Vehicle Routing for a Complex Waste Collection Problem

Iliya Markov[1,2], Sacha Varone[2], Michel Bierlaire[1]

[1]Transport and Mobility Laboratory
School of Architecture, Civil and Environmental Engineering
Ecole Polytechnique Fédérale de Lausanne

[2]Haute Ecole de Gestion de Genève
University of Applied Sciences Western Switzerland (HES-SO)

VeRoLog 2014

Third meeting of the EURO Working Group on
Vehicle Routing and Logistics Optimization

Oslo, June 22-25, 2014

Overview

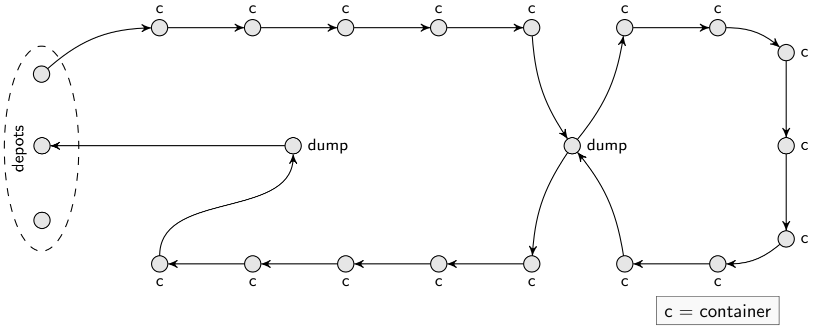
Overview

- A heterogeneous fixed fleet with different:
  - volume capacities
  - weight capacities
  - fixed costs
  - unit-distance running costs
  - unit-time driver wage rates
  - speeds
  - site dependencies (accessibility constraints)

- A heterogeneous fixed fleet with different:
    - volume capacities
    - weight capacities
    - fixed costs
    - unit-distance running costs
    - unit-time driver wage rates
    - speeds
    - site dependencies (accessibility constraints)
- A set of depots
- A set of containers placed at collection points with time windows
- A set of dumps (recycling plants) with time windows

- A heterogeneous fixed fleet with different:
  - volume capacities
  - weight capacities
  - fixed costs
  - unit-distance running costs
  - unit-time driver wage rates
  - speeds
  - site dependencies (accessibility constraints)
- A set of depots
- A set of containers placed at collection points with time windows
- A set of dumps (recycling plants) with time windows
- Maximum tour duration, interrupted by a break
- A tour is a sequence of collections and disposals at the available dumps, with a mandatory disposal before the end of the tour
- A tour need not finish at the depot it started from

Figure 1: Tour illustration



Overview

- VRP with intermediate facilities:
  - VRP with satellite facilities (Bard et al., 1998)
    - no time windows, no driver break, homogeneous fleet
    - branch-and-cut
  - Waste collection VRP (Kim et al., 2006)
    - time windows, driver break, homogeneous fleet
    - simulated annealing
    - Ombuki-Berman et al. (2007) (GA), Benjamin (2011) (VNTS), Buhrkal et al. (2012) (ALNS) improve results by 15-16%
  - MDVRP with inter-depot routes (Crevier et al., 2007)
    - no time windows, no driver break, homogeneous fleet at single depot
    - SP on a pool of single-depot, multi-depot and inter-depot routes
    - Tarantilis et al. (2008) (h-GLS), Hemmelmayr et al. (2013) (VNS) improve results by 1-3%

- VRP with intermediate facilities:
  - VRP with satellite facilities (Bard et al., 1998)
    - no time windows, no driver break, homogeneous fleet
    - branch-and-cut
  - Waste collection VRP (Kim et al., 2006)
    - time windows, driver break, homogeneous fleet
    - simulated annealing
    - Ombuki-Berman et al. (2007) (GA), Benjamin (2011) (VNTS), Buhrkal et al. (2012) (ALNS) improve results by 15-16%
  - MDVRP with inter-depot routes (Crevier et al., 2007)
    - no time windows, no driver break, homogeneous fleet at single depot
    - SP on a pool of single-depot, multi-depot and inter-depot routes
    - Tarantilis et al. (2008) (h-GLS), Hemmelmayr et al. (2013) (VNS) improve results by 1-3%
- Heterogeneous fixed fleet VRP:
  - Proposed by Taillard (1996)
  - Best exact solutions by Baldacci and Mingozzi (2009)
  - Best heuristic solutions by Subramanian et al. (2012) and Penna et al. (2013)

- Contribution to this problem class:
    - Multiple depots
    - Multiple capacities
    - Fixed heterogeneous fleet
    - Realistic cost-based objective function
    - Simplification in the modeling of the dump visits
    - Non-time window constrained break
    - Incentive, rather than enforcement, to go back to the origin depot

Overview

Sets and Parameters:

O' = set of origins
D = set of dumps
$N = O' \cup O'' \cup D \cup P$
K = set of vehicles

O'' = set of destinations
P = set of containers

Sets and Parameters:

O' = set of origins

O'' = set of destinations

D = set of dumps

P = set of containers

$N = O' \cup O'' \cup D \cup P$

K = set of vehicles

π_{ij} = length of edge (i, j)

α_{ijk} = 1 if edge (i, j) is accessible for vehicle k, 0 otherwise

Sets and Parameters:

O' = set of origins

O'' = set of destinations

D = set of dumps

P = set of containers

$N = O' \cup O'' \cup D \cup P$

K = set of vehicles

π_{ij} = length of edge (i, j)

α_{ijk} = 1 if edge (i, j) is accessible for vehicle k, 0 otherwise

τ_{ijk} = travel time of vehicle k on edge (i, j)

ϵ_i = service duration at point i

$[\lambda_i, \mu_i]$ = time window lower and upper bound at point i

Sets and Parameters:

O' = set of origins
D = set of dumps
$N = O' \cup O'' \cup D \cup P$
K = set of vehicles

O'' = set of destinations
P = set of containers

π_{ij} = length of edge (i, j)
α_{ijk} = 1 if edge (i, j) is accessible for vehicle k, 0 otherwise
τ_{ijk} = travel time of vehicle k on edge (i, j)
ϵ_i = service duration at point i
$[\lambda_i, \mu_i]$ = time window lower and upper bound at point i
H = maximum tour duration
η = maximum continuous work limit after which a break is due
δ = break duration

Sets and Parameters:

O'	$=$ set of origins	O''	$=$ set of destinations
D	$=$ set of dumps	P	$=$ set of containers
N	$= O' \cup O'' \cup D \cup P$		
K	$=$ set of vehicles		

π_{ij}	$=$ length of edge (i, j)
α_{ijk}	$=$ 1 if edge (i, j) is accessible for vehicle k, 0 otherwise
τ_{ijk}	$=$ travel time of vehicle k on edge (i, j)
ϵ_i	$=$ service duration at point i
$[\lambda_i, \mu_i]$	$=$ time window lower and upper bound at point i
H	$=$ maximum tour duration
η	$=$ maximum continuous work limit after which a break is due
δ	$=$ break duration
ρ_i^v, ρ_i^w	$=$ volume and weight pickup quantity at point i
Ω_k^v, Ω_k^w	$=$ volume and weight capacity of vehicle k

Sets and Parameters:

O'	$=$ set of origins	O''	$=$ set of destinations
D	$=$ set of dumps	P	$=$ set of containers
N	$= O' \cup O'' \cup D \cup P$		
K	$=$ set of vehicles		

π_{ij}	$=$ length of edge (i, j)
α_{ijk}	$=$ 1 if edge (i, j) is accessible for vehicle k, 0 otherwise
τ_{ijk}	$=$ travel time of vehicle k on edge (i, j)
ϵ_i	$=$ service duration at point i
$[\lambda_i, \mu_i]$	$=$ time window lower and upper bound at point i
H	$=$ maximum tour duration
η	$=$ maximum continuous work limit after which a break is due
δ	$=$ break duration
ρ_i^v, ρ_i^w	$=$ volume and weight pickup quantity at point i
Ω_k^v, Ω_k^w	$=$ volume and weight capacity of vehicle k
ϕ_k	$=$ fixed cost of vehicle k
β_k	$=$ unit-distance running cost of vehicle k
θ_k	$=$ unit-time wage rate of vehicle k

Decision Variables:

$$x_{ijk} = \begin{cases} 1 & \text{if vehicle } k \text{ traverses edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

$$b_{ijk} = \begin{cases} 1 & \text{if vehicle } k \text{ takes a break on edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

$$y_k = \begin{cases} 1 & \text{if vehicle } k \text{ is used} \\ 0 & \text{otherwise} \end{cases}$$

Decision Variables:

$$x_{ijk} = \begin{cases} 1 & \text{if vehicle } k \text{ traverses edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

$$b_{ijk} = \begin{cases} 1 & \text{if vehicle } k \text{ takes a break on edge } (i,j) \\ 0 & \text{otherwise} \end{cases}$$

$$y_k = \begin{cases} 1 & \text{if vehicle } k \text{ is used} \\ 0 & \text{otherwise} \end{cases}$$

S_{ik} = start-of-service time of vehicle k at point i

Q^v_{ik} = cumulative volume on vehicle k at point i

Q^w_{ik} = cumulative weight on vehicle k at point i

$$\text{Min} \quad f = \sum_{k \in K} \left(\phi_k y_k + \beta_k \sum_{i \in N} \sum_{j \in N} \pi_{ij} x_{ijk} + \theta_k \left(\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \right) \right) \tag{1}$$

(8)

$$\text{Min} \quad f = \sum_{k \in K} \left(\phi_k y_k + \beta_k \sum_{i \in N} \sum_{j \in N} \pi_{ij} x_{ijk} + \theta_k \left(\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \right) \right) \tag{1}$$

$$\text{s.t.} \quad \sum_{k \in K} \sum_{j \in D \cup P} x_{ijk} = 1, \qquad \forall i \in P \tag{2}$$

(8)

$$\text{Min} \quad f = \sum_{k \in K} \left(\phi_k y_k + \beta_k \sum_{i \in N} \sum_{j \in N} \pi_{ij} x_{ijk} + \theta_k \left(\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \right) \right) \tag{1}$$

$$\text{s.t.} \quad \sum_{k \in K} \sum_{j \in D \cup P} x_{ijk} = 1, \qquad \forall i \in P \tag{2}$$

$$\sum_{i \in O'} \sum_{j \in N} x_{ijk} = y_k, \qquad \forall k \in K \tag{3}$$

$$\sum_{i \in D} \sum_{j \in O''} x_{ijk} = y_k, \qquad \forall k \in K \tag{4}$$

