The Vehicle Routing Problem with Floating Locations: Formulation and Branch & Price Algorithm

Claudio Gambella[1],
Bissan Ghaddar[2] and Joe Naoum-Sawaya[3]

[1] DEI "Guglielmo Marconi" University of Bologna - Italy
[2] Smarter Cities Technology Centre, IBM Research - Ireland
[3] Ivey Business School - London, Ontario, Canada

Bologna, May 19th, 2016

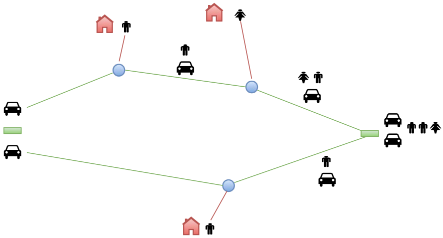
Outline

1. Introduction - The Vehicle Routing Problem with Floating Locations
2. Mathematical Formulation
3. Lagrangian Relaxation
4. Branch-and-Price Algorithm
5. Computational Results
6. Conclusions

The Vehicle Routing Problem with Floating Locations (VRPFL)

General problem description

A set of target points in the plane is allowed to move in the Euclidean plane from its initial location. The goal of the optimization problem is to determine minimum-time vehicle routes starting from the depot and ending in the destination such that all targets are intercepted by a vehicle in a convenient location.



Applications

- Ride-sharing: cars picking-up customers which share a common destination.
- Target tracking problems: e.g., defense, weather monitoring.

VRPFL - Introduction

Challenging aspects

- Finding the optimal target visiting sequence.
- Determining the meeting points between targets and vehicles.

Assumptions

- Common depot for vehicles.
- Common destination for targets.
- Homogeneous fleet (capacity, speed).
- Targets have a maximum speed and can wait for the vehicle arrival.

Problem Variant

VRPFL with fixed line direction: the target locations can move only in a fixed (known) direction.

VRPFL Contributions

- Solving a Mixed Integer Second Order Conic Program (MISOCP) formulation with CPLEX.
- Introducing valid inequalities to strengthen the MISOCP continuous relaxation.
- Developing a Branch-and-Price algorithm based on a Lagrangian Relaxation of the MISOCP.

VRPFL Contributions

- Solving a Mixed Integer Second Order Conic Programming (MISOCP) formulation with CPLEX.
- Introducing valid inequalities to strengthen the MISOCP continuous relaxation.
- Developing a Branch-and-Price algorithm based on a Lagrangian Relaxation of the MISOCP.

VRPFL Formulation: Parameters

n	number of targets
K	number of vehicles available
Q	vehicles capacity
V	vehicles speed
O	coordinates of the depot
D	coordinates of the destination location
q_j	initial position of target j
v_j	target j speed
d_j	target j direction vector

VRPFL Formulation: Decision variables (1)

Continuous variables

M_i^k	coordinates of the i-th meeting point of vehicle k
m_j	coordinates of the meeting point of target j
T_i^k	time required by vehicle k for travelling from M_{i-1}^k to M_i^k
t_j	time required by target j for reaching m_j
λ_j	scalar variable for determining m_j along d_j

VRPFL Formulation: Decision variables (2)

Binary variables

$$x_{ij}^{k} = \begin{cases} 1 & \text{if target } j \text{ is the } i\text{-th target visited by vehicle } k \\ 0 & \text{otherwise} \end{cases}$$

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

VRPFL Formulation: Mathematical Model

min	Vehicle routes travel time
s.t	Vehicle travel time between consecutive meeting points
	Target travel time
	Compatibility of the two sets of meeting points
	Time synchronization in meeting points
	Assignment of targets to vehicles
	Ensuring the target visiting sequence
	Vehicle usage
	Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

s.t Vehicle travel time between consecutive meeting points

Target travel time

Compatibility of the two sets of meeting points

Time synchronization in meeting points

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \; k = 1, \ldots, K$$

Target travel time

Compatibility of the two sets of meeting points

Time synchronization in meeting points

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \; k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

Compatibility of the two sets of meeting points

Time synchronization in meeting points

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \; k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

$$M_i^k = m_j \;\; \text{if } x_{i,j}^k = 1 \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, j = 1, \ldots, n+1$$

Time synchronization in meeting points

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$
\begin{aligned}
\min \quad & \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k \\
\text{s.t} \quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k && i = 1, \dots, Q+1,\ k = 1, \dots, K \\
& \frac{\|m_j - q_j\|}{v_j} \leq t_j && j = 1, \dots, n \\
& M_i^k \geq m_j - C_M(1 - x_{ij}^k) && i = 1, \dots, Q,\ k = 1, \dots, K,\ j = 1, \dots, n+1 \\
& M_i^k \leq m_j + C_M(1 - x_{ij}^k) && i = 1, \dots, Q,\ k = 1, \dots, K,\ j = 1, \dots, n+1
\end{aligned}
$$

