Operations Research Letters 38 (2010) 405–407

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

A note on "Efficient feasibility testing for dial-a-ride problems"

Jiafu Tang [a,*], Yuan Kong [a], Henry Lau [b], Andrew W.H. Ip [b]

[a] Department of Systems Engineering, Key Laboratory of Integrated Automation of Process Industry of MOE, Northeastern University, Shenyang 110004, China

[b] Department of Industrial & Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 16 April 2009
Accepted 26 April 2010
Available online 12 May 2010

Keywords:
Dial-a-ride problems
Feasibility
Ride time constraints

ABSTRACT

Hunsaker and Savelsbergh [B. Hunsaker, M.W.P. Savelsbergh, Efficient feasibility testing for dial-a-ride problems, Operations Research Letters 30 (2002) 169–173.] developed a linear-time algorithm to verify the feasibility for dial-a-ride problems. However, this algorithm may incorrectly declare infeasibility due to ride time constraints in some cases. We propose a revised procedure to address this flaw, but in an $O(n^2)$ worst-case time.

1. Introduction

In a dial-a-ride problem (DARP), there are a number of requests that an item be picked up from a specific location (*pickup* point) and delivered to a specific destination (*delivery* point). A set of routes and schedules must be determined to accommodate the requests towards a minimum total distance under some service-level constraints, such as maximum waiting time constraints and maximum ride time constraints, where the former one gives the maximum allowable waiting time at a stop before departure, and the latter one shows the maximum allowable time between pickup and delivery. The DARP is different from and more complicated than other routing and scheduling problems when constructing a high quality feasible schedule. Thus, how to determine whether there is a feasible schedule for a given route becomes a critical issue.

The DARP problem described in [1] is considered in this study. Let I denote the set of requests and $|I| = n$; each request $i \in I$ has a pickup location i^+ with time window $[e_{i^+}, l_{i^+}]$, a delivery location i^- with time window $[e_{i^-}, l_{i^-}]$ and size d_i. Let $J = \{i^+, i^- : i \in I\} \cup \{0\}$ be the set of all stops, where the symbol 0 represents the initial location. The route starts from a central facility with location 0 in the time interval $[e_0, l_0]$. Let $t_{i^+i^-}$ be the direct travel time to drive from i^+ to i^-. The capacity of a vehicle is denoted as Q, and the maximum waiting time at any stop is W. The maximum ride time from pickup to delivery of request i is $\alpha t_{i^+i^-}$, where $\alpha > 1$ is a constant value. Given a route $R = (0, 1, \ldots, 2n)$ with a sequence of $2n + 1$ stops of J, the arrival and departure time at stop j are denoted as A_j and D_j, respectively.

In the presence of maximum waiting time and maximum ride time constraints, Hunsaker and Savelsbergh [1] proposed a three-pass algorithm to determine whether a feasible schedule for a given route exists in $O(n)$ time. In the Pass 1, capacity, time windows and maximum waiting time constraints are checked in a forward manner. The Pass 2 goes through the stops backwards to spread waiting time and repair ride time. The Pass 3 checks the validity of the attempt at repairing ride time is valid in the forward direction. During these three passes, they claimed that there is no feasible solution that has an arrival or departure time earlier than the computed arrival (or departure) time in the current solution. If the final pass finishes without declaring infeasibility, a feasible schedule with the computed arrival and departure time is constructed, which gives the earliest feasible schedule for the fixed route.

The Pass 1 accounts for capacity and scheduling constraints of time windows and maximum waiting time constraints through the route R in a forward direction. It starts with $D_0 = e_0$, and iteratively computes arrivals $A_{j+1} = D_j + t_{j,j+1}$ and departures $D_{j+1} = \max\{A_{j+1}, e_{j+1}\}$. If it finishes without declaring infeasibility, it determines the values of A_j and D_j at each stop j.

