An Exact Solution Framework for Multi-Trip Vehicle Routing Problems with Time Windows

Autumn School on Advanced BCP Tools

R. Roberti

Dept. of Supply Chain Analytics
Vrije Universiteit Amsterdam

November 21, 2019

Table of Contents

Table of Contents

Introduction

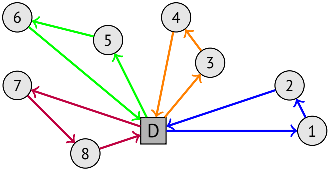
- Most of the literature on the Vehicle Routing Problem (VRP) addresses problems where each vehicle can perform at most one trip per day
- Many contributions on VRPs where vehicles can perform multiple trips have been published in the last decade
- These problems are called Multi-Trip Vehicle Routing Problems (MTVRP)

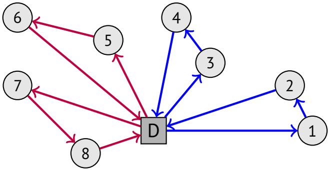
Introduction

- Most of the literature on the Vehicle Routing Problem (VRP) addresses problems where each vehicle can perform at most one trip per day
- Many contributions on VRPs where vehicles can perform multiple trips have been published in the last decade
- These problems are called Multi-Trip Vehicle Routing Problems (MTVRP)

Introduction

- Most of the literature on the Vehicle Routing Problem (VRP) addresses problems where each vehicle can perform at most one trip per day
- Many contributions on VRPs where vehicles can perform multiple trips have been published in the last decade
- These problems are called Multi-Trip Vehicle Routing Problems (MTVRP)





Motivation

- Such an increasing interest in MTVRPs is due to new practices in, e.g., city logistics and last-mile delivery
- The need of limiting noise and pollution in city centers requires the usage of small vans, electric vehicles, and/or drones and forbids large trucks from entering city centers
- The limited capacity/autonomy of these vehicles forces them to perform multiple trips and to return to the depot to reload multiple times over the day

Motivation

- Such an increasing interest in MTVRPs is due to new practices in, e.g., city logistics and last-mile delivery
- The need of limiting noise and pollution in city centers requires the usage of small vans, electric vehicles, and/or drones and forbids large trucks from entering city centers
- The limited capacity/autonomy of these vehicles forces them to perform multiple trips and to return to the depot to reload multiple times over the day

Motivation

- Such an increasing interest in MTVRPs is due to new practices in, e.g., city logistics and last-mile delivery
- The need of limiting noise and pollution in city centers requires the usage of small vans, electric vehicles, and/or drones and forbids large trucks from entering city centers
- The limited capacity/autonomy of these vehicles forces them to perform multiple trips and to return to the depot to reload multiple times over the day

Research Question

Main Research Question to Address in this Talk

What is the best model to solve an MTVRP (with side constraints) to optimality?

Research Question

Main Research Question to Address in this Talk

What is the best model to solve an MTVRP (with side constraints) to optimality?

Based on the state-of-the-art exact methods for lots of VRPs...

Set Partitioning Models!

Table of Contents

Definition of the Multi-Trip VRP I

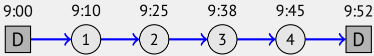
Input Data

- N set of customers
- V vertex set, $V = N \cup \{0\}$, where 0 is the depot
- $\mathcal{A}$ arc set, $\mathcal{A} = \{(i,j) \mid i,j \in V : i \neq j\}$
- $\mathcal{G}$ directed graph, $\mathcal{G} = (V, \mathcal{A})$
- t_{ij} travel time of arc $(i,j) \in \mathcal{A}$
- K fleet of identical capacitated vehicles, $|K| = m$
- q_i demand of customer $i \in N$
- Q vehicle capacity
- T length of the planning horizon

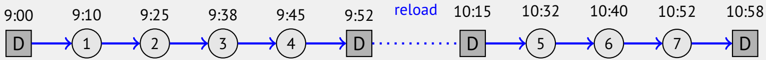
Definition of the Multi-Trip VRP II

Definitions

- A trip is a sequence of customers, whose total demand does not exceed Q, that can be visited by a vehicle in between two visits at the depot, and that has a fixed departure time from the depot



- A journey is a sequence of non-overlapping trips assigned to a vehicle whose total travel time does not exceed T



The MTVRP aims at defining a set of at most m journeys such that:

1. each customer is visited exactly once
2. the total traveled time is minimized

Table of Contents

Models with 3- and 4-index Variables

4-index Variables

$x_{ij}^{kh} \in \{0,1\}$ equal to 1 if trip h of vehicle $k \in K$ traverses arc $(i,j) \in \mathcal{A}$ (0 otherwise)

3-index Variables with Vehicle Index (without Trip Index)

$x_{ij}^{k} \in \{0,1\}$ equal to 1 if vehicle $k \in K$ traverses arc $(i,j) \in \mathcal{A}$ (0 otherwise)

3-index Variables with Trip Index (without Vehicle Index)

$x_{ij}^{h} \in \{0,1\}$ equal to 1 if trip h traverses arc $(i,j) \in \mathcal{A}$ (0 otherwise)

Models with 3- and 4-index Variables

4-index Variables

$x_{ij}^{kh} \in \{0, 1\}$ equal to 1 if trip h of vehicle $k \in K$ traverses arc $(i, j) \in \mathcal{A}$ (0 otherwise)

3-index Variables with Vehicle Index (without Trip Index)

$x_{ij}^{k} \in \{0, 1\}$ equal to 1 if vehicle $k \in K$ traverses arc $(i, j) \in \mathcal{A}$ (0 otherwise)

3-index Variables with Trip Index (without Vehicle Index)

$x_{ij}^{h} \in \{0, 1\}$ equal to 1 if trip h traverses arc $(i, j) \in \mathcal{A}$ (0 otherwise)

Models with 3- and 4-index Variables

4-index Variables

$x_{ij}^{kh} \in \{0, 1\}$ equal to 1 if trip h of vehicle $k \in K$ traverses arc $(i, j) \in \mathcal{A}$ (0 otherwise)

3-index Variables with Vehicle Index (without Trip Index)

$x_{ij}^{k} \in \{0, 1\}$ equal to 1 if vehicle $k \in K$ traverses arc $(i, j) \in \mathcal{A}$ (0 otherwise)

3-index Variables with Trip Index (without Vehicle Index)

$x_{ij}^{h} \in \{0, 1\}$ equal to 1 if trip h traverses arc $(i, j) \in \mathcal{A}$ (0 otherwise)

Models with 3- and 4-index Variables

Pros and Cons

- Polynomial number of variables
- Can be solved with commercial solvers
- Easy to embed additional side constraints

Models with 3- and 4-index Variables

Pros and Cons

- Polynomial number of variables
- Can be solved with commercial solvers
- Easy to embed additional side constraints

- High integrality gaps
- BigM constraints
- Symmetries in the vehicles

2-Index Arc-based Model (Koc and Karaoglan (2011)) I

Variables

$x_{ij} \in \{0,1\}$ equal to 1 if arc $(i,j) \in \mathcal{A}$ is traversed (0 otherwise)

