ELSEVIER
Available online at www.sciencedirect.com
ScienceDirect
Procedia - Social and Behavioral Sciences 111 (2014) 203 – 212
Procedia
Social and Behavioral Sciences

EWGT2013 – 16th Meeting of the EURO Working Group on Transportation

The Multi-Objective Multi-Vehicle Pickup and Delivery Problem with Time Windows

L. Grandinetti[a], F. Guerriero[b,*], F. Pezzella[c], O. Pisacane[c]

[a] *Dipartimento di Ingegneria Informatica, Modellistica, Elettronica e Sistemistica- Università della Calabria-Via P. Bucci, Rende 87036,Italy*
[b] *Dipartimento di Ingegneria Meccanica, Energetica e Gestionale- Università della Calabria-Via P. Bucci, Rende 87036,Italy*
[c] *Dipartimento di Ingegneria dell'Informazione- Università Politecnica delle Marche-Via B. Bianche 12, Ancona 60131, Italy*

Abstract

The *Single Objective Single Vehicle Pickup and Delivery Problem* (*SOSV-PDP*) is a *Vehicle Routing Problem* (*VRP*) in which the vehicle, based at the depot, has to visit exactly once a set of customers with known demand. Each request specifies two locations: an origin for the picking and one for the delivery. The vehicle must start and finish at the depot and the total handled demand must not exceed its capacity. Moreover, for each request, the origin must precede the destination (*precedence constraints*). In the *SOSV-PDP with Time Windows* (*SOSV-PDPTW*), each request specifies also a time window. Therefore, the vehicle has to serve the customer within the time window (*time window constraint*). The *Single Objective Multiple Vehicle-PDPTW* (*SOMV-PDPTW*) is an extension of *SOSV-PDPTW* where customers are served by a fleet (usually homogeneous) of vehicles. Therefore, together with the precedencies, for each request, the origin and the destination have to belong to the same route (*pairing constraints*). In the traditional *SOMV-PDPTW*, only one objective is optimized (usually, the total travel cost); while, in the literature, few multi-objective *MOSV-PDPTW* exist that optimize at most three criteria simultaneously. The contribution of this paper consists in addressing the *MOMV-PDPTW* from both a modeling and methodological point of view. In fact, the *MOMV-PDPTW* is firstly modeled with the aim of optimizing the number of vehicles, the total travel cost and the longest travel cost, simultaneously; then, a two-step solution approach is proposed. In particular, in the first step, a set of feasible routes is generated by properly adapting some meta-heuristics proposed in literature for the *SOMV-PDPTW*; then, set partitioning optimization problems are solved within an *ϵ-constraint framework*. More specifically, each set partitioning problem selects the routes from the feasible set, optimizing one criterion at time, constraining the remaining ones by appropriate upper bounds and satisfying customer requirements. Finally, the second step finds the set of efficient solutions for approximating the *Pareto Fronts*. Computational experiments, carried out on some instances generated in literature, show that our approach determines good quality *Efficient Pareto Fronts* (in terms of number of efficient solutions) and also provides well-diversified efficient sets. This last aspect is properly evaluated by computing the *Spread* metric on each of the instances.

* Corresponding author. Tel.: +39-0984-494620.
E-mail address: francesca.guerriero@unical.it

1877-0428 © 2013 The Authors. Published by Elsevier Ltd.
Selection and/or peer-review under responsibility of Scientific Committee
doi:10.1016/j.sbspro.2014.01.053

Keywords: Multi-Objective Mathematical Programming; Vehicle Routing; Pickup and Delivery; ε-Constraint Method.