(8)

$$\text{Min} \quad f = \sum_{k \in K} \left(\phi_k y_k + \beta_k \sum_{i \in N} \sum_{j \in N} \pi_{ij} x_{ijk} + \theta_k \left(\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \right) \right) \quad (1)$$

$$\text{s.t.} \quad \sum_{k \in K} \sum_{j \in D \cup P} x_{ijk} = 1, \qquad \forall i \in P \quad (2)$$

$$\sum_{i \in O'} \sum_{j \in N} x_{ijk} = y_k, \qquad \forall k \in K \quad (3)$$

$$\sum_{i \in D} \sum_{j \in O''} x_{ijk} = y_k, \qquad \forall k \in K \quad (4)$$

$$\sum_{i \in N} x_{ijk} = 0, \qquad \forall k \in K, j \in O' \quad (5)$$

$$\sum_{j \in N} x_{ijk} = 0, \qquad \forall k \in K, i \in O'' \quad (6)$$

$$(8)$$

$$\text{Min} \quad f = \sum_{k\in K}\left(\phi_k y_k + \beta_k \sum_{i\in N}\sum_{j\in N}\pi_{ij}x_{ijk} + \theta_k\left(\sum_{j\in O''} S_{jk} - \sum_{i\in O'} S_{ik}\right)\right) \tag{1}$$

$$\text{s.t.} \quad \sum_{k\in K}\sum_{j\in D\cup P} x_{ijk} = 1, \qquad \forall i \in P \tag{2}$$

$$\sum_{i\in O'}\sum_{j\in N} x_{ijk} = y_k, \qquad \forall k \in K \tag{3}$$

$$\sum_{i\in D}\sum_{j\in O''} x_{ijk} = y_k, \qquad \forall k \in K \tag{4}$$

$$\sum_{i\in N} x_{ijk} = 0, \qquad \forall k \in K, j \in O' \tag{5}$$

$$\sum_{j\in N} x_{ijk} = 0, \qquad \forall k \in K, i \in O'' \tag{6}$$

$$\sum_{i\in N\setminus O''} x_{ijk} = \sum_{i\in N\setminus O'} x_{jik}, \qquad \forall k \in K, j \in D \cup P \tag{7}$$

$$\tag{8}$$

$$\text{Min} \quad f = \sum_{k \in K} \left(\phi_k y_k + \beta_k \sum_{i \in N} \sum_{j \in N} \pi_{ij} x_{ijk} + \theta_k \left(\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \right) \right) \quad (1)$$

$$\text{s.t.} \quad \sum_{k \in K} \sum_{j \in D \cup P} x_{ijk} = 1, \qquad \forall i \in P \quad (2)$$

$$\sum_{i \in O'} \sum_{j \in N} x_{ijk} = y_k, \qquad \forall k \in K \quad (3)$$

$$\sum_{i \in D} \sum_{j \in O''} x_{ijk} = y_k, \qquad \forall k \in K \quad (4)$$

$$\sum_{i \in N} x_{ijk} = 0, \qquad \forall k \in K, j \in O' \quad (5)$$

$$\sum_{j \in N} x_{ijk} = 0, \qquad \forall k \in K, i \in O'' \quad (6)$$

$$\sum_{i \in N \setminus O''} x_{ijk} = \sum_{i \in N \setminus O'} x_{jik}, \qquad \forall k \in K, j \in D \cup P \quad (7)$$

$$x_{ijk} \leqslant \alpha_{ijk}, \qquad \forall k \in K, i \in N \setminus O'', j \in N \setminus O' \quad (8)$$

$$\text{s.t.} \quad Q^v_{ik} \leqslant \Omega^v_k, \qquad \forall k \in K, i \in P \tag{9}$$

$$Q^w_{ik} \leqslant \Omega^w_k, \qquad \forall k \in K, i \in P \tag{10}$$

(19)

$$
\begin{aligned}
\text{s.t.}\quad & Q_{ik}^{v} \leqslant \Omega_{k}^{v}, && \forall k \in K, i \in P && (9)\\
& Q_{ik}^{w} \leqslant \Omega_{k}^{w}, && \forall k \in K, i \in P && (10)\\
& Q_{ik}^{v} = 0, && \forall k \in K, i \in N \setminus P && (11)\\
& Q_{ik}^{w} = 0, && \forall k \in K, i \in N \setminus P && (12)
\end{aligned}
$$

(19)

$$
\begin{aligned}
\text{s.t.} \quad & Q_{ik}^{v} \leqslant \Omega_{k}^{v}, && \forall k \in K, i \in P && (9)\\
& Q_{ik}^{w} \leqslant \Omega_{k}^{w}, && \forall k \in K, i \in P && (10)\\
& Q_{ik}^{v} = 0, && \forall k \in K, i \in N \setminus P && (11)\\
& Q_{ik}^{w} = 0, && \forall k \in K, i \in N \setminus P && (12)\\
& Q_{ik}^{v} + \rho_{j}^{v} \leqslant Q_{jk}^{v} + \left(1 - x_{ijk}\right) M, && \forall k \in K, i \in N \setminus O'', j \in P && (13)\\
& Q_{ik}^{w} + \rho_{j}^{w} \leqslant Q_{jk}^{w} + \left(1 - x_{ijk}\right) M, && \forall k \in K, i \in N \setminus O'', j \in P && (14)
\end{aligned}
$$

