{ik}, \qquad \forall i, j, k, l \tag{4.3}$$

$$z_{ijkl} \leq y_{jl}, \qquad \forall i,j,k,l \qquad (4.4)$$

$$y_{ik} + y_{jk} \leq 1 + \hat{x}_{ij} + \hat{x}_{ji} \qquad \forall i,j,k \qquad (4.5)$$

$$z_{ijkk} \leq \hat{x}_{ij} \qquad \forall i, j \neq i, k, l \qquad (4.6)$$

$$\sum_{k=1}^{m}\sum_{l=1}^{m}\sum_{j}^{n}\sum_{i\in\{i:a_i\leq t_r\}} f_{ij}z_{ijkl} + \sum_{k=1}^{m}\sum_{l=1}^{m}\sum_{i}^{n}\sum_{j\in\{j:d_j\leq t_r\}} f_{ij}z_{ijkl} \leq C \qquad \forall r \in \{1,2,\ldots,2n\} \qquad (4.7)$$

$$z_{ijkl} = 0 \qquad \forall i,j,k,l : j \neq i, (d_j - a_i - t_{kl}) \leq 0 \qquad (4.8)$$

$$y_{ik} \in \{0,1\}, z_{ijkl} \in \{0,1\} \qquad (4.9)$$

The new constraints (4.5) ensure that if the arrival/departure time windows of two trucks i and j overlap ($\hat{x}_{ij} = \hat{x}_{ji} = 0$), either of them can be docked at a dock k (not both). The optimal solution, s'^{*}, of this problem using the model CROSS-DOCK-R has an optimal objective value of 11 and $z_{1311} = 1, z_{1412} = 1, z_{2321} = 1, z_{2422} = 1, z_{4321} = 1, y_{11} = 1, y_{22} = 1, y_{31} = 1, y_{42} = 1$ while $\hat{x}_{13} = 1, \hat{x}_{14} = 1, \hat{x}_{23} = 1, \hat{x}_{24} = 1$ and the relative gap between the two solutions is 45.45%.

4.1. Why s'^{*} is not feasible in s^{*}

A diagnosis of infeasibility has reported that an *Irreducibly Inconsistent Set (IIS)* of constraints is consisted of $y_{11} + y_{22} <= 1 + z_{1212}$ as the cause of infeasibility.

In s'^{*}, we have $y_{11} = 1,\ y_{22} = 1$. By substituting in constraint (2.6), we have $1 + 1 = y_{11} + y_{22} <= 1 + z_{1212}$ ($1+1 = y_{11}+y_{22} <= 1+z_{2121}$) which forces $z_{2121} = 1$. However, constraint (2.8) has already suggested $z_{1212} = 0$ (and $z_{2121} = 0$) as $d_j - a_i - t_{kl} < 0$ (and $d_i - a_j - t_{lk} < 0$) which is a contradiction.

5. Summary and conclusion

We carefully devised an instance of the problem and showed that the model in Miao *et al.* [5] has some fundamental polyhedral deficiencies and does not correspond to the problem description they proposed therein. This model does not necessarily produce an optimal solution and cannot be used as a reference point to measure the quality of heuristic solutions. The amendment proposed in Daquin *et al.* [1] do not touch the polytope (feasible space) and therefore, all the structural issues remain intact.

Acknowledgements. The author is grateful of the constructive comments provided by anonymous reviewers.

References

[1] C. Daquin, H. Allaoui, G. Goncalves and T. Hsu, Variable neighborhood search based algorithms for crossdock truck assignment. *RAIRO: OR* **55** (2021) 2291–2323.

[2] S. Gelareh, G. Goncalves and R. Monemi, *On truck dock assignment problem with operational time constraint within cross docks.* Working Paper (2015).

[3] S. Gelareh, R.N. Monemi, F. Semet and G. Goncalves, A branch-and-cut algorithm for the truck dock assignment problem with operational time constraints. *Eur. J. Oper. Res.* **249** (2016) 1144–1152.

[4] A. Lim, H. Ma and Z. Miao, Truck dock assignment problem with operational time constraint within crossdocks, edited by M. Ali and R. Dapoigny. In: *Advances in Applied Artificial Intelligence*, volume 4031 of *Lecture Notes in Computer Science.* Springer Berlin Heidelberg (2006) 262–271.

[5] Z. Miao, A. Lim and H. Ma, Truck dock assignment problem with operational time constraint within crossdocks. *Eur. J. Oper. Res.* **192** (2009) 105–115.