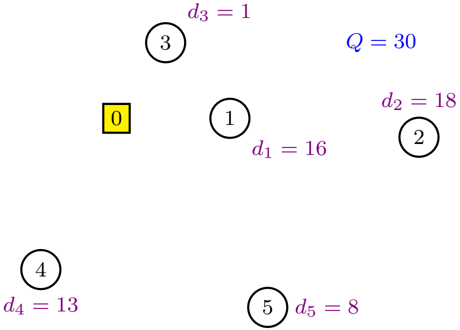
VRPSolver Tutorial

Teobaldo Bulhões, Artur Pessoa, Ruslan Sadykov and
Guillaume Marques, Eduardo Queiroga

November 2019

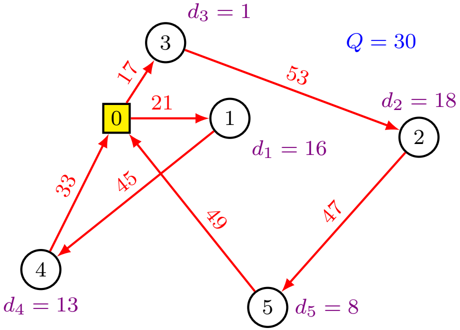
Capacitated Vehicle Routing Problem (CVRP)

- Undirected graph $G' = (V, E)$, $V = \{0, \ldots, n\}$, 0 is the depot, $V_+ = \{1, \ldots, n\}$ are the customers ; positive cost c_e, $e \in E$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting and ending at the depot, visiting all customers and such that the sum of the demands of the customers in a route does not exceed vehicle capacity.



Capacitated Vehicle Routing Problem (CVRP)

- Undirected graph $G' = (V, E)$, $V = \{0, \ldots, n\}$, 0 is the depot, $V_+ = \{1, \ldots, n\}$ are the customers ; positive cost c_e, $e \in E$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting and ending at the depot, visiting all customers and such that the sum of the demands of the customers in a route does not exceed vehicle capacity.



Capacitated Vehicle Routing Problem (CVRP) : Compact Formulation

- Undirected graph $G' = (V, E)$, $V = \{0, \dots, n\}$, 0 is the depot, $V_+ = \{1, \dots, n\}$ are the customers ; positive cost c_e, $e \in E$; positive demand d_i, $i \in V^+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting and ending at the depot, visiting all customers and such that the sum of the demands of the customers in a route does not exceed vehicle capacity.

$$\text{Min} \quad \sum_{e \in E} c_e x_e \tag{1a}$$

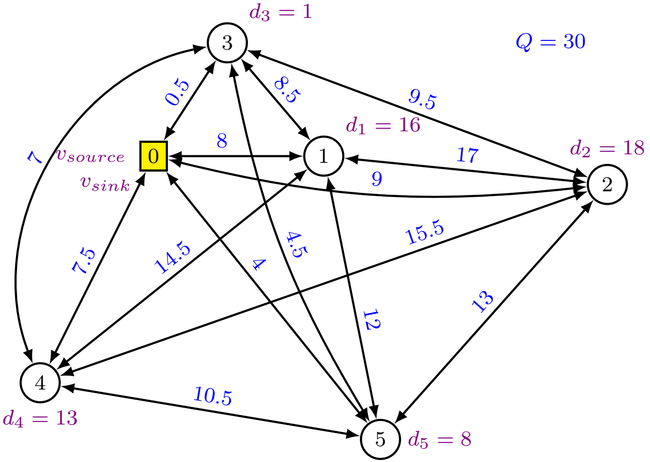
$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2, \qquad i \in V^+; \tag{1b}$$

$$\sum_{e \in \delta(S)} x_e \geq 2 \left\lceil \frac{d(S)}{Q} \right\rceil, \qquad S \subseteq V^+; \tag{1c}$$

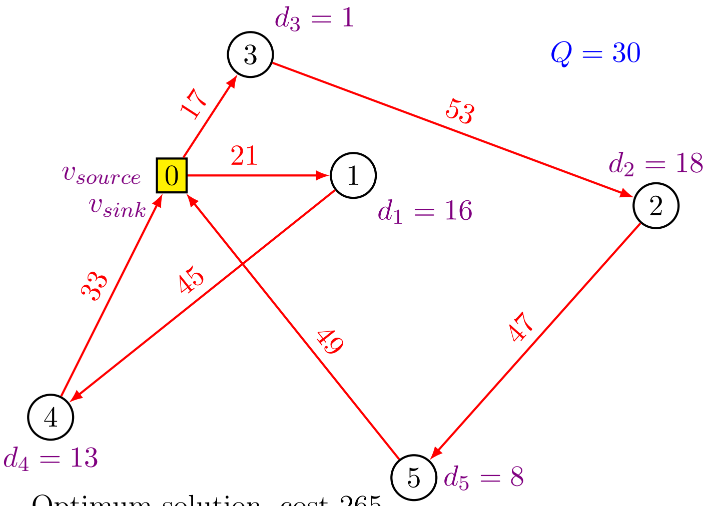
$$x_e \in \mathbb{Z}_+, \qquad e \in E. \tag{1d}$$

