On solving the minmax multiple traveling salesman problem by column generation

Nicola Bianchessi†, Christian Tilk‡, Stefan Irnich‡‡

†Dipartimento di Informatica

UNIVERSITÀ DEGLI STUDI DI MILANO

‡Department of Business Decisions and Analytics

universität wien

‡‡Chair of Logistics Management, Gutenberg School of Management and Economics

JOHANNES GUTENBERG UNIVERSITÄT MAINZ

CG2023 - Montréal, May 16-19, 2023

- "Nowadays, the leading exact algorithms for solving many classes of *vehicle-routing problems* (VRPs, Toth and Vigo [2014]) are branch-price-and-cut algorithms." [Costa et al., 2019].
- In most of the cases, the considered objective function is to minimize the total distance travelled by all vehicles (*minsum* objectives).

- "Nowadays, the leading exact algorithms for solving many classes of *vehicle-routing problems* (VRPs, Toth and Vigo [2014]) are branch-price-and-cut algorithms." [Costa et al., 2019].
- In most of the cases, the considered objective function is to minimize the total distance travelled by all vehicles (*minsum* objectives).
- **Can column generation be effectively applied to address minmax vehicle routing problems?**

- "Nowadays, the leading exact algorithms for solving many classes of *vehicle-routing problems* (VRPs, Toth and Vigo [2014]) are branch-price-and-cut algorithms." [Costa et al., 2019].
- In most of the cases, the considered objective function is to minimize the total distance travelled by all vehicles (*minsum* objectives).
- **Can column generation be effectively applied to address minmax vehicle routing problems?**

  The *minmax* multiple traveling salesman problem (mTSP) [França et al., 1995]:

- "Nowadays, the leading exact algorithms for solving many classes of *vehicle-routing problems* (VRPs, Toth and Vigo [2014]) are branch-price-and-cut algorithms." [Costa et al., 2019].
- In most of the cases, the considered objective function is to minimize the total distance travelled by all vehicles (*minsum* objectives).
- **Can column generation be effectively applied to address minmax vehicle routing problems?**
  The *minmax* multiple traveling salesman problem (mTSP) [França et al., 1995]:
  - The *minsum* mTSP is generalized (directly or indirectly) by several routing problems.

- "Nowadays, the leading exact algorithms for solving many classes of *vehicle-routing problems* (VRPs, Toth and Vigo [2014]) are branch-price-and-cut algorithms." [Costa et al., 2019].
- In most of the cases, the considered objective function is to minimize the total distance travelled by all vehicles (*minsum* objectives).
- **Can column generation be effectively applied to address minmax vehicle routing problems?**
  The *minmax* multiple traveling salesman problem (mTSP) [França et al., 1995]:
  - The *minsum* mTSP is generalized (directly or indirectly) by several routing problems.
  - By addressing compact formulations for the minmax mTSP through commercial MILP solvers, optimal solutions can be found, within 1 hour, only for instances with 10 customers [Sarin et al., 2014, Soylu, 2015].
    The optimality gap for heuristic solutions can be computed only for small instances [see, e.g., He and Hao, 2023].

- Given:
  - a road network;
  - a fleet of homogeneous vehicles housed in a common depot;
  - a set of customers to visit;
- Find vehicle tours (routes) such that:
  - each customer is visited;
  - *the length of the longest tour (route) is minimized*.
- Example:

(a) (b)

A minmax mTSP instance (a) and a corresponding solution (b) with a fleet of 3 vehicles.

Data:

N	Customer set $\{1, \ldots, n\}$
$\{0, n+1\}$	Depot at the beginning and the end of the planning horizon, respectively
$G = (V, A)$	Directed graph modelling the road network: ■ $V = N \cup \{0, n+1\}$ ■ $A = \{(i,j) \in \lvert V\rvert \times \lvert V\rvert : i \neq n+1, j \neq 0\} \setminus \{(0, n+1)\}$
t_{ij}	Length (distance) associated with the traversal of arc $(i,j) \in A$
K	Index set for the vehicles

Variables:

$C \in \mathbb{R}_+$	Length of the longest tour (route) among those assigned to the $\lvert K\rvert$ vehicles
$x_{ij} \in \{0, 1\}$	Vehicle flow along arc $(i,j) \in A$
$T_{ij} \in \mathbb{R}_+$	Cumulated length, at vertex j, of a partial tour (route) covered by a vehicle coming directly from vertex i

Compact model: Arc-flow formulation (known: ∼[Maffioli and Sciomachen, 1997])

$$\min C \tag{1a}$$

$$\sum_{j \in N} x_{0,j} = \sum_{i \in N} x_{i,n+1} = |K| \tag{1b}$$

$$\sum_{(j,i) \in \delta^-(i)} x_{ji} = \sum_{(i,j) \in \delta^+(i)} x_{ij} = 1 \qquad i \in N \tag{1c}$$

$$T_{0j} = t_{0j} x_{0j} \qquad j \in N \tag{1d}$$

$$\sum_{(i,j) \in \delta^+(i)} T_{ij} - \sum_{(j,i) \in \delta^-(i)} T_{ji} = \sum_{(i,j) \in \delta^+(i)} t_{ij} x_{ij} \qquad i \in N \tag{1e}$$

$$T_{i,n+1} \leq C \qquad i \in N \tag{1f}$$

$$t^0_{ij} x_{ij} \leq T_{ij} \leq C^U_{j,n+1} x_{ij} \qquad (i,j) \in A \tag{1g}$$

$$x_{ij} \in \{0,1\} \qquad (i,j) \in A \tag{1h}$$

$$C^L \leq C^1 \leq C^U \text{ (optional)} \tag{1i}$$

where:

- $\delta^+(S)/\delta^-(S)$: Set of arcs $\{(i,j) \in A : i \in S, j \notin S\}/\{(i,j) \in A : i \notin S, j \in S\}$ leaving/entering the set $S \subseteq N$ (with $\delta^+(\{i\}) = \delta^+(i)/\delta^-(\{i\}) = \delta^-(i)$);
- $t^0_{ij} = t_{0i} + t_{ij}$, with $t_{00} = 0$, and $C^U_{j,n+1} = C^U - t_{j,n+1}$, with $t_{n+1,n+1} = 0$.

Path-based formulation (known: [Benavent et al., 2014])

Data:

- P Set of feasible paths for any vehicle
- a_i^p Number of times path $p \in P$ visits customer $i \in N$
- c^p Length of path $p \in P$

Variables:

- $C^k \in \mathbb{R}_+$ Length of the path assigned to vehicle $k \in K$
- $\lambda^{kp} \in \{0,1\}$ Binary variable equal to 1 if path $p \in P$ is assigned to vehicle $k \in K$

$$\min \quad C^1 \tag{2a}$$

$$s.t. \quad \sum_{k \in K} \sum_{p \in P} a_i^p \lambda^{kp} \geq 1 \qquad i \in N \tag{2b}$$

$$\sum_{p \in P} \lambda^{kp} \leq 1 \qquad k \in K \tag{2c}$$

$$\sum_{p \in P} c^p \lambda^{kp} \leq C^k \qquad k \in K \tag{2d}$$

$$C^k \geq C^{k+1} \qquad k \in \{1, \ldots, |K| - 1\} \tag{2e}$$

$$\lambda^{kp} \in \{0, 1\} \qquad k \in K,\ p \in P \tag{2f}$$

$$C^k \geq 0 \qquad k \in K \tag{2g}$$

$$C^L \leq C^1 \leq C^U \text{ (optional)} \tag{2h}$$

- The dual of the linear relaxation of (2) is:

$$\max \quad \sum_{i \in N} \mu_i + \sum_{k \in K} \theta_k \tag{3a}$$

$$s.t. \quad \sum_{i \in N} a_i^p \mu_i + \theta_k + c^p \rho_k \leq 0 \qquad k \in K,\ p \in P \tag{3b}$$

$$-\rho_1 + \sigma_1 \leq 1 \tag{3c}$$

$$-\rho_k - \sigma_{k-1} + \sigma_k \leq 0 \qquad k \in \{2, \ldots, |K| - 2\} \tag{3d}$$

$$-\rho_{|K|} - \sigma_{|K|-1} \leq 0 \tag{3e}$$

$$\mu_i \geq 0 \qquad i \in N \tag{3f}$$

$$\theta_k, \rho_k \leq 0 \qquad k \in K \tag{3g}$$

$$\sigma_k \geq 0 \qquad k \in \{1, \ldots, |K| - 1\} \tag{3h}$$

where $\mu_i \in \mathbb{R}_+$, $\theta_k, \rho_k \in \mathbb{R}_-$, and $\sigma_k \in \mathbb{R}_+$ are the dual variables associated with constraints (2b), (2c), (2d), and (2e), respectively.