(19)

$$
\begin{aligned}
\text{s.t.}\quad & Q_{ik}^{v} \leqslant \Omega_{k}^{v}, && \forall k \in K, i \in P && (9)\\
& Q_{ik}^{w} \leqslant \Omega_{k}^{w}, && \forall k \in K, i \in P && (10)\\
& Q_{ik}^{v} = 0, && \forall k \in K, i \in N \setminus P && (11)\\
& Q_{ik}^{w} = 0, && \forall k \in K, i \in N \setminus P && (12)\\
& Q_{ik}^{v} + \rho_{j}^{v} \leqslant Q_{jk}^{v} + \left(1 - x_{ijk}\right) M, && \forall k \in K, i \in N \setminus O'', j \in P && (13)\\
& Q_{ik}^{w} + \rho_{j}^{w} \leqslant Q_{jk}^{w} + \left(1 - x_{ijk}\right) M, && \forall k \in K, i \in N \setminus O'', j \in P && (14)\\
& S_{ik} + \epsilon_{i} + \delta b_{ijk} + \tau_{ijk} \leqslant S_{jk} + \left(1 - x_{ijk}\right) M, && \forall k \in K, i \in N \setminus O'', j \in N \setminus O' && (15)
\end{aligned}
$$

(19)

$$\text{s.t.} \quad Q^v_{ik} \leqslant \Omega^v_k, \qquad \forall k \in K, i \in P \tag{9}$$

$$Q^w_{ik} \leqslant \Omega^w_k, \qquad \forall k \in K, i \in P \tag{10}$$

$$Q^v_{ik} = 0, \qquad \forall k \in K, i \in N \setminus P \tag{11}$$

$$Q^w_{ik} = 0, \qquad \forall k \in K, i \in N \setminus P \tag{12}$$

$$Q^v_{ik} + \rho^v_j \leqslant Q^v_{jk} + \left(1 - x_{ijk}\right) M, \qquad \forall k \in K, i \in N \setminus O'', j \in P \tag{13}$$

$$Q^w_{ik} + \rho^w_j \leqslant Q^w_{jk} + \left(1 - x_{ijk}\right) M, \qquad \forall k \in K, i \in N \setminus O'', j \in P \tag{14}$$

$$S_{ik} + \epsilon_i + \delta b_{ijk} + \tau_{ijk} \leqslant S_{jk} + \left(1 - x_{ijk}\right) M, \qquad \forall k \in K, i \in N \setminus O'', j \in N \setminus O' \tag{15}$$

$$\left(S_{ik} - \sum_{m \in O'} S_{mk} \right) + \epsilon_i - \eta \leqslant \left(1 - b_{ijk}\right) M, \qquad \forall k \in K, i \in N \setminus O'', j \in N \setminus O' \tag{16}$$

$$\eta - \left(S_{jk} - \sum_{m \in O'} S_{mk} \right) \leqslant \left(1 - b_{ijk}\right) M, \qquad \forall k \in K, i \in N \setminus O'', j \in N \setminus O' \tag{17}$$

$$b_{ijk} \leqslant x_{ijk}, \qquad \forall k \in K, i, j \in N \tag{18}$$

$$\left(\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \right) - \eta \leqslant \left(\sum_{\substack{i \in N \setminus O'' \\ j \in N \setminus O'}} b_{ijk} \right) M, \qquad \forall k \in K \tag{19}$$

$$\text{s.t.} \quad \lambda_i \sum_{j \in N \setminus O'} x_{ijk} \leqslant S_{ik} \leqslant \mu_i \sum_{j \in N \setminus O'} x_{ijk}, \qquad \forall k \in K, i \in N \setminus O'' \tag{20}$$

(23)

$$\text{s.t.} \quad \lambda_i \sum_{j \in N \setminus O'} x_{ijk} \leqslant S_{ik} \leqslant \mu_i \sum_{j \in N \setminus O'} x_{ijk}, \qquad \forall k \in K, i \in N \setminus O'' \tag{20}$$

$$\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \leqslant \text{H}, \qquad \forall k \in K \tag{21}$$

(23)

$$\text{s.t.} \quad \lambda_i \sum_{j \in N \setminus O'} x_{ijk} \leqslant S_{ik} \leqslant \mu_i \sum_{j \in N \setminus O'} x_{ijk}, \qquad \forall k \in K, i \in N \setminus O'' \tag{20}$$

$$\sum_{j \in O''} S_{jk} - \sum_{i \in O'} S_{ik} \leqslant \mathrm{H}, \qquad \forall k \in K \tag{21}$$

$$x_{ijk}, y_k, b_{ijk} \in \{0, 1\}, \qquad \forall k \in K, i, j \in N \tag{22}$$

$$Q^v_{ik}, Q^w_{ik}, S_{ik} \geqslant 0, \qquad \forall k \in K, i \in N \tag{23}$$