The Pass 2 starts with the final arrival time A_{2n}, which has been determined in Pass 1, and goes backwards to spread waiting time with Eqs. (1) and (2) for each stop j as follows:

$$D_j = A_{j+1} - t_{j,j+1} \tag{1}$$

$$A_j = \max\{A_j, D_j - W\}. \tag{2}$$

When the stop j becomes a pickup point i^+, it tries to repair departure time to a later time using following rules:

$$\text{if} \quad \Delta \equiv (D_{i^-} - D_{i^+}) - \alpha t_{i^+i^-} > 0 \tag{3}$$

$$\text{then} \quad D_{i^+} = D_{i^+} + \Delta \tag{4}$$

where $D_{i^-} - D_{i^+}$ is the ride time for the request i as defined in [1]. Of course, it is more rational to consider ride time as $A_{i^-} - D_{i^+}$,

* Corresponding author.
E-mail address: jftang@mail.neu.edu.cn (J. Tang).

0167-6377/$ – see front matter
doi:10.1016/j.orl.2010.05.002

Table 1
The parameters of counterexample.

j	(x_j, y_j) (km)	$[e_j, l_j]$ (min)	d_j (kg)	j	(x_j, y_j) (km)	$[e_j, l_j]$ (min)	d_j (kg)
0	(−4,3)	[0, 1440]	0				
1^+	(0, 0)	[305, 345]	5	1^-	(8, 0)	[310, 350]	−5
2^+	(4, 3)	[320, 360]	5	2^-	(4,−3)	[335, 375]	−5

when $A_{i^-} \geq e_{i^-}$, however it seems hard to handle the wait time when $A_{i^-} < e_{i^-}$.

Thereafter, to check whether there is enough waiting time to shift D_{i^+} according to Eq. (4), it also keeps track of the total waiting time $\hat{W}$ from the stop j to the end of the route. This value is initialized as $\hat{W} = 0$ at stop $2n$, and updated as follows:

$$\hat{W} = \hat{W} + (D_j - A_j).$$

The Pass 3 starts with D_0, and then updates $A_j = D_{j-1} + t_{j-1,j}$ and $D_j = \max\{A_j, D_j\}$ in a forward direction. The value of Δ is computed when arriving at a delivery point i^-. If $\Delta > 0$, the route is infeasible since the ride time constraint for request i is not satisfied.

There are two implicit but unwarranted assumptions considered in [1]. The first one is that there is no waiting time at stop $2n$. However, there is waiting time at stop $2n$ when $D_{2n} > A_{2n}$. In this case, the total waiting time $\hat{W}$ should be initialized as $\hat{W} = D_{2n} - A_{2n}$ rather than $\hat{W} = 0$ in the Pass 2. The second one is that the predetermined D_{i^-} is still unchanged when Hunsaker and Savelsbergh compute Δ in the Pass 2. However, the predetermined D_{i^-} may have been changed to a later time. Therefore, they have a flaw in checking ride time constraints. Under some circumstances, the algorithm may incorrectly declare a route infeasible. This short note focuses on addressing this flaw based on relevant analysis as follow.

2. A flaw in Hunsaker and Savelsbergh's algorithm

For the convenience of analysis, we introduce some additional notations. Let $jR = (j, \ldots, 2n)$ denote the sub-route from stop j to the end of the route, and A_j^r and D_j^r be the arrival and departure time at stop j determined in Pass r ($r = 1, 2, 3$), respectively.

Note that there are two things to do in Pass 2: spreading waiting time and repairing ride time. Remember that waiting time is spread with Eqs. (1) and (2) and that ride time is repaired by Eq. (3) and (4). The value of D_j^2 is determined by Eqs. (1) and (4), and it therefore may change D_k^2 ($\forall k \in (j+1)R$) in the backward pass. Given this possibility, analysis on these two equations is given as follows. Eq. (1) depends on the constant $t_{j,j+1}$ (the required travel time from stop j to stop $j+1$) and thus maintains the validity of D_k^2 ($\forall k \in (j+1)R$). However, the value of Δ in Eq. (4) only shows the ride time relation from pickup to delivery of a request when the stop j becomes a pickup point, i.e., $j = i^+$. Considering travel times required to visit stop k ($\forall k \in (j+1)R$), there might not be enough room to postpone D_j^2, and some of the predetermined D_k^2 ($\forall k \in (j+1)R$) may be changed to a later time. Therefore, when computing Δ in the backward pass, it is possible that pickups between $succ(i^+)$ and $pred(i^-)$ have changed the predetermined time $D_{i^-}^2$ to a later time.

