Scheduling in Dial-A-Ride Problems

Maria Bresich

PhD-Seminar

September 12, 2023

ALGORITHMS AND COMPLEXITY GROUP

Outline

Scheduling Problem

- Problem Motivation
- Scheduling in DARP & E-ADARP

Solving Approaches

- Exact Approaches
- Heuristic Approaches
- Comparison

Problem Motivation

Scheduling (sub-)problems occur in routing problems where time is relevant:

- Vehicle Routing Problem with Time Windows (VRPTW)
- Pickup and Delivery Problem with Time Windows (PDPTW)
- Dial-A-Ride Problem (DARP)
- Electric Autonomous Dial-A-Ride Problem (E-ADARP)

Time windows: a time window $[e_i, l_i]$ for a location i determines

- the earliest time e_i when service can take place
- the latest time l_i when service can take place

Problem Motivation: Feasibility Testing

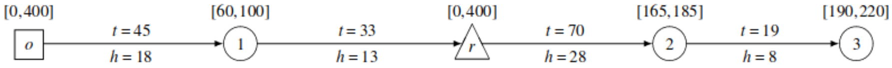

Figure: Route with time windows (taken from Desaulniers et al. (2016)).

Decision Problem: Given a route (a sequence of locations), does there exist a feasible schedule satisfying all time windows and maximum user ride time constraints?

- can be solved in linear time (Hunsaker and Savelsbergh (2002))
- does not optimize

Optimization Problem: Given a route, find a feasible schedule satisfying all time windows and maximum user ride time constraints and minimizing route duration.

- needed for route evaluation

Example: DARP & E-ADARP I

Cordeau and Laporte (2003); Bongiovanni et al. (2019)

Definition (Standard Dial-A-Ride Problem)

Given: n users with transportation requests from a pickup to a drop-off location, a fleet of m vehicles
Task: Design m vehicle routes serving all requests, s.t. the total routing cost is minimized and certain constraints are satisfied.

Definition (Static Electric Autonomous DARP)

Given: n users with transportation requests from a pickup to a drop-off location, a fleet of m electric autonomous vehicles
Task: Design m vehicle routes serving all requests, s.t. the total travel time and excess ride time of all users are minimized and certain constraints are satisfied.

Example: DARP & E-ADARP II

Cordeau and Laporte (2003); Bongiovanni et al. (2019)

Service related constraints:

- time windows for pickup and drop-off locations
- maximum user ride time

Scheduling:

- DARP: determine the departure time from the depot and the time at which service should begin at each location such that time windows and maximum user ride time constraints are satisfied and route duration is minimized
- e-ADARP: additionally determine time for partial recharging while also minimizing user excess ride time
  - $\rightarrow$ scheduling and battery management
- delays (sometimes) beneficial

Solving Approaches

Exact:

- linear programming (LP)
- labeling algorithm

Heuristic:

- suboptimal solutions possible
- often multiple steps
- often based on forward time slack (Savelsbergh (1992))
  - time span how far the service time of a location can be shifted forward in time (from the latest time l_i) without causing the route to become infeasible

Problem: incorrect infeasibility declarations

Exact Approaches - LP Formulations I Bongiovanni et al. (2023)

- check feasibility regarding time windows and maximum user ride time
- compute concrete time values (service times, waiting times)

LP1:

- directly minimizes user excess ride time
- computes service times
- inputs:
  - d_i = service duration at location i
  - $t_{i,j}$ = travel time from location i to location j
  - u_i = maximum ride time of user i
- decision variables: T_i = service start time at location i

$$\min \sum_{i \in \{1,\dots,n\}} (T_{D_i} - T_{P_i} - d_{P_i} - t_{P_i,D_i}) \tag{1}$$

$$\text{s.t.} \quad T_i + t_{i,i+1} + d_i \leq T_{i+1} \qquad \forall i \in \{1,2,\dots,\bar{M}-1\} \tag{2}$$

$$T_{D_i} - T_{P_i} - d_{P_i} \leq u_{P_i} \qquad \forall i \in \{1,\dots,n\} \tag{3}$$

$$e_i \leq T_i \leq l_i \qquad \forall i \in \{1,2,\dots,\bar{M}\} \tag{4}$$

Exact Approaches - LP Formulations II Bongiovanni et al. (2023)