Time synchronization in meeting points

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \ k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

$$M_i^k \geq m_j - C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \ k = 1, \ldots, K, \ j = 1, \ldots, n+1$$

$$M_i^k \leq m_j + C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \ k = 1, \ldots, K, \ j = 1, \ldots, n+1$$

$$M_{Q+1}^k = y_k D + (1 - y_k) O \qquad k = 1, \ldots, K$$

Time synchronization in meeting points

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \; k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

$$M_i^k \geq m_j - C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n+1$$

$$M_i^k \leq m_j + C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n+1$$

$$M_{Q+1}^k = y_k D + (1 - y_k) O \qquad k = 1, \ldots, K$$

$$\sum_{i'=1}^{i} T_{i'}^k \geq t_j \quad \text{if } x_{i,j}^k = 1 \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n$$

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1,\ k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

$$M_i^k \geq m_j - C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n+1$$

$$M_i^k \leq m_j + C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n+1$$

$$M_{Q+1}^k = y_k D + (1 - y_k) O \qquad k = 1, \ldots, K$$

$$\sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) \qquad i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n$$

Assignment of targets to vehicles

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$
\begin{aligned}
\min \quad & \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k \\
\text{s.t} \quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k && i = 1, \dots, Q+1, \; k = 1, \dots, K \\
& \frac{\|m_j - q_j\|}{v_j} \leq t_j && j = 1, \dots, n \\
& M_i^k \geq m_j - C_M(1 - x_{ij}^k) && i = 1, \dots, Q, \; k = 1, \dots, K, \; j = 1, \dots, n+1 \\
& M_i^k \leq m_j + C_M(1 - x_{ij}^k) && i = 1, \dots, Q, \; k = 1, \dots, K, \; j = 1, \dots, n+1 \\
& M_{Q+1}^k = y_k D + (1 - y_k) O && k = 1, \dots, K \\
& \sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) && i = 1, \dots, Q, \; k = 1, \dots, K, \; j = 1, \dots, n \\
& \sum_{k=1}^{K} \sum_{i=1}^{Q} x_{i,j}^k = 1 && j = 1, \dots, n
\end{aligned}
$$

Ensuring the target visiting sequence

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

s.t

$$\frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \; k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

$$M_i^k \geq m_j - C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n+1$$

$$M_i^k \leq m_j + C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n+1$$

$$M_{Q+1}^k = y_k D + (1 - y_k) O \qquad k = 1, \ldots, K$$

$$\sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n$$

$$\sum_{k=1}^{K} \sum_{i=1}^{Q} x_{i,j}^k = 1 \qquad j = 1, \ldots, n$$

$$\sum_{j'=1}^{n} x_{i,j'}^k \geq x_{i+1,j}^k \qquad j = 1, \ldots, n, \; i = 1, \ldots, Q-1, \; k = 1, \ldots, K$$

Vehicle usage

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$
\begin{aligned}
\min \quad & \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k \\
\text{s.t} \quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k && i = 1, \dots, Q+1,\ k = 1, \dots, K \\
& \frac{\|m_j - q_j\|}{v_j} \leq t_j && j = 1, \dots, n \\
& M_i^k \geq m_j - C_M(1 - x_{ij}^k) && i = 1, \dots, Q,\ k = 1, \dots, K,\ j = 1, \dots, n+1 \\
& M_i^k \leq m_j + C_M(1 - x_{ij}^k) && i = 1, \dots, Q,\ k = 1, \dots, K,\ j = 1, \dots, n+1 \\
& M_{Q+1}^k = y_k D + (1 - y_k) O && k = 1, \dots, K \\
& \sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) && i = 1, \dots, Q,\ k = 1, \dots, K,\ j = 1, \dots, n \\
& \sum_{k=1}^{K} \sum_{i=1}^{Q} x_{i,j}^k = 1 && j = 1, \dots, n \\
& \sum_{j'=1}^{n} x_{i,j'}^k \geq x_{i+1,j}^k && j = 1, \dots, n,\ i = 1, \dots, Q-1,\ k = 1, \dots, K \\
& y_k = \sum_{j=1}^{n} x_{1,j}^k && k = 1, \dots, K
\end{aligned}
$$

Reaching the final destination

VRPFL Formulation: Mathematical Model

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1, \; k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

$$M_i^k \geq m_j - C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n+1$$

$$M_i^k \leq m_j + C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n+1$$

$$M_{Q+1}^k = y_k D + (1 - y_k) O \qquad k = 1, \ldots, K$$

$$\sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) \qquad i = 1, \ldots, Q, \; k = 1, \ldots, K, \; j = 1, \ldots, n$$

$$\sum_{k=1}^{K} \sum_{i=1}^{Q} x_{i,j}^k = 1 \qquad j = 1, \ldots, n$$

$$\sum_{j'=1}^{n} x_{i,j'}^k \geq x_{i+1,j}^k \qquad j = 1, \ldots, n, \; i = 1, \ldots, Q-1, \; k = 1, \ldots, K$$

$$y_k = \sum_{j=1}^{n} x_{1,j}^k \qquad k = 1, \ldots, K$$

$$\sum_{j=1}^{n+1} x_{i,j}^k = y_k \qquad i = 2, \ldots, Q, \; k = 1, \ldots, K$$