Single graph

$G = (V, A)$, $A = \{(i,j),(j,i) : \{i,j\} \in E\}$, $v_{\text{source}} = v_{\text{sink}} = 0$;
$R = R_M = \{1\}$; $q_{a,1} = (d_i + d_j)/2$, $a = (i,j) \in A$ (define $d_0 = 0$) ;
$l_{i,1} = 0, u_{i,1} = Q, i \in V$;



Capacitated Vehicle Routing Problem (CVRP) : solution



Optimum solution, cost 265

VRPSolver Model for CVRP

Formulation

Integer variables x_e, $e \in E$.

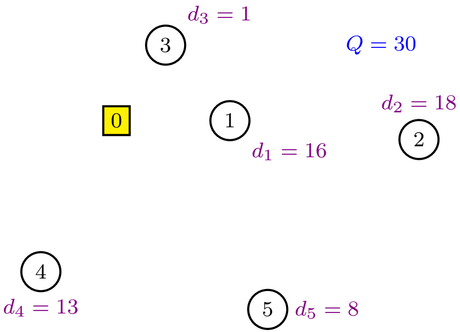
$$\text{Min} \quad \sum_{e \in E} c_e x_e \tag{2a}$$

$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2, \qquad i \in V_+. \tag{2b}$$

$L = \lceil \sum_{i=1}^{n} d_i / Q \rceil$, $U = n$; $M(x_e) = \{(i,j),(j,i)\}$, $e = \{i,j\} \in E$.

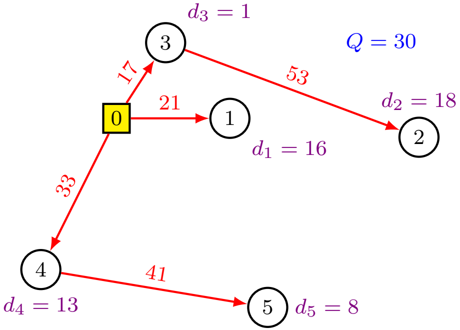
Open Vehicle Routing Problem (OVRP)

- Directed graph $G' = (V, A')$, $V = \{0, \dots, n\}$, 0 is the depot, $V_+ = \{1, \dots, n\}$ are the customers and A' have no arcs ending at the depot ; positive cost c_a, $a \in A'$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting at the depot, visiting all customers such that the sum of the demands of the customers in a route does not exceed vehicle capacity.



Open Vehicle Routing Problem (OVRP)

- Directed graph $G' = (V, A')$, $V = \{0, \ldots, n\}$, 0 is the depot, $V_+ = \{1, \ldots, n\}$ are the customers and A' have no arcs ending at the depot ; positive cost c_a, $a \in A'$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting at the depot, visiting all customers such that the sum of the demands of the customers in a route does not exceed vehicle capacity.



Open Vehicle Routing Problem (OVRP) : Compact Formulation

- Directed graph $G' = (V, A')$, $V = \{0, \ldots, n\}$, 0 is the depot, $V_+ = \{1, \ldots, n\}$ are the customers and A' have no arcs ending at the depot ; positive cost c_a, $a \in A'$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting at the depot, visiting all customers such that the sum of the demands of the customers in a route does not exceed vehicle capacity.

$$\begin{aligned}
\text{Min} \quad & \sum_{a \in A'} c_a x_a & & & \text{(3a)} \\
\text{S.t.} \quad & \sum_{a \in \delta^-(i)} x_a = 1, & & i \in V^+; & \text{(3b)} \\
& \sum_{a \in \delta^-(S)} x_a \geq \left\lceil \frac{d(S)}{Q} \right\rceil, & & S \subseteq V^+; & \text{(3c)} \\
& x_a \in \mathbb{Z}_+, & & a \in A'. & \text{(3d)}
\end{aligned}$$

How to Adapt the CVRP Model for the OVRP ?

Single graph

$G = (V, A)$, $A = \{(i,j),(j,i) : \{i,j\} \in E\}$, $v_{\text{source}} = v_{\text{sink}} = 0$;
$R = R_M = \{1\}$; $q_{a,1} = (d_i + d_j)/2$, $a = (i,j) \in A$ (define $d_0 = 0$) ;
$l_{i,1} = 0, u_{i,1} = Q, i \in V$;

Formulation

Integer variables x_e, $e \in E$.

$$\text{Min} \quad \sum_{e \in E} c_e x_e \tag{4a}$$

$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2, \qquad i \in V_+. \tag{4b}$$

$L = \lceil \sum_{i=1}^{n} d_i/Q \rceil$, $U = n$; $M(x_e) = \{(i,j),(j,i)\}$,
$e = \{i,j\} \in E$.

How to Adapt the CVRP Model for the OVRP ?