$x'_{ij} \in \{0,1\}$ equal to 1 if a vehicle visits customers $i,j \in N$ $(i \neq j)$ consecutively with a stop at the depot in between (0 otherwise)

$\ell_i \in \mathbb{R}_+$ load on board after visiting customer $i \in N$

$a_i \in \mathbb{R}_+$ arrival time at customer $i \in N$

2-Index Arc-based Model (Koc and Karaoglan (2011)) II

$\min \sum_{(i,j)\in\mathcal{A}} t_{ij}x_{ij}$		[Minimize travel times]	(1a)
s.t. $\sum_{(i,j)\in\mathcal{A}} x_{ij} = 1$	$i \in N$	[Serve each customer]	(1b)
$\sum_{(i,j)\in\mathcal{A}} x_{ij} = \sum_{(j,i)\in\mathcal{A}} x_{ji}$	$i \in V$	[Flow conservation]	(1c)
$\ell_i + q_j \leq \ell_j + Q(1 - x_{ij})$	$i \in N\ j \in V$	[Subtour + Load on board]	(1d)
$a_i + t_{ij} \leq a_j + T(1 - x_{ij})$	$i \in V\ j \in N$	[Subtour + Arrival time]	(1e)
$a_i + (t_{i0} + t_{0j}) \leq a_j + T(1 - x'_{ij})$	$i, j \in N : i \neq j$	[Arrival time depot visit]	(1f)
$t_{0i} \leq a_i \leq T - t_{i0}$	$i \in N$	[Planning horizon]	(1g)
$\sum_{j\in N} x'_{ij} \leq x_{i0}$	$i \in N$	[Link x with x']	(1h)
$\sum_{j\in N} x'_{ij} \leq x_{0j}$	$j \in N$	[Link x with x']	(1i)
$\sum_{(0,j)\in\mathcal{A}} x_{0j} - \sum_{i,j\in N : i\neq j} x'_{ij} \leq m$		[Number of vehicles]	(1j)
$x_{ij} \in \{0, 1\}$	$(i, j) \in \mathcal{A}$		(1k)
$x'_{ij} \in \{0, 1\}$	$i, j \in N : i \neq j$		(1l)
$q_i \leq \ell_i \leq Q, \quad a_i \in \mathbb{R}_+$	$i \in N$		(1m)

2-Index Arc-based Model (Koc and Karaoglan (2011))

Pros and Cons

- Polynomial number of variables (much fewer than 3- and 4-index models)
- Can be solved with commercial solvers
- Easy to embed side constraints

2-Index Arc-based Model (Koc and Karaoglan (2011))

Pros and Cons

- Polynomial number of variables (much fewer than 3- and 4-index models)
- Can be solved with commercial solvers
- Easy to embed side constraints

- High integrality gaps
- BigM constraints
- Instances with 50 customers are already difficult to close

Trip-based Model (Mingozzi, Roberti, and Toth (2013))

$\mathcal{H}$ set of all feasible trips

c_h cost of trip $h \in \mathcal{H}$

α_{ih} trip $h \in \mathcal{H}$ serves customer $i \in N$ ($\alpha_{ih} = 1$) or not ($\alpha_{ih} = 0$)

d_h duration of trip $h \in \mathcal{H}$

Variables

$x_{hk} \in \{0, 1\}$ trip $h \in \mathcal{H}$ is assigned to vehicle $k \in K$ ($x_{hk} = 1$) or not ($x_{hk} = 0$)

$$\min \sum_{h \in \mathcal{H}} c_h \sum_{k \in K} x_{hk} \quad \text{[Minimize travel costs]} \tag{2a}$$

$$\text{s.t.} \sum_{h \in \mathcal{H}} \sum_{k \in K} \alpha_{ih} x_{hk} = 1 \qquad i \in N \quad \text{[Serve each customer]} \tag{2b}$$

$$\sum_{h \in \mathcal{H}} d_h x_{hk} \leq T \qquad k \in K \quad \text{[Planning horizon]} \tag{2c}$$

$$x_{hk} \in \{0, 1\} \qquad h \in \mathcal{H} \quad k \in K \tag{2d}$$

Trip-based Model (Mingozzi, Roberti, and Toth (2013))

$\mathcal{H}$ set of all feasible trips

c_h cost of trip $h \in \mathcal{H}$

α_{ih} trip $h \in \mathcal{H}$ serves customer $i \in N$ ($\alpha_{ih} = 1$) or not ($\alpha_{ih} = 0$)

d_h duration of trip $h \in \mathcal{H}$

Variables

$x_{hk} \in \{0, 1\}$ trip $h \in \mathcal{H}$ is assigned to vehicle $k \in K$ ($x_{hk} = 1$) or not ($x_{hk} = 0$)

$$\min \sum_{h\in\mathcal{H}} c_h \sum_{k\in K} x_{hk} \quad \text{[Minimize travel costs]} \tag{2a}$$

$$\text{s.t.} \sum_{h\in\mathcal{H}} \sum_{k\in K} \alpha_{ih} x_{hk} = 1 \qquad i \in N \quad \text{[Serve each customer]} \tag{2b}$$

$$\sum_{h\in\mathcal{H}} d_h x_{hk} \leq T \qquad k \in K \quad \text{[Planning horizon]} \tag{2c}$$

$$x_{hk} \in \{0, 1\} \qquad h \in \mathcal{H} \quad k \in K \tag{2d}$$

Trip-based Model (Mingozzi, Roberti, and Toth (2013))

$\mathcal{H}$ set of all feasible trips

c_h cost of trip $h \in \mathcal{H}$

α_{ih} trip $h \in \mathcal{H}$ serves customer $i \in N$ ($\alpha_{ih} = 1$) or not ($\alpha_{ih} = 0$)

d_h duration of trip $h \in \mathcal{H}$

Variables

$x_{hk} \in \{0, 1\}$ trip $h \in \mathcal{H}$ is assigned to vehicle $k \in K$ ($x_{hk} = 1$) or not ($x_{hk} = 0$)

$$\min \sum_{h \in \mathcal{H}} c_h \sum_{k \in K} x_{hk} \qquad \text{[Minimize travel costs]} \tag{2a}$$

$$\text{s.t.} \sum_{h \in \mathcal{H}} \sum_{k \in K} \alpha_{ih} x_{hk} = 1 \qquad i \in N \qquad \text{[Serve each customer]} \tag{2b}$$

$$\sum_{h \in \mathcal{H}} d_h x_{hk} \leq T \qquad k \in K \qquad \text{[Planning horizon]} \tag{2c}$$

$$x_{hk} \in \{0, 1\} \qquad h \in \mathcal{H} \quad k \in K \tag{2d}$$

Trip-based Model (Mingozzi, Roberti, and Toth (2013))

Pros and Cons

- Small integrality gaps
- Instances with 100-120 customers can be closed
- Easy to embed side constraints

Trip-based Model (Mingozzi, Roberti, and Toth (2013))