Consider a route segment $U = (i^+, \ldots, h^+, \ldots, i^-, \ldots, h^-)$ with pickup stops i^+ and h^+, and delivery points i^- and h^-. Let $T_{u,v} = \sum_{j=u}^{v-1} t_{j,j+1}$ be the total travel time from stop u to stop v along route segment U. If $D_{h^+}^2$ is adjusted according to Eq. (4), such that $D_{h^+}^2 + T_{h^+i^-} > D_{i^-}^2$, then $D_{i^-}^2$ becomes unfeasible and should be updated to $D_{h^+}^2 + T_{h^+i^-}$ although it is still a lower bound on a feasible departure time. In addition, if $D_{h^+}^2 + T_{h^+i^-}$ is the departure time at i^-, the earliest departure time at pickup i^+ that satisfies the maximum ride time limit, $\alpha t_{i^+i^-}$, would be $D_{h^+}^2 + T_{h^+i^-} - \alpha t_{i^+i^-}$, denoted as $\overline{D_{i^+}}$. In the case that $D_{h^+}^2 + T_{h^+i^-} > D_{i^-}^2$, $D_{i^+}^3 < \overline{D_{i^+}}$ and $\overline{D_{i^+}} + T_{i^+h^+} \leq D_{h^+}^2$, by Pass 3 in [1], the following condition would be met.

$$A_{h^+}^3 = D_{i^+}^3 + T_{i^+,h^+} < \overline{D_{i^+}} + T_{i^+,h^+} \leq D_{h^+}^2$$

$$D_{h^+}^3 = \max\{A_{h^+}^3, D_{h^+}^2\} = D_{h^+}^2;$$

$$A_{i^-}^3 = D_{h^+}^3 + T_{h^+i^-} = D_{h^+}^2 + T_{h^+i^-};$$

$$D_{i^-}^3 = \max\{A_{i^-}^3, D_{i^-}^2\} = \max\{D_{h^+}^2 + T_{h^+i^-}, D_{i^-}^2\} = D_{h^+}^2 + T_{h^+i^-};$$

and

$$\Delta = D_{i^-}^3 - D_{i^+}^3 > D_{h^+}^2 + T_{h^+i^-} - \overline{D_{i^+}} = \alpha t_{i^+i^-}.$$

Thus, the route is infeasible. In this case, however, the route has a feasible schedule when we change $D_{i^+}^3$ to $\overline{D_{i^+}}$. There are two reasons that there is enough waiting time between i^+ and i^-, since

$$\overline{D_{i^+}} + T_{i^+h^+} + T_{h^+i^-} \leq D_{h^+}^2 + T_{h^+i^-} = D_{i^-}^3;$$

and the ride time limit can also be satisfied, since

$$D_{i^-}^3 - \overline{D_{i^+}} = D_{h^+}^2 + T_{h^+i^-} - (D_{h^+}^2 + T_{h^+i^-} - \alpha t_{i^+i^-}) = \alpha t_{i^+i^-}.$$

Therefore, in the case that $D_{h^+}^2 + T_{h^+i^-} > D_{i^-}^2$, $D_{i^+}^3 < \overline{D_{i^+}}$ and $\overline{D_{i^+}} + T_{i^+h^+} \leq D_{h^+}^2$, the algorithm in [1] incorrectly declares infeasibility when checking ride time constraint of request i. The following, example is given to explain this case.