Single graph

$G = (V, A)$, $A = \{(i,j),(j,i) : \{i,j\} \in E\}$, $v_{\text{source}} = v_{\text{sink}} = 0$;
$R = R_M = \{1\}$; $q_{a,1} = (d_i + d_j)/2$, $a = (i,j) \in A$ (define $d_0 = 0$) ;
$l_{i,1} = 0, u_{i,1} = Q, i \in V$;

Formulation

Integer variables x_a, $a \in A$.

$$\text{Min} \quad \sum_{a \in A} c'_a x_a \tag{5a}$$

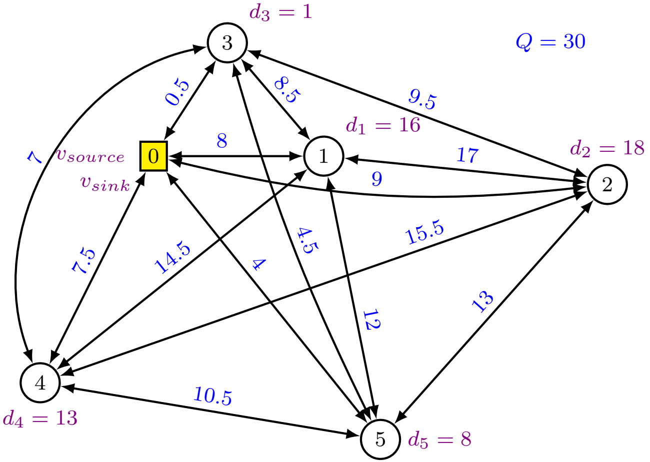
$$\text{S.t.} \quad \sum_{a \in \delta^-(i)} x_a = 1, \qquad i \in V_+. \tag{5b}$$

$L = \lceil \sum_{i=1}^{n} d_i / Q \rceil$, $U = n$; $M(x_a) = \{(i,j)\}$, $a = (i,j) \in A$.
$c'_a = c_a$ if $a \notin \delta^-(0)$, and 0 otherwise.

Open Vehicle Routing Problem (OVRP) : graph

Single graph

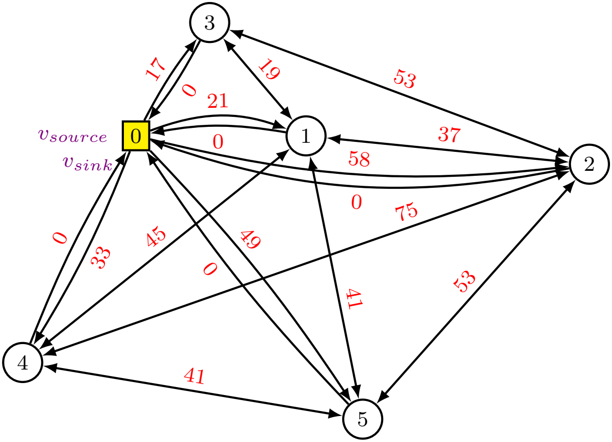
$G = (V, A)$, $A = A' \cup \{(i,0) : i \in V^+\}$, $c_{i0} = 0, i \in V^+$; $v_{\text{source}} = v_{\text{sink}} = 0$;
$R = R_M = \{1\}$; $q_{a,1} = (d_i + d_j)/2$, $a = (i,j) \in A$ (define $d_0 = 0$) ;
$l_{i,1} = 0, u_{i,1} = Q, i \in V$;



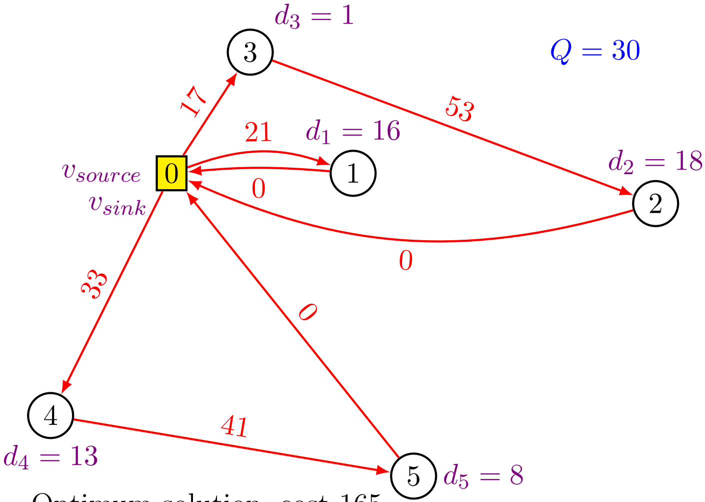
Open Vehicle Routing Problem (OVRP) : graph

Single graph

$G = (V, A)$, $A = A' \cup \{(i, 0) : i \in V^+\}$, $c_{i0} = 0, i \in V^+$; $v_{\text{source}} = v_{\text{sink}} = 0$;
$R = R_M = \{1\}$; $q_{a,1} = (d_i + d_j)/2$, $a = (i, j) \in A$ (define $d_0 = 0$) ;
$l_{i,1} = 0, u_{i,1} = Q, i \in V$;



Open Vehicle Routing Problem (OVRP) : solution



Optimum solution, cost 165

VRPSolver Model for OVRP

Formulation

Integer variables x_a, $a \in A$.