Pros and Cons

- Small integrality gaps
- Instances with 100-120 customers can be closed
- Easy to embed side constraints

- Exponential number of variables
- Symmetries in the vehicles
- Column generation/branch(-and-cut)-and-price needed
- Additional constraints can make the pricing problem difficult

Journey-based Model (Mingozzi, Roberti, and Toth (2013))

$\mathcal{R}$ set of all feasible journeys

c_r cost of journey $r \in \mathcal{R}$

α_{ir} journey $r \in \mathcal{R}$ serves customer $i \in N$ ($\alpha_{ir} = 1$) or not ($\alpha_{ir} = 0$)

Variables

$x_r \in \{0, 1\}$ journey $r \in \mathcal{R}$ is selected ($x_r = 1$) or not ($x_r = 0$)

$$\min \sum_{r \in \mathcal{R}} c_r x_r \qquad \text{[Minimize travel costs]} \qquad (3a)$$

$$\text{s.t.} \sum_{r \in \mathcal{R}} \alpha_{ir} x_r = 1 \quad i \in N \qquad \text{[Serve each customer]} \qquad (3b)$$

$$\sum_{r \in \mathcal{R}} x_r \leq m \qquad \text{[Number of vehicles]} \qquad (3c)$$

$$x_r \in \{0, 1\} \quad r \in \mathcal{R} \qquad (3d)$$

Journey-based Model (Mingozzi, Roberti, and Toth (2013))

$\mathcal{R}$ set of all feasible journeys

c_r cost of journey $r \in \mathcal{R}$

α_{ir} journey $r \in \mathcal{R}$ serves customer $i \in N$ ($\alpha_{ir} = 1$) or not ($\alpha_{ir} = 0$)

Variables

$x_r \in \{0, 1\}$ journey $r \in \mathcal{R}$ is selected ($x_r = 1$) or not ($x_r = 0$)

$$\min \sum_{r \in \mathcal{R}} c_r x_r \qquad \text{[Minimize travel costs]} \tag{3a}$$

$$\text{s.t.} \sum_{r \in \mathcal{R}} \alpha_{ir} x_r = 1 \quad i \in N \qquad \text{[Serve each customer]} \tag{3b}$$

$$\sum_{r \in \mathcal{R}} x_r \leq m \qquad \text{[Number of vehicles]} \tag{3c}$$

$$x_r \in \{0, 1\} \quad r \in \mathcal{R} \tag{3d}$$

Journey-based Model (Mingozzi, Roberti, and Toth (2013))

$\mathcal{R}$ set of all feasible journeys

c_r cost of journey $r \in \mathcal{R}$

α_{ir} journey $r \in \mathcal{R}$ serves customer $i \in N$ ($\alpha_{ir} = 1$) or not ($\alpha_{ir} = 0$)

Variables

$x_r \in \{0, 1\}$ journey $r \in \mathcal{R}$ is selected ($x_r = 1$) or not ($x_r = 0$)

$$\min \sum_{r \in \mathcal{R}} c_r x_r \qquad \text{[Minimize travel costs]} \tag{3a}$$

$$\text{s.t.} \sum_{r \in \mathcal{R}} \alpha_{ir} x_r = 1 \quad i \in N \qquad \text{[Serve each customer]} \tag{3b}$$

$$\sum_{r \in \mathcal{R}} x_r \leq m \qquad \text{[Number of vehicles]} \tag{3c}$$

$$x_r \in \{0, 1\} \qquad r \in \mathcal{R} \tag{3d}$$

Journey-based Model (Mingozzi, Roberti, and Toth (2013))

Pros and Cons

- Small integrality gaps (smaller than trip-based model)
- Instances with 100-120 customers can be closed
- Easy to embed additional side constraints

Journey-based Model (Mingozzi, Roberti, and Toth (2013))

Pros and Cons

- Small integrality gaps (smaller than trip-based model)
- Instances with 100-120 customers can be closed
- Easy to embed additional side constraints

- Exponential number of variables
- Column generation/branch(and-cut)-and-price needed
- Pricing problem more difficult than trip-based model
- Additional constraints can make the pricing problem (even more) difficult

Table of Contents

Main Side Constraints and Academic Extensions

- Time Windows: each customer $i \in N$ must be visited within a time interval $[a_i, b_i]$
- Service-Dependent Loading Times: vehicle loading time at the depot depends on the customers visited in the next trip
- Limited Trip Duration: maximum time between the departure from the depot and the arrival time at the last customer of the trip
- Profits: a profit p_i is associated with each customer $i \in N$; hierarchical objective function: maximize profit first; minimize routing cost second

Main Side Constraints and Academic Extensions

- Time Windows: each customer $i \in N$ must be visited within a time interval $[a_i, b_i]$
- Service-Dependent Loading Times: vehicle loading time at the depot depends on the customers visited in the next trip
- Limited Trip Duration: maximum time between the departure from the depot and the arrival time at the last customer of the trip
- Profits: a profit p_i is associated with each customer $i \in N$; hierarchical objective function: maximize profit first; minimize routing cost second

Main Side Constraints and Academic Extensions

- Time Windows: each customer $i \in N$ must be visited within a time interval $[a_i, b_i]$
- Service-Dependent Loading Times: vehicle loading time at the depot depends on the customers visited in the next trip
- Limited Trip Duration: maximum time between the departure from the depot and the arrival time at the last customer of the trip
- Profits: a profit p_i is associated with each customer $i \in N$; hierarchical objective function: maximize profit first; minimize routing cost second

Main Side Constraints and Academic Extensions

- Time Windows: each customer $i \in N$ must be visited within a time interval $[a_i, b_i]$
- Service-Dependent Loading Times: vehicle loading time at the depot depends on the customers visited in the next trip
- Limited Trip Duration: maximum time between the departure from the depot and the arrival time at the last customer of the trip
- Profits: a profit p_i is associated with each customer $i \in N$; hierarchical objective function: maximize profit first; minimize routing cost second

Main Side Constraints and Academic Extensions

Reference	Time Windows	Service-Dependent Loading Times	Limited Trip Duration	Profits
Exact Methods				
Azi, Gendreau, and Potvin (2010)	✓	✓	✓	✓
Macedo et al. (2011)	✓	✓	✓	✓
Hernandez et al. (2014)	✓	✓	✓	
Hernandez et al. (2016)	✓	✓		
Heuristic Methods				
Azi, Gendreau, and Potvin (2014)	✓	✓	✓	✓
Wang, Liang, and Hu (2014)	✓	✓	✓	✓
Cattaruzza, Absi, and Feillet (2016a)	✓	✓		
Anaya-Arenas et al. (2016)	✓		✓	

From Cattaruzza, Absi, and Feillet (2016b)

Table of Contents

Trip-based Model (Hernandez et al. (2016))

$\mathcal{H}$ set of all feasible trips

c_h cost of trip $h \in \mathcal{H}$

α_{ih} trip $h \in \mathcal{H}$ serves customer $i \in N$ ($\alpha_{ih} = 1$) or not ($\alpha_{ih} = 0$)