Pricing problem (cont'd)

- The reduced cost of path $p \in P$, when assigned to vehicle $k \in K$, is defined as:

$$-\sum_{i \in N} a_i^p \mu_i - \theta_k - c^p \rho_k \tag{4}$$

- A distinct pricing problem (PP) for each $k \in K$:

$$\min_{p \in P} \bar{c}_p(\boldsymbol{\mu}, \rho_k)$$

where $\bar{c}_p(\boldsymbol{\mu}, \rho_k) = -\sum_{i \in N} a_i^p \mu_i - c^p \rho_k$.

  - A path $p \in P$ represents a negative reduced cost solution for PP_k, $k \in K$, if $\bar{c}_p(\boldsymbol{\mu}, \rho_k) < \theta_k$.
  - Each PP corresponds to an Elementary Shortest Path Problem with Resource Constraints (ESPPRC), which is known to be NP-hard in the strong sense (Dror [1994]), and can be solved by means of state-of-the-art techniques ([Costa et al., 2019]).
  - Any available upper bound C^U for C^1 can be used to restrict the feasible regions of the PPs.

Can we somehow exploit the fact that all the pricing problems share the same feasible region?

Observation 1 (Empirical)

At each column generation iteration, it often happens that some dual variables ρ_k, $k \in K$, have the same value (Benavent et al. [2014], Bianchessi and Tresoldi [2021], Bianchessi et al. [2022]).

Can we somehow exploit the fact that all the pricing problems share the same feasible region?

Observation 1 (Empirical)

At each column generation iteration, it often happens that some dual variables ρ_k, $k \in K$, have the same value (Benavent et al. [2014], Bianchessi and Tresoldi [2021], Bianchessi et al. [2022]).

(i) The column generation iteration can be properly designed in order to exploit Observation 1.

Algorithm 1: Column generation iteration

Input: Index set fot the vehicles K, dual variables $(\boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{\rho})$, set of pricing problems $\{PP_1, \ldots, PP_{|K|}\}$.

Output: Sets of pareto-optimal solutions $\bar{S}_k$ with negative reduced cost for all pricing problems PP_k, $k \in K$.

Define set $K' = \{k_1, \ldots, k_{|K|}\}$ such that $\rho_{k_i} \leq \rho_{k_j}$ for each $i < j$;
$\bar{S}_k \leftarrow \emptyset$ for each $k \in K$; $i \leftarrow 1$;
while $i \leq |K|$ **do**
  if $(i == 1)$ *or* $(|\rho_{k_i} - \rho_{k_{i-1}}| > \epsilon)$ **then**
    Solve PP_{k_i} computing set S_{k_i}; /* S_k: Set of pareto-optimal solutions of PP_k */
    for $s \in S_{k_i}$ **do**
      $j \leftarrow i$;
      while $(j \leq |K|)$ *and* $(|\rho_{k_i} - \rho_{k_j}| \leq \epsilon)$ **do**
        if (s is a negative reduced cost solution for PP_{k_j}) **then**
          $\bar{S}_{k_i} \leftarrow \bar{S}_{k_i} \cup \{s\}$;
        $j \leftarrow j + 1$;
  $i \leftarrow i + 1$;

Can we somehow exploit the fact that all the pricing problems share the same feasible region?

Observation 1 (Empirical)

At each column generation iteration, it often happens that some dual variables ρ_k, $k \in K$, have the same value (Benavent et al. [2014], Bianchessi and Tresoldi [2021], Bianchessi et al. [2022]).

(i) The column generation iteration can be properly designed in order to exploit Observation 1.

Algorithm 1: Column generation iteration

Input: Index set fot the vehicles K, dual variables $(\boldsymbol{\mu}, \boldsymbol{\theta}, \boldsymbol{\rho})$, set of pricing problems $\{PP_1, \ldots, PP_{|K|}\}$.

Output: Sets of pareto-optimal solutions $\bar{S}_k$ with negative reduced cost for all pricing problems PP_k, $k \in K$.

```
Define set K' = {k_1, ..., k_|K|} such that ρ_{k_i} ≤ ρ_{k_j} for each i < j;
S̄_k ← ∅ for each k ∈ K; i ← 1;
while i ≤ |K| do
    if (i == 1) or (|ρ_{k_i} − ρ_{k_{i−1}}| > ε) then
        Solve PP_{k_i} computing set S_{k_i};   /* S_k: Set of pareto-optimal solutions of PP_k */
        for s ∈ S_{k_i} do
            j ← i;
            while (j ≤ |K|) and (|ρ_{k_i} − ρ_{k_j}| ≤ ε) do
                if (s is a negative reduced cost solution for PP_{k_j}) then
                    S̄_{k_i} ← S̄_{k_i} ∪ {s};
                j ← j + 1;
    i ← i + 1;
```

(ii) The branching scheme must allow the PPs to share the same feasible region at every node of the branch-and-bound tree.

Notation

$(\bar{\boldsymbol{\lambda}}, \bar{\mathbf{C}})$ Optimal fractional solution to the current linear relaxation of (2)

$\bar{P}$ Set of paths associated with variables $\bar{\boldsymbol{\lambda}} > 0$

$\hat{P}$ Superset of $\bar{P}$ defined according to some criteria

$$
\begin{aligned}
C^U = \min \quad & C^1 && && \text{(5a)}\\
s.t. \quad & \sum_{k \in K} \sum_{p \in \hat{P}} a_i^p \lambda^{kp} \geq 1 && i \in N && \text{(5b)}\\
& \sum_{p \in \hat{P}} \lambda^{kp} \leq 1 && k \in K && \text{(5c)}\\
& \sum_{p \in \hat{P}} c^p \lambda^{kp} \leq C^k && k \in K && \text{(5d)}\\
& C^k \geq C^{k+1} && k \in \{1, \ldots, |K| - 1\} && \text{(5e)}\\
& \lambda^{kp} \in \{0, 1\} && k \in K,\ p \in \hat{P} && \text{(5f)}
\end{aligned}
$$

Notation

$(\bar{\boldsymbol{\lambda}}, \bar{\mathbf{C}})$ Optimal fractional solution to the current linear relaxation of (2)

$\bar{P}$ Set of paths associated with variables $\bar{\boldsymbol{\lambda}} > 0$

$\hat{P}$ Superset of $\bar{P}$ defined according to some criteria

$$
\begin{aligned}
C^U = \min \quad & C^1 & & & (5a)\\
s.t. \quad & \sum_{k \in K} \sum_{p \in \hat{P}} a_i^p \lambda^{kp} \geq 1 & & i \in N & (5b)\\
& \sum_{p \in \hat{P}} \lambda^{kp} \leq 1 & & k \in K & (5c)\\
& \sum_{p \in \hat{P}} c^p \lambda^{kp} \leq C^k & & k \in K & (5d)\\
& C^k \geq C^{k+1} & & k \in \{1, \ldots, |K| - 1\} & (5e)\\
& \lambda^{kp} \in \{0, 1\} & & k \in K,\ p \in \hat{P} & (5f)
\end{aligned}
$$

Improving UB values allow to:

- restrict the PPs' feasible region, potentially constraining the LB to improve;
- speed up the solution algorithm for the PPs.

Given:

prop Property that uniquely determines a representative vertex for each path $p \in P$ (e.g., "**Property 1**: *The last customer vertex visited before reaching directly the depot vertex* $n+1$.", "**Property 2**: *The first customer vertex reached directly after leaving the depot vertex* 0.", etc.)

$f^{prop}: N \times P \rightarrow \mathbb{B}$ $f^{prop}(i,p) = 1$ iff vertex i is the representative vertex of path p according to property *prop*

Path-based formulation

Given:

prop Property that uniquely determines a representative vertex for each path $p \in P$ (e.g., "**Property 1**: *The last customer vertex visited before reaching directly the depot vertex* $n+1$.", "**Property 2**: *The first customer vertex reached directly after leaving the depot vertex* 0.", etc.)

