Appendix A. Fastening the insertion heuristic

Masson et al. (2013) proposed a *fast feasibility check* (FFC) procedure for interdependent routes. It extends the *forward slack time* procedure introduced by Savelsbergh (1992). In the VRPTR case, an additional modeling effort is required to take into account that all WRs depend on each other (as the same worker can be transported through multiple WRs).

Appendix A.1. Modeling: aggregated nodes

For the vehicle routes, only the entry and exit points of a WR are relevant (as all intermediate nodes are not visited by the vehicles). Accordingly, we introduce two aggregated nodes for a WR: one for the drop-off (at the beginning of the WR) and one for the pick-up (at the end of the WR). These nodes are visited by the vehicles and gather consolidated information about the WR (total duration and resulting time window).

Let $\Omega = \bigcup_{w \in W} \Omega_w$ be the set of all WRs, where Ω_w represents the ordered set of WRs performed by worker $w \in W$ in her/his schedule ($\kappa_{i+1} \in \Omega_w$ is directly performed after WR κ_i in the schedule of worker w, $\forall i < |\Omega_w|$). $w(\kappa)$ is the worker associated with κ, and $i(\kappa)$ is the position of WR κ in the worker's planning. Furthermore, $\rho(\kappa)$ (resp. $\sigma(\kappa)$) represents the predecessor (resp. successor) of κ in the corresponding worker planning. $\rho(\kappa)$ (resp. $\sigma(\kappa)$) is $\emptyset$ if κ is the first (resp. the last) WR done. $[e_\kappa, l_\kappa]$ and p_κ denotes the time window of κ (to serve all jobs of κ on time) and the total processing time (i.e., all processing times, walking times, and waiting times along κ).

$\mathcal{D}$ (resp. $\mathcal{P}$) is the set of all drop-off (resp. pick-up) points. $D_\kappa \in \mathcal{D}$ (resp. $P_\kappa \in \mathcal{P}$) is the drop-off (resp. pick-up) point of κ. $O_W \subset \mathcal{P}$ (resp. $O'_W \subset \mathcal{D}$) represents the set of worker pick-up (resp. drop-off) points at the depot. Moreover, O_K (resp. O'_K) denotes the set of the first (resp. the last) nodes visited by the vehicles. Finally, $\mathcal{M} = \mathcal{P} \cup \mathcal{D} \cup O_K \cup O'_K$ denotes the set of all aggregated nodes for a given solution.

A transportation request arises between the end of a WR κ and the beginning of the next WR $\sigma(\kappa)$, denoted by $(P_\kappa, D_{\sigma(\kappa)})$. Furthermore, transportation requests are required between the pick-up at the depot and the drop-off at the beginning of the first WR as well as between the pick-up in the last WR (of any worker's planning) and the final drop-off at the depot: (P_{O_w}, D_{κ_1}) (resp. $(P_{\kappa_{|\omega_w|}}, D_{0'_w}))$ for the transportation between the depot (resp. last WR $\kappa_{|\omega_w|}$) and the first WR κ_1 (resp. the depot).

For each $v \in \mathcal{M}$, $k(v)$ is the route that visits v, and $i(v)$ is the position of v in the route. $\rho(v)$ (resp. $\sigma(v)$) denotes the predecessor (resp. successor) of v in the route. Finally, for each $v \in \mathcal{D} \cup \mathcal{P}$, $\kappa(v)$ denotes the WR that contains v. Workers' pick-up and drop-off times are set to be null. Each $v \in \mathcal{M}$ can be characterized by an associated time window $[e_v, l_v]$, which corresponds to the time a car must drop off a worker to have an on-time arrival for the jobs composing $\kappa(v)$. For each $v \in \mathcal{D}$, we have $e_v = e_{\kappa(v)}$ and $l_v = l_{\kappa(v)}$, whereas for each $v \in \mathcal{P}$, we have $l_v = \infty$, and e_v depends on the drop-off time at $D(v)$.