Extension:

$$z_{ijk} = \begin{cases} 1 & \text{if } i \text{ is the origin and } j \text{ the destination of vehicle } k \\ 0 & \text{otherwise} \end{cases}$$

Ψ = weight of relocation term

$$\text{Min} \quad f = \text{Objective (1)} + \Psi \sum_{k \in K} \sum_{i \in O'} \sum_{j \in O''} \left(\beta_k \pi_{ji} + \theta_k \tau_{jik} \right) z_{ijk} \tag{24}$$

s.t. Constraints (2) to (23)

$$\sum_{m \in P} x_{imk} + \sum_{m \in D} x_{mjk} - 1 \leqslant z_{ijk}, \qquad \forall k \in K, i \in O', j \in O'' \tag{25}$$

$$z_{ijk} = \{0, 1\}, \qquad \forall k \in K, i \in O', j \in O'' \tag{26}$$

Overview

- For small instances, common solver for the MILP formulation enhanced by valid inequalities and elimination rules, including:
    - Impossible traversals
    - Time window infeasible traversals
    - Latest start/earliest finish
    - Minimum tour duration
    - Symmetry breaking for subsets of identical vehicles
    - Minimum/maximum number of dump visits

- For small instances, common solver for the MILP formulation enhanced by valid inequalities and elimination rules, including:
    - Impossible traversals
    - Time window infeasible traversals
    - Latest start/earliest finish
    - Minimum tour duration
    - Symmetry breaking for subsets of identical vehicles
    - Minimum/maximum number of dump visits
- For realistic-size instances, a local search heuristic admitting infeasible intermediate solutions
- Feasibility is defined in terms of three criteria:
    - Time-window feasibility
    - Duration feasibility
    - Capacity feasibility

- For small instances, common solver for the MILP formulation enhanced by valid inequalities and elimination rules, including:
  - Impossible traversals
  - Time window infeasible traversals
  - Latest start/earliest finish
  - Minimum tour duration
  - Symmetry breaking for subsets of identical vehicles
  - Minimum/maximum number of dump visits
- For realistic-size instances, a local search heuristic admitting infeasible intermediate solutions
- Feasibility is defined in terms of three criteria:
  - Time-window feasibility
  - Duration feasibility
  - Capacity feasibility
- The quality of the heuristic is assessed by benchmarking its results to the optimal ones on small instances, and in comparison to executed tours.

Figure 2: Temporal feasibility algorithm

Data: tour k as a sequence of points $1, \ldots, n$ after a change
Result: start-of-service times, waiting times and temporal feasibility of tour k

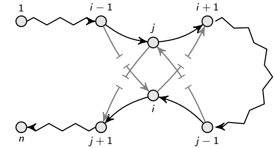
```
set S_1k to earliest possible;
for i = 2...n in tour k do
    // Calculate tentative start-of-service times
    S_ik = S_(i-1)k + ε_(i-1) + τ_(i-1)ik;
    // Insert break
    if S_(i-1)k ⩽ S_1k + η and S_ik + ε_i > S_1k + η then
        S_ik = S_ik + δ;
    end
    // Calculate waiting times
    if S_ik < λ_i then
        w_ik = λ_i − S_ik;
        S_ik = λ_i;
    else
        w_ik = 0;
    end
end
```

Figure 2: Temporal feasibility algorithm, cont'd

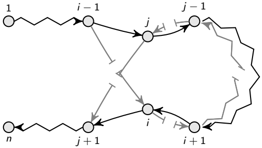
```
// Check time window feasibility
if S_ik ≤ μ_i, ∀i then
    // Forward time slack reduction
    for i = n...2 in tour k do
        S'_(i−1)k = S_(i−1)k;
        S_(i−1)k = min(S_(i−1)k + w_ik, μ_(i−1));
        w_(i−1)k = w_(i−1)k + (S_(i−1)k − S'_(i−1)k);
        w_ik = w_ik − (S_(i−1)k − S'_(i−1)k);
    end
    w_1k = 0;
    // Check duration feasibility
    if S_nk − S_1k ≤ H then
        tour k is temporally feasible;
    else
        tour k is (duration) infeasible;
    end
else
    tour k is (time-window) infeasible;
end
```