$f^{prop}: N \times P \rightarrow \mathbb{B}$ $f^{prop}(i,p) = 1$ iff vertex i is the representative vertex of path p according to property *prop*

Variables:

$C \in \mathbb{R}_+$ Length of the longest path among those assigned to $|K|$ vehicles

$\lambda^p \in \{0,1\}$ Binary variable equal to 1 if path $p \in P$ is assigned to one of the $|K|$ vehicles

$$\min \quad C \tag{6a}$$

$$s.t. \quad \sum_{p \in P} a_i^p \lambda^p \geq 1 \qquad i \in N \tag{6b}$$

$$\sum_{p \in P} \lambda^p \leq |K| \tag{6c}$$

$$\sum_{p \in P: f^{prop}(i,p)=1} c^p \lambda^p \leq C \qquad i \in N \tag{6d}$$

$$\lambda^p \in \{0,1\} \qquad p \in P \tag{6e}$$

$$C^L \leq C \leq C^U \text{ (optional)} \tag{6f}$$

- A unique pricing problem (PP):

$$\min_{p \in P} \bar{c}_p(\boldsymbol{\mu}, \boldsymbol{\rho}, \theta)$$

where:

- $\bar{c}_p(\boldsymbol{\mu}, \boldsymbol{\rho}, \theta) = -\sum_{i \in N} a_i^p \mu_i - \sum_{i \in N} c^p \rho_i f^{prop}(i, p) - \theta$
- $\mu_i \in \mathbb{R}_+$, $\theta \in \mathbb{R}_-$, and $\rho_i \in \mathbb{R}_-$ are the dual variables associated with constraints (6b), (6c), and (6d), respectively, in the linear relaxation of (6).

The PP corresponds to an ESPPRC with linear length costs (that depend on the path's representative vertex) and possible side constraints.

- A unique pricing problem (PP):

$$\min_{p \in P} \bar{c}_p(\boldsymbol{\mu}, \boldsymbol{\rho}, \theta)$$

where:

  - $\bar{c}_p(\boldsymbol{\mu}, \boldsymbol{\rho}, \theta) = -\sum_{i \in N} a_i^p \mu_i - \sum_{i \in N} c^p \rho_i f^{prop}(i, p) - \theta$
  - $\mu_i \in \mathbb{R}_+$, $\theta \in \mathbb{R}_-$, and $\rho_i \in \mathbb{R}_-$ are the dual variables associated with constraints (6b), (6c), and (6d), respectively, in the linear relaxation of (6).

  The PP corresponds to an ESPPRC with linear length costs (that depend on the path's representative vertex) and possible side constraints.

- The model of the PP varies on the basis of the considered property *prop*. W.r.t. Property 1:

$$\min \; -\sum_{i \in N} \mu_i y_i - \sum_{i \in N} \rho_i T_{i,n+1} - \theta \tag{7a}$$

$$\sum_{j \in N} x_{0,j} = \sum_{i \in N} x_{i,n+1} = 1 \tag{7b}$$

$$\sum_{(j,i) \in \delta^-(i)} x_{ji} = \sum_{(i,j) \in \delta^+(i)} x_{ij} = y_i \qquad i \in N \tag{7c}$$

$$T_{0j} = t_{0j} x_{0j} \qquad j \in N \tag{7d}$$

$$\sum_{(i,j) \in \delta^+(i)} T_{ij} - \sum_{(j,i) \in \delta^-(i)} T_{ji} = \sum_{(i,j) \in \delta^+(i)} t_{ij} x_{ij} \qquad i \in N \tag{7e}$$

$$t_{ij}^0 x_{ij} \leq T_{ij} \leq C_{j,n+1}^U x_{ij} \qquad (i,j) \in A \tag{7f}$$

$$y_i \in \{0,1\} \qquad i \in N \tag{7g}$$

$$x_{ij} \in \{0,1\} \qquad (i,j) \in A \tag{7h}$$

We address the ng-path relaxation (Baldacci et al. [2011]) of the PP:

- Labels of the form $(t, \mathbf{B}, R, i)$. (Initial label $(0, \mathbf{0}, 0, 0)$ at vertex 0.)
- Extension of label $L = (t, \mathbf{B}, R, i)$ along arc $(i, j) \in V$:
  1. Feasibility check:
     - $t + t_{ij} \leq C^U - 1$
     - $(j = n+1)$ or $(j \notin N_i)$ or $(j \in N_i$ and $B(j) = 0)$
  2. Resulting label $L' = (t', \mathbf{B}', R', j)$:
     - $t' = t + t_{ij}$
     - $B'(u) = \begin{cases} 0 \text{ for all } u \in V \setminus \{0, n+1\} \text{ if } j = n+1 \\ 0 \text{ for all } u \notin N_i \cap N_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ B(u) \text{ for all } u \in N_i \cap N_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ 1 \text{ if } j \neq n+1 \text{ and } u = j \end{cases}$
     - $R' = \begin{cases} R - \mu_j & \text{if } j \neq n+1 \\ R - \rho_i t' & \text{if } j = n+1 \end{cases}$
- Dominance: $L' = (t', \mathbf{B}', R', j) \preceq L'' = (t'', \mathbf{B}'', R'', j)$ if $t' \leq t''$, $B'(u) \leq B''(u)$ for each $u \in N_i$, and $R' \leq R''$.

We address the ng-path relaxation (Baldacci et al. [2011]) of the PP:

- Labels of the form $(t, \mathbf{B}, R, i)$. (Initial label $(0, \mathbf{0}, 0, 0)$ at vertex 0.)
- Extension of label $L = (t, \mathbf{B}, R, i)$ along arc $(i, j) \in V$:
  1. Feasibility check:
     - $t + t_{ij} \leq C^U - 1$
     - $(j = n+1)$ or $(j \notin N_i)$ or $(j \in N_i$ and $B(j) = 0)$
  2. Resulting label $L' = (t', \mathbf{B}', R', j)$:
     - $t' = t + t_{ij}$
     - $B'(u) = \begin{cases} 0 \text{ for all } u \in V \setminus \{0, n+1\} \text{ if } j = n+1 \\ 0 \text{ for all } u \notin N_i \cap N_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ B(u) \text{ for all } u \in N_i \cap N_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ 1 \text{ if } j \neq n+1 \text{ and } u = j \end{cases}$
     - $R' = \begin{cases} R - \mu_j & \text{if } j \neq n+1 \\ R - \rho_i t' & \text{if } j = n+1 \end{cases}$
- Dominance: $L' = (t', \mathbf{B}', R', j) \preceq L'' = (t'', \mathbf{B}'', R'', j)$ if $t' \leq t''$, $B'(u) \leq B''(u)$ for each $u \in N_i$, and $R' \leq R''$.

Observation 2

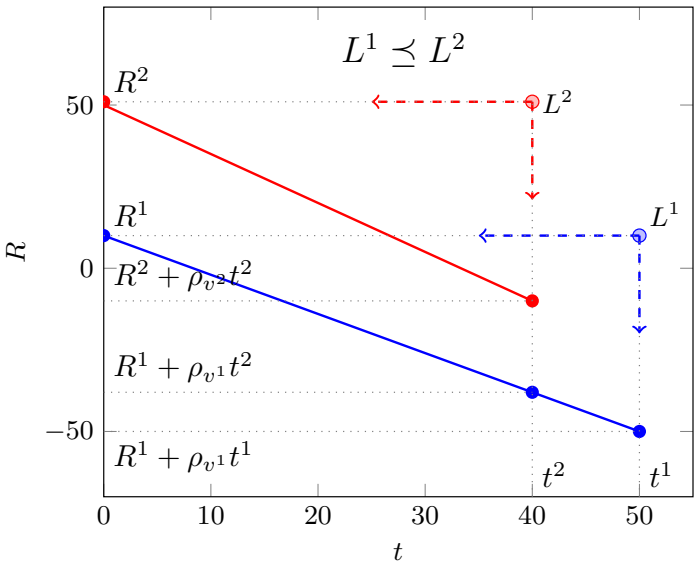
Information concerning the (i) representative vertex and (ii) final length of the path become known at the same time.