VRPFL Formulation: Mathematical Model for the Fixed Line Direction case

$$\min \quad \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k$$

$$\text{s.t} \quad \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k \qquad i = 1, \ldots, Q+1,\ k = 1, \ldots, K$$

$$\frac{\|m_j - q_j\|}{v_j} \leq t_j \qquad j = 1, \ldots, n$$

Meeting points restricted on a line

$$M_i^k \geq m_j - C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n+1$$

$$M_i^k \leq m_j + C_M(1 - x_{ij}^k) \qquad i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n+1$$

$$M_{Q+1}^k = y_k D + (1 - y_k) O \qquad k = 1, \ldots, K$$

$$\sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) \qquad i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n$$

$$\sum_{k=1}^{K} \sum_{i=1}^{Q} x_{i,j}^k = 1 \qquad j = 1, \ldots, n$$

$$\sum_{j'=1}^{n} x_{i,j'}^k \geq x_{i+1,j}^k \qquad j = 1, \ldots, n,\ i = 1, \ldots, Q-1,\ k = 1, \ldots, K$$

$$y_k = \sum_{j=1}^{n} x_{1,j}^k \qquad k = 1, \ldots, K$$

$$\sum_{j=1}^{n+1} x_{i,j}^k = y_k \qquad i = 2, \ldots, Q,\ k = 1, \ldots, K$$

VRPFL Formulation: Mathematical Model for the Fixed Line Direction case

$$
\begin{array}{lll}
\min & \displaystyle\sum_{k=1}^{K}\sum_{i=1}^{Q+1} T_i^k & \\
\text{s.t} & \dfrac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k & i = 1,\dots,Q+1,\ k = 1,\dots,K \\
& \dfrac{\|m_j - q_j\|}{v_j} \leq t_j & j = 1,\dots,n \\
& m_j - q_j = \lambda_j\, d_j & j = 1,\dots,n \\
& M_i^k \geq m_j - C_M(1 - x_{ij}^k) & i = 1,\dots,Q,\ k = 1,\dots,K,\ j = 1,\dots,n+1 \\
& M_i^k \leq m_j + C_M(1 - x_{ij}^k) & i = 1,\dots,Q,\ k = 1,\dots,K,\ j = 1,\dots,n+1 \\
& M_{Q+1}^k = y_k D + (1 - y_k) O & k = 1,\dots,K \\
& \displaystyle\sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) & i = 1,\dots,Q,\ k = 1,\dots,K,\ j = 1,\dots,n \\
& \displaystyle\sum_{k=1}^{K}\sum_{i=1}^{Q} x_{i,j}^k = 1 & j = 1,\dots,n \\
& \displaystyle\sum_{j'=1}^{n} x_{i,j'}^k \geq x_{i+1,j}^k & j = 1,\dots,n,\ i = 1,\dots,Q-1,\ k = 1,\dots,K \\
& y_k = \displaystyle\sum_{j=1}^{n} x_{1,j}^k & k = 1,\dots,K \\
& \displaystyle\sum_{j=1}^{n+1} x_{i,j}^k = y_k & i = 2,\dots,Q,\ k = 1,\dots,K
\end{array}
$$

VRPFL Formulation: Mathematical Model for the Fixed Line Direction case

$$
\begin{aligned}
\min \quad & \sum_{k=1}^{K} \sum_{i=1}^{Q+1} T_i^k \\
\text{s.t} \quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \leq T_i^k && i = 1, \ldots, Q+1,\ k = 1, \ldots, K \\
& \frac{\|d_j\| \lambda_j}{v_j} \leq t_j && j = 1, \ldots, n \\
& m_j - q_j = \lambda_j \, d_j && j = 1, \ldots, n \\
& M_i^k \geq m_j - C_M(1 - x_{ij}^k) && i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n+1 \\
& M_i^k \leq m_j + C_M(1 - x_{ij}^k) && i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n+1 \\
& M_{Q+1}^k = y_k D + (1 - y_k) O && k = 1, \ldots, K \\
& \sum_{i'=1}^{i} T_{i'}^k \geq t_j - C_T(1 - x_{ij}^k) && i = 1, \ldots, Q,\ k = 1, \ldots, K,\ j = 1, \ldots, n \\
& \sum_{k=1}^{K} \sum_{i=1}^{Q} x_{i,j}^k = 1 && j = 1, \ldots, n \\
& \sum_{j'=1}^{n} x_{i,j'}^k \geq x_{i+1,j}^k && j = 1, \ldots, n,\ i = 1, \ldots, Q-1,\ k = 1, \ldots, K \\
& y_k = \sum_{j=1}^{n} x_{1,j}^k && k = 1, \ldots, K \\
& \sum_{j=1}^{n+1} x_{i,j}^k = y_k && i = 2, \ldots, Q,\ k = 1, \ldots, K
\end{aligned}
$$

VRPFL Formulation: Mathematical Model

Model features

- Mixed Integer Second Order Conic Programming (MISOCP) formulation:
  - Linear objective function
  - $(Q+1)K+n$ second-order conic constraints in the general case; $(Q+1)K$ second-order conic constraints in the fixed line direction case
  - Linear constraints
  - Continuous and integer variables
- Polynomial in the problem size

Implementation

- C callable libraries of CPLEX 12.6.1.