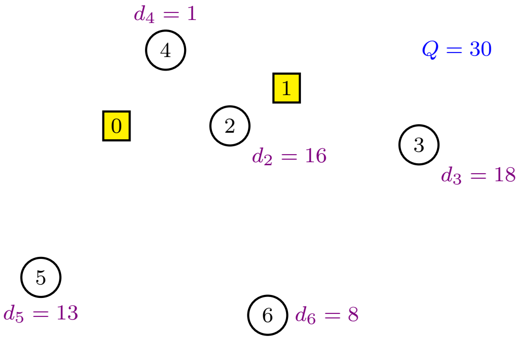
$$\text{Min} \quad \sum_{a \in A} c_a x_a \tag{6a}$$

$$\text{S.t.} \quad \sum_{a \in \delta^-(i)} x_a = 1, \qquad i \in V^+. \tag{6b}$$

$L = \lceil \sum_{i=1}^{n} d_i / Q \rceil$, $U = n$; $M(x_a) = \{a\}$, $a \in A$.

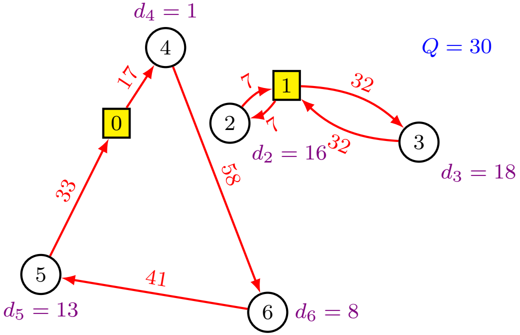
Multi-Depot Vehicle Routing Problem (MDVRP)

- Graph $G' = (V, E)$, $V = \{0, \ldots, n+m-1\}$, $D = \{0, 1, \ldots, m\}$ is a set of depots, $V_+ = \{m+1, \ldots, n+m-1\}$ are the customers ; $E = \{\{i, j\} : i, j \in V, i < j, i$ or j is not a depot$\}$; cost c_e, $e \in E$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting and ending at the same depot, visiting all customers and such that the sum of the demands of the customers in a route does not exceed capacity.



Multi-Depot Vehicle Routing Problem (MDVRP)

- Graph $G' = (V, E)$, $V = \{0, \ldots, n+m-1\}$, $D = \{0, 1, \ldots, m\}$ is a set of depots, $V_+ = \{m+1, \ldots, n+m-1\}$ are the customers ; $E = \{\{i,j\} : i,j \in V, i < j, i \text{ or } j \text{ is not a depot}\}$; cost c_e, $e \in E$; positive demand d_i, $i \in V_+$; vehicle capacity Q.
- Find a minimum cost set of routes, starting and ending at the same depot, visiting all customers and such that the sum of the demands of the customers in a route does not exceed capacity.



How to Adapt the CVRP Model for the MDVRP ?

Single graph

$G = (V, A)$, $A = \{(i,j),(j,i) : \{i,j\} \in E\}$, $v_{\text{source}} = v_{\text{sink}} = 0$;
$R = R_M = \{1\}$; $q_{a,1} = (d_i + d_j)/2$, $a = (i,j) \in A$ (define $d_0 = 0$) ;
$l_{i,1} = 0, u_{i,1} = Q, i \in V$;

Formulation

Integer variables x_e, $e \in E$.

$$\text{Min} \quad \sum_{e \in E} c_e x_e \tag{7a}$$

$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2, \qquad i \in V_+. \tag{7b}$$

$L = \lceil \sum_{i=1}^{n} d_i/Q \rceil$, $U = n$; $M(x_e) = \{(i,j),(j,i)\}$,
$e = \{i,j\} \in E$.

How to Adapt the CVRP Model for the MDVRP ?

Multiple graphs

$G^k = (V^k, A^k)$ for each $k \in D$,
$A^k = \{(v_i^k, v_j^k), (v_j^k, v_i^k) : \{i,j\} \in E, i \neq D \setminus k\}$, $v_{\text{source}} = v_{\text{sink}} = v_k^k$;
$R^k = R_M^k = \{1\}$; $q_{a,1}^k = (d_{v_i^k} + d_{v_j^k})/2$, $a = (v_i^k, v_j^k) \in A^k$ (define $d_{v_k^k} = 0$) ;
$l_{v_i^k,1} = 0, u_{v_i^k,1} = Q, v_i^k \in V^k$.

Formulation

Integer variables x_e, $e \in E$, with no edge between depots.

