The Multi Trip Container Truck Scheduling with Synchronized Empties Re-positioning in a Dry-port Setting

S. Fazi[a,*], S. K. Choudhary [b]

[a,*] Delft University of Technology, Delft, the Netherlands
s.fazi@tudelft.nl

[b] Georgia Institute of Technology, Atlanta, USA
schoudhary66@gatech.edu

[*] Corresponding author

Extended abstract submitted for presentation at the 11th Triennial Symposium on Transportation Analysis conference (TRISTAN XI)
June 19-25, 2022, Mauritius Island

March 24, 2022

Keywords: Dry-port; daily truck scheduling; empty containers management; column-and-row generation.

1 INTRODUCTION

Truck transport plays a pivotal role in inland container supply chains, due to fast transport times, and direct connection with (almost) all destinations (Fazi *et al.*, 2020, Zhang *et al.*, 2020). Also, in the context of empty containers, trucks are crucial to re-position them quickly at the sea port side in order to avoid usage penalties for shippers from shipping lines and to meet the request of shippers in the hinterland. It is a challenge for truck operators to minimize trucking costs and to devise schedules that fit with the highly diverse requirements of the hinterland supply chain.

In this study, we propose a multi-trip full-truck container scheduling problem that considers a general inland container setting, where a central dry-port has to satisfy a set of requests from a set of shippers located in its hinterland. A shipper may request full or empty containers to be delivered or picked up within time windows. Origins and destinations include, besides the shipper's premises, also the dry-port and the sea port. Full containers if unloaded directly may be re-used as empty containers in the network to satisfy other requests or can be re-positioned at the dry-port to fill up the available limited stock, also used to satisfy empty requests. If necessary, the trucks can retrieve empty containers from a local (empty) depot, obviously generating extra transport costs. The goal is to satisfy the shippers' requests at a minimum cost with a fleet of trucks, which may perform multiple trips during the day. The challenge is also the synchronization of the trucks' trips for the re-usage of empty containers in order to use as much as possible the available stock of containers at the dry-port.

In the literature, the problem of container truck routing and scheduling, known also as drayage problem, has been addressed in a few studies. In general, trucks are bounded to very short trips since the carried containers are related to a single shipping request and that trucks can in general only carry one or two containers (Imai *et al.*, 2006). Several variants have been developed. Consideration of single and multiple starting depots, reposition of empty containers, time windows, multiple container sizes, etc. Among others, see as main examples: Caris & Janssens (2009), Zhang *et al.* (2020) and Zhang *et al.* (2011). However, available studies have not considered, or at least not together, empty container inventory management and multi-trip scheduling problems. The former has been considered mostly as relocation problem and with empty containers as unlimited resource; Zhang *et al.* (2020) have addressed the problem, but developed a non-linear formulation.

After modelling the problem mathematically with a mixed integer linear programming (MILP) formulation, we propose an exact approach based on a column-and-row generation algorithm embedded in a branch-and-price framework. The row generation is needed since the stock of empty containers at the dry-port can change dynamically over time; therefore, trucks may influence each other trips. Numerical experiments follow to assess the performance of the methodology against the MILP formulation. In this regard, classical Solomon's instances have been adapted to the problem.

2 Mathematical model

We consider a network $G(N, \mathcal{A})$ with $\mathcal{A}$ the set of arcs and with N set of nodes that consists of the dry-port (node 0), the depot for empties (node 1), and a node for each shipper. S identifies the subset of shippers within N. C_{ij} is the travelling distance between node i and j. The sea port node is "bypassed" in our network, meaning that if a shipper requires an inbound or outbound connection to it, the distance from/to the other nodes includes the detour through the sea port.