VRPFL with fixed line direction: Optimal solution visualization

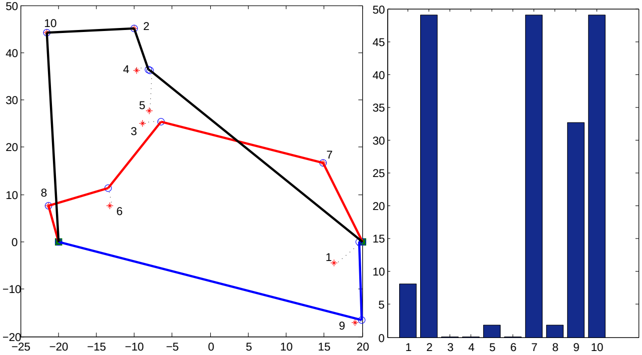


Figure: Instance with $n = 10, K = 3, Q = 4$: solution plot and waiting times

Lagrangian Relaxation

$$
\begin{aligned}
\textbf{[LR}(\mu)\textbf{]} \ \min \quad & \sum_{k=1}^{K}\sum_{i=1}^{Q+1} T_i^k \\
\text{s.t} \quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \le T_i^k && i = 1,\dots,Q+1,\ k = 1,\dots,K \\
& \frac{\|m_j - q_j\|}{v_j} \le t_j && j = 1,\dots,n \\
& M_i^k \ge m_j - C_M(1 - x_{ij}^k) && i = 1,\dots,Q,\ k = 1,\dots,K,\ j = 1,\dots,n+1 \\
& M_i^k \le m_j + C_M(1 - x_{ij}^k) && i = 1,\dots,Q,\ k = 1,\dots,K,\ j = 1,\dots,n+1 \\
& M_{Q+1}^k = y_k \cdot D + (1 - y_k) \cdot O && k = 1,\dots,K \\
& \sum_{i'=1}^{i} T_{i'}^k \ge t_j - C_T(1 - x_{ij}^k) && i = 1,\dots,Q,\ k = 1,\dots,K,\ j = 1,\dots,n \\
& \sum_{k=1}^{K}\sum_{i=1}^{Q} x_{i,j}^k = 1 && j = 1,\dots,n \quad \longrightarrow \quad \mu_j \\
& \sum_{j'=1}^{n} x_{i,j'}^k \ge x_{i+1,j}^k && j = 1,\dots,n,\ i = 1,\dots,Q-1,\ k = 1,\dots,K \\
& y_k = \sum_{j=1}^{n} x_{1,j}^k && k = 1,\dots,K \\
& \sum_{j=1}^{n+1} x_{i,j}^k = y_k && i = 2,\dots,Q,\ k = 1,\dots,K
\end{aligned}
$$

Lagrangian Relaxation

$$
\begin{aligned}
\textbf{[LR(}\mu\textbf{)]}\ \ \min \quad & \sum_{k=1}^{K}\sum_{i=1}^{Q+1} T_i^k + \sum_{j=1}^{n}\mu_j\Big(1-\sum_{i=1}^{Q}\sum_{k=1}^{K}x_{i,j}^k\Big) \\
\text{s.t} \quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \le T_i^k && i=1,\ldots,Q+1,\ k=1,\ldots,K \\
& \frac{\|m_j - q_j\|}{v_j} \le t_j && j=1,\ldots,n \\
& M_i^k \ge m_j - C_M(1-x_{ij}^k) && i=1,\ldots,Q,\ k=1,\ldots,K,\ j=1,\ldots,n+1 \\
& M_i^k \le m_j + C_M(1-x_{ij}^k) && i=1,\ldots,Q,\ k=1,\ldots,K,\ j=1,\ldots,n+1 \\
& M_{Q+1}^k = y_k\cdot D + (1-y_k)\cdot O && k=1,\ldots,K \\
& \sum_{i'=1}^{i} T_{i'}^k \ge t_j - C_T(1-x_{ij}^k) && i=1,\ldots,Q,\ k=1,\ldots,K,\ j=1,\ldots,n \\
& \sum_{j'=1}^{n} x_{i,j'}^k \ge x_{i+1,j}^k && j=1,\ldots,n,\ i=1,\ldots,Q-1,\ k=1,\ldots,K \\
& y_k = \sum_{j=1}^{n} x_{1,j}^k && k=1,\ldots,K \\
& \sum_{j=1}^{n+1} x_{i,j}^k = y_k && i=2,\ldots,Q,\ k=1,\ldots,K
\end{aligned}
$$