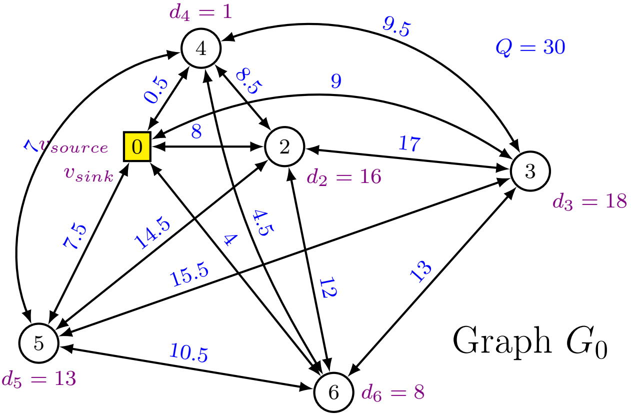
$$\text{Min} \quad \sum_{e \in E} c_e x_e \tag{8a}$$

$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2, \qquad i \in V_+. \tag{8b}$$

$L^k = 0$, $U^k = n$, for each $k \in D$;
$M(x_e) = \{(v_i^k, v_j^k), (v_j^k, v_i^k) : v_i^k, v_j^k \in G^k\}$, $e = \{i,j\} \in E$.

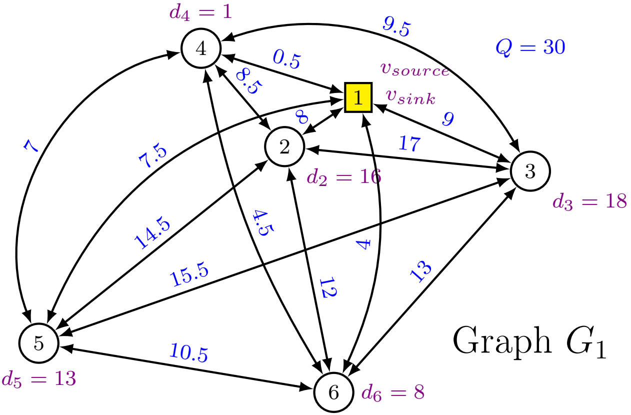
Multiple graphs

$G^k = (V^k, A^k)$ for each $k \in D$,
$A^k = \{(v_i^k, v_j^k), (v_j^k, v_i^k) : \{i,j\} \in E, i \neq D \setminus k\}$, $v_{\text{source}} = v_{\text{sink}} = v_k^k$;
$R^k = R_M^k = \{1\}$; $q_{a,1}^k = (d_{v_i^k} + d_{v_j^k})/2$, $a = (v_i^k, v_j^k) \in A^k$ (define $d_{v_k^k} = 0$) ;
$l_{v_i^k,1} = 0, u_{v_i^k,1} = Q, v_i^k \in V^k$.



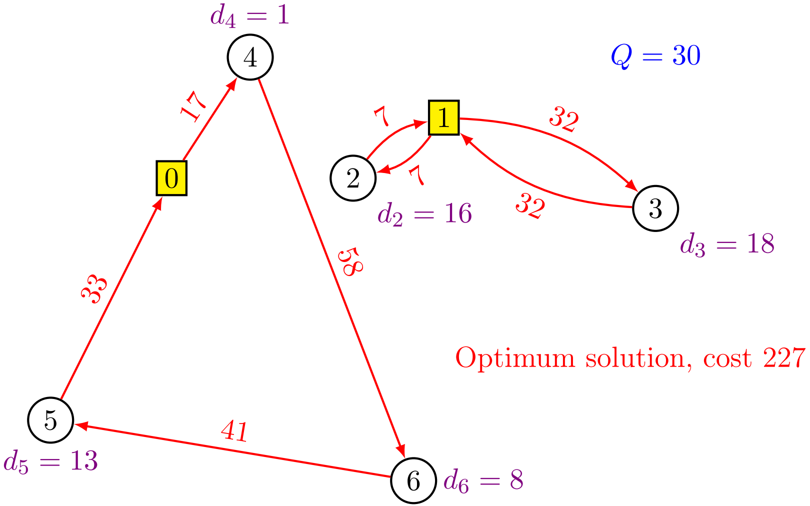
Multiple graphs

$G^k = (V^k, A^k)$ for each $k \in D$,
$A^k = \{(v_i^k, v_j^k), (v_j^k, v_i^k) : \{i,j\} \in E, i \neq D \setminus k\}$, $v_{\text{source}} = v_{\text{sink}} = v_k^k$;
$R^k = R_M^k = \{1\}$; $q_{a,1}^k = (d_{v_i^k} + d_{v_j^k})/2$, $a = (v_i^k, v_j^k) \in A^k$ (define $d_{v_k^k} = 0$) ;
$l_{v_i^k,1} = 0, u_{v_i^k,1} = Q, v_i^k \in V^k$.



Graph G_1

Multi-Depot Vehicle Routing Problem (MDVRP) : solution



Bin Packing Problem (BPP)

Data : Set T of items ; bin capacities Q ; item weight w_t, $t \in T$.

Goal : Find a packing using the minimum number of bins, such that, the total weight of the items in a bin does not exceed its capacity.