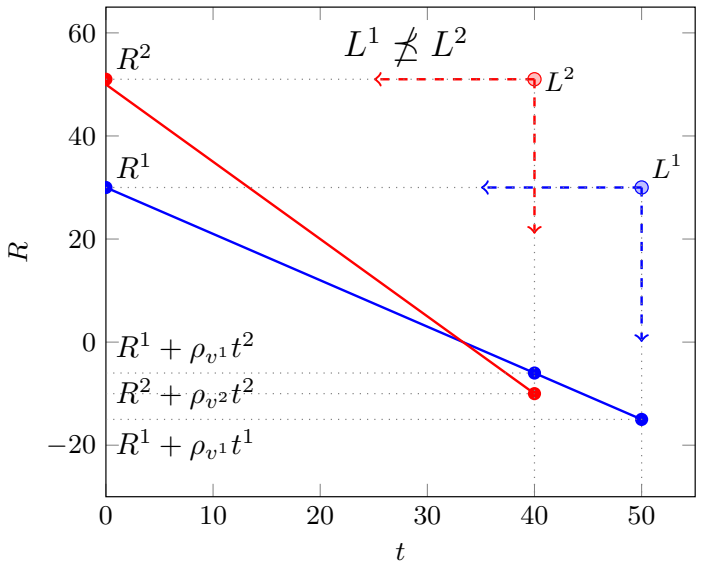
Backward search:

- Labels of the form $(t, \mathbf{B}, R, v, i)$.
  (Initial label $(C^U - 1, \mathbf{0}, 0, -1, n+1)$ at vertex $n+1$.)
- Extension of label $L = (t, \mathbf{B}, R, v, i)$ along arc $(i, j) \in V$:
  1. Feasibility check:
     - $t - t_{ij} \geq 0$
     - $(i = 0)$ or $(i \notin N_j)$ or $(i \in N_j$ and $B(i) = 0)$
  2. Resulting label $L' = (t', \mathbf{B}', R', v', j)$:
     - $t' = t - t_{ij}$
     - $B'(u) = \begin{cases} 0 \text{ for all } u \in V \setminus \{0, n+1\} \text{ if } i = 0 \\ 0 \text{ for all } u \notin N_i \cap N_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ B(u) \text{ for all } u \in N_i \cap N_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ 1 \text{ if } i \neq 0 \text{ and } u = i \end{cases}$
     - $v' = \begin{cases} i \text{ if } j = n+1 \\ v \text{ otherwise} \end{cases}$
     - $R' = \begin{cases} R - \rho_i(C^U - 1) \text{ if } j = n+1 \\ R - \mu_j \text{ if } i \neq 0 \text{ and } j \neq n+1 \\ R + \rho_v t' \text{ if } i = 0 \end{cases}$

Backward search (cont'd):

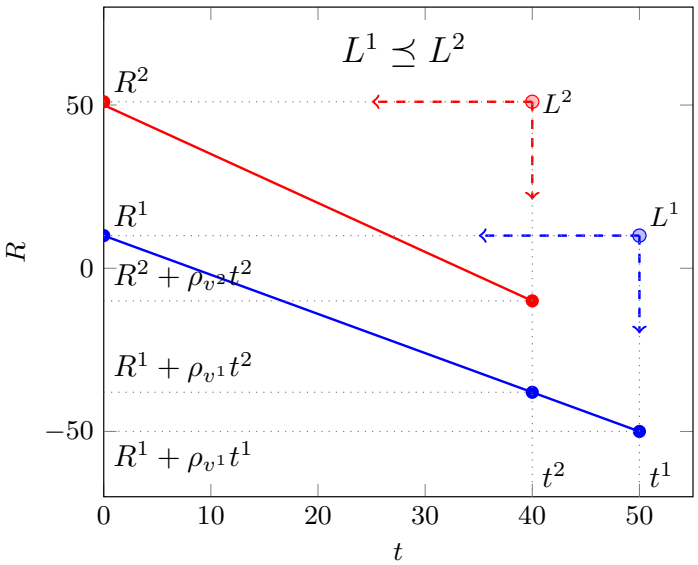
- Dominance: $L^1 = (t^1, \mathbf{B}^1, R^1, v^1, j) \preceq L^2 = (t^2, \mathbf{B}^2, R^2, v^2, j)$ if:
  - (i) $t^1 \geq t^2$
  - (ii) $B^1(u) \leq B^2(u)$ for each $u \in N_j$
  - (iii) $R^1 \leq R^2$
  - (iv) $R^1 + \rho_{v^1} t^2 \leq R^2 + \rho_{v^2} t^2$

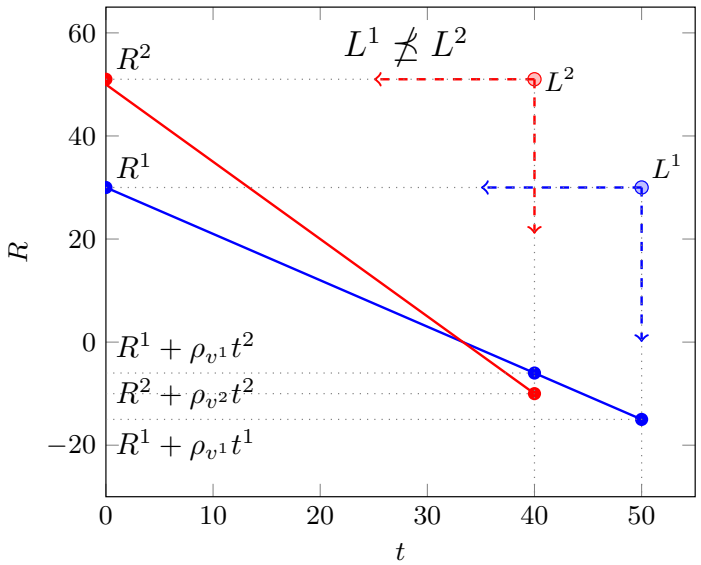




Backward search (cont'd):

- Dominance: $L^1 = (t^1, \mathbf{B}^1, R^1, v^1, j) \preceq L^2 = (t^2, \mathbf{B}^2, R^2, v^2, j)$ if:
  - (i) $t^1 \geq t^2$
  - (ii) $B^1(u) \leq B^2(u)$ for each $u \in N_j$
  - (iii) $R^1 \leq R^2$
  - (iv) $R^1 + \rho_{v^1} t^2 \leq R^2 + \rho_{v^2} t^2$





Merge of labels $L^f = (t^f, \mathbf{B}^f, R^f, i)$ and $L^b = (t^b, \mathbf{B}^b, R^b, v^b, i)$ at $i \in N$:

- Feasibility check: (i) $t^f \leq t^b$ and (ii) $B^f(u) + B^b(u) < 1$ for each $u \in N_i$, $u \neq i$
- Resulting reduced cost: $R^f + R^b + \rho_{v^b}(t^b - t^f)$

- `BP-M6` - The BP algorithm based on model (6) - Property 1 - in which:
  - the ng-path relaxation of the PP is solved (the size of the ng-neighborhoods is equal to 5);
  - the bi-directional search is applied on top of the label setting dynamic programming;
  - the RMH is applied at each node of the tree;
- `CPLEX-M1` - The CPLEX solver addressing model (1) plus constraints

$$t_{0,i}x_{0,i} + t_{i,n+1}x_{i,n+1} \leq C \quad i \in N \tag{8}$$

  and with parameters:
  - `IloCplex::Param::Threads=6`;
  - `IloCplex::ParallelMode=1`;

- `BP-M6` - The BP algorithm based on model (6) - Property 1 - in which:
  - the ng-path relaxation of the PP is solved (the size of the ng-neighborhoods is equal to 5);
  - the bi-directional search is applied on top of the label setting dynamic programming;
  - the RMH is applied at each node of the tree;
- `CPLEX-M1` - The CPLEX solver addressing model (1) plus constraints

$$t_{0,i}x_{0,i} + t_{i,n+1}x_{i,n+1} \leq C \quad i \in N \tag{8}$$

  and with parameters:
  - `IloCplex::Param::Threads=6`;
  - `IloCplex::ParallelMode=1`;

Constraints $C^L \leq C^1 \leq C^U$ are considered both in (6) and (1), with:

- $C^L = C_1^L = \max_{i \in N}(t_{0,i} + t_{i,n+1})$
- $C^U = C_1^U$

- We considered the instances of Set I proposed by [He and Hao, 2022]: 8 TSP instances (`att532`, `kroA200`, `lin318`, `mtsp100`, `mtsp150`, `mtsp51`, `pcb1173`, `rat783`) addressed with different values of $|K|$.
- For each of the TSP instances, we generated an instance considering the first $|N| \in \{10, 20, 30, 40, 50\}$ customers listed in the TSP instance, to address with values of $|K|$ such that (i) $3 \leq |K| \leq 10$ and (ii) $\frac{|N|}{|K|} \geq 3$:
  - Combinations:

$\|N\|$	Possible values of $\|K\|$
10	3
20	3,4,5,6
30	3,4,5,6,7,8,9,10
40	3,4,5,6,7,8,9,10
50	3,4,5,6,7,8,9,10

  - 29 combinations for a total of 232 ($29 \cdot 8$) instances.

- SW: C++/CPLEX 20.1; the code was compiled in release mode with Microsoft Visual Studio Community 2022 (64-bit) (Visual C++ 2022).
- HW: 64-bit Windows machine, with the Intel processor i7-6700K, 4.00 GHz, and 32 GB of RAM.
- Overall execution time limit: 900 seconds.

