Dr. Niels Lindner (lindner@zib.de)
Pedro Maristany de las Casas (maristany@zib.de)

Lecture 13

January 20, 2020

6.2 Single-Depot Aperiodic Vehicle Scheduling

In an abstract formulation, the input for (single-depot) aperiodic vehicle scheduling consists of

- a finite set $\mathcal{T}$ of *trips*,
- an acyclic relation $\preceq$ on $\mathcal{T} \times \mathcal{T}$, i.e., for all chains $t_1 \preceq t_2 \preceq \cdots \preceq t_r$ with $r \geq 2$ holds $t_1 \neq t_r$.

For a pair $(t_1, t_2) \in \mathcal{T} \times \mathcal{T}$, $t_1 \preceq t_2$ should hold if and only if a vehicle can serve trip t_2 after having served t_1. For example, suppose that the set $\mathcal{T}$ of trips comes with a time information time information $\tau_{\text{dep}}, \tau_{\text{arr}} : \mathcal{T} \to \mathbb{R}$ indicating departure times at the first stop of a trip, and arrival times at the last stop, respectively. Then one may define

$$t_1 \preceq t_2 \quad :\Leftrightarrow \quad \tau_{\text{dep}}(t_2) - \tau_{\text{arr}}(t_1) \geq \tau_{\min}(t_1, t_2),$$

where $\tau_{\min}(t_1, t_2) > 0$ is a minimum turnaround time, e.g., the length of a deadhead trip from the last stop of t_2 to the first stop of t_1, or the minimum driver break duration.

Definition 1. *An* aperiodic vehicle schedule *is a collection* $S = \{s_1, \ldots, s_k\}$ *of chains*

$$s_i = t_{i,1} \preceq t_{i,2} \preceq \cdots \preceq t_{i,r_i}, \quad i = 1, \ldots, k,$$

such that each trip in $\mathcal{T}$ *occurs in exactly one chain in* S.

The number of vehicles *of an aperiodic vehicle schedule* $S = \{s_1, \ldots, s_k\}$ *is defined as* $\nu(S) := k$.

Definition 2. *Given* $(\mathcal{T}, \preceq)$ *as above, the* single-depot aperiodic vehicle scheduling problem *is to find an aperiodic vehicle schedule* S *minimizing* $\nu(S)$.

Remark 3. It is clear that an aperiodic vehicle schedule exists, e.g., the trivial schedule $S = \{t \mid t \in T\}$ with $\nu(S) = |\mathcal{T}|$.

Network flow model

We will now model the single-depot aperiodic vehicle scheduling problem as a minimum cost network flow problem: To this end, we build an event-activity network $\mathcal{N}(\mathcal{T}, \preceq)$ as follows:

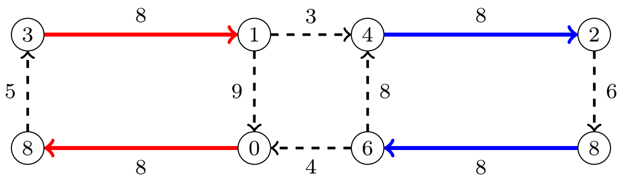
(1) Create two events p and q (*depot nodes*).

(2) For each $t \in \mathcal{T}$, add a *pull-out activity* (p, d_t), a *driving activity* (d_t, a_t) and a *pull-in activity* (a_t, q).