LP2:

- minimizes user excess ride time by minimizing waiting times
- computes waiting times
- additional inputs:
  - L_i = sum of loads up to location i
  - t_i^{early} / t_i^{late} = earliest / latest service start time at location i
- decision variables: t_i^{wait} = waiting time at location i

$$\min \quad \sum_{i=1}^{\bar{M}} L_i t_i^{\text{wait}} \tag{5}$$

$$\text{s.t.} \quad \sum_{j=1}^{i} t_j^{\text{wait}} \geq t_i^{\text{early}} - \sum_{j=1}^{i-1} t_{j,j+1} - \sum_{j=1}^{i-1} d_j - t_1^{\text{early}} \qquad \forall i \in \{1, 2, \dots, \bar{M}\} \tag{6}$$

$$\sum_{j=1}^{i} t_j^{\text{wait}} \leq t_i^{\text{late}} - \sum_{j=1}^{i-1} t_{j,j+1} - \sum_{j=1}^{i-1} d_j - t_1^{\text{early}} \qquad \forall i \in \{1, 2, \dots, \bar{M}\} \tag{7}$$

$$\sum_{j=i+1}^{D_i} t_j^{\text{wait}} \leq u_i - \sum_{j=i}^{D_i-1} t_{j,j+1} - \sum_{j=i+1}^{D_i-1} d_j \qquad \forall i \in \mathcal{P} \tag{8}$$

Exact Approaches - LP Formulations III

Note: some inconsistencies and errors in paper (preprint)

- Ex.: LP1 does not consider loads in objective function but LP2 does!

$$\sum_{i \in \{1,\ldots,n\}} \left(T_{D_i} - T_{P_i} - d_{P_i} - t_{P_i,D_i} \right) \not\equiv \sum_{i=1}^{\bar{M}} L_i t_i^{\mathrm{wait}} \tag{9}$$

→ formulations are not equivalent
→ adjustments and corrections necessary before usage

Exact Approaches - LP Formulations IV Bongiovanni et al. (2023)

Extension for battery management:

- checks feasibility regarding battery constraints
- computes charging times
- additional inputs:
  - Q = vehicle battery capacity
  - r = minimum end battery level ratio
  - α_s = recharge rate at charging station s
  - $\beta_{i,j}$ = battery consumption between locations i, j
- additional decision variables:
  - B_i = battery level at location i
  - E_s = charging time at charging station s

Exact Approaches - LP Formulations IV
Bongiovanni et al. (2023)

Extension for battery management:

- checks feasibility regarding battery constraints
- computes charging times

LP1:

$$B_i = B_{\text{init}} \qquad i = 1 \tag{10}$$

$$B_{i+1} \leq B_i - \beta_{i,i+1} \qquad \forall i \in \{1, \dots, M-1\} \setminus \mathcal{S} \tag{11}$$

$$B_{i+1} \geq B_i - \beta_{i,i+1} \qquad \forall i \in \{1, \dots, M-1\} \setminus \mathcal{S} \tag{12}$$

$$B_{s+1} \leq B_s + \alpha_s E_s - \beta_{s,s+1} \qquad \forall s \in \mathcal{S} \tag{13}$$

$$B_{s+1} \geq B_s + \alpha_s E_s - \beta_{s,s+1} \qquad \forall s \in \mathcal{S} \tag{14}$$

$$Q \geq B_s + \alpha_s E_s \qquad \forall s \in \mathcal{S} \tag{15}$$

$$B_i \geq rQ \qquad i = M \tag{16}$$

$$E_s \leq T_{s+1} - t_{s,s+1} - T_s \qquad \forall s \in \mathcal{S} \tag{17}$$

$$E_s \geq T_{s+1} - t_{s,s+1} - T_s \qquad \forall s \in \mathcal{S} \tag{18}$$

$$B_i \geq 0 \qquad \forall i \in \mathcal{I} \tag{19}$$

$$E_s \geq 0 \qquad \forall s \in \mathcal{S} \tag{20}$$

Exact Approaches - LP Formulations IV Bongiovanni et al. (2023)

Extension for battery management:

- checks feasibility regarding battery constraints
- computes charging times

LP2:

$$
\begin{aligned}
&B_i = B_{\text{init}} && i = 1 && (10)\\
&B_{i+1} \leq B_i - \beta_{i,i+1} && \forall i \in \{1, \dots, M-1\} \setminus \mathcal{S} && (11)\\
&B_{i+1} \geq B_i - \beta_{i,i+1} && \forall i \in \{1, \dots, M-1\} \setminus \mathcal{S} && (12)\\
&B_{s+1} \leq B_s + \alpha_s E_s - \beta_{s,s+1} && \forall s \in \mathcal{S} && (13)\\
&B_{s+1} \geq B_s + \alpha_s E_s - \beta_{s,s+1} && \forall s \in \mathcal{S} && (14)\\
&Q \geq B_s + \alpha_s E_s && \forall s \in \mathcal{S} && (15)\\
&B_i \geq rQ && i = M && (16)\\
&E_s \leq t_{s+1}^{\text{wait}} && \forall s \in \mathcal{S} && (17)\\
&E_s \geq t_{s+1}^{\text{wait}} && \forall s \in \mathcal{S} && (18)\\
&B_i \geq 0 && \forall i \in \mathcal{I} && (19)\\
&E_s \geq 0 && \forall s \in \mathcal{S} && (20)
\end{aligned}
$$

Exact Approaches - Labeling Algorithm I Su et al. (2023)

- based on forward labeling algorithm for EVRPTW (Desaulniers et al. (2016))
- minimizes excess user ride time
- checks feasibility regarding time window and battery constraints
- linear time complexity
- computes bounds for time values
- does not compute concrete time values
  - $\rightarrow$ solves the decision problem
  - $\rightarrow$ charging times can be extracted
  - $\rightarrow$ other times have to be derived

Exact Approaches - Labeling Algorithm II Su et al. (2023)

Each node $i \in \mathcal{R}$ of a route $\mathcal{R}$ has a label with 4 resource attributes:

$$L_i := \{(T_i^{rch_s})_{s \in S}, T_i^{tMin}, T_i^{tMax}, T_i^{rtMax}\}$$

- $T_i^{rch_s}$: number of times charging station $s \in S$ is visited up to i
- T_i^{tMin}: earliest service start time at i assuming minimum recharges
- T_i^{tMax}: earliest service start time at i assuming maximum recharges
- T_i^{rtMax}: maximum charging time to fully recharge at i assuming minimum recharges

Initial label: $\{(0, \ldots, 0), 0, 0, 0\}$
Use resource extension functions (REFs) to compute the succeeding label L_j from the previous label L_i:

$$T_j^{rch_s} = T_i^{rch_s} + \begin{cases} 1, & \text{if } j = s \\ 0, & \text{otherwise.} \end{cases} \tag{21}$$

Exact Approaches - Labeling Algorithm III Su et al. (2023)

A route $\mathcal{R}$ is feasible if and only if $\forall j \in \mathcal{R}$, the label L_j satisfies:

$$T_j^{tMin} \leq l_j \tag{22}$$

$$T_j^{tMin} \leq T_j^{tMax} \tag{23}$$

$$T_j^{rch_s} \leq 1 \tag{24}$$

$$T_j^{rtMax} \leq \begin{cases} (1-\gamma)H, & \text{if } j \in F \\ H, & \text{otherwise.} \end{cases} \tag{25}$$

Challenge: incorrect infeasibility declarations

- Ex.: battery infeasibility declarations because possibility of charging directly before destination depots is disregarded

Heuristic Approaches I

Cordeau and Laporte (2003): 8-step scheduling procedure

- based on forward slack times
- does not minimize excess user ride time
- incorrect infeasibility declarations and suboptimal solutions possible

Heuristic Approaches I

Cordeau and Laporte (2003): 8-step scheduling procedure

- based on forward slack times
- does not minimize excess user ride time
- incorrect infeasibility declarations and suboptimal solutions possible

1. Set $D_0 := e_0$.
2. Compute A_i, W_i, B_i and D_i and for each vertex v_i in the route.
3. Compute F_0.
4. Set $D_0 := e_0 + \min\{F_0, \sum_{0<p<q} W_p\}$.
5. Update A_i, W_i, B_i and D_i for each vertex v_i in the route.
6. Compute L_i for each request assigned to the route.
7. For every vertex v_j that corresponds to the origin of a request j
   (a) Compute F_j.
   (b) Set $B_j := B_j + \min\{F_j, \sum_{j<p<q} W_p\}$; $D_j := B_j + d_j$.
   (c) Update A_i, W_i, B_i and D_i, for each vertex v_i that comes after v_j in the route.
   (d) Update the ride time L_i for each request i whose destination vertex is after vertex v_j.
8. Compute changes in violations of vehicle load, route duration, time window and ride time constraints.