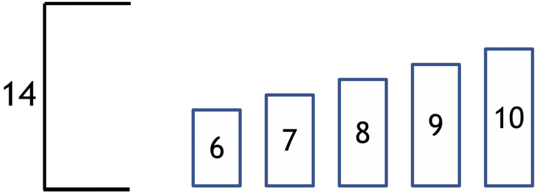
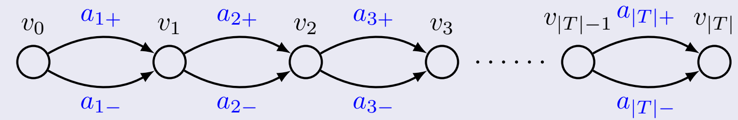


FIGURE – Toy instance

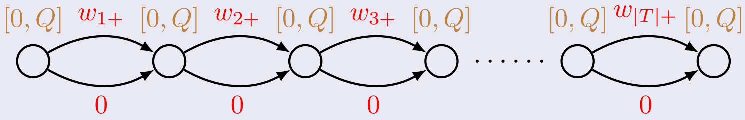
Bin Packing Problem (BPP)

Graph G



- Capacity is the only one resource with consumption : $q_{a_{t+}} = w_t$, $q_{a_{j-}} = 0$, $t \in T$
- Consumption bounds $[0, Q]$ for all nodes

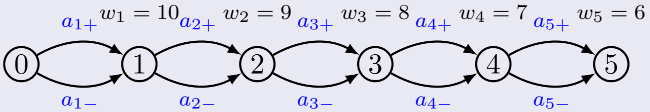
RCSP Subproblem



Bin Packing Problem (BPP)

Toy instance with $T = 5$ and $Q = 14$

Graph



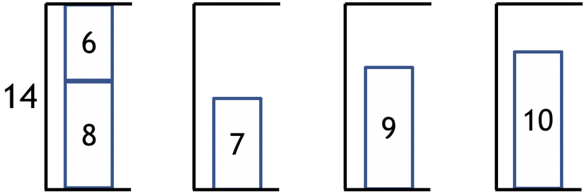
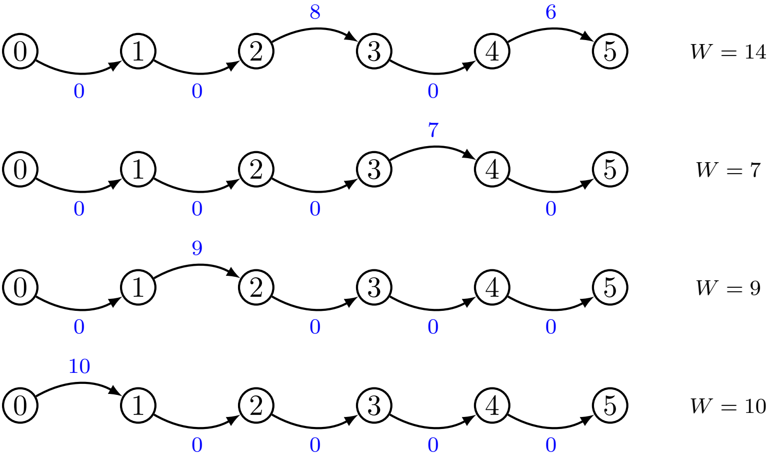


FIGURE – Toy instance solution

Bin Packing Problem (BPP)

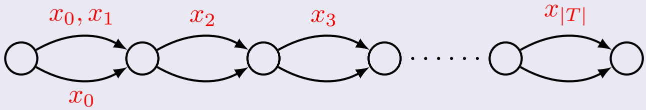
Toy instance with $T = 5$ and $Q = 14$, solution :



Instance `toy.txt`, objective value = 4 bins

Bin Packing Problem (BPP)

Arc mapping



Formulation and Additional Elements

$$\begin{aligned} \text{Min} \quad & x_0 \\ \text{S.t.} \quad & x_t = 1, \qquad t \in T; \end{aligned}$$

- Subproblem cardinality : $L = 0$, $U = \infty$
- Packing sets : $\mathcal{B} = \cup_{t \in T}\{\{a_{t+}\}\}$
- Branching over accumulated resource consumption and, if still needed, by Ryan and Foster rule
- Enumeration is on

Variable Sized Bin Packing Problem (VSBPP)

Data : Set T of items ; Set B of bin types ; bin capacity $Q^k, k \in B$; bin cost $c_k, k \in B$; bin availability $s_k, k \in B$; item weight w_t, $t \in T$.

Goal : Find a packing minimizing the cost with bins, such that, the total weight of the items in a bin does not exceed its capacity and the availability of bins is not violated.