Appendix A.2. Vehicle constraints

A vehicle route is an ordered set of aggregated nodes that must satisfy the following constraints, where $D(v)_{v \in \mathcal{P} \cup O_W}$ (resp. $P(v)_{v \in \mathcal{D} \cup 0'_W}$) designates the drop-off (resp. pick-up) in the pick-up and drop-off couple. More precisely, $D(v) = P_{\rho(\kappa(v))}, \forall v \in \mathcal{P}$, and $D(v) = P_{0,w(\kappa(v))}, \forall v \in O_W$. $\mathbb{1}_{i=P} = 1$ (resp. $\mathbb{1}_{i=D} = 1$) if i is a pick-up (resp. drop-off) aggregated node, and 0 otherwise. Constraints (A.1) ensure that the nodes of a pick-up and drop-off couple are managed by the same vehicle, and the pick-up must occur before the drop-off. Constraints (A.2) ensure that a vehicle cannot move without its associated driver by scheduling the driver's pick-up directly after her/his drop-off in the vehicle route. Constraints (A.3) ensure that the vehicle capacity q is never exceeded. A set of routes is feasible if the above constraints are satisfied and if it fulfills the temporal constraints that are detailed in the next subsection.

$$k(v) = k(D(v)) \text{ and } i(v) < i(D(v)), \qquad \forall v \in O_W \cup \mathcal{P} \tag{A.1}$$

$$k(D_{\kappa(v)}) = k(P_{\kappa(v)}) \text{ and } i(P_{\kappa(v)}) = i(D_{\kappa(v)}) + 1, \qquad \forall v \in \mathcal{D}/\ w(\kappa(v)) \text{ is a driver} \tag{A.2}$$

$$\sum_{v \in k} (\mathbb{1}_{v=P} - \mathbb{1}_{v=D}) \leq q, \qquad \forall k \in K \tag{A.3}$$

Appendix A.3. Temporal constraints

A solution to the VRPTW is feasible if and only if each of its routes satisfies temporal feasibility. The temporal constraints are modeled using a *Simple Temporal Problem* (as described by Dechter et al. (1991)), for which efficient algorithms and representations exist in the literature. Temporal constraints are expressed as follows in Equations (A.4)–(A.6), where h_v represents the service time at the aggregated node $v \in \mathcal{M}$:

$$h_{\sigma(v)} \geq h_v + \tau^d_{v,\sigma(v)}, \qquad \forall v \in \mathcal{M} \backslash O'_K \tag{A.4}$$

$$h_{P_\kappa} \geq \max\{h_{D_\kappa}, e_{D_\kappa}\} + p_\kappa, \qquad \forall \kappa \in \Omega \tag{A.5}$$

$$h_v \leq l_v, \qquad \forall v \in \mathcal{M} \tag{A.6}$$

Equations (A.4) set the temporal constraints in a route, for which the arrival time at a node depends on the departure time at the previous node. Equations (A.5) specify the time at which a worker is available to be picked up after completing a WR. The time at which a worker starts working on a WR depends on both the drop-off time h_{D_κ} and on the time window e_{D_κ} of the WR. Finally, Equations (A.6) state that the service time cannot start after the end of the corresponding time window.

This set of equations can be modeled with a precedence graph, called G^p, where constraints of type $h_u - h_v \geq a_{uv}$ (a_{uv} is a real number) represent an arc from u to v with a cost of a_{uv}. Node o is introduced to represent the beginning of the planning horizon, and, for every drop-off point $D \in \mathcal{D}$, a virtual node $D^{(dup)}$ is introduced to get rid of the max function in Equations (A.5). $\mathcal{D}^{(dup)}$ is the set of duplicated nodes. Equations (A.4) to (A.6) can therefore be rewritten as follows:

$$h_{\sigma(v)} - h_v \geq \tau^d_{v,\sigma(v)}, \qquad \forall v \in \mathcal{M} \backslash O'_K \tag{A.7}$$

$$h_{P_\kappa} - h_{D^{(dup)}_\kappa} \geq p_\kappa, \qquad \forall \kappa \in \Omega \tag{A.8}$$

$$h_{D^{(dup)}_\kappa} - h_{D_\kappa} \geq 0, \qquad \forall \kappa \in \Omega \tag{A.9}$$

$$h_{D^{(dup)}_\kappa} - h_o \geq e_v, \qquad \forall \kappa \in \Omega \tag{A.10}$$

$$h_o - h_v \geq -l_v, \qquad \forall v \in \mathcal{M} \tag{A.11}$$

$$h_v \geq 0, \qquad \forall \kappa \in \Omega \tag{A.12}$$

$$h_o = 0 \tag{A.13}$$

Checking the feasibility of the VRPTR set of temporal constraints is equivalent to showing that there is no cycle of negative length in the precedence graph. This can be done using the so-called BFCT algorithm, which has a complexity of $\mathcal{O}(|\mathcal{M}| \times |A'|)$ (Cherkassky et al. 2009). For any solution satisfying the temporal constraints, the precedence graph is a direct acyclic graph.