1. Introduction

The *MOMV-PDPTW* aims at serving a set of *n* customers by using a fleet of *m* vehicles. In this paper, the fleet is homogeneous and therefore, all the vehicles have the same capacity *Q* (i.e., maximum number of customers to transport) and the same average speed. A *route* is here associated to only one vehicle and has to start/end from/to the *depot*. Each customer specifies the *pickup* and the *delivery* point together with a *time window* for the service. In addition, each service could be required for more than one customer and therefore, the number of passengers to transport (*demand*) is also specified. In this work, the *total travel cost (c)*, the *travel cost of the longest route* (i.e., *makespan* denoted by Ψ), and the number of used vehicles (μ) are simultaneously optimized. The transport network is modeled by a directed graph $G=<V,A>$, where $V=\{0, 1, ..., n, n+1, ..., 2n, 2n+1\}$ is the vertex set and A is the arc set. Both 0 and $2n+1$ represent the depot. The subset $P \subset V$ of vertices consists of the pickup points whereas $D \subset V$ (such that $P \cap D = \emptyset$) contains the delivery points. For each $p \in P$, $q_p > 0$ represents its demand and $[a_p, b_p]$ the corresponding time window; for each $d \in D$, $q_d < 0$ denotes its demand and $[a_d, b_d]$ the related time window. It is also assumed that $|P|=|D|$ and thus, to each $p \in P$, a $d \in D$ is associated such that $q_p=|q_d|$ (i.e., the customers have to be picked-up at p and delivered at d). Among all the arcs $(i, j) \in A$ only those satisfying the following condition are included in the network: $t_{0i}+s_i+t_{ij} \le b_j$, where t_{ij} denotes the time distance between i and j ($i \in V$, $j \in V$) whereas s_i represents the service time $\forall i \in V$. Finally, $\forall p \in P$, *dev(p)* denotes its delivery point whereas $\forall d \in D$, *pick(d)* represents its pickup point. In this paper, an approximation of the *Efficient Pareto Fronts* is determined applying the *ϵ-constraint* method (Chankong and Haimes,1983) whose main idea is to solve a sequence of constrained single-objective problems. This choice is motivated by the fact that this method has been successfully applied to a variety of VRPs (we cite, for example, the recent paper of Grandinetti, Guerriero, Laganà & Pisacane, 2012). The main contributions of this work consist in: proposing a single-objective formulation (*M1*) of *PDPTW* with the aim of minimizing a weighted sum that properly combines μ and c; extending *M1* to a multi-objective formulation (*M2*) with the aim of minimizing μ, c and Ψ simultaneously; formulating a Mixed Integer Linear Programming (*MILP*) model (*M3*) for *MOMV-PDPTW*, based on binary route-variables and on the continuous variable Ψ, starting from *M2* and finally, determining an approximation of the efficient solution set by applying an *ϵ-constraint* method to *M3* that is formulated considering μ as an input parameter. Indeed, the proposed approach starts with the maximum available number of vehicles ($\mu = \overline{m} = m$) and it iteratively decreases μ until the problem becomes infeasible. The remainder of this paper is organized as follows. Section 2 briefly overviews the literature; Section 3 introduces *M3,* whereas the proposed *ϵ-constraint* approach is described in Section 4. Section 5 shows some computational results on a set of instances taken from the literature and finally, Section 6 concludes the work.

Nomenclature

SETS

N	set of $2n$ customers
P	set of n pickup points
D	set of n delivery points
K	set of m identical vehicles
V	set of the $2n+2$ vertices
A	set of feasible arcs
Ω	set of feasible routes

INPUT DATA	
Q	capacity of each vehicle
δ	fixed cost of using a vehicle
M	Distance matrix $M \in R^{(2n+2)\times(2n+2)}$ such that m_{ij} is the distance between the vertices i and j
C	cost matrix $C \in R^{(2n+2)\times(2n+2)}$ such that c_{ij} is the travel cost between the vertices i and j
S	service time matrix $S \in R^{(2n+2)\times 1}$ such that s_i is the service time required by the vertex i
$\overline{Q}$	demand matrix $\overline{Q} \in R^{(2n)\times 1}$ such that q_i is the demand of the vertex i
T	time distance matrix $T \in R^{(2n+2)\times(2n+2)}$ such that t_{ij} is the time distance between the vertices i and j
Z	binary matrix $Z \in R^{\|P\|\times\|\Omega\|}$ such that $z_{i\omega}$ is equal to 1 if the pickup point i is served in the route ω
OUTPUT DATA	
X	assignments matrix $X \in R^{(2n+2)\times(2n+2)\times m}$ such that x_{ij}^{k} is equal to 1 if vehicle k uses the arc (i,j)
Ψ	makespan
Π	routes matrix $\Pi \in R^{\|\Omega\|\times 1}$ such that π_{ω} is equal to 1 if the feasible route $\omega \in \Omega$ is selected
η	number of non-dominated solutions