Lagrangian Relaxation

$$
\begin{aligned}
\textbf{[LR}(\mu)\textbf{]}\quad \min\quad & \sum_{k=1}^{K}\sum_{i=1}^{Q+1} T_i^k + \sum_{j=1}^{n}\mu_j\Big(1-\sum_{i=1}^{Q}\sum_{k=1}^{K}x_{i,j}^k\Big) \\
\text{s.t}\quad & \frac{\|M_i^k - M_{i-1}^k\|}{V} \le T_i^k && i=1,\dots,Q+1,\ k=1,\dots,K \\
& \frac{\|m_j - q_j\|}{v_j} \le t_j && j=1,\dots,n \\
& M_i^k \ge m_j - C_M(1-x_{ij}^k) && i=1,\dots,Q,\ k=1,\dots,K,\ j=1,\dots,n+1 \\
& M_i^k \le m_j + C_M(1-x_{ij}^k) && i=1,\dots,Q,\ k=1,\dots,K,\ j=1,\dots,n+1 \\
& M_{Q+1}^k = y_k\cdot D + (1-y_k)\cdot O && k=1,\dots,K \\
& \sum_{i'=1}^{i} T_{i'}^k \ge t_j - C_T(1-x_{ij}^k) && i=1,\dots,Q,\ k=1,\dots,K,\ j=1,\dots,n \\
& \sum_{i=1}^{Q} x_{i,j}^k \le 1 && j=1,\dots,n,\ k=1,\dots,K \\
& \sum_{j'=1}^{n} x_{i,j'}^k \ge x_{i+1,j}^k && j=1,\dots,n,\ i=1,\dots,Q-1,\ k=1,\dots,K \\
& y_k = \sum_{j=1}^{n} x_{1,j}^k && k=1,\dots,K \\
& \sum_{j=1}^{n+1} x_{i,j}^k = y_k && i=2,\dots,Q,\ k=1,\dots,K
\end{aligned}
$$

Lagrangian Decomposition

- The Lagrangian Relaxation $\mathbf{LR}(\mu)$ decomposes into K identical subproblems $\mathbf{SP}(\mu)$:

$$
\begin{array}{lll}
\textbf{[SP}(\mu)\textbf{]} \quad \min & \displaystyle\sum_{i=1}^{Q+1} T_i - \sum_{j=1}^{n}\sum_{i=1}^{Q} x_{i,j}\mu_j & \\
\text{s.t} & \displaystyle\frac{\|M_i - M_{i-1}\|}{V} \leq T_i & i = 1, \dots, Q+1 \\
& \displaystyle\frac{\|m_j - q_j\|}{v_j} \leq t_j & j = 1, \dots, n \\
& M_i \geq m_j - C_M(1 - x_{ij}) & i = 1, \dots, Q,\ j = 1, \dots, n+1 \\
& M_i \leq m_j + C_M(1 - x_{ij}) & i = 1, \dots, Q,\ j = 1, \dots, n+1 \\
& M_{Q+1} = y \cdot D + (1 - y) \cdot O & \\
& \displaystyle\sum_{i'=1}^{i} T_{i'} \geq t_j - C_T(1 - x_{ij}) & i = 1, \dots, Q, j = 1, \dots, n \\
& \displaystyle\sum_{i=1}^{Q} x_{i,j} \leq 1 & j = 1, \dots, n \\
& \displaystyle\sum_{j'=1}^{n} x_{i,j'} \geq x_{i+1,j} & j = 1, \dots, n,\ i = 1, \dots, Q-1 \\
& \displaystyle y = \sum_{j=1}^{n} x_{1,j} & \\
& \displaystyle\sum_{j=1}^{n+1} x_{i,j} = y & i = 2, \dots, Q
\end{array}
$$

Lagrangian Bound Problem

Lagrangian Relaxation Value

The value $v(LR(\mu))$ is then computed as:

$$v(LR(\mu)) = K \cdot v(SP(\mu)) + \sum_{j=1}^{n} \mu_j$$

Lagrangian Bound

The Lagrangian Bound $v(LR)$ is the solution of the optimization problem: $\max_{\mu \in \mathbb{R}^n} v(LR(\mu))$.

Determination of the Lagrangian Bound

The Lagrangian Bound $v(LR)$ is determined with a cutting plane/column generation procedure.

Solving the Lagrangian Bound Problem (1)

Said H the set of feasible solutions of $\mathbf{SP(\mu)}$, the Lagrangian Bound problem can be written as:

$$\max_{\mu}\Big\{\sum_{j=1}^{n}\mu_j + K\min_{h=1,\dots,H}\Big\{\sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\Big\}\Big\}.$$

Defining

$$\theta := \min_{h=1,\dots,H}\Big\{\sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\Big\},$$

the Lagrangian Bound can be determined by solving the Lagrangian Master Problem (LMP):

$$\textbf{[LMP]}\ \max_{\mu}\ \sum_{j=1}^{n}\mu_j + K\theta$$

$$s.t\quad \theta \le \sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}x_{i,j}^{(h)}\mu_j \quad h\in H$$

Since the set H is not fully known beforehand, an iterative procedure starting from a relaxation of LMP (**RLMP**) is developed.