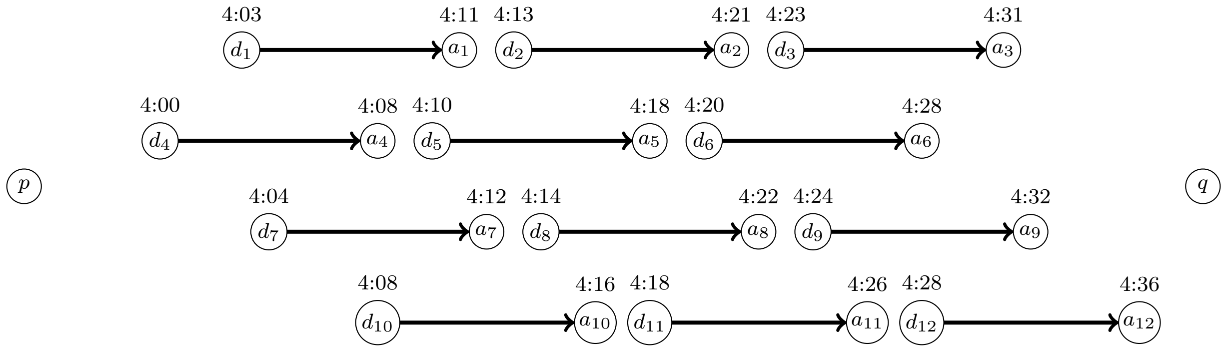
(3) For each $(t_1, t_2) \in \mathcal{T}$ with $t_1 \preceq t_2$, add a *turnaround activity* (a_{t_1}, d_{t_2}).

Example 4. Consider the following event-activity network $\mathcal{P}$ with periodic timetable for a period time of $T = 10$:

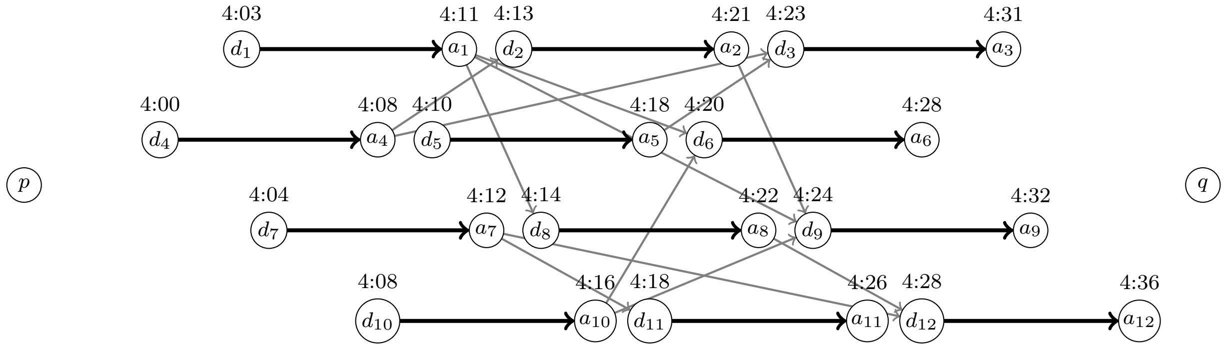
Dr. Niels Lindner (lindner@zib.de)
Pedro Maristany de las Casas (maristany@zib.de)



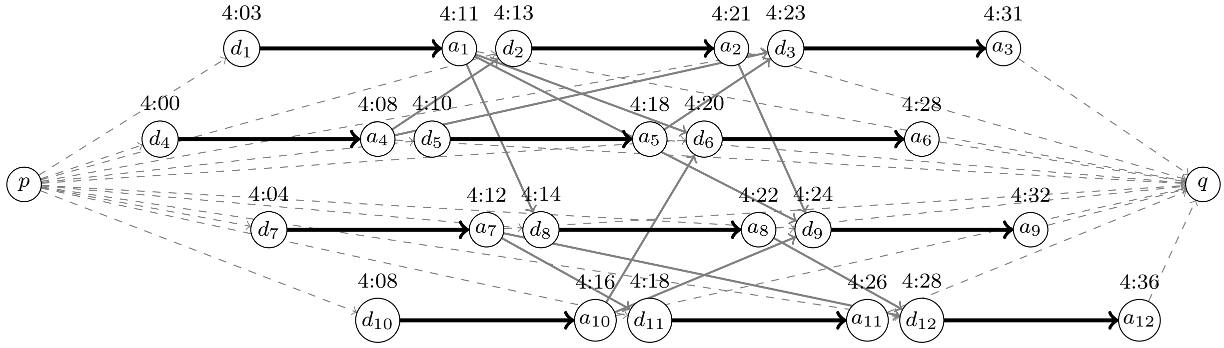
In total, 4 trips are operated every 10 minutes. Construct $\mathcal{T}$ as the set of all $12 = 3 \cdot 4$ trips starting between 4:00 (included) and 4:30 (excluded). Then the depot nodes and driving activities of the event-activity network $\mathcal{N}(\mathcal{T}, \preceq)$ are as follows:



As relation $\preceq$, we allow a turnaround between two trips t_1 and t_2 in $\mathcal{T}$ with corresponding driving activities e_1 and e_2 in $\mathcal{P}$ if and only if the departure of t_2 is not earlier than the arrival of t_1 and there is a turnaround activity in $\mathcal{P}$ from the target of e_1 to the source of e_2. For example, the trip 4:03→4:11 is connected to 4:14→4:22 and 4:24→4:32, but not to 4:04→4:12 or 4:18→4:26. Adding the turnaround activites, $\mathcal{N}(\mathcal{T}, \preceq)$ looks as follows:



Finally, we introduce the pull-out and pull-in activities. They model driving a vehicle from a depot to the first trip, and from the last trip back to the depot.



Dr. Niels Lindner (lindner@zib.de)
Pedro Maristany de las Casas (maristany@zib.de)

Remark 5. As $\preceq$ is acyclic, the network $\mathcal{N}(\mathcal{T}, \preceq)$ is acyclic, i.e., it contains no directed circuits.

Theorem 6. *Given $(\mathcal{T}, \preceq)$ as above, the single-depot aperiodic vehicle scheduling problem is solved by finding a minimum value p-q-flow on $\mathcal{N}(\mathcal{T}, \preceq)$ covering each driving activity exactly once.*

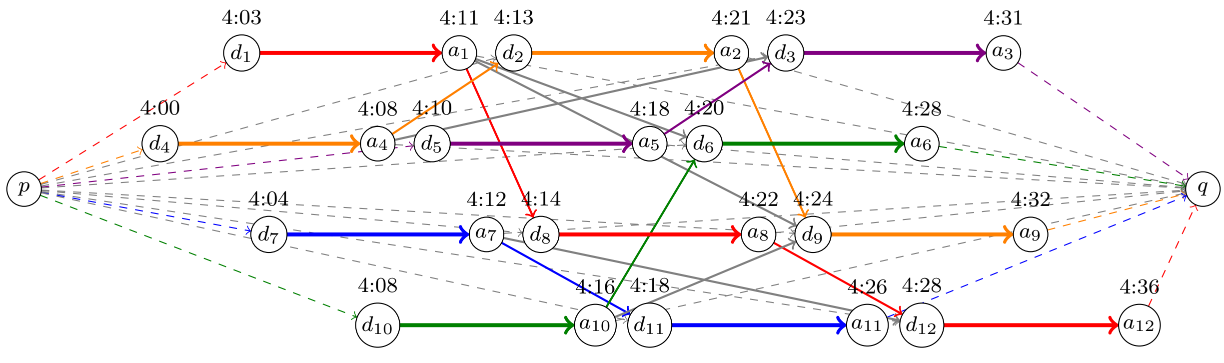
Proof. Since any driving activity is covered exactly once and $\mathcal{N}(\mathcal{T}, \preceq)$ is acyclic, no feasible p-q-flow contains a circulation and decomposes into edge-disjoint p-q-paths. Consequently, there is a one-to-one correspondence between vehicle schedules for $(\mathcal{T}, \preceq)$ and feasible p-q-flows in $\mathcal{N}(\mathcal{T}, \preceq)$, where a chain corresponds to a p-q-path. The number of vehicles equals the number of p-q-paths in the flow and can be measured by the total outflow at p, i.e., the value of the flow. $\square$