2. Related works

In the traditional *VRP* (see Toth & Vigo, 2002), a fleet of vehicles, based at the depot, handles customer demands known in advance. The vehicles have to be routed in such a way that all the customers are visited excatly once; each route starts/ends at/to the depot and the total served demand does not exceed vehicle capacity (*capacity constraints*). A large number of scientific contributions have addressed this problem from both a modeling and methodological point of view: Araque, Kudva, Morin and Pekny (1994) have proposed a *Branch&Cut* algorithm for the identical customer *VRP* transforming it into an equivalent *Path-Partitioning Problem*; Bramel and Simchi-Levi (2000) have designed set-partitioning based solution approaches for the *capacitated VRP* (*CVRP*); Baldacci, Hadjiconstantinou and Mingozzi (2004) have introduced an IP formulation of *CVRP* and Ropke (2005) has described both heuristic and exact solution approaches for the *VRP* and for its variants (e.g., *VRP* with *Time Window* (*VRPTW*), *Pickup and Delivery Problem* (*PDP*) and *PDP* with *Time Window* (*PDPTW*)). In the *VRPTW*, together with the demand, each customer also specifies a *time window* for the service. It means that the vehicle cannot serve it before the beginning and after the end of the *time window* (i.e., *hard time window constraints*). In other variants, it could be allowed violating these bounds by imposing *soft time window constraints*. Lysgaard (2006) has recently introduced a set of *reachability cuts* for the *VRPTW* that are closely related to the ones derived from the precedence constraints in the Asymmetric Traveling Salesman Problem with Time Windows. In the *PDP*, each request is associated to a pickup and to a delivery point: the former is the location at which a certain demand has to be picked up whereas the latter is the point at which the same demand has to be delivered. Each pickup point has to be visited always before of its delivery point (*precedence constraint*) and both of them have to belong to the same route (*payring constraint*). The problem is also known in literature as *Dial and Ride Problem* (*DARP*) and it has been extensively analyzed: Cordeau (2006) has proposed a mixed-IP formulation together with a *Branch&Cut* algorithm. In the *single-vehicle PDP*, only one vehicle has to satisfy the whole demand. However, a more general version consists in using more than one vehicle and it is denoted as the *Multiple Vehicles PDP* (*MV-PDP*): Lu and Dessouky (2004) have addressed the *MV-PDP* for minimizing *c* by fixing the vehicle cost. They have also developed a solution approach without tight constraints, modeling it as a 0-1 IP problem and a *Branch&Cut* algorithm. A further generalization of the *PDP* is the *PDP* with *Time Windows* (*PDPTW*) in which, together with the demand, the pickup and the delivery, a time window is also associated to each request. The *PDPTW* has been analyzed in Ropke, Cordeau and Laporte (2007) where two formulations have been proposed imposing a limit on the elapsed time between the pickup and the delivery. Moreover, they have also introduced valid inequalities to strengthen these formulationsthat have been used within *Branch&Cut* algorithms. Ropke and Cordeau (2009) have proposed a *Branch&Cut&Price* algorithm for the *PDPTW* and they have computed the lower bounds applying a column generation to the LP relaxation of a set partitioning formulation. Instead, Li and Lim (2003) have designed a *Tabu-embedded Simulated Annealing Algorithm* (*TSAA*) for the *MV-PDPTW* that restarts a search from the current best solution after several non-improving iterations. The computational results have underlined that *TSAA* is the best performing approach to solve large scale *MV-PDPTW* and therefore, it will be considered hereafter as reference heuristic. However, these scientific contributions have considered the *Single-objective PDPTW*. In this paper, a *Multi-objective MV-PDPTW* (*MOMV-PDPTW*) is addressed for minimizing c, Ψ and μ. At the best of our kwnoledge, the only existing contribution for *MOMV-PDPTW* is the paper of Harbaoui, Kammarti, Ksouri and Borne (2011) in which a genetic algorithm for minimizing c and the total tardiness time has been proposed. However, the minimization of μ has been not taken into consideration and the quality of the proposed approach has not been investigated by using mathematical metrics. The main goal of this paper is to define an approximate *є-constraint* based solution approach for *MOMV-PDPTW* whose effectiveness will be discussed with reference to the number of non-dominated solutions (η) and to the diversity of this set by considering the *Spread* metric. The experimental analysis will be conducted on a sub-set of the istances proposed in Li and Lim (2003).

3. A mathematical model for the MOMV-PDPTW

In this Section, *MOMV-PDPTW* is mathematically modeled in order to minimize c, Ψ and μ. In particular, Section 3.1 presents a mathematical formulation for *SOMV-PDPTW* (*M1*) whereas Section 3.2 extends *M1* to *MOMV-PDPTW* (*M2*).