Solving the Lagrangian Bound Problem (1)

Said H the set of feasible solutions of $\mathbf{SP}(\mu)$, the Lagrangian Bound problem can be written as:

$$\max_{\mu}\left\{\sum_{j=1}^{n}\mu_j + K \min_{h=1,\dots,H}\left\{\sum_{i=1}^{Q+1} T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\right\}\right\}.$$

Defining

$$\theta := \min_{h=1,\dots,H}\left\{\sum_{i=1}^{Q+1} T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\right\},$$

the Lagrangian Bound can be determined by solving the Lagrangian Master Problem (LMP):

$$\textbf{[LMP]} \quad \max_{\mu} \quad \sum_{j=1}^{n}\mu_j + K\theta$$

$$s.t \quad \theta \le \sum_{i=1}^{Q+1} T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q} x_{i,j}^{(h)}\mu_j \quad h \in H$$

Since the set H is not fully known beforehand, an iterative procedure starting from a relaxation of LMP (**RLMP**) is developed.

Solving the Lagrangian Bound Problem (1)

Said H the set of feasible solutions of **SP(μ)**, the Lagrangian Bound problem can be written as:

$$\max_{\mu}\left\{\sum_{j=1}^{n}\mu_j + K\min_{h=1,\ldots,H}\left\{\sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\right\}\right\}.$$

Defining

$$\theta := \min_{h=1,\ldots,H}\left\{\sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\right\},$$

the Lagrangian Bound can be determined by solving the Lagrangian Master Problem (LMP):

$$\textbf{[LMP]}\ \max_{\mu}\quad \sum_{j=1}^{n}\mu_j + K\theta$$

$$s.t\quad \theta \le \sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}x_{i,j}^{(h)}\mu_j \quad h \in H$$

Since the set H is not fully known beforehand, an iterative procedure starting from a relaxation of LMP (**RLMP**) is developed.

Solving the Lagrangian Bound Problem (1)

Said H the set of feasible solutions of $\mathbf{SP}(\mu)$, the Lagrangian Bound problem can be written as:

$$\max_{\mu}\Big\{\sum_{j=1}^{n}\mu_j + K\min_{h=1,\ldots,H}\Big\{\sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\Big\}\Big\}.$$

Defining

$$\theta := \min_{h=1,\ldots,H}\Big\{\sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}\mu_j x_{i,j}^{(h)}\Big\},$$

the Lagrangian Bound can be determined by solving the Lagrangian Master Problem (LMP):

$$\textbf{[LMP]}\ \max_{\mu} \quad \sum_{j=1}^{n}\mu_j + K\theta$$

$$s.t \quad \theta \leq \sum_{i=1}^{Q+1}T_i^{(h)} - \sum_{j=1}^{n}\sum_{i=1}^{Q}x_{i,j}^{(h)}\mu_j \quad h \in H$$

Since the set H is not fully known beforehand, an iterative procedure starting from a relaxation of LMP (**RLMP**) is developed.

Solving the Lagrangian Bound Problem (2)

Cutting plane procedure

Iterative scheme:

- Solve the current RLMP $\rightarrow$ Solution μ
- Solve subproblem SP(μ) $\rightarrow$ Solution h corresponds to cut h in LMP.
- Add cut h to RLMP.

Bounds computation:

- $\sum_{j=1}^{n} \mu_j + K\theta$ is an upper bound on the Lagrangian Bound
- $K \cdot v(SP(\mu)) + \sum_{j=1}^{n} \mu_j$ is a lower bound on the Lagrangian Bound

Termination criterion:

- Difference between lower and upper bound is sufficiently small.

Branch-and-Price algorithm

Branch-and-Price motivation

The Lagrangian Bound solution is not necessarily feasible for the assignment constraints → Branch & Price framework.

Branching constraints

- Given two targets j_1 and j_2 assigned to multiple vehicles, two child nodes (1) and (2) are created:

$$\sum_{i=1}^{Q} x_{i,j_1}^k = \sum_{i=1}^{Q} x_{i,j_2}^k \quad k = 1, \dots, K, \tag{1}$$

$$\sum_{i=1}^{Q} x_{i,j_1}^k + \sum_{i=1}^{Q} x_{i,j_2}^k \leq 1 \quad k = 1, \dots, K. \tag{2}$$

- The branching constraints are compatible with the subproblem decomposition.

Branch-and-Price

Tree exploration

- Depth-first strategy.
- Initial incumbent provided by a simple assignment heuristic.
- Each node is solved with the cutting plane procedure.
- Fathoming rules:
  - Node feasible for the assignment constraints $\rightarrow$ Valid upper bound $\rightarrow$ Possible incumbent update.
  - Node infeasible for the branching constraints.
  - Node solution is worse than the incumbent.
- Termination criterion: List of open nodes is empty.

Implementation

- Branch & Price implemented in *C*.
- Master problems and subproblems solved by CPLEX 12.6.1 using *C* callable libraries.

Test Instances

18 randomly generated instances:

- $n = 10, \dots, 20$;
- $K = 3, 4, 5$;
- $Q = \lceil \frac{n}{K} \rceil + 2$;
- q_j are randomly generated in the Euclidean space centered at $(0, 0)$ and with width 50 and height 100.
- $O = (-20, 0)$, $D = (20, 0)$.
- V generated from $\mathcal{U}[2, 3]$.
- v_j generated from $\mathcal{U}[0.1, 1]$.
- d_j represented by a random angle.