In particular, the single-depot aperiodic vehicle scheduling problem can be solved by the following integer program on $\mathcal{N}(\mathcal{T}, \preceq) = (V, E)$ with driving activities $E_d \subseteq E$:

$$\begin{aligned}
&\text{Minimize} && \sum_{e \in \delta^+(p)} f_e && \\
&\text{s.t.} && \sum_{e \in \delta^+(v)} f_e - \sum_{e \in \delta^-(v)} f_e = 0, && v \in V \setminus \{p, q\}, \\
& && f_e = 1, && e \in E_d, \\
& && f_e \in \{0, 1\}, && e \in E \setminus E_d.
\end{aligned}$$

Remark 7. This is a standard network flow problem, so the constraint matrix of the above integer program is totally unimodular. This means that the LP relaxation, i.e., relaxing to $f_e \in [0, 1]$, is in fact integral.

Example 8. For the above example, this is a minimum value p-q-flow:



The flow decomposes into 5 p-q-paths corresponding to 5 trip chains. Note that we already considered this example in the context of periodic vehicle scheduling, where we also found 5 as the minimum number of vehicles.

Matching interpretation

As in periodic vehicle scheduling, there is a matching view on aperiodic vehicle scheduling.

Lemma 9. *Consider* $\mathcal{N}(\mathcal{T}, \preceq) = (V, E)$ *with driving activities* E_d *and turnaround activities* E_t*. Then the following numbers are equal:*

(1) *The minimum value of a* p*-*q*-flow covering each* $e \in E_d$ *exactly once.*

(2) $|E_d| - |M|$*, where* M *is a maximum cardinality matching in the subnetwork* (V, E_t)*.*

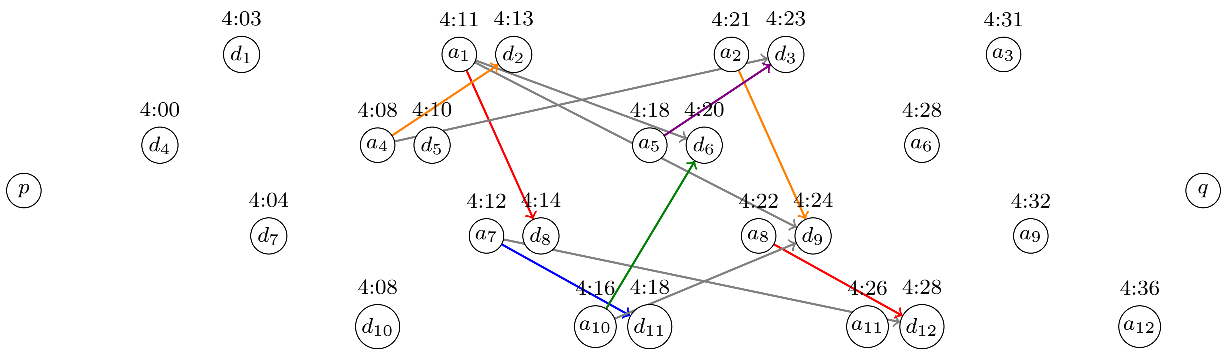
Proof. (1) $\geq$ (2): Let f be an optimal feasible p-q-flow with value ν. Then f decomposes into ν paths which are edge-disjoint, and even pairwise vertex-disjoint outside of p and q. Any such path uses activities in the following pattern:

pull-out $\to$ driving $\to$ turnaround $\to$ driving $\to$ turnaround $\to \ldots \to$ driving $\to$ pull-in.