3.1. A mathematical model for the SOMV-PDPTW

SOMV-PDPTW is mathematically formulated as a *MILP* introducing the following decision variables: x_{ij}^k a binary variable equal to 1 if the arc (i,j) belongs to the route k (i.e., the vehicle k uses it), 0 otherwise; B_i^k a continuous variable that represents the time at which the vehicle k starts serving the vertex i; Q_i^k an integer variable that denotes the remaining capacity of the vehicle k before leaving the vertex i; y^k a binary variable equal to 1 if the vehicle k is used. The mathematical formulation of *SOMV-PDPTW* assumes the following form:

$$\min \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} c_{ij}^k x_{ij}^k + \delta \sum_{k \in K} y^k \quad (1)$$

$$\sum_{k \in K} \sum_{j \in V | j \neq i} x_{ij}^k = 1 \; \forall i \in P \quad (2)$$

$$\sum_{j \in V | j \neq i} x_{ij}^k - \sum_{j \in V | j \neq i} x_{dev(i)j}^k = 0 \; \forall i \in P, k \in K \quad (3)$$

$$\sum_{j \in V | j \neq 0} x_{0j}^k = y^k \; \forall k \in K \quad (4)$$

$$\sum_{j\in V|j\neq 2n+1} x_{j(2n+1)}^k = y^k \;\forall k \in K \quad (5)$$

$$\sum_{i\in V}\sum_{j\in V|j\neq i} x_{ij}^k \le W y^k \;\forall k \in K \quad (6)$$

$$\sum_{i\in V}\sum_{j\in V|j\neq i} x_{ij}^k \ge y^k \;\forall k \in K \quad (7)$$

$$\sum_{j\in V|i\neq j} x_{ji}^k - \sum_{j\in V|i\neq j} x_{ij}^k = 0 \;\forall k \in K, i \in PUD \quad (8)$$

$$B_j^k - B_i^k \ge (s_i + t_{ij}) x_{ij}^k - W(1 - x_{ij}^k) \;\forall i, j \in V \mid i \neq j, k \in K \quad (9)$$

$$Q_j^k - Q_i^k \ge q_j x_{ij}^k - W(1 - x_{ij}^k) \;\forall i, j \in V \mid i \neq j, k \in K \quad (10)$$

$$B_i^k - B_{dev(i)}^k + t_{idev(i)} \sum_{j\in V} x_{ij}^k \le 0 \;\forall i \in P, k \in K \quad (11)$$

$$B_i^k \ge a_i \sum_{j\in V} x_{ij}^k + \phi_i \;\forall i \in V, k \in K \quad (12)$$

$$B_i^k \le b_i \sum_{j\in V} x_{ij}^k \;\forall i \in V, k \in K \quad (13)$$

$$\max\{0, q_i\} \sum_{j\in V} x_{ij}^k \le Q_i^k \le \min\{Q, Q + q_i\} \sum_{j\in V} x_{ij}^k \;\forall i \in V, k \in K \quad (14)$$

$$x_{ij}^k \in \{0,1\} \;\forall i, j \in V \mid i \neq j, k \in K \quad (15) \qquad y^k \in \{0,1\} \;\forall k \in K \quad (16)$$

The objective function (1) to minimize combines c and μ by introducing a parameter δ that represents the cost of using a vehicle (assumed equal for all vehicles); the constraints (2) impose that each request has to be satisfied whereas the constraints (3) ensure the *pairing conditions*. The constraints (4) and (5) assure that for each route, only one arc leaves the depot and only one arc enters it, respectively. The constraints (6) and (7) link the x-variables to the y-variables: if at least one arc *(i,j)* traversed by the vehicle k exists, then k has to be considered used ($y^k = 1$); on the other hand, if k does not traverse the arc *(i,j)*, then the corresponding x-variable is 0. Moreover, W is a large non-negative scalar. The constraints (8) are the *flow conservation constraints* $\forall i \in V \mid i \neq 0 \wedge i \neq 2n+1$ and they impose that $\forall i \in V, i \neq 0 \wedge i \neq 2n+1, k \in K$, the number of ingoing arcs has to be equal to the number of outgoing arcs. The constraints (9) ensure that each vertex in a route has to be served after its predecessor. The constraints (10) properly update the remaining capacity of a vehicle before leaving a vertex, whereas the constraints (11) impose the *precedence constraints*. The constraints (12) assure that the lower bounds on each time window are satisfied. In order to consider the same framework of Li and Lim (2003), a threshold ϕ_i is introduced for representing the *flexibility degree* of the vertex i. The constraints (13) impose that the upper bounds on the time windows are satisfied. The constraints (14) assure that Q is never exceeded in each route. Finally, the constraints (15) and (16) impose the binary conditions on the x-variables and y-variables, respectively. This formulation significantly extends the one proposed in Ropke and Cordeau (2009) explicitly minimizing μ and modifying the time window constraints.