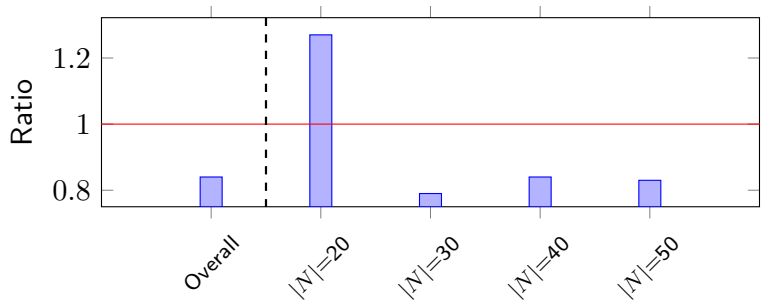
Analysis of BP-M6: Impact of bi-directional search

- BP-M6 compared against BP-M6-$\overline{\text{BI}}$.
- Solution of the linear relaxation of (6) with $C^L = 0$.

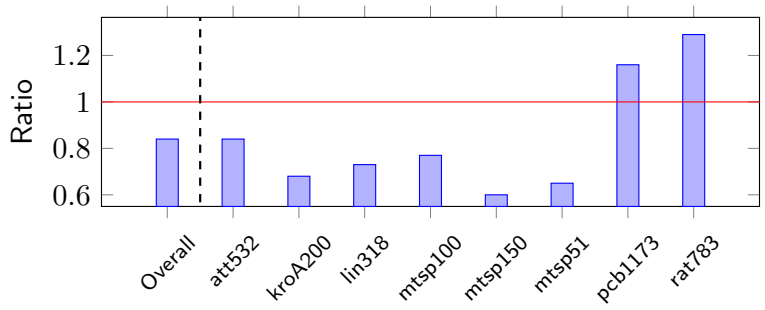
Analysis of BP-M6: Impact of bi-directional search

- BP-M6 compared against BP-M6-$\overline{\text{BI}}$.
- Solution of the linear relaxation of (6) with $C^L = 0$.

Geometric means of ratios of solution times $Sol.\ time^{\text{BP-M6}}/Sol.\ time^{\text{BP-M6-}\overline{\text{BI}}}$



(a) Categories defined w.r.t. $|N|$



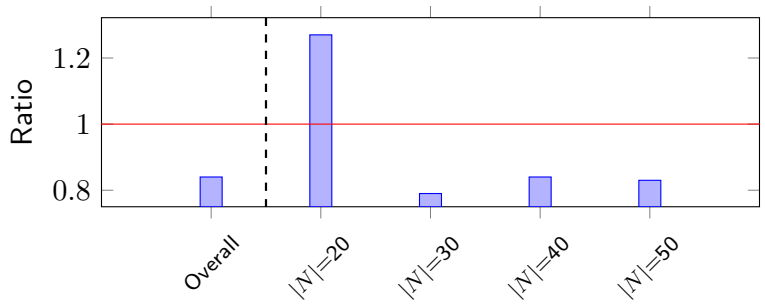
(b) Categories defined w.r.t. the original instances from which the benchmark instances are derived

($Sol.\ time^{\text{BP-M6-}\overline{\text{BI}}}$ or $Sol.\ time^{\text{BP-M6}}$ greater than 5 seconds - 155/232 instances)

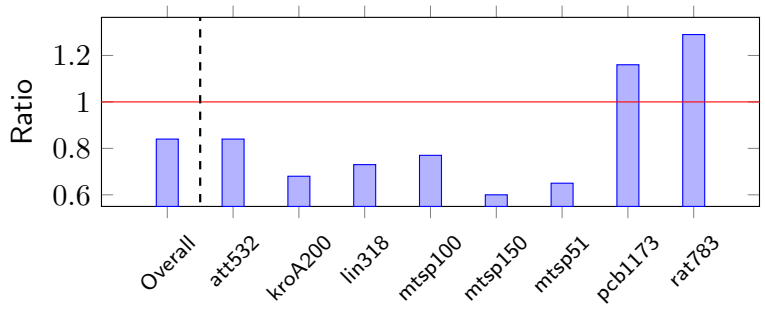
Analysis of BP-M6: Impact of bi-directional search

- BP-M6 compared against BP-M6-$\overline{\text{BI}}$.
- Solution of the linear relaxation of (6) with $C^L = 0$.

Geometric means of ratios of solution times $Sol.\ time^{\text{BP-M6}}/Sol.\ time^{\text{BP-M6-}\overline{\text{BI}}}$



(a) Categories defined w.r.t. $|N|$



(b) Categories defined w.r.t. the original instances from which the benchmark instances are derived

($Sol.\ time^{\text{BP-M6-}\overline{\text{BI}}}$ or $Sol.\ time^{\text{BP-M6}}$ greater than 5 seconds - 155/232 instances)

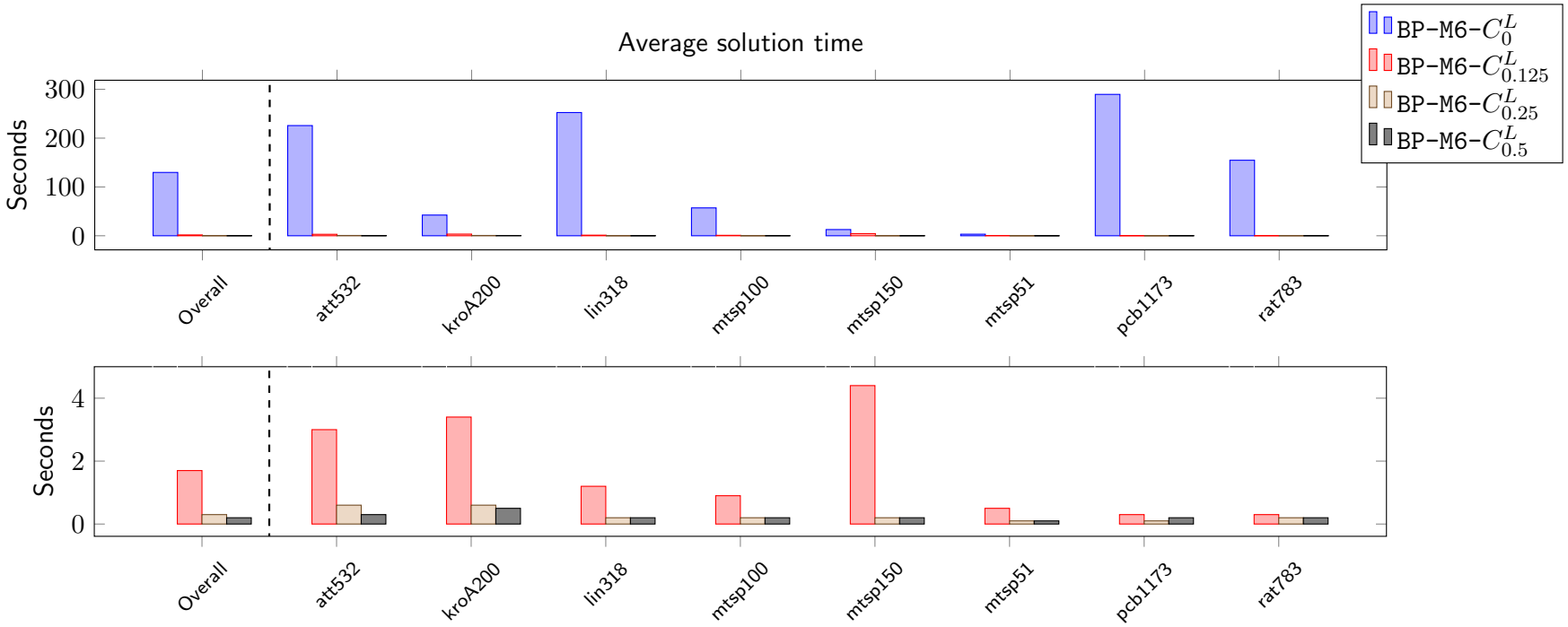
- Bi-directional search is beneficial.

Analysis of BP-M6: Impact of the initial C^L value

- Solution of the linear relaxation of (6) with $C^L = C^L_f = \lceil C^L_1 \cdot f \rceil$, $f \in \{0, 0.125, 0.25, 0.5\}$.