Machine features

QEMU Virtual CPU version 0.14.1 @ 2.40 GHz.

Time limit of 2 hours.

	Branch-and-Price				CPLEX			
Instance Name	Upper Bound	Lower Bound	Gap(%)	CPU Time (s)	Upper Bound	Lower Bound	Gap(%)	CPU Time (s)
p_10_3_6	72.69	72.69	0%	**183.84**	72.69	72.69	0%	229.61
p_10_4_5	62.14	62.14	0%	**70.38**	62.14	62.14	0%	449.27
p_10_5_4	71.71	71.71	0%	**161.28**	71.71	71.71	0%	542.51
p_12_3_6	76.91	76.91	0%	**377.43**	76.91	76.91	0%	579.70
p_12_4_5	91.36	91.36	0%	**1209.43**	91.36	84.05	*8%	>7200
p_12_5_5	80.08	80.08	0%	**532.05**	80.08	80.08	0%	4107.24
p_14_3_7	84.47	84.47	0%	**3841.94**	84.47	84.47	0%	6961.36
p_14_4_6	97.62	97.62	0%	**5131.43**	103.97	90.45	13%	>7200
p_14_5_5	80.18	80.18	0%	**377.12**	80.18	73.77	*8%	>7200
p_16_3_8	181.58	78.08	57%	>7200	**85.84**	**78.97**	8%	>7200
p_16_4_6	75.85	75.85	0%	**6635.41**	76.71	75.18	2%	>7200
p_16_5_6	91.68	91.68	0%	**6084.35**	96.7	64.79	33%	>7200
p_18_3_8	217.47	**97.86**	55%	>7200	**132.58**	86.18	35%	>7200
p_18_4_7	**80.68**	**75.03**	7%	>7200	87.64	60.47	31%	>7200
p_18_5_6	76.18	76.18	0%	**1132.36**	78.73	59.05	25%	>7200
p_20_3_9	193.88	50.41	74%	>7200	**104.74**	**67.03**	36%	>7200
p_20_4_7	216.63	**106.15**	51%	>7200	**127.02**	83.83	34%	>7200
p_20_5_6	**128.35**	**114.23**	11%	>7200	133.24	97.27	27%	>7200

* indicates that the upper bound is the optimal solution however the gap is due to the lower bound.

- B&P solved 12 instances to optimality, while CPLEX solved 5 instances.
- B&P found a better lower bound for 4 instances, while CPLEX found a better upper bound for 4 instances.

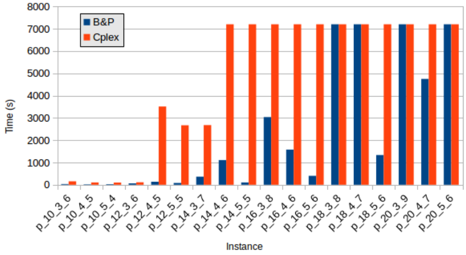
Comparison between Branch-and-Price and CPLEX for VRPFL with fixed line direction.

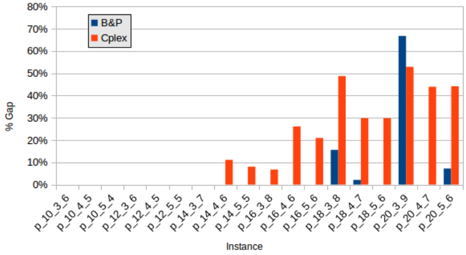
	Branch-and-Price				CPLEX			
Instance Name	Upper Bound	Lower Bound	Gap(%)	CPU Time (s)	Upper Bound	Lower Bound	Gap(%)	CPU Time (s)
p_10_3_6	85.51	85.51	0%	**31.40**	85.51	85.51	0%	154.09
p_10_4_5	72.06	72.06	0%	**20.56**	72.06	72.06	0%	104.20
p_10_5_4	83.33	83.33	0%	**25.95**	83.33	83.33	0%	99.32
p_12_3_6	94.55	94.55	0%	**60.57**	94.55	94.55	0%	111.76
p_12_4_5	109.34	109.34	0%	**137.00**	109.34	109.34	0%	3510.18
p_12_5_5	91.55	91.55	0%	**79.52**	91.55	91.55	0%	2669.57
p_14_3_7	101.43	101.43	0%	**361.97**	101.44	101.44	0%	2676.76
p_14_4_6	105.19	105.19	0%	**1109.85**	113.99	101.45	11%	>7200
p_14_5_5	89.31	89.31	0%	**104.39**	89.97	82.77	8%	>7200
p_16_3_8	100.63	100.63	0%	**3035.48**	100.63	93.59	*7%	>7200
p_16_4_6	97.91	97.91	0%	**1577.55**	111.85	82.77	26%	>7200
p_16_5_6	105.37	105.37	0%	**401.00**	124.07	98.02	21%	>7200
p_18_3_8	**167.57**	**140.76**	16%	>7200	198.62	101.3	49%	>7200
p_18_4_7	**85.96**	**84.24**	2%	>7200	99.84	69.89	30%	>7200
p_18_5_6	97.93	97.93	0%	**1333.93**	110.27	77.19	30%	>7200
p_20_3_9	248.08	**81.87**	67%	>7200	**159.38**	74.91	53%	>7200
p_20_4_7	128.94	128.94	0%	**4750.29**	169.41	94.87	44%	>7200
p_20_5_6	**137.36**	**127.74**	7%	>7200	187.12	104.79	44%	>7200

* indicates that the upper bound is the optimal solution however the gap is due to the lower bound.