3.2. A mathematical model for the MOMV-PDPTW

In this section, *MOMV-PDPTW* is modeled considering the following three objective functions to optimize:

$$\min \sum_{i\in V}\sum_{j\in V|j\neq i}\sum_{k\in K} c_{ij}^{k} x_{ij}^{k} \quad (17) \qquad \min \sum_{k\in K} y^{k} \quad (18) \qquad \min \Psi \quad (19)$$

where (17), (18) and (19) represent the minimization of c, μ and Ψ, respectively. The set of constraints (2)—(16) are also included in the model together with the constraints (20) for imposing that $\Psi>0$ is greater or equal to the total travel cost of each designed route:

$$\Psi \geq \sum_{i\in V}\sum_{j\in V|j\neq i} c_{ij}^{k} \quad \forall k \in K \quad (20)$$

Therefore, the model taken into consideration hereafter is (17)—(19),(2)—(16),(20).

4. An approximate ∈-constraint based solution approach for the MOMV-PDPTW

Since the model (17)—(19), (2)—(16),(20) cannot be efficiently solved, the *Efficient Pareto Fronts* have been approximated by defining the following optimization model based on binary routes variables in which: Ω represents a set of feasible routes, c_ω denotes the cost associated to the route $\omega\in\Omega$, π_ω is a binary decision variable equal to 1 if the route ω is selected and Z is a binary input matrix in which $z_{i\omega}$ equal to 1 if the pickup i is served by the feasible route ω:

$$\min \sum_{\omega\in\Omega} c_\omega \pi_\omega \quad (21) \qquad \min \Psi \quad (22) \qquad \min \sum_{\omega\in\Omega} \pi_\omega \quad (23)$$

$$\sum_{\omega\in\Omega} z_{i\omega}\pi_\omega = 1 \; \forall i \in P \quad (24)$$

$$\pi_\omega \in \{0,1\} \forall \omega \in \Omega \quad (25)$$

where (21), (22) and (23) minimize the total travel cost, the makespan and the number of used vehicles, respectively; the constraints (24) impose that each pickup is served by one and only one vehicle. Therefore, for a given instance Γ, the approximate *∈-constraint* based approach (see Grandinetti et al. 2012, for example) firstly determines valid lower and upper bounds on μ, c and Ψ. Then, it properly populates Ω with feasible routes generated by *TSAA*. Finally, it selects from Ω a set of routes by solving single objective problems (Section 4.1) properly derived from (21)—(25) The lower bound on μ ($\underline{m}$) is set equal to the one of the best solution obtained by *TSAA,* whereas the lower and upper bounds on c and Ψ ($\underline{c},\underline{\Psi}$ and $\bar{c},\bar{\Psi}$, respectively) are equal to the best and the worst values detected by *TSAA*. Indeed, since *TSAA* optimizes a weighted sum that combines μ, c, the total waiting time and the total schedule time, it has been executed different times varying these weights. The best and the worst values have been determined over all the performed runs.

4.1. Generating non-dominated solutions

In the proposed approximate *∈-constraint* approach, μ comes into play as a parameter (i.e., fixed at each macro-iteration). Since it varies from $\underline{m}$ to $\bar{m}$, the optimization process minimizes all the three objectives simultaneously. Then, two single-objective formulations (*P1* and *P2*) are deduced by (21)—(25). For each value of μ, the problem *P1* minimizes c, and imposes an upper limit on Ψ. Therefore, its objective function to minimize is (21), subject to (24) and (25) and the following constraints:

$$\Psi \geq c_\omega \pi_\omega \quad \forall \omega \in \Omega \quad (26)$$

$$\sum_{\omega\in\Omega} \pi_\omega \leq m \quad (27)$$

$$\Psi \leq \varepsilon_1 \quad (28)$$

where constraints (26) ensure that Ψ is greater or equal to the cost of each route, condition (27) imposes a limit on the number of selected routes, whereas the constraint (28) imposes an upper bound on Ψ . On the other hand, for a given value of μ, *P2* ensures the satisfaction of an upper limit on c imposing (24)--(27), minimizing (22) and introducing the following constraint on the value of c

$$\sum_{\omega\in\Omega} c_{\omega}\pi_{\omega} \le \varepsilon_2 \quad (29)$$

The complete algorithm is outlined in the following scheme:
Step 1: Populate Ω;
Step 2: For each $\mu\in[\underline{m},\overline{m}]$

Step 2.1: For each $\varepsilon_1 \in [\underline{\Psi},\overline{\Psi}]$ F←Optimize *P1*;

Step 2.2: For each $\varepsilon_2 \in [\underline{c},\overline{c}]$ F←Optimize *P2*;

Step 3: F← Remove the dominated solutions.
where F contains the non-dominated solutions and approximates the *Efficient Pareto front*. It is worth noting that Ω contains also the *singletons* that is routes in which only one couple (pickup, delivery) is served. This ensures that a feasible solution can be always found. The value of μ varies in its range with step equal to 1. Finally, ε_1 and ε_2 are varied in their ranges by a step equal to 1 since this value ensures that a sequence of $\in$-constrain problems generates one feasible solution for each point of the Pareto front (Theorem 3 of Berube, Gendreau and Potvin (2009)).