A counterexample

Consider a route $P = (0, 1^+, 2^+, 1^-, 2^-)$. Let (x_j, y_j) ($\forall j \in P$) denote the coordinate of stop. Table 1 shows the values of parameters of the example, where distance is measured in kilometers (km), time in minutes (min) and demand in kilograms (kg). The direct travel time between stops is assumed as the same value of Euclidean distance. We set $\alpha = 2$, $W = 20$ (min), and $Q = 20$ (kg). With the algorithm in [1], we find that the route terminates with a declaration of infeasibility when we check if there is enough waiting time ($\Delta > \hat{W}$) at pickup point 2^+ in Pass 2. In this case, $D_{2^+}^2 = 320$, $A_{2^-}^2 = 330$, $D_{2^-}^2 = 335$, $\Delta = 3$ and $\hat{W} = 0$. In fact, there is enough waiting time when we initialize $\hat{W} = D_{2^-}^2 - A_{2^-}^2 = 5$ rather than $\hat{W} = 0$. However, if we initialize $\hat{W} = D_{2^-}^2 - A_{2^-}^2$, the process also terminates with an infeasible departure schedule (304, 309, 323, 328, 335) at the final pass because the ride time limit of the first request is violated, $(328 - 309) - 2 * 8 > 0$. In this case, $D_{2^+}^2 = 323$, $D_{1^-}^2 = 325$, $D_{1^+}^3 = 309$ and $\overline{D_{1^+}} = 312$. Thus, $D_{2^+}^2 + t_{2^+1^-} > D_{1^-}^2$, $D_{1^+}^3 < \overline{D_{1^+}}$ and $\overline{D_{1^+}} + t_{1^+2^+} \leq D_{2^+}^2$. But there exist at least a feasible schedule, e.g. the departure schedule (307, 312, 323, 328, 335), when $D_{1^+}^3 = \overline{D_{1^+}} = 312$. Therefore, the algorithm in [1] makes a wrong declaration of infeasibility when checking ride time constraint of the first request.

3. Revised procedure

Based on the above analysis, two changes are required on the original Pass 2 of the algorithm [1]. The first one is to initialize $\hat{W} = D_{2n} - A_{2n}$ instead of $\hat{W} = 0$. The other is to update D_k ($\forall k \in (j+1)R$) whenever D_j ($\forall j \in R$, and $j = i^+$) is adjusted by Eq. (4).

The updating process can be given as follows. For $k = succ(i^+)$ to $2n$, set

$D_k = \max\{D_k, D_{k-1} + t_{k-1,k}\}$.

Combining the changes, the pass 2 can be replaced by the revised one as follows.

Procedure: revised pass 2

(a) *Set* $\hat{W} = D_{2n} - A_{2n}$ *and* $j = 2n - 1$.
(b) Set $D_j = A_{j+1} - t_{j,j+1}$ and $A_j = \max\{A_j, D_j - W\}$.
(c) If $j = i^+$:
  i. Set $\Delta = (D_{i^-} - D_{i^+}) - \alpha t_{i^+,i^-}$.
  ii. *If* $\Delta > 0$, *then an adjustment is required for drive time. If* $\Delta > \hat{W}$, *then the route is infeasible; stop. Otherwise, set* $D_j = D_j + \Delta$, $A_j = \max\{A_j, D_j - W\}$, $\hat{W} = \hat{W} - \Delta$, *and for* $k = succ(i^+)$ *to* $2n$, *set* $D_k = \max\{D_k, D_{k-1} + t_{k-1,k}\}$.
  iii. If $D_j > L_j$, then the route is infeasible. Stop.
(d) Set $\hat{W} = \hat{W} + (D_j - A_j)$.
(e) If $j > 0$, set $j = j - 1$ and return to (b)

where L_j is the latest possible departure time at stop j. In comparison with the original algorithm in [1], the changes are given in step 2(a) and step 2(c) ii.

Since each updating process takes linear time (as it involves a forward procedure from $succ(i^+)$ to $2n$) and there are n pickups in the route, this revised procedure increases the worst-case time to $O(n^2)$. In the future, how to develop an efficient algorithm with time complexity of $O(n)$ is a promising issue.

Acknowledgements

This research is financially supported by the Natural Science Foundation of China (NSFC 70625001, 70721001), the Fundamental Research Funds for the Central Universities of MOE (N090204001), Ph.D. program of MOE (20060145009), National Basic Research Program of China (2009CB320601). The authors are greatly indebted to the Editor, AE and referees for their constructive suggestions and patience during the revision.

References

[1] B. Hunsaker, M.W.P. Savelsbergh, Efficient feasibility testing for dial-a-ride problems, Operations Research Letters 30 (2002) 169–173.