We define six types of shippers depending on whether they request an inbound or outbound full or empty container. In particular a shipper type is identified with the symbol $\Diamond^{ab}$, where a stands for the inbound request and b for the outbound. For example, a shipper $\Diamond^{FE}$ receives a full container and releases immediately an empty one; a shipper $\Diamond^{E\varnothing}$ simply requests an empty container and releases nothing. Next, we define the set of trucks K which can carry one container at a time. This is a common assumption in the drayage literature, due to the larger use of 40ft units. Also, we define $R = 1 \dots |R|$ as the set of possible trips. If $i < j$ with $i, j \in R$, then trip i will be scheduled before trip j. For each shipper $i \in S$, the time window is represented by $[A_i, D_i]$. Finally, a number of empty containers E is initially available at the dry-port.

With regard to variables, x_{ij}^{kr} is the binary routing variable for truck k in trip r and arc (i, j). Time variable t_i^{kr} ($\in \mathbb{R}^+$) is for node $i = 0$ the departure time, whereas for $i \neq 0$ is the arrival time. t_{end}^{kr} ($\in \mathbb{R}^+$) is the end time of trip r of truck k. Concerning variables keeping track of the inventory available at the dry-port, binary variable $z^{\alpha\beta,kr}$ equals 1 if truck α in trip β starts after truck k in trip r, and is relevant when "$\alpha\beta$" starts with an empty container removal from the dry-port, and "kr" ends by filling the dry-port with an empty container. Finally, binary variable $y^{\alpha\beta,kr}$ is 1 if truck/trip "$\alpha\beta$" starts after "kr", and is relevant if both trips started with an empty container removal from the dry-port.

We formulate the problem as follows:

$$\min \sum_{k\in K}\sum_{r\in R}\sum_{(ij)\in\mathcal{A}} x_{ij}^{kr} C_{ij} \tag{1}$$

$$\sum_{(0i)\in\mathcal{A}} x_{0i}^{kr} \leq 1 \qquad \forall k \in K, r \in R \tag{2}$$

$$\sum_{j:(ji)\in\mathcal{A}} x_{ji}^{kr} = \sum_{j:(ij)\in\mathcal{A}} x_{ij}^{kr} \qquad \forall k \in K, r \in R, i \in N \tag{3}$$

$$\sum_{k\in K}\sum_{r\in R}\sum_{j:(ji)\in\mathcal{A}} x_{ji}^{kr} = 1 \qquad \forall i \in S \tag{4}$$

$$t_i^{kr} + C_{ij} \leq t_j^{kr} + M(1 - x_{ij}^{kr}) \qquad \forall k \in K, r \in R, i \in N, j \in N/\{0\} \tag{5}$$

$$t_i^{kr} + C_{i0} - M(1 - x_{i0}^{kr}) \leq t_{end}^{kr} \qquad \forall k \in K, r \in R, i \in S \tag{6}$$

$$(A_i + C_{i0})x_{i0}^{kr} \leq t_{end}^{kr} \qquad \forall k \in K, r \in R, i \in S \tag{7}$$

$$t_i^{kr} \leq D_i \qquad \forall k \in K, r \in R, i \in S \tag{8}$$

$$t_j^{kr} \geq (A_i + C_{ij})x_{ij}^{kr} \qquad \forall k \in K, r \in R, i \in S, j \in S \tag{9}$$

$$t_0^{kr+1} \geq t_{end}^{kr} \qquad \forall k \in K, r \in 1 \ldots |R| - 1 \tag{10}$$

$$t_0^{kr} \leq t_{end}^{kr} \qquad \forall k \in K, r \in R \tag{11}$$

$$t_0^{\alpha\beta} - t_0^{kr} \leq M y^{\alpha\beta,kr} + M(1 - \sum_{i:(0i) \in \diamondsuit^{E\varnothing} \cup \diamondsuit^{EF}} x_{0i}^{\alpha\beta}) + M(1 - \sum_{i:(0i) \in \diamondsuit^{E\varnothing} \cup \diamondsuit^{EF}} x_{0i}^{kr}) \qquad \forall k, \alpha \in K, r, \beta \in R \tag{12}$$

$$y^{\alpha\beta,kr} + y^{kr,\alpha\beta} = 1 \qquad \forall k, \alpha \in K, r, \beta \in R \tag{13}$$