5. Computational results

The proposed approach has been implemented in *Java* and all the optimization models have been solved by using *ILOG CPLEX* library (*release10.1*). The experiments have been carried on a cluster of 6 *Pentium* machines, each supported by *Intel(R) Xeon(R)* processor technology (*X5680*) and clocked at 3.33 GHz, with a totally 50 GB RAM. The computational analysis has been performed in two steps. Firstly, an accurate and intensive sensitivity analysis has been performed on the proposed model (1)—(16) by properly varying δ and □$_i$ (for each vertex i) in order to analyze their effects on the performances. According to the obtained results, we can conclude that the model is more sensible to □$_i$. Then, a set of instances (available at http://www.sintef.no/Projectweb/TOP/PDPTW/Li--Lim-benchmark/) has been taken into consideration: *LR101*, *LR102*, *LR103*, *LR104*, *LR105*, *LR106* and *LRC105*. All these instances present a fleet of 25 vehicles whose capacity is equal to 200. Moreover, the value of $|N|$ is specified in Table 1 for each instance and the distance between two nodes is computed by using the Euclidean metric. In the same table, the number of feasible routes generated for each instance is also given. Since at the best of our knowledge, *MOMV-PDPTW* has not been previously analyzed in the literature with reference to our objectives, the *Efficient Pareto Fronts* are unknown. This consequently prevents us to apply metrics generally used to measure the extent of convergence to the *Efficient Optimal Set*. Therefore, in order to conclude about the quality of our method, the *Spread* has been used as a metric. It has been introduced in Deb (2001) and given a solution *Sol*, it is defined as:

$$Spread(Sol) = \frac{d_f + d_l + \sum_{i=1}^{\eta-1}\left|d_i - \overline{d}\right|}{d_f + d_l + (\eta-1)\overline{d}}$$

where d_f and d_l are the *Euclidean* distances between the extreme solutions and the boundary solutions of *Sol*; $\overline{d}$ is, instead, the average of all distances d_i ($i=1,..\eta-1$). The smaller its value, the better the diversity of the efficient set. Denoting by *E1* and *E2* the extreme points of each efficient set (characterized by μ, Ψ and c), the obtained results are detailed in Table 1. The computational results underline that the considered approach has detected a significant value of η (equal to 9, on average) with a very low *Spread* (equal to 0.82, on average). Some approximated *Efficient Pareto Fronts* are plotted in the figures of Appendix A.

Table 1. Computational results

Problem	|N|		E1			E2		η	|Ω|	Spread
		μ	Ψ	c	μ	Ψ	c			
LR101	106	25	187	2622.34	23	478	2635.42	13	11,336	1.05
LR102	110	25	208	2915.96	24	332	2732.82	12	10,888	0.82
LR103	104	25	218	2650.12	23	252	2530.01	7	10,291	0.73
LR104	104	25	223	2494.16	21	276	2577.87	10	5,335	0.83
LR105	106	25	173	2521.65	24	365	2387.37	11	14,166	0.85
LR106	104	25	162	2430.46	24	177	2265.26	5	11,778	0.76
LRC105	108	25	370	3071.54	24	370	3443.42	4	8,330	0.70

6. Conclusions and future works

This paper presented an approximate *∈-constraint* approach for *MOMV-PDPTW*. The computational experiments conducted on some instances taken from the literature clearly highlighted that it detects good quality approximated *Efficient Pareto Fronts* (in terms of η and the *Spread*). However, since the quality of the feasible solutions of Ω strongly affects the algorithm performances, it will be significant to diversify Ω designing heuristic approaches (alternatively to *TSSA*). In addition, the experimental campaign will be extended to other instances described in literature. These topics are the subject of ongoing research.

References

Araque, J. R., Kudva, G., Morin, T. L. & Pekny, J. F. (1994). A branch and cut algorithm for the vehicle routing problem. *Annals of Operations Research, 50,37–59*.

Baldacci, R., Hadjiconstantinou, E. & Mingozzi, A. (2004). An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. *Operations Research, 52, 723—738*.

Berube, J.F.,Gendreau, M., & Potvin, J.Y. (2009). An exact ∈-constraint method for bi-objective combinatorial optimization problems: application to the travelling salesman problem with profits. *European Journal of Operational Research, 194,39–50*.