Figure: 8-step scheduling procedure by Cordeau and Laporte (2003).

Heuristic Approaches I

Cordeau and Laporte (2003): 8-step scheduling procedure

- based on forward slack times
- does not minimize excess user ride time
- incorrect infeasibility declarations and suboptimal solutions possible

Parragh et al. (2009): modified 8-step scheduling procedure

- adapted computation of forward slack times
- minimizing excess user ride time
  - $\rightarrow$ increases solution quality
  - $\rightarrow$ more restrictive regarding feasibility

Heuristic Approaches II

Molenbruch et al. (2017): 4-step scheduling heuristic

- minimizing excess user ride time
- steps:
  1. backward loop: service time for pickup locations
  2. forward loop: service time drop-off locations
  3. forward loop: adjust service times regarding travel time feasibility
  4. multiple loops: adjust service times further for travel time feasibility
- fewer incorrect infeasibility declarations and suboptimal solutions

Heuristic Approaches III

Bongiovanni et al. (2023):

- scheduling heuristic
  - minimizing excess user ride time
  - based on definition of waiting times from LP2
  - suboptimal solutions possible
  - extended with recourse heuristic to recover feasibility in cases of incorrect infeasibility declarations
- battery management heuristic
  - recharge as much as possible as early as possible to ensure battery feasibility

Computational Results - Scheduling

Tests with E-ADARP instances:

- approaches by Bongiovanni et al. (2023):
  - heuristic: always optimal solutions
  - LP: only 2–4x slower than heuristic
- approach by Cordeau and Laporte (2003):
  - few incorrect infeasibility declarations
  - many suboptimal solutions
  - average deviations of up to 106%
- approach by Parragh et al. (2009):
  - high quality solutions
  - many incorrect infeasibility declarations
- approach by Molenbruch et al. (2017):
  - almost always optimal solutions
  - on average faster than Bongiovanni's algorithm

Discussion & Questions

Thank you!

References I

Claudia Bongiovanni, Mor Kaspi, and Nikolas Geroliminis. The electric autonomous dial-a-ride problem. *Transportation Research Part B: Methodological*, 122:436–456, April 2019.

Claudia Bongiovanni, Nikolas Geroliminis, and Mor Kaspi. A ride time-oriented scheduling algorithm for dial-a-ride problems. (arXiv:2211.07347), Apr 2023. arXiv:2211.07347 [cs, math].

Jean-François Cordeau and Gilbert Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B: Methodological*, 37(6):579–594, July 2003.

Guy Desaulniers, Fausto Errico, Stefan Irnich, and Michael Schneider. Exact algorithms for electric vehicle-routing problems with time windows. *Operations Research*, 64(6):1388–1405, Dec 2016.

Brady Hunsaker and Martin Savelsbergh. Efficient feasibility testing for dial-a-ride problems. *Operations Research Letters*, 30 (3):169—-173, Jun 2002.

Yves Molenbruch, Kris Braekers, An Caris, and Greet Vanden Berghe. Multi-directional local search for a bi-objective dial-a-ride problem in patient transportation. *Computers & Operations Research*, 77:58–71, Jan 2017.

Sophie N. Parragh, Karl F. Doerner, Richard F. Hartl, and Xavier Gandibleux. A heuristic two-phase solution approach for the multi-objective dial-a-ride problem. *Networks*, 54(4):227–242, 2009.

Martin Savelsbergh. The vehicle routing problem with time windows: Minimizing route duration. *INFORMS Journal on Computing*, 4:146—-154, May 1992.

Yue Su, Nicolas Dupin, and Jakob Puchinger. A deterministic annealing local search for the electric autonomous dial-a-ride problem. *European Journal of Operational Research*, 309(3):1091–1111, Sep 2023.