$$(2 - y^{\alpha\beta,\gamma\delta} + y^{kr,\alpha\beta}) \geq 1 - y^{kr,\gamma\delta} \qquad \forall k, \alpha, \gamma \in K, r, \beta, \delta \in R \tag{14}$$

$$t_0^{\alpha\beta} - t_{end}^{kr} \leq M z^{\alpha\beta,kr} + M(1 - \sum_{i:(i0) \in \diamondsuit^{FE}} x_{i0}^{kr}) + M(1 - \sum_{i:(0i) \in \diamondsuit^{E\varnothing} \cup \diamondsuit^{EF}} x_{0i}^{\alpha\beta}) \qquad \forall k, \alpha \in K, r, \beta \in R \tag{15}$$

$$z^{\alpha\beta,kr} \leq \sum_{i:(0i) \in \diamondsuit^{E\varnothing} \cup \diamondsuit^{EF}} x_{0i}^{\alpha\beta} \qquad \forall k, \alpha \in K, r, \beta \in R \tag{16}$$

$$z^{\alpha\beta,kr} \leq \sum_{i:(i0) \in \diamondsuit^{E\varnothing}} x_{i0}^{kr} \qquad \forall k, \alpha \in K, r, \beta \in R \tag{17}$$

$$t_{end}^{kr} - t_0^{\alpha\beta} \leq M(1 - z^{\alpha\beta,kr}) \qquad \forall k, \alpha \in K, r, \beta \in R \tag{18}$$

$$x_{0i}^{kr} \leq E - \sum_{\alpha \in K} \sum_{\beta \in R} y^{kr,\alpha\beta} + \sum_{\alpha \in K} \sum_{\beta \in R} z^{kr,\alpha\beta} + (1 - x_{0i}^{kr}) * M \qquad \forall i \in \diamondsuit^{E\varnothing} \cup \diamondsuit^{EF}, k \in K, r \in R \tag{19}$$

The objective function (1) minimizes the total routing cost. Constraints from (2) to (4) are the classical VRP constraints. Inequalities (5) compute the departure of a truck at a certain node, whereas (6) the end of the trip. With (7), (8) and (9) we impose that the departure time of a truck from a shipper is within its time window. (10) imposes that a trip must start after the previous has ended. With (11) the end of a trip is after its start. (12) defines variables $y^{\alpha\beta,kr}$, whereas (13) and (14) avoid wrong values in case of equal departure times. Inequalities (15) define variable $z^{\alpha\beta,kr}$. This takes value 1 if trip r of truck k ends before trip β of truck α and if "$\alpha\beta$" reduces the inventory of the inland terminal of 1 empty container and, finally, if "kr" increases the inventory by bringing a container from $\diamondsuit^{FE}$. From constraints (16) to (18) we prevent $z^{\alpha\beta,kr}$ to take value 1 when not necessary. Finally, (19) imposes that the net amount of empty containers E is not exceeded.

3 Solution framework

For the proposed problem, we develop a tailored column-and-row generation approach. The row generation is required since the availability of empty containers is dynamic and trucks can affect each other routes. This algorithm has been proven to be quite effective in a number of papers, see Maher (2016) and Li & Jia (2019). Therefore, our approach consists in dynamically adding rows (i.e., constraints) to the master problem, related to critical events occurring in the added paths, i.e, removal of containers from the dry-port; where a path is defined as a sequence of trips.

Like every classical column generation approach, the algorithm solves first the relaxed master problem. The pricing problem can be treated as an Elementary Shortest Path Problem with Resource Constraints, which is NP-hard in the strong sense. To solve it, we develop an exact algorithm based on the Pulse algorithm, proposed by Lozano *et al.* (2016). To get integer solutions, we embed the column-and-row generation in a branch-and-price framework. Hence, we develop a classical branching scheme based on arcs, and apply column generation at each node.