Figure 3: Neighborhood operators

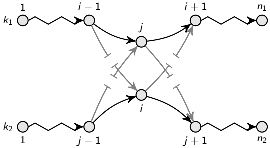
Single-tour 1-1 exchange



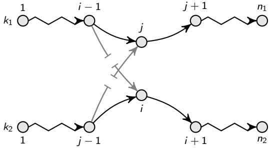
Single-tour 2-opt



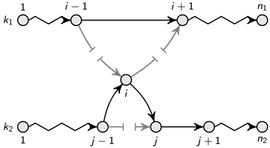
Inter-tour 1-1 exchange



Inter-tour 2-opt



Inter-tour reinsert

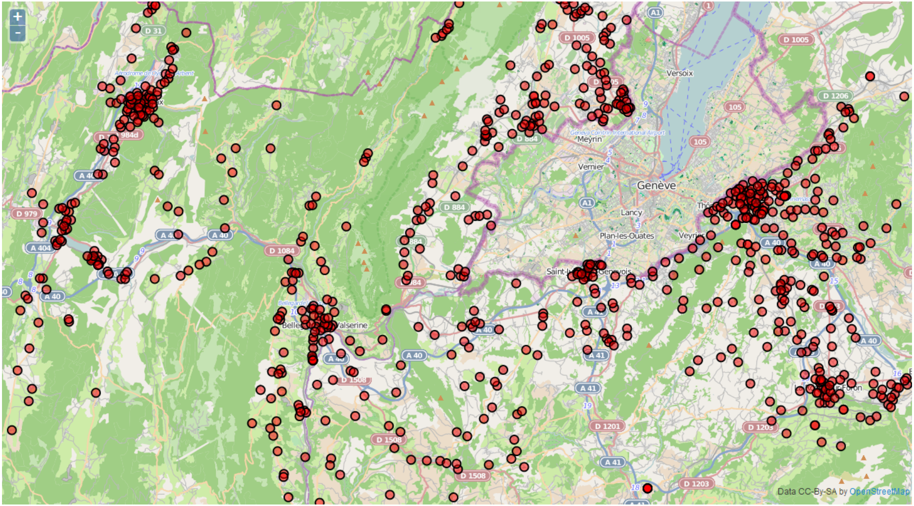


- Tour construction – feasibility preserving insertion:
  - At every iteration an unassigned container is inserted at the point that yields the smallest increase in the objective value
  - When container insertions would violate capacity, a dump is inserted using the same logic
  - A dump insertion should allow for at least one subsequent temporally feasible container insertion

- Tour construction – feasibility preserving insertion:
  - At every iteration an unassigned container is inserted at the point that yields the smallest increase in the objective value
  - When container insertions would violate capacity, a dump is inserted using the same logic
  - A dump insertion should allow for at least one subsequent temporally feasible container insertion
- Tour improvement - local search admitting intermediate infeasibility:
  - The cost of an infeasible solution is multiplied by $(1 + \mathit{infPenalty})^{\mathit{numInf}}$, where *infPenalty* is a percent penalty for being infeasible and *numInf* is the number of infeasible solutions visited in a given operator loop
  - The application of an inter-tour operator is followed by single-tour improvement of the affected tours, if the solution is feasible
  - Every operator is applied for *maxOpIter* iterations and *maxOpNonImpIter* non-improving iterations, before changing to the next operator
  - Both single-tour and multi-tour improvement run for *maxIter* iterations and *maxNonImpIter* non-improving iterations
  - The resulting tour schedule is the best found during all iterations

Overview

Comparison to random instances based on real data from a French collector



- 5 instances extracted randomly from real underlying data
- 3 versions of each instance:
  - No time windows (iX)
  - Wide time windows (iX_tw) - randomly assigned
  - Narrow time windows (iX_ntw) - randomly assigned
- 1 depot, 1 dump, 2 identical vehicles

- 5 instances extracted randomly from real underlying data
- 3 versions of each instance:
  - No time windows (iX)
  - Wide time windows (iX_tw) - randomly assigned
  - Narrow time windows (iX_ntw) - randomly assigned
- 1 depot, 1 dump, 2 identical vehicles
- Tests on 2.60 GHz Intel Core i7, 8GB of RAM
  - Local search heuristic coded in Java
  - Model solved on Gurobi 5.6.2 warm-started with the solutions from the local search heuristic
  - Solver time limit set to 1000 sec

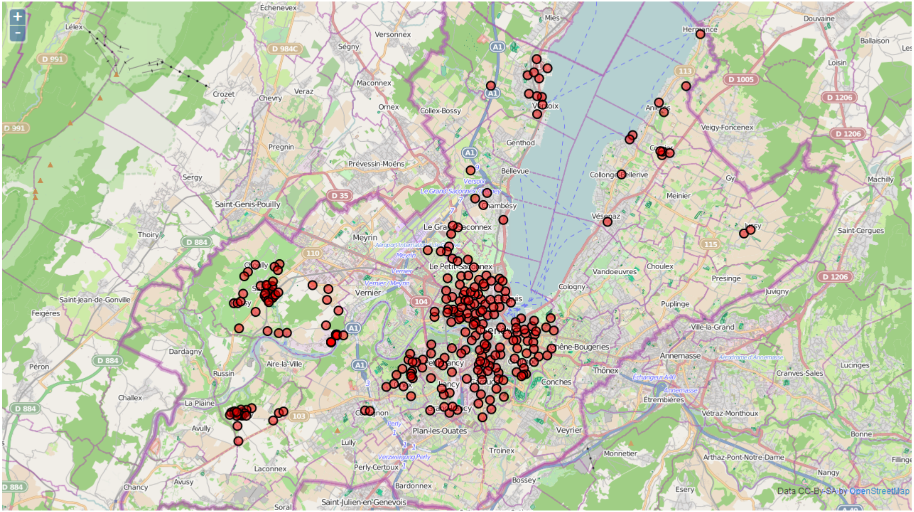
Table 1: Comparison between heuristic and solver on random instances
infPenalty = 5%, *maxOpIter* = 29, *maxOpNonImpIter* = 13, *maxIter* = 5, *maxNonImpIter* = 1

