IV
Mathematical Formulations

In this chapter, two different mathematical formulations for the VRPTWEOS are presented. These formulations were modeled over (discretized) time indexed graphs. At first, the notation used in those formulations is introduced. Then we present an assignment model which was used to solve the real case scenario described in section III.2. Finally, we describe a flow formulation, which we use later in a decomposition approach.

IV.1 Notation

This section introduces the notation used in this chapter to describe the mathematical formulations for the VRPTWEOS. We will indistinctively refer to the customers as *jobs* and to the vehicles as *mobile equipment* as convenient.

– $J = \{0, \dots, n\}$ is the set of jobs

– $E = \{1, \dots, k\}$ is the set of types of equipment

– $H = \{0, \dots, T\}$ is the set of time periods in planning horizon

– $M_e = \{1, \dots, k_e\}$ is the set of (identical) machines of type $e \in E$

– $E_j = \{e_{j1}, e_{j2}, \dots, e_{jk}\}$ is the set of equipment types required by job j

– $S_j = \{t_j^{init}, \dots, t_j^{end}\}$ denotes the time window of job j.

– k_e refers to the number of machines of type e available at the beginning of opperation.

– p_j denotes the service time of job $j \in J$

– s_{ij} corresponds to the transition time between job locations $i \in J$ and $j \in J$

IV.2 Assignment Formulation

This formulation was modeled over a bipartite graph $G = (U, V, A)$ where U and V are disjoint sets of vertices and A is the set of arcs. Vertices $u \in U$ correspond to jobs $j \in J$, and vertices $v \in V$ correspond to machines $m \in M_e$ for all $e \in E$ and time periods $t \in H$.

Notice that if a job j does not require machines of equipment type e then the associated arcs $(u, v) \in A$ can be eliminated (not created). Similarly $\forall j \in J$ and $\forall t \notin S_j$ the associated arcs can also be eliminated. Figure IV.1 shows an example of such graph.

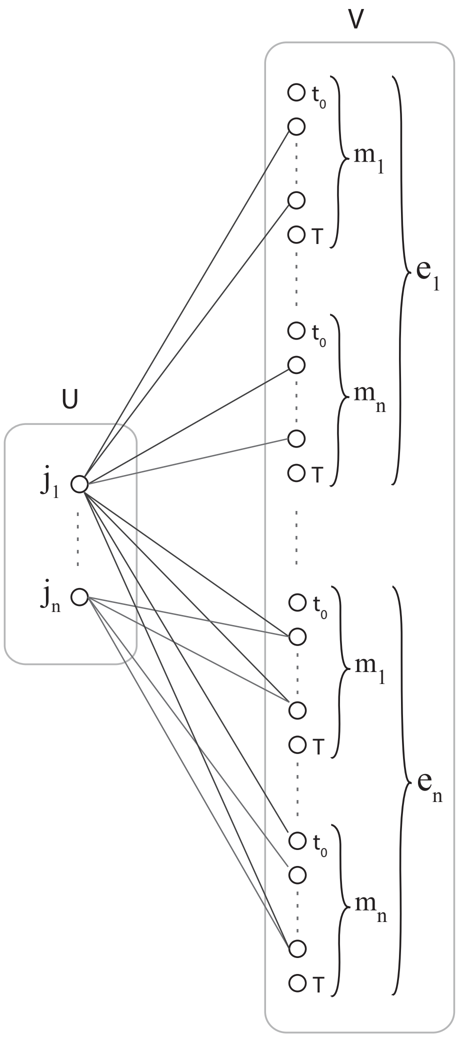

Figure IV.1: Bipartite Graph for the Assignment Model

For the assignment formulation we introduce two types of binary variables. Those variables are:

- $y_{jt} \in \{0,1\}$, $j \in J$, $t \in S_j$, equals 1 if job j starts to be executed at time t, 0 otherwise
- $x_{mejt} \in \{0,1\}$, $j \in J$, $t \in S_j$, $e \in E_j$, $m \in M_e$, equals 1 if machine m of type e starts to execute job j at time period t, 0 otherwise.

The formulation is presented below.

$$\text{Maximize} \sum_{j \in J} \sum_{t \in S_j} y_{jt} \tag{IV.1}$$

subject to

$$\sum_{t \in S_j} y_{jt} \leq 1 \quad \forall j \in J \tag{IV.2}$$

$$\sum_{m \in M_e} x_{mejt} - y_{jt} = 0 \quad \forall j \in J, t \in S_j, e \in E_j \tag{IV.3}$$

$$x_{mejt} + \sum_{t'=t}^{t+p_j+s_{ji}-1} x_{meit'} \leq 1 \quad \forall j, i \in J, t \in S_j, e \in E_j, m \in M_e \tag{IV.4}$$

The objective function (IV.1) maximizes the number of jobs executed. Constraints (IV.2) are the packing constraints, stating that each job may be executed at most once. Constraints (IV.3) are the synchronization constraints, which guarantee that for each job j either all or none of the required equipment will start executing the task at time t. Constraints (IV.4) are the resource constraints, stating that at any given time period, each machine is being used by a single job, traveling between a pair of jobs or idle.

Notice that this formulation disregards the total traveling time of mobile equipment as an explicit decision factor. This is no problem, as this formulation was developed to tackle the real case scenario described in section III.2. As previously mentioned, due to a limited fleet of vehicles, the main goal of that application was to serve as much demands as possible without concerning with traveling times.