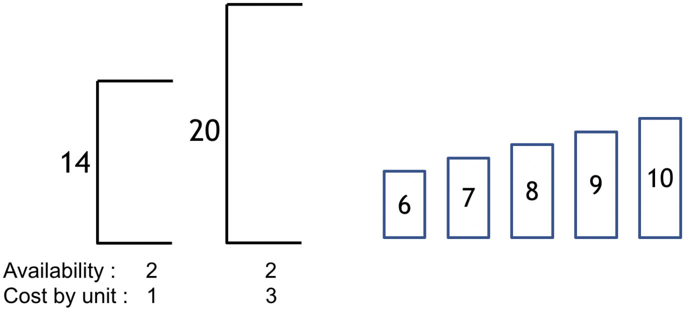


FIGURE – Toy instance

Variable Sized Bin Packing Problem (VSBPP)

Data : Set T of items ; Set B of bin types ; bin capacity $Q^k, k \in B$; bin cost $c_k, k \in B$; bin availability $s_k, k \in B$; item weight w_t, $t \in T$.

Goal : Find a packing minimizing the cost with bins, such that, the total weight of the items in a bin does not exceed its capacity and the availability of bins is not violated.

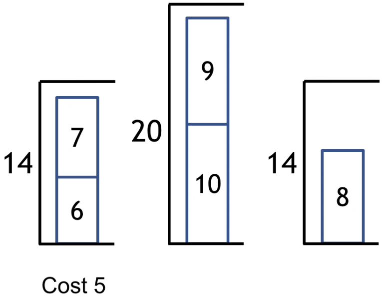
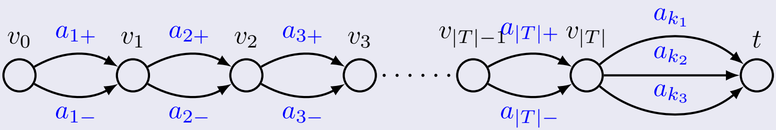


FIGURE – Toy instance solution

How to Adapt the BPP Model for the VSBPP ?

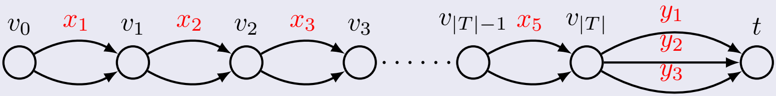
Graph G



- Capacity is the only one resource with consumption : $q_{a_{t+}} = w_t,\ q_{a_{j-}} = 0,\ t \in T$
- **Consumption bounds** $[0, \max_{k \in B} Q^k]$ for nodes v_t, $t \in T$
- **Consumption bounds** $[0, Q^k]$ for arcs a_k, $k \in B$

How to Adapt the BPP Model for the VSBPP ?

Arc mapping



Model

Let $x_t = 1$ if item t assigned, let y_k be the number of bin of type $k \in B$ used.

$$\text{Min} \quad \sum_{k\in B} c_k y_k \tag{9a}$$

$$\text{S.t.} \quad x_t = 1, \qquad t \in T \tag{9b}$$

$$y_k \le s_k, \qquad k \in B \tag{9c}$$

Black and White Traveling Salesman Problem (BWTSP)

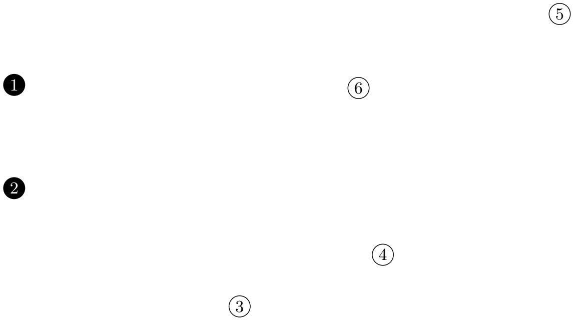
- Let B be the set of black nodes, W be the set of white nodes, a complete graph $G = (B \cup W, E)$, and $Q \in \mathbb{N}^+$.
- Find a shortest Hamiltonian tour, visiting all vertices and such that the number of white vertices between any two customers not exceed value Q.

How to model that with one subproblem creating paths that :

- start with a black node,
- visit at most Q white nodes,
- and finish with a black node ?

Black and White Traveling Salesman Problem (BWTSP)

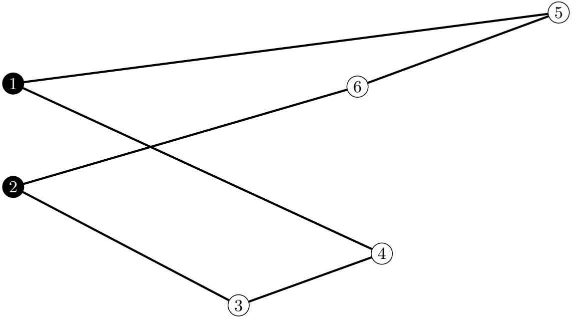
$B = \{1, 2\}$ and $Q = 2$



Instance `toy.tsp`

Black and White Traveling Salesman Problem (BWTSP)