A path using r driving activities hence contains $r - 1$ turnaround activities. The flow f covers all driving activities exactly once, so it contains $|E_d|$ driving activities and $|E_d| - \nu$ turnaround activities. By the structure of $\mathcal{N}(\mathcal{T}, \preceq)$, restricting f to (V, E_t) is a matching. If M is a maximum cardinality matching in (V, E_t), we obtain hence $|M| \geq |E_d| - \nu$, i.e., $\nu \geq |E_d| - |M|$.

(2) $\geq$ (1): Let M be a maximum cardinality matching in (V, E_t). Consider the p-q-flow f' obtained by the paths (p, d_t, a_t, q) for all trips $t \in T$, i.e., the flow corresponding to the trivial schedule. This flow has value $|E_d|$. Pick an edge $(a_{t_1}, d_{t_2}) \in M$ and replace the two paths (p, d_{t_1}, a_{t_1}, q) and (p, d_{t_2}, a_{t_2}, q) by the single path $(p, d_{t_1}, a_{t_1}, d_{t_2}, a_{t_2}, q)$, so that the value of f' reduces by 1. This way, continue by replacing the two paths containing two matched trips by a single path for each edge of M. The resulting p-q-flow is feasible and has value $|E_d| - |M|$. Consequently, $|E_d| - |M| \geq \nu$. □

Example 10. In the running example, this is the maximum cardinality matching obtained from the above optimal p-q-flow:



In (V, E_t), only the 7 arrival vertices $a_1, a_2, a_4, a_5, a_7, a_8, a_9, a_{10}$ are non-isolated. All of these vertices are matched, so that we obtain even a perfect matching (of the non-isolated vertices). As there are 12 driving activities, the minimal number of vehicles equals $12 - 7 = 5$.

Summary: The single-depot vehicle scheduling problem can be solved by either computing a minimum value network flow covering all driving activities exactly once, or by finding a maximum cardinality matching of the subgraph given by the turnaround activities.

Dr. Niels Lindner (lindner@zib.de)
Pedro Maristany de las Casas (maristany@zib.de)

Comparison of periodic and aperiodic scheduling

Consider an event-activity network $\mathcal{P}$ with driving activities $E_d(\mathcal{P})$ and turnaround activities $E_t(\mathcal{P})$. Suppose we are given a periodic timetable π on $\mathcal{P}$ w.r.t. some period time T with corresponding activity durations $x \geq 0$. For an integer $n \in \mathbb{N}$, let $\mathcal{N}_n$ be the event-activity network modeling the aperiodic vehicle scheduling problem for all periodic trips starting between time 0 (included) and time $n \cdot T$ (excluded) as in the running example above. Formally, we set

$$\mathcal{T}_n := E_d(\mathcal{P}) \times \{0, 1, \dots, n-1\},$$

and for $t_1 = (v_1 w_1, i_1), t_2 = (v_2 w_2, i_2) \in \mathcal{T}_n$, we define

$$t_1 \preceq t_2 \quad :\Leftrightarrow \quad \pi_{v_2} + i_2 T \geq \pi_{v_1} + x_{v_1 w_1} + i_1 T \quad \text{and} \quad w_1 v_2 \in E_t(\mathcal{P}),$$

and let $\mathcal{N}_n := \mathcal{N}(\mathcal{T}_n, \preceq)$. Intuitively, we compare the "real aperiodic" departure time at v_2 of the periodic trip departing within the interval $[i_2 T, i_2 T + T)$ with the "real aperiodic" arrival time at w_1 of the periodic trip departing within $[i_1 T, i_1 T + T)$.

Lemma 11. *Let S_p be an optimal periodic vehicle schedule for $(\mathcal{P}, T, \pi, x)$ and let $S_{a,n}$ be an optimal aperiodic vehicle schedule for $(\mathcal{T}_n, \preceq)$. Then $\nu(S_{a,n}) \leq \nu(S_p)$.*

Exercise. Find an example where $\nu(S_{a,n}) < \nu(S_p)$. In particular, $M_{a,n}$ is not a maximum cardinality matching of $E_t(\mathcal{N}_n)$.