4 Preliminary numerical tests

We chose the classical Solomon's instances for the Vehicle Routing Problem with Time Windows with 25 nodes. We selected C1 type of instances for a total of 9 instances. These instances have clustered customers and narrow time windows. While the given coordinates for the depot (i.e., dry-port in our setting) are (40,50), the coordinates of the empty depot were set to (0,0), and the sea terminal ones to (70,80). These coordinates were chosen to penalize the need to reach the empty depot. The customers' types were generated randomly, as well as the origin and destination of the cargo, be it the dry-port or

Table 1 – *Results on 25 nodes adapted Solomon's instances. UB and LB stand respectively for upper bound and lower bound for the minimization problem. Timings are expressed in seconds.*

	CPLEX MILP				B&P			
Instance	UB	LB	Gap	Time	UB	LB	Gap	Time
C101	1698	1494	0.12	5310	1698	1698	0	307
C102	1118	1118	0.00	146	1118	1118	0	0.57
C103	1120	1120	0.00	50.1	1120	1120	0	5.06
C104	767	767	0.00	662	767	767	0	24.91
C105	1869	1361	0.27	2348	1869	1869	0	1.46
C106	1497	1375	0.08	419	1497	1497	0	4.22
C107	1264	1264	0.00	28.27	1264	1264	0	0.83
C108	938	938	0.00	18.32	938	938	0	4.2
C109	968	968	0.00	27.86	968	968	0	4.01

the sea port. In Table 1, we report the results of the experimentation. We compare the performances of the MILP formulation (CPLEX MILP) and our algorithm, referred to as B&P.

From the table, we can appreciate the good performance of both methods, though the B&P outperforms CPLEX in terms of both the overall speed and quality of the solution. All nine instances are solved to optimality with B&P, whereas CPLEX struggles for three of them.

5 Conclusions

In this paper, we have studied a particular drayage problem which is typical in several inland container supply chains. The multitrip component and the presence of constraints to avoid stockout have not received much attention in the drayage literature. We have modelled the problem with a MILP formulation and proposed an exact approach based on column-and-row generation. The first experiments on classical instances with 25 nodes are promising and the exact method outperformed CPLEX both in terms of quality and speed. For future research, we expect to further refine the algorithm and process even larger instances. Finally, if real-world data will become available, it would be interesting to assess the impact of empty container re-positioning in this context and to assess the goodness of current planning practices.

References

Caris, An, & Janssens, Gerrit K. 2009. A local search heuristic for the pre-and end-haulage of intermodal container terminals. *Computers & Operations Research*, **36**(10), 2763–2772.

Fazi, Stefano, Fransoo, Jan C, Van Woensel, Tom, & Dong, Jing-Xin. 2020. A variant of the split vehicle routing problem with simultaneous deliveries and pickups for inland container shipping in dry-port based systems. *Transportation Research Part E: Logistics and Transportation Review*, **142**, 102057.

Imai, Akio, Sasaki, Kazuya, Nishimura, Etsuko, & Papadimitriou, Stratos. 2006. Multi-objective simultaneous stowage and load planning for a container ship with container rehandle in yard stacks. *European Journal of Operational Research*, **171**(2), 373–389.

Li, Shuqin, & Jia, Shuai. 2019. The seaport traffic scheduling problem: Formulations and a column-row generation algorithm. *Transportation Research Part B: Methodological*, **128**, 158–184.

Lozano, Leonardo, Duque, Daniel, & Medaglia, Andrés L. 2016. An exact algorithm for the elementary shortest path problem with resource constraints. *Transportation Science*, **50**(1), 348–357.

Maher, Stephen J. 2016. Solving the integrated airline recovery problem using column-and-row generation. *Transportation Science*, **50**(1), 216–239.

Zhang, Ruiyou, Yun, Won Young, & Moon, Il Kyeong. 2011. Modeling and optimization of a container drayage problem with resource constraints. *International Journal of Production Economics*, **133**(1), 351–359.

Zhang, Ruiyou, Huang, Chao, & Wang, Junwei. 2020. A novel mathematical model and a large neighborhood search algorithm for container drayage operations with multi-resource constraints. *Computers & Industrial Engineering*, **139**, 106143.