τ_{th} trip $h \in \mathcal{H}$ is active at time $t \in [a_0, b_0]$ ($\tau_{th} = 1$) or not ($\tau_{th} = 0$)

Variables

$x_h \in \{0,1\}$ trip $h \in \mathcal{H}$ is selected ($x_h = 1$) or not ($x_h = 0$)

$$\min \sum_{h \in \mathcal{H}} c_h x_h \qquad \text{[Minimize travel costs]} \tag{4a}$$

$$\text{s.t.} \sum_{h \in \mathcal{H}} \alpha_{ih} x_h = 1 \qquad i \in N \qquad \text{[Serve each customer]} \tag{4b}$$

$$\sum_{h \in \mathcal{H}} \tau_{th} x_h \leq m \qquad t \in [a_0, b_0] \qquad \text{[No overlaps]} \tag{4c}$$

$$x_h \in \{0,1\} \qquad h \in \mathcal{H} \tag{4d}$$

Trip-based Model (Hernandez et al. (2016))

$\mathcal{H}$ set of all feasible trips

c_h cost of trip $h \in \mathcal{H}$

α_{ih} trip $h \in \mathcal{H}$ serves customer $i \in N$ ($\alpha_{ih} = 1$) or not ($\alpha_{ih} = 0$)

τ_{th} trip $h \in \mathcal{H}$ is active at time $t \in [a_0, b_0]$ ($\tau_{th} = 1$) or not ($\tau_{th} = 0$)

Variables

$x_h \in \{0, 1\}$ trip $h \in \mathcal{H}$ is selected ($x_h = 1$) or not ($x_h = 0$)

$$\min \sum_{h \in \mathcal{H}} c_h x_h \qquad \text{[Minimize travel costs]} \tag{4a}$$

$$\text{s.t.} \sum_{h \in \mathcal{H}} \alpha_{ih} x_h = 1 \qquad i \in N \qquad \text{[Serve each customer]} \tag{4b}$$

$$\sum_{h \in \mathcal{H}} \tau_{th} x_h \leq m \qquad t \in [a_0, b_0] \qquad \text{[No overlaps]} \tag{4c}$$

$$x_h \in \{0, 1\} \qquad h \in \mathcal{H} \tag{4d}$$

Trip-based Model (Hernandez et al. (2016))

$\mathcal{H}$ set of all feasible trips

c_h cost of trip $h \in \mathcal{H}$

α_{ih} trip $h \in \mathcal{H}$ serves customer $i \in N$ ($\alpha_{ih} = 1$) or not ($\alpha_{ih} = 0$)

τ_{th} trip $h \in \mathcal{H}$ is active at time $t \in [a_0, b_0]$ ($\tau_{th} = 1$) or not ($\tau_{th} = 0$)

Variables

$x_h \in \{0, 1\}$ trip $h \in \mathcal{H}$ is selected ($x_h = 1$) or not ($x_h = 0$)

$$\min \sum_{h \in \mathcal{H}} c_h x_h \qquad \text{[Minimize travel costs]} \tag{4a}$$

$$\text{s.t.} \sum_{h \in \mathcal{H}} \alpha_{ih} x_h = 1 \qquad i \in N \qquad \text{[Serve each customer]} \tag{4b}$$

$$\sum_{h \in \mathcal{H}} \tau_{th} x_h \leq m \qquad t \in [a_0, b_0] \qquad \text{[No overlaps]} \tag{4c}$$

$$x_h \in \{0, 1\} \qquad h \in \mathcal{H} \tag{4d}$$

Trip-based Model (Hernandez et al. (2016))

Pros and Cons

- Small integrality gaps
- Easy to embed additional side constraints defining the feasibility of the trips

Trip-based Model (Hernandez et al. (2016))

Pros and Cons

- Small integrality gaps
- Easy to embed additional side constraints defining the feasibility of the trips

- Exponential number of variables
- Column generation/branch(-and-cut)-and-price needed
- Side constraints make the pricing problem difficult
- Constraints (4c) to add in a cutting plane fashion
- Instances with 25 customers can be out of reach

Journey-based Model (Hernandez et al. (2014, 2016))

$\mathcal{R}$ set of all feasible journeys

c_r cost of journey $r \in \mathcal{R}$

α_{ir} journey $r \in \mathcal{R}$ serves customer $i \in N$ ($\alpha_{ir} = 1$) or not ($\alpha_{ir} = 0$)

Variables

$x_r \in \{0, 1\}$ journey $r \in \mathcal{R}$ is selected ($x_r = 1$) or not ($x_r = 0$)

$$\min \sum_{r \in \mathcal{R}} c_r x_r \quad \text{[Minimize travel costs]} \tag{5a}$$

$$\text{s.t.} \sum_{r \in \mathcal{R}} \alpha_{ir} x_r = 1 \quad i \in N \quad \text{[Serve each customer]} \tag{5b}$$

$$\sum_{r \in \mathcal{R}} x_r \leq m \quad \text{[Number of vehicles]} \tag{5c}$$

$$x_r \in \{0, 1\} \quad r \in \mathcal{R} \tag{5d}$$

Journey-based Model (Hernandez et al. (2014, 2016))

- $\mathcal{R}$ set of all feasible journeys
- c_r cost of journey $r \in \mathcal{R}$
- α_{ir} journey $r \in \mathcal{R}$ serves customer $i \in N$ ($\alpha_{ir} = 1$) or not ($\alpha_{ir} = 0$)

Variables

$x_r \in \{0, 1\}$ journey $r \in \mathcal{R}$ is selected ($x_r = 1$) or not ($x_r = 0$)

$$\min \sum_{r \in \mathcal{R}} c_r x_r \quad \text{[Minimize travel costs]} \tag{5a}$$

$$\text{s.t.} \sum_{r \in \mathcal{R}} \alpha_{ir} x_r = 1 \quad i \in N \quad \text{[Serve each customer]} \tag{5b}$$

$$\sum_{r \in \mathcal{R}} x_r \leq m \quad \text{[Number of vehicles]} \tag{5c}$$

$$x_r \in \{0, 1\} \quad r \in \mathcal{R} \tag{5d}$$

Journey-based Model (Hernandez et al. (2014, 2016))

$\mathcal{R}$ set of all feasible journeys

c_r cost of journey $r \in \mathcal{R}$

α_{ir} journey $r \in \mathcal{R}$ serves customer $i \in N$ ($\alpha_{ir} = 1$) or not ($\alpha_{ir} = 0$)

Variables

$x_r \in \{0, 1\}$ journey $r \in \mathcal{R}$ is selected ($x_r = 1$) or not ($x_r = 0$)

$$\min \sum_{r \in \mathcal{R}} c_r x_r \qquad \text{[Minimize travel costs]} \tag{5a}$$

$$\text{s.t.} \sum_{r \in \mathcal{R}} \alpha_{ir} x_r = 1 \quad i \in N \qquad \text{[Serve each customer]} \tag{5b}$$

$$\sum_{r \in \mathcal{R}} x_r \leq m \qquad \text{[Number of vehicles]} \tag{5c}$$

$$x_r \in \{0, 1\} \qquad r \in \mathcal{R} \tag{5d}$$

Journey-based Model (Hernandez et al. (2014, 2016))