Bramel, J. & Simchi-Levi, D. (2000). Set-covering-based algorithms for the capacitated VRP. In P. Toth, & D. Vigo (Eds), *The Vehicle Routing Problem*. SIAM, Philadelphia, PA.

Chankong, V., & Haimes, Y.Y. (1983). Multiobjective decision making.*Theory and methodology*. North Holland Series in System Science and Engineering.

Cordeau, J. F. (2006). A Branch-and-Cut Algorithm for the Dial-a-Ride Problem. *Operations Research, 54 (3), 573-586*.

Deb, K. (2001). Multi-Objective Optimization Using Evolutionary Algorithms. *First Edition, Chichester, Uk: Wiley*.

Grandinetti, L., Guerriero, F., Laganà, D. & Pisacane, O. (2012). An optimization-based heuristic for the Multi-objective Undirected Capacitated Arc Routing Problem. *Computers & Operations Research*, *39, 2300–2309*.

Harbaoui Dridi, I., Kammarti, R., Ksouri, M. & Borne P. (2011). Multi-Objective Optimization for the m-PDPTW: Aggregation Method With Use of Genetic Algorithm and Lower Bounds. *International Journal of Computers Communications & Control*, *6(2), 246-257*.

Li, H.,& Lim, A. (2003). A Metaheuristic for the Pickup and Delivery Problem with Time Windows. *International Journal on Artificial Intelligence Tools, 12(2), 173—186*.

Lu, Q. & Dessouky, M. (2004). An Exact Algorithm for the Multiple Vehicle Pickup and Delivery Problem. *Transportation Science, 38(4), 503-514*.

Lysgaard, J. (2006). Reachability cuts for the vehicle routing problem with time windows. *European Journal of Operational Research*, *175 (1), 210-223*.

Ropke, S. (2005). Heuristic and exact algorithms for vehicle routing problems. Ph.D. Thesis, Computer science department at the University of Copenhagen (DIKU).

Ropke, S., Cordeau, J. F. & Laporte, G. (2007). Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks, 49(4), 258-272*.
Ropke, S. & Cordeau, J. F. (2009). Branch and Cut and Price for the Pickup and Delivery Problem with Time Windows. *Transportation Science*, *43(3), 267-286*.
Toth, P., & Vigo, D. (Eds.) (2001). The Vehicle Routing Problem. *Society for Industrial and Applied Mathematics*, Philadelphia, PA, USA.

Appendix A.