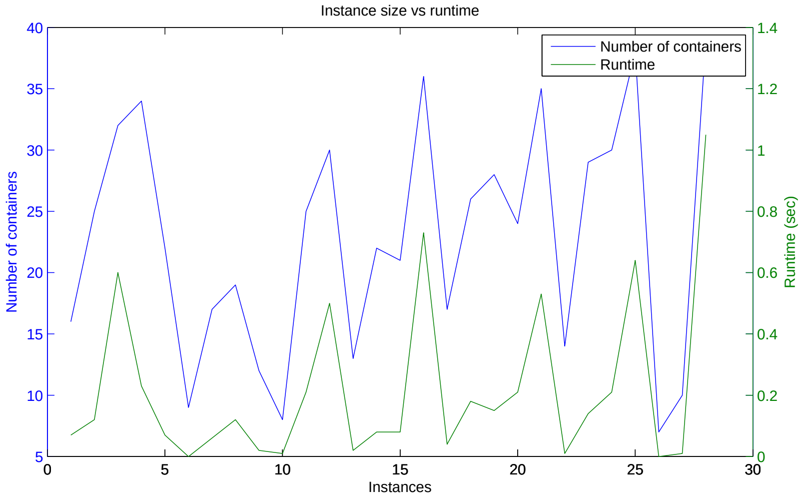
	Heuristic		Solver				
Instance	Objective	Runtime (sec.)	Objective	L Bound	MIP gap (%)	Runtime (sec.)	Opt gap (%)
i1	214.849	0.130	214.849	214.849	0.000	375.562	0.000
i1_tw	252.825	0.040	252.825	252.825	0.000	4.038	0.000
i1_ntw	394.817	0.100	394.817	394.817	0.000	0.922	0.000
i2	249.317	0.010	249.317	249.317	0.000	400.032	0.000
i2_tw	257.583	0.000	257.583	257.582	0.000	2.306	0.000
i2_ntw	439.769	0.200	439.769	439.769	0.000	2.420	0.000
i3	240.133	0.000	240.133	76.004	68.349	1000.000	0.000
i3_tw	245.457	0.010	245.457	245.457	0.000	2.894	0.000
i3_ntw	444.589	0.090	444.589	444.589	0.000	2.446	0.000
i4	138.643	0.010	138.643	138.643	0.000	521.509	0.000
i4_tw	140.204	0.000	140.204	140.204	0.000	7.660	0.000
i4_ntw	179.537	0.010	179.537	179.537	0.000	2.849	0.000
i5	220.770	0.000	220.770	129.834	41.190	1000.000	0.000
i5_tw	233.211	0.000	233.211	233.211	0.000	3.501	0.000
i5_ntw	405.622	0.170	405.622	405.622	0.000	3.051	0.000

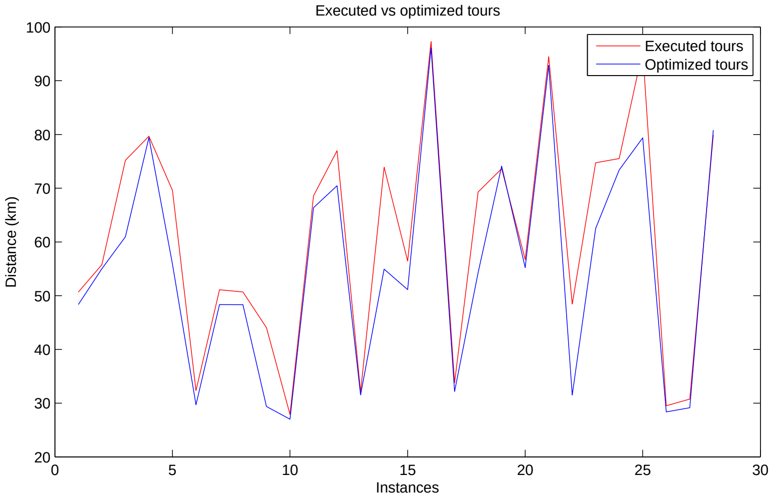
Table 1: Comparison between heuristic and solver on random instances
infPenalty = 5%, *maxOpIter* = 29, *maxOpNonImpIter* = 13, *maxIter* = 5, *maxNonImpIter* = 1