Pros and Cons

- Small integrality gaps (smaller than trip-based model)
- Easy to embed additional side constraints both related to trips and journeys

Journey-based Model (Hernandez et al. (2014, 2016))

Pros and Cons

- Small integrality gaps (smaller than trip-based model)
- Easy to embed additional side constraints both related to trips and journeys

- Exponential number of variables
- Column generation/branch(and-cut)-and-price needed
- Pricing problem more difficult than trip-based model
- Instances with 25 customers can be out of reach

The Concept of Structure

Definition of Structure

A structure $s = (0, i_1, i_2, \ldots, i_{\mu_s}, 0)$ is an ordered set of μ_s customers that can be visited in between two visits at the depot and can start from the depot within time interval $[e_s, \ell_s]$, such that:

1. capacity constraints are satisfied
2. the duration d_s and the cost c_s are constant for each departure time from the depot within $[e_s, \ell_s]$
3. the duration d_s is the minimum duration to serve the set of customers in the given order

Structure-based Model (Paradiso et al. (2019))

$\mathcal{S}$ set of all feasible structures

c_s cost of structure $s \in \mathcal{S}$

α_{is} structure $s \in \mathcal{S}$ serves $i \in N$ ($\alpha_{is} = 1$) or not ($\alpha_{is} = 0$)

Variables

$x_s \in \{0, 1\}$ structure $s \in \mathcal{S}$ is selected ($x_s = 1$) or not ($x_s = 0$)

$$\min \sum_{s \in \mathcal{S}} c_s x_s \quad \text{[Minimize travel costs]} \tag{6a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \quad i \in N \quad \text{[Serve each customer]} \tag{6b}$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \quad \widehat{\mathcal{S}} \subseteq \mathcal{S} \quad \text{[Structure feasibility constraints]} \tag{6c}$$

$$x_s \in \{0, 1\} \quad s \in \mathcal{S} \tag{6d}$$

where $\eta_m(\widehat{\mathcal{S}})$ is the maximum number of structures of the set $\widehat{\mathcal{S}}$ that can be simultaneously in a solution given the number of vehicles m

Structure-based Model (Paradiso et al. (2019))

$\mathcal{S}$ set of all feasible structures

c_s cost of structure $s \in \mathcal{S}$

α_{is} structure $s \in \mathcal{S}$ serves $i \in N$ ($\alpha_{is} = 1$) or not ($\alpha_{is} = 0$)

Variables

$x_s \in \{0, 1\}$ structure $s \in \mathcal{S}$ is selected ($x_s = 1$) or not ($x_s = 0$)

$$\min \sum_{s \in \mathcal{S}} c_s x_s \quad \text{[Minimize travel costs]} \tag{6a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \quad i \in N \quad \text{[Serve each customer]} \tag{6b}$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \quad \widehat{\mathcal{S}} \subseteq \mathcal{S} \quad \text{[Structure feasibility constraints]} \tag{6c}$$

$$x_s \in \{0, 1\} \quad s \in \mathcal{S} \tag{6d}$$

where $\eta_m(\widehat{\mathcal{S}})$ is the maximum number of structures of the set $\widehat{\mathcal{S}}$ that can be simultaneously in a solution given the number of vehicles m

Structure-based Model (Paradiso et al. (2019))

$\mathcal{S}$ set of all feasible structures

c_s cost of structure $s \in \mathcal{S}$

α_{is} structure $s \in \mathcal{S}$ serves $i \in N$ ($\alpha_{is} = 1$) or not ($\alpha_{is} = 0$)

Variables

$x_s \in \{0, 1\}$ structure $s \in \mathcal{S}$ is selected ($x_s = 1$) or not ($x_s = 0$)

$$\min \sum_{s \in \mathcal{S}} c_s x_s \quad \text{[Minimize travel costs]} \tag{6a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \quad i \in N \quad \text{[Serve each customer]} \tag{6b}$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \quad \widehat{\mathcal{S}} \subseteq \mathcal{S} \quad \text{[Structure feasibility constraints]} \tag{6c}$$

$$x_s \in \{0, 1\} \quad s \in \mathcal{S} \tag{6d}$$

where $\eta_m(\widehat{\mathcal{S}})$ is the maximum number of structures of the set $\widehat{\mathcal{S}}$ that can be simultaneously in a solution given the number of vehicles m

Structure-based Model (Paradiso et al. (2019))

Pros and Cons

- Small integrality gaps
- Easy to embed additional side constraints related to trips
- Fewer variables than trip-based (and journey-based) model

Structure-based Model (Paradiso et al. (2019))

Pros and Cons

- Small integrality gaps
- Easy to embed additional side constraints related to trips
- Fewer variables than trip-based (and journey-based) model

- Exponential number of variables
- Column generation/branch(-and-cut)-and-price needed
- Constraints (6c) to add in a cutting plane fashion

Trip vs Journey vs Structure (-based Models)

	Trip	Journey	Structure
Integrality gap	👍	👍👍	👍(👍)
Number of variables	👎👎	👎👎👎	👎
Number of constraints	👎	👍	👎👎
Trip-related constraints	👍	👍	👍
Journey-related constraints	👎	👍	👎
Complexity of algorithms	👎	👎	👎

Sketch of an Exact Method based on Structure-based Model

Paradiso et al. (2019)

1. Compute SP Bound: solve LP relaxation of (7) without (7c) to compute dual sol. $\boldsymbol{u}^1$ of cost LB_1
2. Enumerate Structures: enumerate structures ($\tilde{\mathcal{S}}$) of red. cost $\leq \mathrm{UB} - \mathrm{LB}_1$ w.r.t. $\boldsymbol{u}^1$, where UB is a guessed upper bound
3. Compute SP plus Relaxed SFC: solve LP relaxation of (7) with relaxed (7c) to compute dual sol. $\boldsymbol{u}^2$ of cost LB_2
4. Reduce set of structures: remove from $\tilde{\mathcal{S}}$ structures of red. cost $> \mathrm{UB} - \mathrm{LB}_2$ w.r.t. $\boldsymbol{u}^2$
5. Branch-and-cut: solve (7) by replacing $\mathcal{S}$ with $\tilde{\mathcal{S}}$
6. Optimality check: if no feasible sol. of cost $\leq$ UB exists, increase UB and go to Step 2

Structure-based Model

$$\min \sum_{s \in \mathcal{S}} c_s x_s \tag{7a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \qquad i \in N \tag{7b}$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \qquad \widehat{\mathcal{S}} \subseteq \mathcal{S} \tag{7c}$$

$$x_s \in \{0, 1\} \qquad s \in \mathcal{S} \tag{7d}$$

Sketch of an Exact Method based on Structure-based Model

Paradiso et al. (2019)