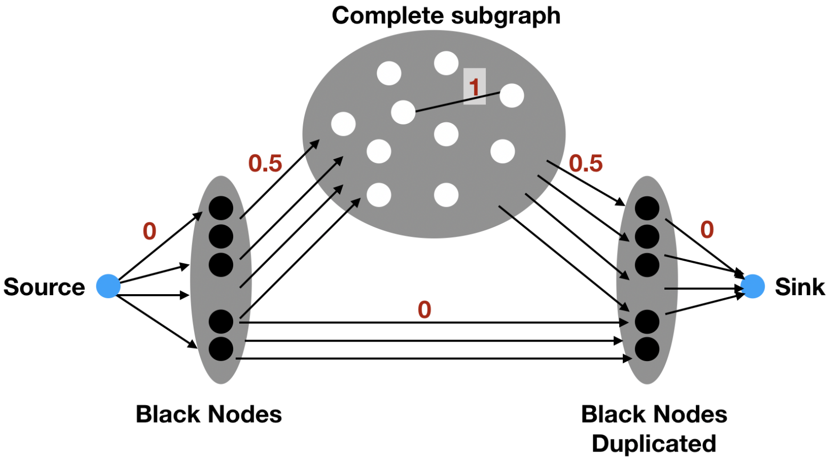
$B = \{1, 2\}$ and $Q = 2$



Instance `toy.tsp`, cost 3381

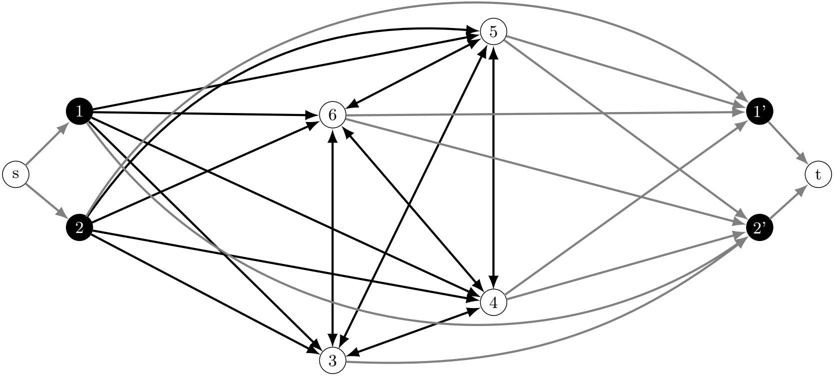
Black and White Traveling Salesman Problem (BWTSP)

Graph representation of the model



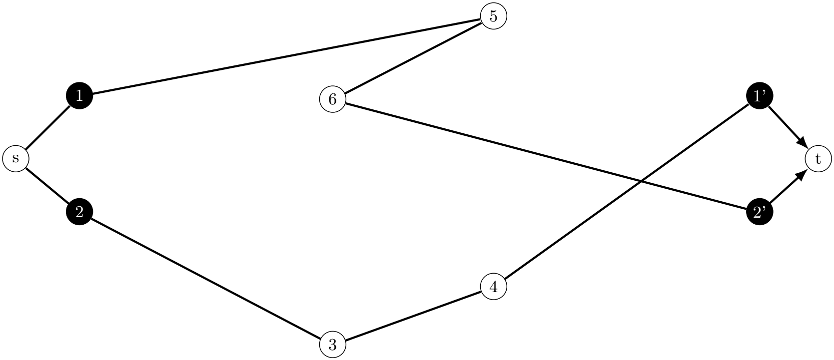
- Resource bounds on nodes : $[0, Q]$
- Resource consumption of arcs

Graph representation of the model, $B = \{1, 2\}$ and $Q = 2$



Instance `toy.tsp`

Graph representation of the model, $B = \{1, 2\}$ and $Q = 2$



Instance `toy.tsp`, cost 3381

Black and White Traveling Salesman Problem (BWTSP)

- Let B be the set of black nodes, W be the set of white nodes, a complete graph $G = (B \cup W, E)$, and $Q \in \mathbb{N}^+$.
- We denote $V = B \cup W$, c_e the cost of using $e \in E$

Formulation

Integer variables x_e, $e \in E$

$$\text{Min} \quad \sum_{e \in E} c_e x_e \tag{10a}$$

$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2, \qquad i \in V \tag{10b}$$

$L = U = Q$;
$M(x_e) = \{\text{edges representing } e \text{ in the subproblem graph}\}$

Is this model works? Try to print a solution.

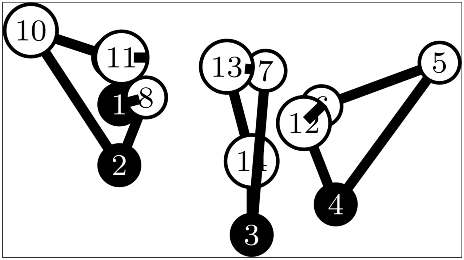


FIGURE – Example of a feasible solution to the previous model

We need **subtour elimination constraints** :
Consider black node $1 \in B$,

$$\sum_{i \in V_1 \cup \{1\}} \sum_{j \in V_2 \cup \{b\}} x_{ij} \geq 2 \quad b \in B \backslash \{1\}, V_1 \cup V_2 = V \backslash \{1, b\}, V_1 \cap V_2 = \emptyset$$

Separation by looking for mincut between pairs of black nodes $(1, b)$, $b \in B \setminus \{1\}$