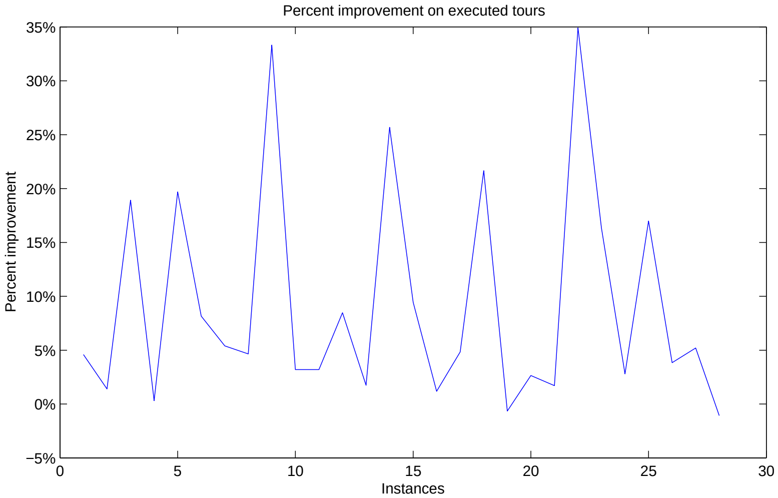
	Heuristic		Solver				
Instance	Objective	Runtime (sec.)	Objective	L Bound	MIP gap (%)	Runtime (sec.)	Opt gap (%)
i1	214.849	0.130	214.849	214.849	0.000	375.562	0.000
i1_tw	252.825	0.040	252.825	252.825	0.000	4.038	0.000
i1_ntw	394.817	0.100	394.817	394.817	0.000	0.922	0.000
i2	249.317	0.010	249.317	249.317	0.000	400.032	0.000
i2_tw	257.583	0.000	257.583	257.582	0.000	2.306	0.000
i2_ntw	439.769	0.200	439.769	439.769	0.000	2.420	0.000
i3	240.133	0.000	240.133	76.004	**68.349**	**1000.000**	0.000
i3_tw	245.457	0.010	245.457	245.457	0.000	2.894	0.000
i3_ntw	444.589	0.090	444.589	444.589	0.000	2.446	0.000
i4	138.643	0.010	138.643	138.643	0.000	521.509	0.000
i4_tw	140.204	0.000	140.204	140.204	0.000	7.660	0.000
i4_ntw	179.537	0.010	179.537	179.537	0.000	2.849	0.000
i5	220.770	0.000	220.770	129.834	**41.190**	**1000.000**	0.000
i5_tw	233.211	0.000	233.211	233.211	0.000	3.501	0.000
i5_ntw	405.622	0.170	405.622	405.622	0.000	3.051	0.000

Comparison to executed tours from a Swiss collector




Instance size vs runtime
Number of containers
Runtime
Number of containers
Runtime (sec)
Instances
40
35
30
25
20
15
10
5
0
5
10
15
20
25
30
1.4
1.2
1
0.8
0.6
0.4
0.2
0


Executed vs optimized tours
Executed tours
Optimized tours
Distance (km)
Instances
100
90
80
70
60
50
40
30
20
0
5
10
15
20
25
30


Percent improvement on executed tours
35%
30%
25%
20%
15%
10%
5%
0%
−5%
Percent improvement
0
5
10
15
20
25
30
Instances

Overview

- Conclusions:
  - Mathematical model
  - Local search heuristic
  - The heuristic performs favorably with a zero optimality gap compared to the small random instances and an average improvement of 9% compared to the executed tours.

- Conclusions:
  - Mathematical model
  - Local search heuristic
  - The heuristic performs favorably with a zero optimality gap compared to the small random instances and an average improvement of 9% compared to the executed tours.
- Future work
  - Mathematical model improvement to solve larger instances for benchmarking
  - Development of efficient vehicle-to-tour evaluation and assignment procedures to respond to the challenge posed by the heterogeneous fleet
  - Sensitivity analysis of the parameters
  - Container level prediction algorithms based on data from level sensors
  - Development of an inventory routing system with dynamic periodicity

Thank you for your attention!
Questions?

Overview

Baldacci, R. and Mingozzi, A. (2009). A unified exact method for solving different classes of vehicle routing problems. *Mathematical Programming*, 120(2):347–380.

Bard, J. F., Huang, L., Dror, M., and Jaillet, P. (1998). A branch and cut algorithm for the VRP with satellite facilities. *IIE Transactions*, 30(9):821–834.

Benjamin, A. M. (2011). *Metaheuristics for the Waste Collection Vehicle Routing Problem with Time Windows*. PhD thesis, Department of Mathematical Sciences, Brunel University.

Buhrkal, K., Larsen, A., and Ropke, S. (2012). The waste collection vehicle routing problem with time windows in a city logistics context. *Procedia-Social and Behavioral Sciences*, 39:241–254.

Crevier, B., Cordeau, J.-F., and Laporte, G. (2007). The multi-depot vehicle routing problem with inter-depot routes. *European Journal of Operational Research*, 176(2):756–773.

Hemmelmayr, V., Doerner, K. F., Hartl, R. F., and Rath, S. (2013). A heuristic solution method for node routing based solid waste collection problems. *Journal of Heuristics*, 19(2):129–156.

Kim, B. I., Kim, S., and Sahoo, S. (2006). Waste collection vehicle routing problem with time windows. *Computers & Operations Research*, 33(12):3624–3642.

Ombuki-Berman, B. M., Runka, A., and Hanshar, F. T. (2007). Waste collection vehicle routing problem with time windows using multi-objective genetic algorithms. In *Proceedings of the Third IASTED International Conference on Computational Intelligence*, CI '07, pages 91–97, Anaheim, CA, USA.

Penna, P., Subramanian, A., and Ochi, L. (2013). An iterated local search heuristic for the heterogeneous fleet vehicle routing problem. *Journal of Heuristics*, 19(2):201–232.

Subramanian, A., Penna, P. H. V., Uchoa, E., and Ochi, L. S. (2012). A hybrid algorithm for the heterogeneous fleet vehicle routing problem. *European Journal of Operational Research*, 221(2):285–295.

Taillard, E. D. (1996). A heuristic column generation method for heterogeneous fleet VRP. *Publication CRT-96-13*.

Tarantilis, C. D., Zachariadis, E. E., and Kiranoudis, C. T. (2008). A hybrid guided local search for the vehicle-routing problem with intermediate replenishment facilities. *INFORMS Journal on Computing*, 20(1):154–168.