1. Compute SP Bound: solve LP relaxation of (7) without (7c) to compute dual sol. $\boldsymbol{u}^1$ of cost LB_1
2. Enumerate Structures: enumerate structures ($\tilde{\mathcal{S}}$) of red. cost $\leq \mathrm{UB} - \mathrm{LB}_1$ w.r.t. $\boldsymbol{u}^1$, where UB is a guessed upper bound
3. Compute SP plus Relaxed SFC: solve LP relaxation of (7) with relaxed (7c) to compute dual sol. $\boldsymbol{u}^2$ of cost LB_2
4. Reduce set of structures: remove from $\tilde{\mathcal{S}}$ structures of red. cost $> \mathrm{UB} - \mathrm{LB}_2$ w.r.t. $\boldsymbol{u}^2$
5. Branch-and-cut: solve (7) by replacing $\mathcal{S}$ with $\tilde{\mathcal{S}}$
6. Optimality check: if no feasible sol. of cost $\leq$ UB exists, increase UB and go to Step 2

Structure-based Model

$$\min \sum_{s \in \mathcal{S}} c_s x_s \tag{7a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \qquad i \in N \tag{7b}$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \qquad \widehat{\mathcal{S}} \subseteq \mathcal{S} \tag{7c}$$

$$x_s \in \{0, 1\} \qquad s \in \mathcal{S} \tag{7d}$$

Sketch of an Exact Method based on Structure-based Model

Paradiso et al. (2019)

1. Compute SP Bound: solve LP relaxation of (7) without (7c) to compute dual sol. $\boldsymbol{u}^1$ of cost LB_1
2. Enumerate Structures: enumerate structures ($\tilde{\mathcal{S}}$) of red. cost $\leq \mathrm{UB} - \mathrm{LB}_1$ w.r.t. $\boldsymbol{u}^1$, where UB is a guessed upper bound
3. Compute SP plus Relaxed SFC: solve LP relaxation of (7) with relaxed (7c) to compute dual sol. $\boldsymbol{u}^2$ of cost LB_2
4. Reduce set of structures: remove from $\tilde{\mathcal{S}}$ structures of red. cost $> \mathrm{UB} - \mathrm{LB}_2$ w.r.t. $\boldsymbol{u}^2$
5. Branch-and-cut: solve (7) by replacing $\mathcal{S}$ with $\tilde{\mathcal{S}}$
6. Optimality check: if no feasible sol. of cost $\leq$ UB exists, increase UB and go to Step 2

Structure-based Model

$$\min \sum_{s \in \mathcal{S}} c_s x_s \quad (7a)$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \qquad i \in N \quad (7b)$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \qquad \widehat{\mathcal{S}} \subseteq \mathcal{S} \quad (7c)$$

$$x_s \in \{0,1\} \qquad s \in \mathcal{S} \quad (7d)$$

Sketch of an Exact Method based on Structure-based Model

Paradiso et al. (2019)

1. Compute SP Bound: solve LP relaxation of (7) without (7c) to compute dual sol. $\boldsymbol{u}^1$ of cost LB_1
2. Enumerate Structures: enumerate structures ($\tilde{\mathcal{S}}$) of red. cost $\leq \text{UB} - \text{LB}_1$ w.r.t. $\boldsymbol{u}^1$, where UB is a guessed upper bound
3. Compute SP plus Relaxed SFC: solve LP relaxation of (7) with relaxed (7c) to compute dual sol. $\boldsymbol{u}^2$ of cost LB_2
4. Reduce set of structures: remove from $\tilde{\mathcal{S}}$ structures of red. cost $> \text{UB} - \text{LB}_2$ w.r.t. $\boldsymbol{u}^2$
5. Branch-and-cut: solve (7) by replacing $\mathcal{S}$ with $\tilde{\mathcal{S}}$
6. Optimality check: if no feasible sol. of cost $\leq$ UB exists, increase UB and go to Step 2

Structure-based Model

$$\min \sum_{s\in\mathcal{S}} c_s x_s \qquad (7a)$$

$$\text{s.t.} \sum_{s\in\mathcal{S}} \alpha_{is} x_s = 1 \qquad i \in N \qquad (7b)$$

$$\sum_{s\in\widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \qquad \widehat{\mathcal{S}} \subseteq \mathcal{S} \qquad (7c)$$

$$x_s \in \{0,1\} \qquad s \in \mathcal{S} \qquad (7d)$$

Sketch of an Exact Method based on Structure-based Model

Paradiso et al. (2019)

1. Compute SP Bound: solve LP relaxation of (7) without (7c) to compute dual sol. $\boldsymbol{u}^1$ of cost LB_1
2. Enumerate Structures: enumerate structures ($\tilde{\mathcal{S}}$) of red. cost $\leq \text{UB} - \text{LB}_1$ w.r.t. $\boldsymbol{u}^1$, where UB is a guessed upper bound
3. Compute SP plus Relaxed SFC: solve LP relaxation of (7) with relaxed (7c) to compute dual sol. $\boldsymbol{u}^2$ of cost LB_2
4. Reduce set of structures: remove from $\tilde{\mathcal{S}}$ structures of red. cost $> \text{UB} - \text{LB}_2$ w.r.t. $\boldsymbol{u}^2$
5. Branch-and-cut: solve (7) by replacing $\mathcal{S}$ with $\tilde{\mathcal{S}}$
6. Optimality check: if no feasible sol. of cost $\leq$ UB exists, increase UB and go to Step 2

Structure-based Model

$$\min \sum_{s\in\mathcal{S}} c_s x_s \tag{7a}$$

$$\text{s.t.} \sum_{s\in\mathcal{S}} \alpha_{is} x_s = 1 \qquad i \in N \tag{7b}$$

$$\sum_{s\in\widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \qquad \widehat{\mathcal{S}} \subseteq \mathcal{S} \tag{7c}$$

$$x_s \in \{0,1\} \qquad s \in \mathcal{S} \tag{7d}$$

Sketch of an Exact Method based on Structure-based Model

Paradiso et al. (2019)

1. Compute SP Bound: solve LP relaxation of (7) without (7c) to compute dual sol. $\boldsymbol{u}^1$ of cost LB_1
2. Enumerate Structures: enumerate structures ($\tilde{\mathcal{S}}$) of red. cost $\leq \text{UB} - \text{LB}_1$ w.r.t. $\boldsymbol{u}^1$, where UB is a guessed upper bound
3. Compute SP plus Relaxed SFC: solve LP relaxation of (7) with relaxed (7c) to compute dual sol. $\boldsymbol{u}^2$ of cost LB_2
4. Reduce set of structures: remove from $\tilde{\mathcal{S}}$ structures of red. cost $> \text{UB} - \text{LB}_2$ w.r.t. $\boldsymbol{u}^2$
5. Branch-and-cut: solve (7) by replacing $\mathcal{S}$ with $\tilde{\mathcal{S}}$
6. Optimality check: if no feasible sol. of cost $\leq$ UB exists, increase UB and go to Step 2