For real demand scenarios, the number of variables created in this model was about 80K variables for 100 jobs in a planning horizon of about 2000 time instants. This number is "small" when compared to a flow formulation which explicitly considers all transitions between jobs, for the sake of considering traveling time in the objective function. By contrast, the number of resource constraints in the assignment formulation is greater than their counterpart in a flow formulation, which are the flow constraints. However, notice that resource constraints are only needed for each pair of jobs that have conflicting time windows.

The developed application had to run using a general MIP solver on hardware with limited memory. In practice we found that the assignment formulation was smaller and better suited for solving this problem under the given conditions. To date, all real case scenarios are solved to optimality within few minutes.

IV.3 Time Indexed Flow Formulation

The flow formulation was modeled over a multi-graph $G = (V, A)$ where V is the set of vertices and A is the set of arcs. Each vertex $v \in V$ represents a customer $j \in J$ in a specific time instant $t \in H$. There will be an arc $((i, t_1), (j, t_2 = t_1 + p_i + s_{ij}))^e \in A$ for each pair of customers i, j that require vehicles of type $e \in E$. Notice that an arc can be eliminated if time instant $t_2 \notin S_j$. An example of such a graph is given in Figure IV.2.

- $y_{jt} \in \{0, 1\}$, is 1 if customer j is served at time t, 0 otherwise.
- $x_{jite} \in \{0, 1\}$, is 1 if a vehicle of type e is serving customer j at time t and transiting to customer i immediatedly after serving j, 0 otherwise.
- $w_{jte} \in \{0, 1\}$, is 1 if a vehicle of type e is waiting at customer j at time t, 0 otherwise.

$$\text{Minimize} \sum_{j \in J} \sum_{i \in J} \sum_{t \in S_j} \sum_{e \in E_j} s_{ij} x_{jite} \tag{IV.5}$$

Subject to:

$$\sum_{t \in S_j} y_{jt} = 1 \ \forall j \in J \setminus \{0\} \tag{IV.6}$$

$$\sum_{i \in J} x_{jite} - y_{jt} = 0 \ \forall j \in J, t \in S_j, e \in E_j \tag{IV.7}$$

$$\sum_{i \in J \setminus \{0\}} x_{0i0e} = k_e \forall e \in E \tag{IV.8}$$

$$\sum_{i \in J} x_{jite} + w_{jte} - \sum_{i \in J} x_{ij(t-p_i-s_{ij})e} - w_{j(t-1)e} = 0 \ \forall j \in J, t \in S_j, e \in E_j \tag{IV.9}$$

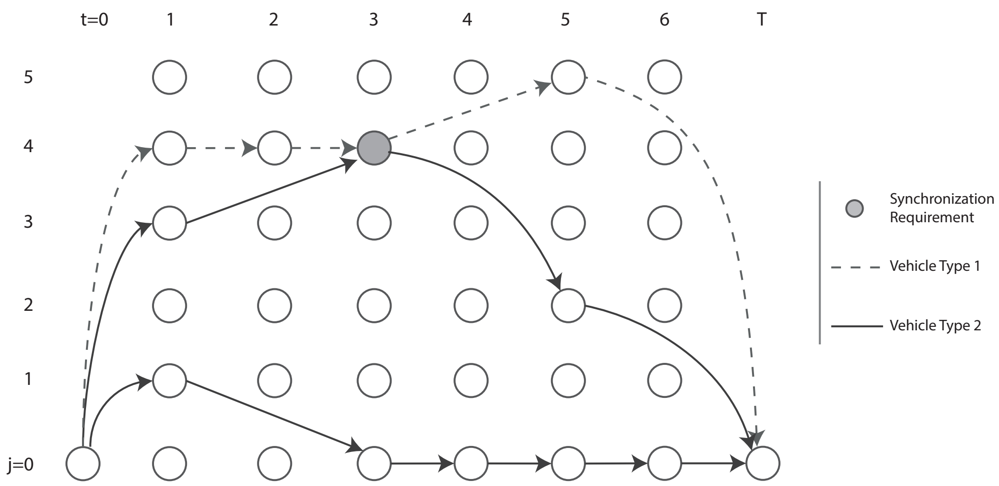

Figure IV.2: Example solution of the flow formulation

The objective function (IV.5) minimizes the total travel time of vehicles. Constraints (IV.6) are the assignment constraints, which guarantee that each customer is served exactly once. Constraints (IV.7) are the synchronization constraints, stating that for any given job j at any given time instant t either all vehicles required by j start operating at time t or none of them will. Constraints (IV.8) enforce that for each vehicle type $e \in E$ all available vehicles of that type must leave the depot, i.e. there will be k_e routes for each vehicle type $e \in E$. Constraints (IV.9) are the flow constraints, which state that all vehicles that arrive at a customer j at some particular time t must leave the customer in some posterior time.

The main difference of the flow model with respect to the assignment model is that it considers the traveling time as a decision factor. As this is similar to what is done for the VRPTW, the same benchmark instances used for the VRPTW can be tested for this model. Furthermore, those instances can be adapted to include synchronization requirements. Thus it is possible to compare the performance of the formulation with and without the existence of synchronization.

In the following chapter we describe an algorithmic approach based on the Dantzig-Wolfe decomposition of this time indexed flow formulation.