Analysis of BP-M6: Impact of the initial C^L value

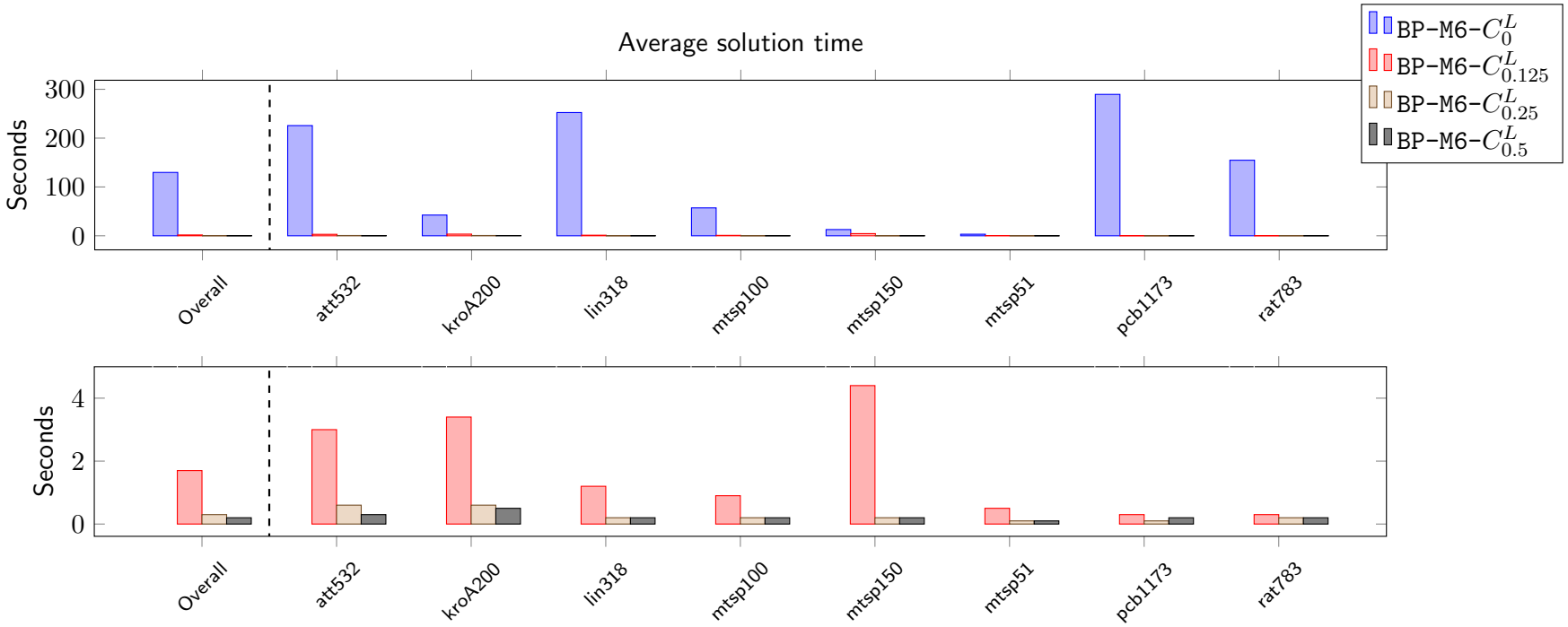
- Solution of the linear relaxation of (6) with $C^L = C^L_f = \lceil C^L_1 \cdot f \rceil$, $f \in \{0, 0.125, 0.25, 0.5\}$.



Categories defined w.r.t. the original instances from which the benchmark instances are derived

Analysis of BP-M6: Impact of the initial C^L value

- Solution of the linear relaxation of (6) with $C^L = C^L_f = \lceil C^L_1 \cdot f \rceil$, $f \in \{0, 0.125, 0.25, 0.5\}$.



Categories defined w.r.t. the original instances from which the benchmark instances are derived

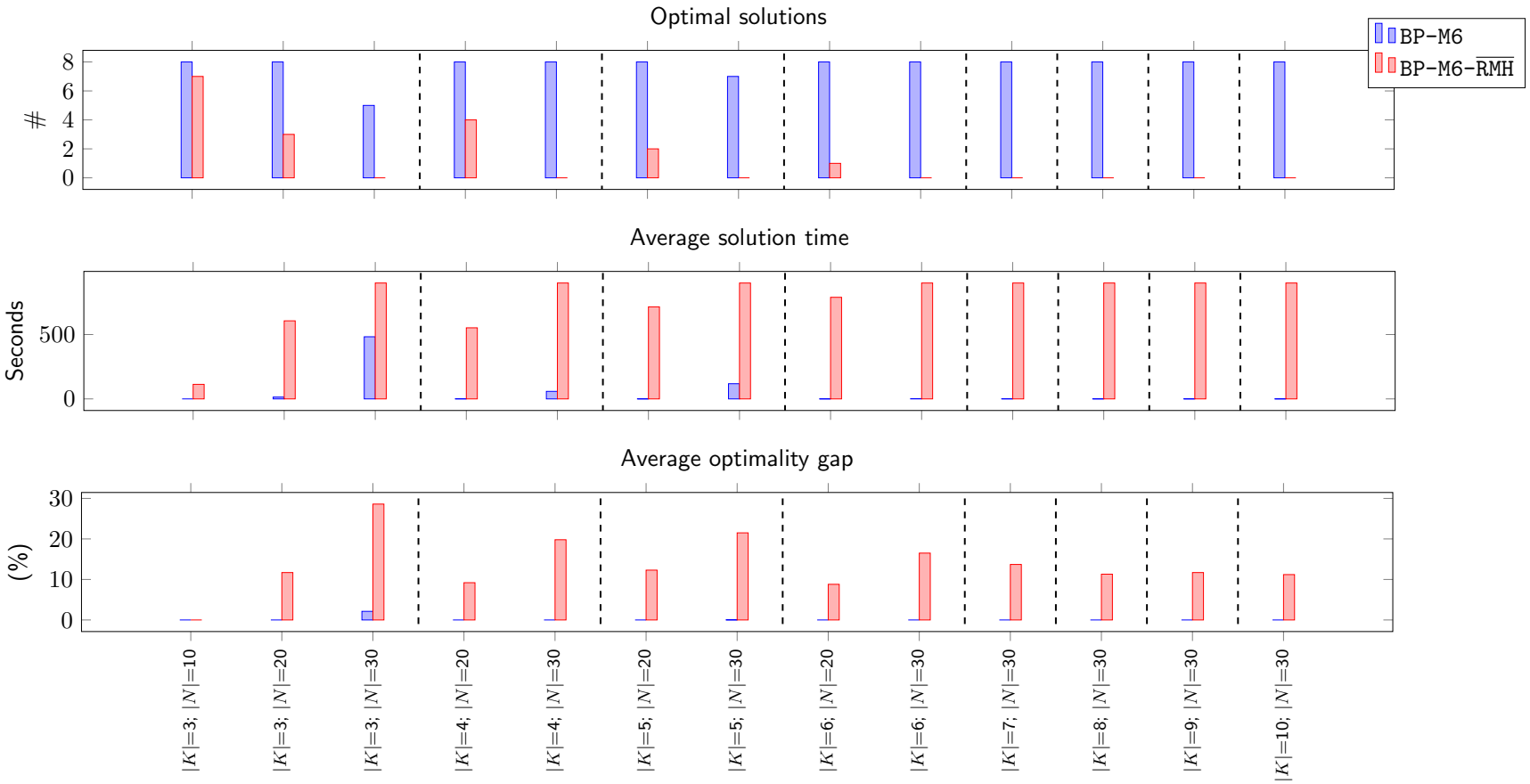
- Lower bound values have huge impacts on both (i) the column generation algorithm and (ii) the label setting dynamic programming algorithm.

Analysis of BP-M6: Impact of the restricted master heuristic (RMH)

- BP-M6 compared against BP-M6-$\overline{\text{RMH}}$.
- Subsets of instances with $|N| \leq 30$.