Structure-based Model

$$\min \sum_{s \in \mathcal{S}} c_s x_s \tag{7a}$$

$$\text{s.t.} \sum_{s \in \mathcal{S}} \alpha_{is} x_s = 1 \qquad i \in N \tag{7b}$$

$$\sum_{s \in \widehat{\mathcal{S}}} x_s \leq \eta_m(\widehat{\mathcal{S}}) \qquad \widehat{\mathcal{S}} \subseteq \mathcal{S} \tag{7c}$$

$$x_s \in \{0, 1\} \qquad s \in \mathcal{S} \tag{7d}$$

Table of Contents

Computational Results

MTVRP with Time Windows, Loading Times

			Trip-based Hernandez et al. (2016) Intel Core i7 2670QM			Journey-based Hernandez et al. (2016) Intel Core i7 2670QM			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	8	2.24	8	108	2.12	7	805	0.73	8	19
R	25	11	2.41	11	646	1.19	7	6,925	0.78	11	115
RC	25	8	5.41	6	6,671	2.86	5	2,963	1.91	8	880
C	40	8							1.51	7	2,170
R	40	11							0.41	10	418
RC	40	8							0.83	8	872
C	50	8							1.41	3	3,577
R	50	11							-	0	-
RC	50	8							0.59	7	312

Computational Results

MTVRP with Time Windows, Loading Times

			Trip-based Hernandez et al. (2016) Intel Core i7 2670QM			Journey-based Hernandez et al. (2016) Intel Core i7 2670QM			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\lvert N \rvert$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	8	2.24	8	108	2.12	7	805	0.73	8	19
R	25	11	2.41	11	646	1.19	7	6,925	0.78	11	115
RC	25	8	5.41	6	6,671	2.86	5	2,963	1.91	8	880
C	40	8							1.51	7	2,170
R	40	11							0.41	10	418
RC	40	8							0.83	8	872
C	50	8							1.41	3	3,577
R	50	11							-	0	-
RC	50	8							0.59	7	312

Computational Results

MTVRP with Time Windows, Loading Times

			Trip-based Hernandez et al. (2016) Intel Core i7 2670QM			Journey-based Hernandez et al. (2016) Intel Core i7 2670QM			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\lvert N \rvert$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	8	2.24	8	108	2.12	7	805	0.73	8	19
R	25	11	2.41	11	646	1.19	7	6,925	0.78	11	115
RC	25	8	5.41	6	6,671	2.86	5	2,963	1.91	8	880
C	40	8							1.51	7	2,170
R	40	11							0.41	10	418
RC	40	8							0.83	8	872
C	50	8							1.41	3	3,577
R	50	11							-	0	-
RC	50	8							0.59	7	312

Computational Results

MTVRP with Time Windows, Loading Times

			Trip-based Hernandez et al. (2016) Intel Core i7 2670QM			Journey-based Hernandez et al. (2016) Intel Core i7 2670QM			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	8	2.24	8	108	2.12	7	805	0.73	8	19
R	25	11	2.41	11	646	1.19	7	6,925	0.78	11	115
RC	25	8	5.41	6	6,671	2.86	5	2,963	1.91	8	880
C	40	8							1.51	7	2,170
R	40	11							0.41	10	418
RC	40	8							0.83	8	872
C	50	8							1.41	3	3,577
R	50	11							-	0	-
RC	50	8							0.59	7	312

Computational Results

MTVRP with Time Windows, Loading Times

			Trip-based Hernandez et al. (2016) Intel Core i7 2670QM			Journey-based Hernandez et al. (2016) Intel Core i7 2670QM			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	8	2.24	8	108	2.12	7	805	0.73	8	19
R	25	11	2.41	11	646	1.19	7	6,925	0.78	11	115
RC	25	8	5.41	6	6,671	2.86	5	2,963	1.91	8	880
C	40	8							1.51	7	2,170
R	40	11							0.41	10	418
RC	40	8							0.83	8	872
C	50	8							1.41	3	3,577
R	50	11							-	0	-
RC	50	8							0.59	7	312

Computational Results

MTVRP with Time Windows, Loading Times, Limited Trip Duration

			Trip-based Hernandez et al. (2014) Intel Core 2 Duo 2.10GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	16	1.91	16	420	0.38	16	14
R	25	22	0.76	22	33	0.25	22	2
RC	25	16	2.35	11	18	0.49	16	2
C	40	16	1.25	13	511	0.48	16	151
R	40	19	1.43	12	1,738	1.06	19	220
RC	40	2	-	0	-	0.67	2	11
C	50	16				0.22	16	62
R	50	22				0.22	22	20
RC	50	16				0.28	16	11

Computational Results

MTVRP with Time Windows, Loading Times, Limited Trip Duration

			Trip-based Hernandez et al. (2014) Intel Core 2 Duo 2.10GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	16	1.91	16	420	0.38	16	14
R	25	22	0.76	22	33	0.25	22	2
RC	25	16	2.35	11	18	0.49	16	2
C	40	16	1.25	13	511	0.48	16	151
R	40	19	1.43	12	1,738	1.06	19	220
RC	40	2	-	0	-	0.67	2	11
C	50	16				0.22	16	62
R	50	22				0.22	22	20
RC	50	16				0.28	16	11

Computational Results

MTVRP with Time Windows, Loading Times, Limited Trip Duration

			Trip-based Hernandez et al. (2014) Intel Core 2 Duo 2.10GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	16	1.91	16	420	0.38	16	14
R	25	22	0.76	22	33	0.25	22	2
RC	25	16	2.35	11	18	0.49	16	2
C	40	16	1.25	13	511	0.48	16	151
R	40	19	1.43	12	1,738	1.06	19	220
RC	40	2	-	0	-	0.67	2	11
C	50	16				0.22	16	62
R	50	22				0.22	22	20
RC	50	16				0.28	16	11

Computational Results

MTVRP with Time Windows, Loading Times, Limited Trip Duration

			Trip-based Hernandez et al. (2014) Intel Core 2 Duo 2.10GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	16	1.91	16	420	0.38	16	14
R	25	22	0.76	22	33	0.25	22	2
RC	25	16	2.35	11	18	0.49	16	2
C	40	16	1.25	13	511	0.48	16	151
R	40	19	1.43	12	1,738	1.06	19	220
RC	40	2	-	0	-	0.67	2	11
C	50	16				0.22	16	62
R	50	22				0.22	22	20
RC	50	16				0.28	16	11

Computational Results

MTVRP with Time Windows, Loading Times, Limited Trip Duration

			Trip-based Hernandez et al. (2014) Intel Core 2 Duo 2.10GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
Group	$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
C	25	16	1.91	16	420	0.38	16	14
R	25	22	0.76	22	33	0.25	22	2
RC	25	16	2.35	11	18	0.49	16	2
C	40	16	1.25	13	511	0.48	16	151
R	40	19	1.43	12	1,738	1.06	19	220
RC	40	2	-	0	-	0.67	2	11
C	50	16				0.22	16	62
R	50	22				0.22	22	20
RC	50	16				0.28	16	11