Figure A.8 presents the precedence graph associated with Figure 1 using the above-introduced notation. In Figure 1, the solution with carpooling and walking contains three WRs, which can be denoted as $\Omega = \{\kappa_1 = \{j_1\}, \kappa_2 = \{j_2, j_3\}, \kappa_3 = \{j_4\}\}$. It involves six pick-up and drop-off couples denoted as $(P_{O_{w_1}}, D_{\kappa_1})$, $(P_{\kappa_1}, D_{0'_{w_1}})$, $(P_{O_{w_2}}, D_{\kappa_2})$, $(P_{\kappa_2}, D_{0'_{w_2}})$, $(P_{O_{w_3}}, D_{\kappa_3})$, and $(P_{\kappa_3}, D_{0'_{w_3}})$.

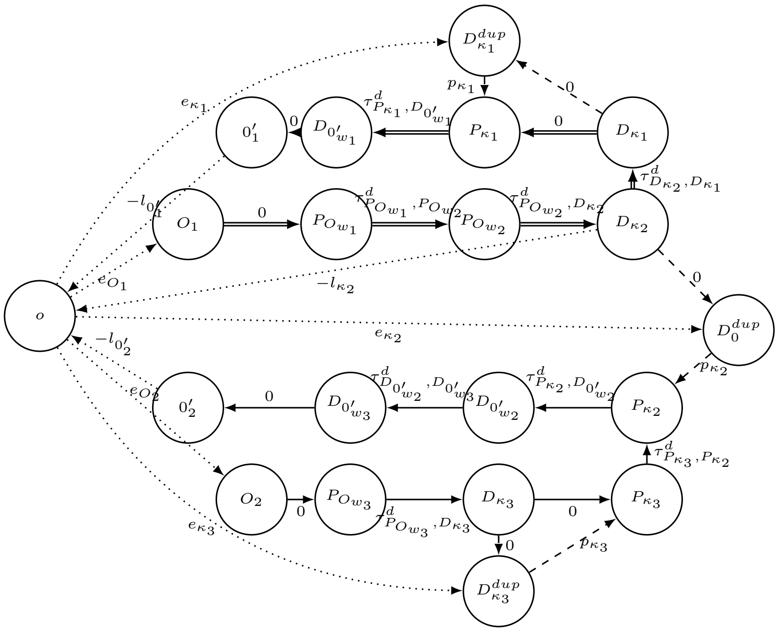

Figure A.8: Precedence graph representing the VRPTR solution of Figure 1. Dotted arcs represent time window constraints (for the sake of clarity, not all time window constraints are drawn), dashed arcs represent precedence constraints due to WRs, and both double and normal arcs represent precedence constraints due to the routes. The order of the nodes in the route must satisfy the constraints in Equations (A.1)–(A.3).

The FFC procedure pre-computes, for each aggregated node $v \in \mathcal{M}$, the earliest service time (h_v), the latest departure time (λ_v), and the matrix of waiting times between all aggregated nodes $((\Phi_{uv})_{u,v\in\mathcal{M}})$. As the precedence graph G^p represents a feasible solution, it does not contain any cycle of positive weight; therefore, the longest path is the shortest path in $-G^p$, where the arcs of $-G^p$ have the opposite weight of the arcs in G^p. The precedence graph is a direct acyclic graph, where the shortest paths can be computed in linear time. $(\Phi_{uv})_{u,v\in\mathcal{M}}$ is computed as the shortest paths in the precedence graph, where the arcs are weighted with the waiting time in the solution at the terminal node of the arc (i.e., the waiting time at node v is equal to $\max\{0, e_v - h_v\}$), and it can be computed in $\mathcal{O}(n^2)$. All these shortest paths are computed once, and then the $\mathcal{O}(n^4)$ insertion positions are tested in constant time.