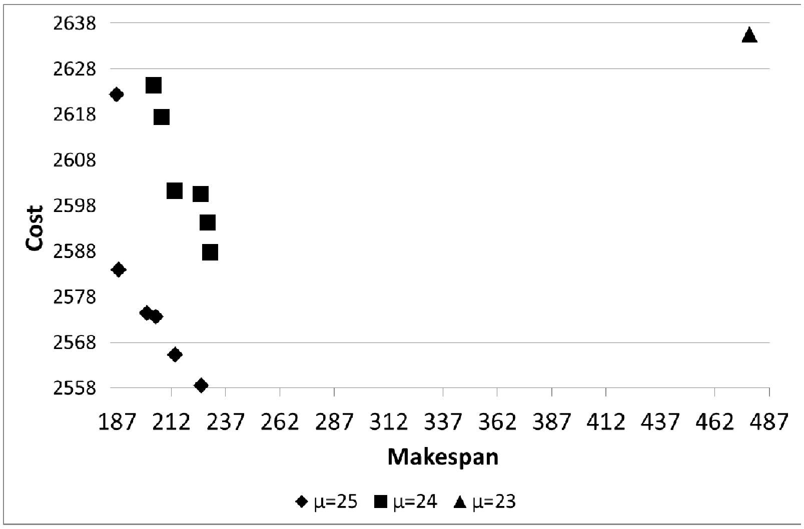

Fig. 1. Efficient Pareto Front of LR101

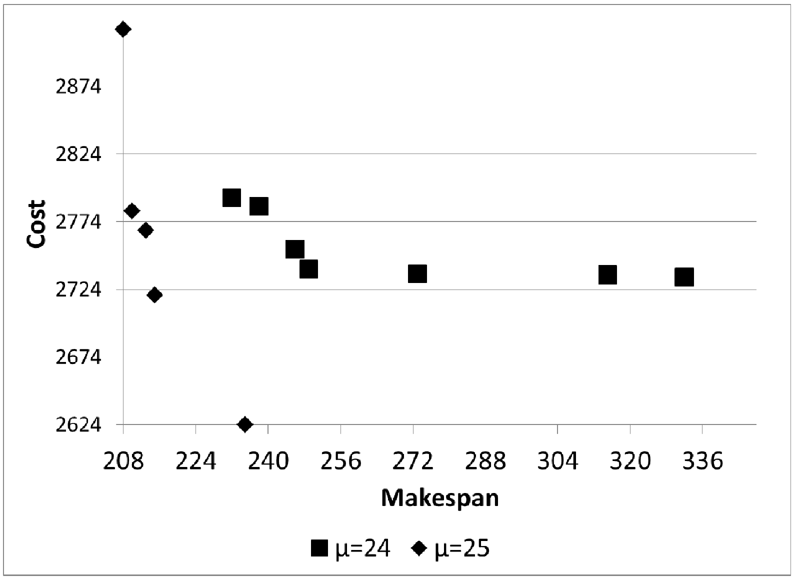

Fig. 2. Efficient Pareto Front of LR102

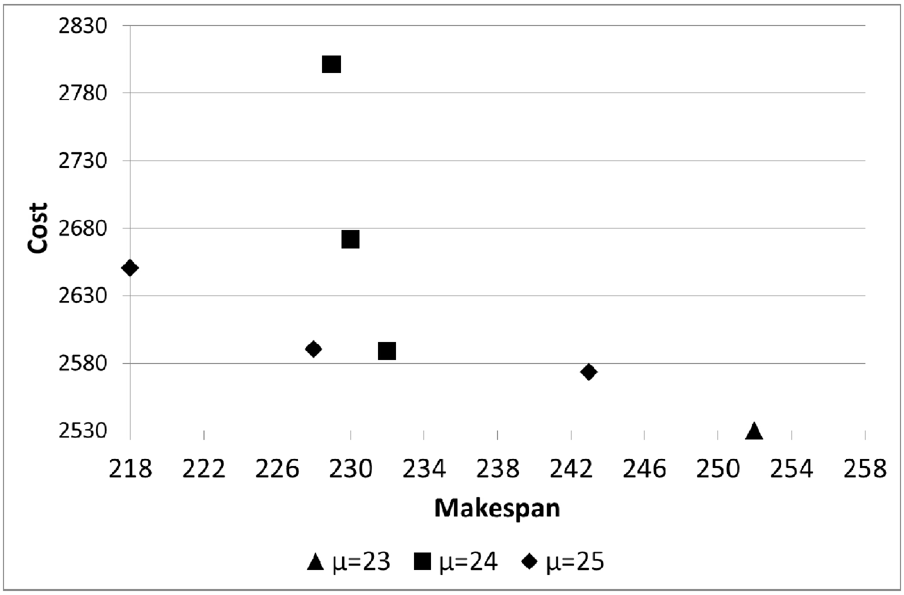

Fig. 3. Efficient Pareto Front of LR103

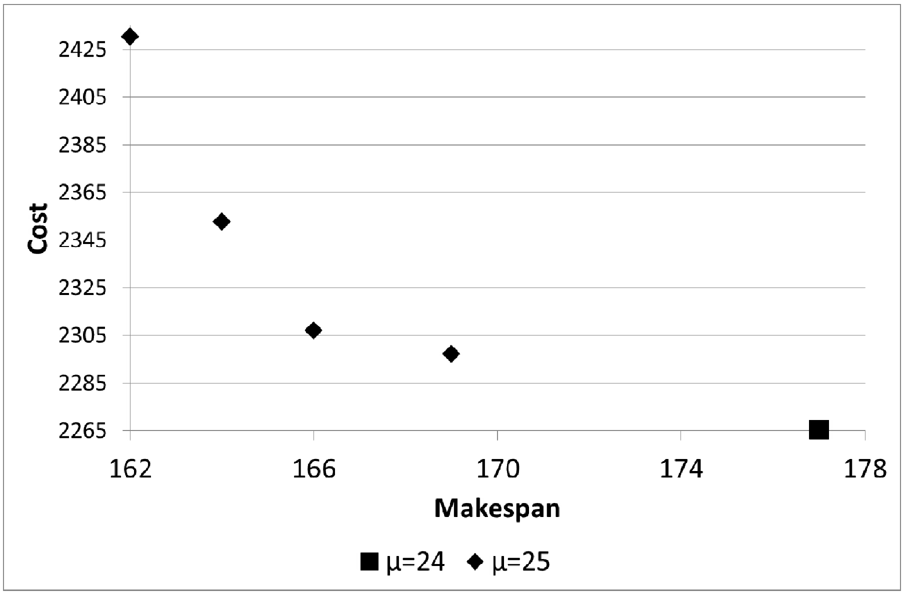

Fig. 4. Efficient Pareto Front of LR106