Analysis of BP-M6: Impact of the restricted master heuristic (RMH)

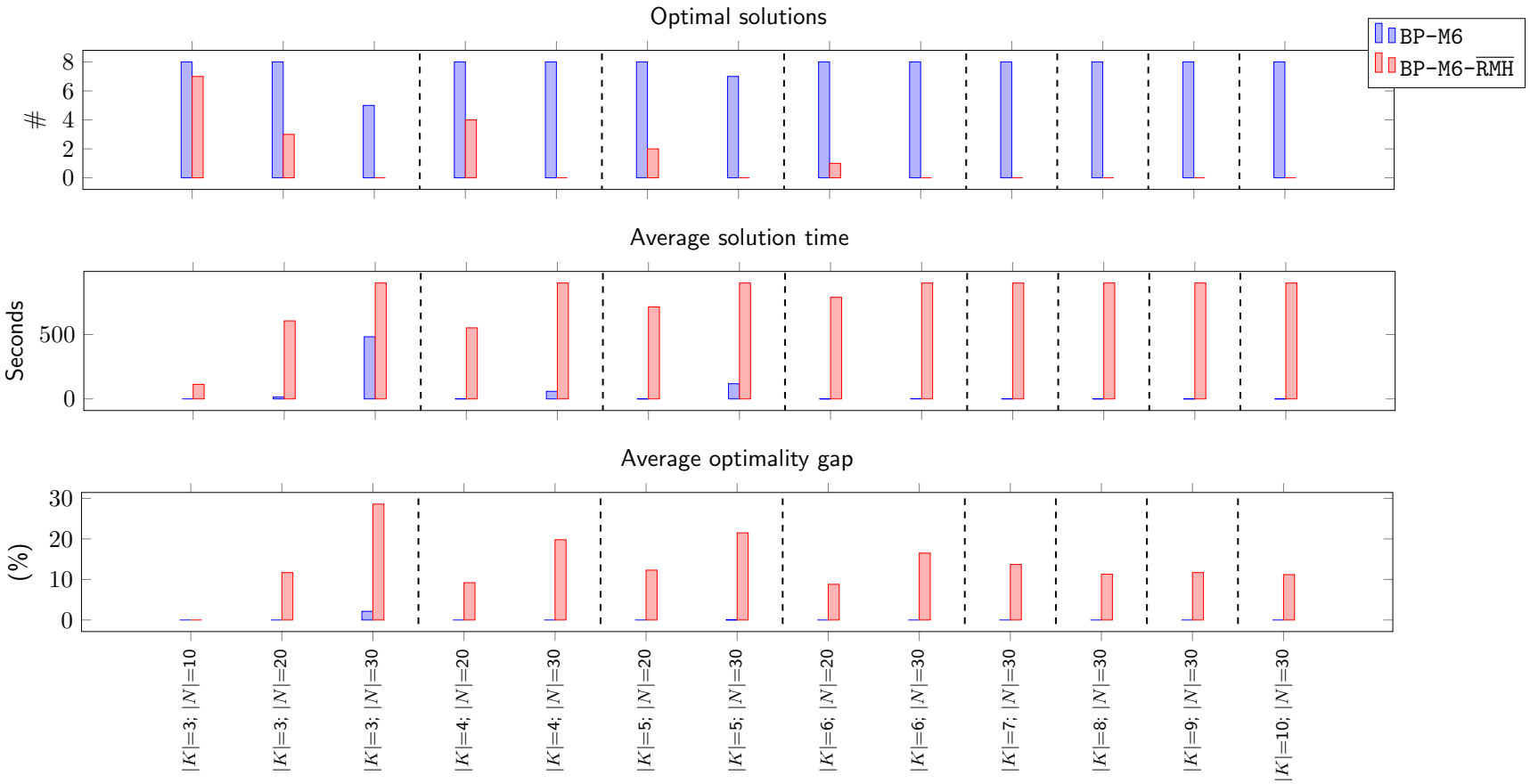
- BP-M6 compared against BP-M6-$\overline{\text{RMH}}$.
- Subsets of instances with $|N| \leq 30$.



Categories defined w.r.t. $|K|$; $|N|$ combinations

Analysis of BP-M6: Impact of the restricted master heuristic (RMH)

- BP-M6 compared against BP-M6-$\overline{\text{RMH}}$.
- Subsets of instances with $|N| \leq 30$.



Categories defined w.r.t. $|K|$; $|N|$ combinations

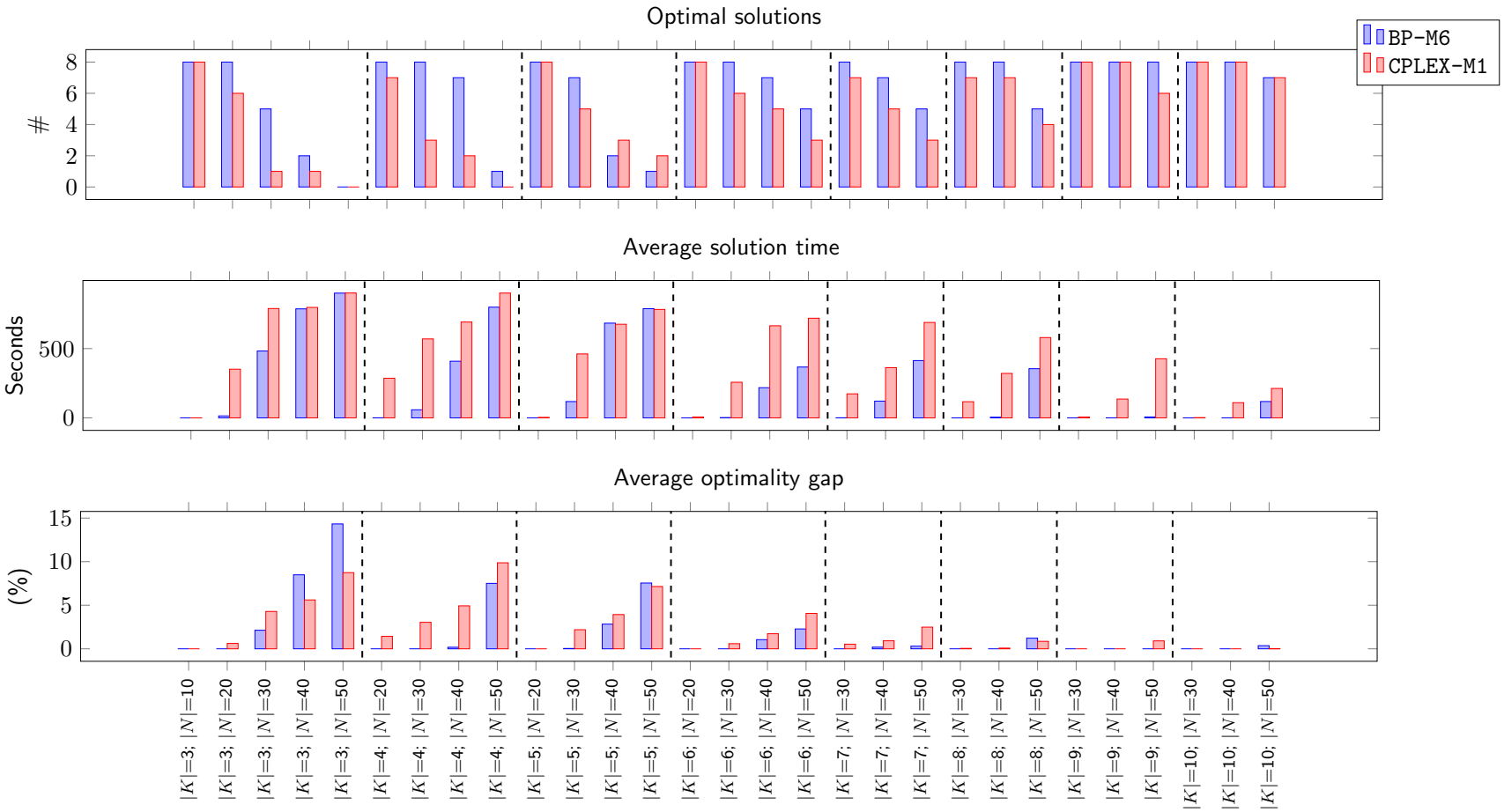
- Embedding primal bound heuristics into the BP algorithm is fundamental; RMHs come almost for free.

Assessment of BP-M6

- BP-M6 compared against CPLEX-M1.