Computational Results

Drone Routing Problem

		Arc-based Cheng et al. (2018) Intel X5650 2.67GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
10	10	4.49	10	0	0.40	10	0
15	10	5.47	4	9	1.28	10	1
20	10	3.69	5	18	0.86	10	2
25	37	2.68	22	59	0.62	37	1
30	10		0		0.52	10	4
35	10		0		0.44	10	11
40	37	3.83	4	4,168	0.20	37	5
45	10		0		0.36	10	13
50	5		0		1.72	5	275

Computational Results

Drone Routing Problem

		Arc-based Cheng et al. (2018) Intel X5650 2.67GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
10	10	4.49	10	0	0.40	10	0
15	10	5.47	4	9	1.28	10	1
20	10	3.69	5	18	0.86	10	2
25	37	2.68	22	59	0.62	37	1
30	10		0		0.52	10	4
35	10		0		0.44	10	11
40	37	3.83	4	4,168	0.20	37	5
45	10		0		0.36	10	13
50	5		0		1.72	5	275

Computational Results

Drone Routing Problem

		Arc-based Cheng et al. (2018) Intel X5650 2.67GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
$\lvert N \rvert$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
10	10	4.49	10	0	0.40	10	0
15	10	5.47	4	9	1.28	10	1
20	10	3.69	5	18	0.86	10	2
25	37	2.68	22	59	0.62	37	1
30	10		0		0.52	10	4
35	10		0		0.44	10	11
40	37	3.83	4	4,168	0.20	37	5
45	10		0		0.36	10	13
50	5		0		1.72	5	275

Computational Results

Drone Routing Problem

		Arc-based Cheng et al. (2018) Intel X5650 2.67GHz			Structure-based Paradiso et al. (2019) Virtual CPU 2.59GHz		
$\|N\|$	Inst	%Gap	Opt	T_{tot}	%Gap	Opt	T_{tot}
10	10	4.49	10	0	0.40	10	0
15	10	5.47	4	9	1.28	10	1
20	10	3.69	5	18	0.86	10	2
25	37	2.68	22	59	0.62	37	1
30	10		0		0.52	10	4
35	10		0		0.44	10	11
40	37	3.83	4	4,168	0.20	37	5
45	10		0		0.36	10	13
50	5		0		1.72	5	275

Table of Contents

Conclusions

- Increasing interest in MTVRPs, mainly motivated by city logistics and last-mile delivery
- Trip-based and journey-based models are effective to solve the MTVRP
- To handle side constraints, structure-based models seem the better choice, even better than set-partitioning models

Conclusions

- Increasing interest in MTVRPs, mainly motivated by city logistics and last-mile delivery
- Trip-based and journey-based models are effective to solve the MTVRP
- To handle side constraints, structure-based models seem the better choice, even better than set-partitioning models

Conclusions

- Increasing interest in MTVRPs, mainly motivated by city logistics and last-mile delivery
- Trip-based and journey-based models are effective to solve the MTVRP
- To handle side constraints, structure-based models seem the better choice, even better than set-partitioning models

Open Questions

- Research on MTVRPs is scarce and 50-customer instances are already challenging, how can large instances be solved?
- Are there better models (maybe models not based on arcs, structures, trips, or journeys)?
- Can we use models not based on arcs or routes to solve other VRPs?

R. Paradiso, R. Roberti, D. Laganá, W. Dullaert. An Exact Solution Framework for Multi-Trip Vehicle Routing Problems with Time Windows. *Operations Research* (forthcoming)

Open Questions

- Research on MTVRPs is scarce and 50-customer instances are already challenging, how can large instances be solved?
- Are there better models (maybe models not based on arcs, structures, trips, or journeys)?
- Can we use models not based on arcs or routes to solve other VRPs?

R. Paradiso, R. Roberti, D. Laganá, W. Dullaert. An Exact Solution Framework for Multi-Trip Vehicle Routing Problems with Time Windows. *Operations Research* (forthcoming)

Open Questions

- Research on MTVRPs is scarce and 50-customer instances are already challenging, how can large instances be solved?
- Are there better models (maybe models not based on arcs, structures, trips, or journeys)?
- Can we use models not based on arcs or routes to solve other VRPs?

R. Paradiso, R. Roberti, D. Laganá, W. Dullaert. An Exact Solution Framework for Multi-Trip Vehicle Routing Problems with Time Windows. *Operations Research* (forthcoming)

Open Questions

- Research on MTVRPs is scarce and 50-customer instances are already challenging, how can large instances be solved?
- Are there better models (maybe models not based on arcs, structures, trips, or journeys)?
- Can we use models not based on arcs or routes to solve other VRPs?

R. Paradiso, R. Roberti, D. Laganá, W. Dullaert. An Exact Solution Framework for Multi-Trip Vehicle Routing Problems with Time Windows. *Operations Research* (forthcoming)

References I

Anaya-Arenas A, Chabot T, Renaud J, Ruiz A, 2016 Biomedical sample transportation in the province of quebec: a case study. International Journal of Production Research 54(2):602–615.

Azi N, Gendreau M, Potvin JY, 2010 An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. European Journal of Operational Research 202(3):756–763.

Azi N, Gendreau M, Potvin JY, 2014 An adaptive large neighborhood search for a vehicle routing problem with multiple routes. Computers and Operations Research 41(1):167–173.

Cattaruzza D, Absi N, Feillet D, 2016a The multi-trip vehicle routing problem with time windows and release dates. Transportation Sci. 50(2):676–693.

Cattaruzza D, Absi N, Feillet D, 2016b Vehicle routing problems with multiple trips. 4OR 14(3):223–259.

Cheng C, Adulyasak Y, Rousseau L, 2018 Formulations and exact algorithms for drone routing problem. Technical Report CIRRELT-2018-31, CIRRELT.

Hernandez F, Feillet D, Giroudeau R, Naud O, 2014 A new exact algorithm to solve the multi-trip vehicle routing problem with time windows and limited duration. 4OR 12(3):235–259.

Hernandez F, Feillet D, Giroudeau R, Naud O, 2016 Branch-and-price algorithms for the solution of the multi-trip vehicle routing problem with time windows. European Journal of Operational Research 249(2):551–559.

Koc C, Karaoglan I, 2011 A branch and cut algorithm for the vehicle routing problem with multiple use of vehicles. Proceedings of the 41st international conference on computers and industrial engineering, 554–559 (Los Angeles, USA).

References II

Macedo R, Alves C, De Carvalho J, Clautiaux F, Hanafi S, 2011 Solving the vehicle routing problem with time windows and multiple routes exactly using a pseudo-polynomial model. European Journal of Operational Research 214(3):536–545.

Mingozzi A, Roberti R, Toth P, 2013 An exact algorithm for the multitrip vehicle routing problem. INFORMS Journal on Computing 25(2):193–207.

Paradiso R, Roberti R, Laganá D, Dullaert W, 2019 An exact solution framework for multi-trip vehicle routing problems with time windows. Operations Research (forthcoming).

Wang Z, Liang W, Hu X, 2014 A metaheuristic based on a pool of routes for the vehicle routing problem with multiple trips and time windows. Journal of the Operational Research Society 65(1):37–48.