- B&P solved 14 instances to optimality, while CPLEX solved 7 instances.
- VRPFL with fixed line direction is computationally less challenging to solve than the general VRPFL.

VRPFL with fixed direction: Branch-and-Price vs. CPLEX - Time and Gap





Details of the Branch-and-Price performance for the general VRPFL

		Root Node			Other Nodes†		
Instance Name	Number of Nodes	Iterations	CPU Master Problem	CPU Subproblem	Avg. Iterations	Avg. CPU Master Problem	Avg. CPU Subproblem
p_10_3_6	17	29	<0.01	60.18	5	0.02	7.71
p_10_4_5	1	37	0.10	70.26	-	-	-
p_10_5_4	41	30	<0.01	25.26	4	<0.01	3.40
p_12_3_6	1	63	<0.01	377.39	-	-	-
p_12_4_5	147	27	<0.01	62.71	4	<0.01	7.85
p_12_5_5	67	39	<0.01	88.7	5	<0.01	6.72
p_14_3_7	1	109	0.02	3841.8	-	-	-
p_14_4_6	157	51	<0.01	532.1	6	<0.01	29.48
p_14_5_5	15	50	<0.01	143.99	7	<0.01	16.65
p_16_3_8	1	97	0.11	>7200	-	-	-
p_16_4_6	83	66	<0.01	1182.2	7	<0.01	66.49
p_16_5_6	119	51	0.03	484.03	7	<0.01	47.45
p_18_3_8	1	67	0.01	>7200	-	-	-
p_18_4_7	39	62	<0.01	2275.28	7	<0.01	134.43
p_18_5_6	1	81	0.01	1132.23	-	-	-
p_20_3_9	1	39	<0.01	>7200	-	-	-
p_20_4_7	1	103	0.02	>7200	-	-	-
p_20_5_6	187	77	0.01	2254.2	5	<0.01	27.35

† indicates the average results over all the nodes except the root node in the branch-and-price tree.

- indicates that the branch-and-price stopped at the root node.

- Computational bottleneck is in solving the MISOCP subproblems.
- Child nodes are less time-consuming thanks to warm starting techniques.

Details of the Branch-and-Price performance for VRPFL with fixed line direction.

		Root Node			Other Nodes†		
Instance Name	Number of Nodes	Iterations	CPU Master Problem	CPU Subproblem	Avg. Iterations	Avg. CPU Master Problem	Avg. CPU Subproblem
p_10_3_6	1	34	<0.01	31.39	-	-	-
p_10_4_5	1	27	<0.01	20.55	-	-	-
p_10_5_4	21	22	<0.01	6.91	4	<0.01	0.95
p_12_3_6	1	40	<0.01	60.55	-	-	-
p_12_4_5	39	30	<0.01	27.42	5	<0.01	2.88
p_12_5_5	9	34	<0.01	30.26	9	<0.01	6.15
p_14_3_7	1	56	<0.01	361.91	-	-	-
p_14_4_6	45	57	<0.01	230.12	7	<0.01	19.99
p_14_5_5	7	38	<0.01	52.75	8	<0.01	8.60
p_16_3_8	1	83	0.01	3035.33	-	-	-
p_16_4_6	41	62	<0.01	353.82	7	<0.01	30.58
p_16_5_6	3	63	<0.01	314.19	12	<0.01	43.37
p_18_3_8	48	83	0.01	2665.69	5	<0.01	97.05
p_18_4_7	45	76	0.01	1409.48	12	<0.01	144.74
p_18_5_6	21	68	0.03	536.00	9	<0.01	39.89
p_20_3_9	1	58	0.01	7199.86	-	-	-
p_20_4_7	7	95	0.39	2549.12	16	0.07	366.69
p_20_5_6	349	73	0.01	589.63	6	<0.01	19.00

† indicates the average results over all the nodes except the root node in the branch-and-price tree.

- indicates that the branch-and-price stopped at the root node.

- For 5 instances, the Lagrangian Bound at the root node is the VRPFL optimal value.
- Branch-and-Price explored a limited number of nodes to find the optimal solutions.

Conclusions

Summary

- We studied a dynamic variant of VRP and developed a novel mathematical formulation.
- We implemented a Branch-and-Price algorithm based on a Lagrangian Relaxation.
- Numerical testing show the effectiveness of the Branch & Price approach against CPLEX.

Future research directions

- Test CPLEX and Branch-and-Price on instances with different capacities.
- Investigate approaches to efficiently solve the single vehicle routing with floating targets problem may highly improve the performance of the presented Branch-and-Price approach.
- Other applications?

Thank you! Questions?