Assessment of BP-M6

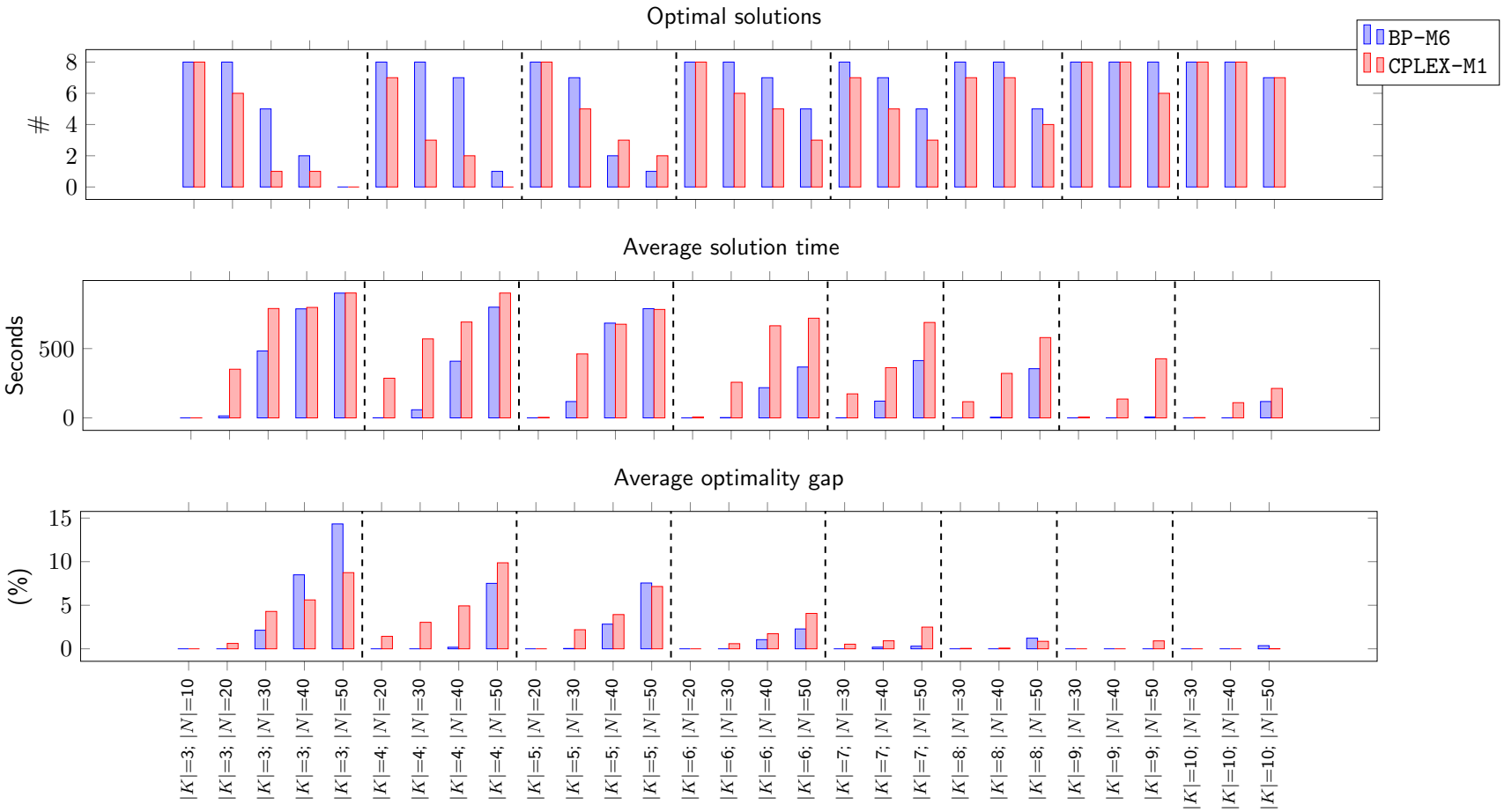
- BP-M6 compared against CPLEX-M1.



Categories defined w.r.t. $|K|$; $|N|$ combinations

Assessment of BP-M6

- BP-M6 compared against CPLEX-M1.



Categories defined w.r.t. $|K|$; $|N|$ combinations

- BP-M6 is superior to CPLEX-M1.

- Improved the state-of-the-art concerning the exact solution of the minmax mTSP.
- Identified what seem to be some key components/features to embed in BP algorithms addressing minmax VRPs.

- Improved the state-of-the-art concerning the exact solution of the minmax mTSP.
- Identified what seem to be some key components/features to embed in BP algorithms addressing minmax VRPs.
- Future work:
  - Implement/evaluate the BP algorithm based on model (2).
  - Implement/evaluate the BP algorithm based on model (6) - **Property 3**: *The representative vertex is the customer vertex at which merge is performed when applying the bi-directional search to solve the PP.*
    - (↑) Observation 2 applies for both the fw and bw search.
    - (↓) No more possible to consider dynamic half-way points in bi-directional search [Tilk et al., 2017].
    - (↓) The representative vertex of a path has to be kept consistent w.r.t. improving UB values.
  - Check for improving LB values at each node of the B&B tree on the basis of the arcs enforced by branching decisions.
  - ...

- Improved the state-of-the-art concerning the exact solution of the minmax mTSP.
- Identified what seem to be some key components/features to embed in BP algorithms addressing minmax VRPs.
- Future work:
  - Implement/evaluate the BP algorithm based on model (2).
  - Implement/evaluate the BP algorithm based on model (6) - **Property 3**: *The representative vertex is the customer vertex at which merge is performed when applying the bi-directional search to solve the PP.*
    - (↑) Observation 2 applies for both the fw and bw search.
    - (↓) No more possible to consider dynamic half-way points in bi-directional search [Tilk et al., 2017].
    - (↓) The representative vertex of a path has to be kept consistent w.r.t. improving UB values.
  - Check for improving LB values at each node of the B&B tree on the basis of the arcs enforced by branching decisions.
  - ...

... thanks for your attention!!!

R. Baldacci, A. Mingozzi, and R. Roberti. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, 59(5):1269–1283, 2011.

Enrique Benavent, Ángel Corberán, Guy Desaulniers, François Lessard, Isaac Plana, and José M. Sanchis. A branch-price-and-cut algorithm for the min-max k-vehicle windy rural postman problem. *Networks*, 63(1): 34–45, 2014. doi: 10.1002/net.21520.

Nicola Bianchessi and Emanuele Tresoldi. A stand-alone branch-and-price algorithm for identical parallel machine scheduling with conflicts. *Computers and Operations Research*, 136:105464, 2021. ISSN 0305-0548. doi: 10.1016/j.cor.2021.105464.

Nicola Bianchessi, Ángel Corberán, Isaac Plana, Miguel Reula, and José M. Sanchis. The min-max close-enough arc routing problem. *European Journal of Operational Research*, 300(3):837–851, 2022. ISSN 0377-2217. doi: 10.1016/j.ejor.2021.10.047.

Luciano Costa, Claudio Contardo, and Guy Desaulniers. Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science*, 53(4):946–985, 2019. doi: 10.1287/trsc.2018.0878.

M. Dror. Note on the complexity of the shortest path models for column generation in vrptw. *Operations Research*, 42:977–978, 1994.

Paulo M. França, Michel Gendreau, Gilbert Laporte, and Felipe M. Müller. The m-traveling salesman problem with minmax objective. *Transportation Science*, 29(3):267–275, 1995. doi: 10.1287/trsc.29.3.267.

Pengfei He and Jin-Kao Hao. Hybrid search with neighborhood reduction for the multiple traveling salesman problem. *Computers & Operations Research*, 142:105726, 2022. ISSN 0305-0548. doi: https://doi.org/10.1016/j.cor.2022.105726.

Pengfei He and Jin-Kao Hao. Memetic search for the minmax multiple traveling salesman problem with single and multiple depots. *European Journal of Operational Research*, 307(3):1055–1070, 2023. ISSN 0377-2217. doi: 10.1016/j.ejor.2022.11.010.

C. Joncour, S. Michel, R. Sadykov, D. Sverdlov, and F. Vanderbeck. Column generation based primal heuristics. *Electronic Notes in Discrete Mathematics*, 36:695–702, 2010.

Francesco Maffioli and Anna Sciomachen. A mixed-integer model for solving ordering problems with side constraints. *Annals of Operations Research*, 69(0):277–297, Jan 1997. ISSN 1572-9338. doi: 10.1023/A:1018989130169. URL https://doi.org/10.1023/A:1018989130169.

Giovanni Righini and Matteo Salani. Symmetry helps: Bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints. *Discrete Optimization*, 3(3):255–273, 2006.

References

Subhash C. Sarin, Hanif D. Sherali, Jason D. Judd, and Pei-Fang (Jennifer) Tsai. Multiple asymmetric traveling salesmen problem with and without precedence constraints: Performance comparison of alternative formulations. *Computers & Operations Research*, 51:64–89, 2014. ISSN 0305-0548. doi: 10.1016/j.cor.2014.05.014.

Banu Soylu. A general variable neighborhood search heuristic for multiple traveling salesmen problem. *Computers & Industrial Engineering*, 90:390–401, 2015. ISSN 0360-8352. doi: 10.1016/j.cie.2015.10.010.

Christian Tilk, Ann-Kathrin Rothenbächer, Timo Gschwind, and Stefan Irnich. Asymmetry matters: Dynamic half-way points in bidirectional labeling for solving shortest path problems with resource constraints faster. *European Journal of Operational Research*, 261(2):530–539, 2017. doi: 10.1016/j.ejor.2017.03.017.

P. Toth and D. Vigo, editors. *Vehicle Routing: Problems, Methods, and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2014.