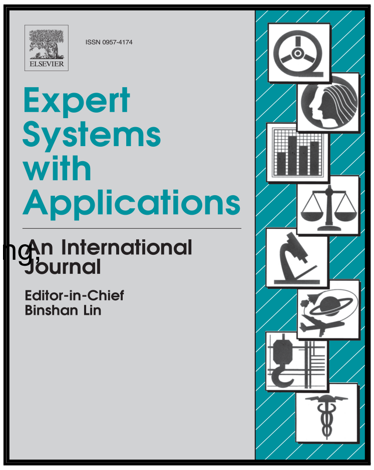
Journal Pre-proof



The Bi-objective Periodic Closed Loop Network Design Problem

Elham Jelodari Mamaghani Writing - Original Draft Writing - Review & Editing,
Soheil Davari Writing - Original Draft Writing - Review & Editing

PII: S0957-4174(19)30785-7
DOI: https://doi.org/10.1016/j.eswa.2019.113068
Reference: ESWA 113068

To appear in: *Expert Systems With Applications*

Received date: 29 August 2019
Revised date: 30 October 2019
Accepted date: 30 October 2019

Please cite this article as: Elham Jelodari Mamaghani Writing - Original Draft Writing - Review & Editing, Soheil Davari Writing - Original Draft Writing - Review & Editing, The Bi-objective Periodic Closed Loop Network Design Problem, *Expert Systems With Applications* (2019), doi: https://doi.org/10.1016/j.eswa.2019.113068

Highlights

- The problem of designing a closed loop supply chain is addressed.
- Simultaneous pickup and delivery as well as time windows are considered.
- A bi-objective integer linear mathematical model is proposed.
- The performance of NSGA-II and NRGA are compared to solve the problem.

The Bi-objective Periodic Closed Loop Network Design Problem

Elham Jelodari Mamaghani[a] and Soheil Davari[b]

[a]Industerial System Optimization Laboratory, Charles Delaunary Institute and UMR CNRS 6281, Uuniversité de Technologie de Troyes, Troyes 1004, France
[b]Hertfordshire Business School, University of Hertfordshire, Hatfield, AL10 9AB, United Kingdom
*elham.jelodari_mamaghani@utt.fr; s.davari@herts.ac.uk

October 31, 2019

Abstract. Reverse supply chains are becoming a crucial part of retail supply chains given the recent reforms in the consumers' rights and the regulations by governments. This has motivated companies around the world to adopt zero-landfill goals and move towards circular economy to retain the product's value during its whole life cycle. However, designing an efficient closed loop supply chain is a challenging undertaking as it presents a set of unique challenges, mainly owing to the need to handle pickups and deliveries at the same time and the necessity to meet the customer requirements within a certain time limit. In this paper, we model this problem as a bi-objective periodic location routing problem with simultaneous pickup and delivery as well as time windows and examine the performance of two procedures, namely NSGA-II and NRGA, to solve it. The goal is to find the best locations for a set of depots, allocation of customers to these depots, allocation of customers to service days and the optimal routes to be taken by a set of homogeneous vehicles to minimise the total cost and to minimise the overall violation from the customers' defined time limits. Our results show that while there is not a significant difference between the two algorithms in terms of diversity and number of solutions generated, NSGA-II outperforms NRGA when it comes to spacing and runtime.

Keywords. Network design, Closed loop supply chain, Periodic location-routing problem, Simultaneous pickup and delivery, Time window, Bi-objective.

1 Introduction

The retail e-commerce sales worldwide is estimated to almost quadruple from 2014 to 2021 (Statista, 2019) and the so-called Amazon effect has disrupted the way customers shop, bringing about massive challenges to the retail managers around the globe. As a result, retailers need to re-engineer their last mile delivery to keep up with the rapid change in the market and to have a sustainable business. A major challenge for retailers is return management which is estimated to cost £60 billion and $260 billion to British and American retailers a year (Financial Times, 2016; CNBC, 2016) due to product damage, obsolescence and devaluation. This phenomenon is often called "return tsunami" which refers to an ever-increasing number of customers who are willing to try a product before deciding on buying them or not. Meanwhile, poor return options can lead to "basket abandonment" which can even lead to brand abandonment in the fiercely competitive markets of the 21st century. Additionally, there is a growing amount of pressure from consumers and non-profit organisations urging the need for incorporating circular economy throughout a product's life-cycle. This has led to a proliferation of studies on circular economy and motivated investments by governments such as €320 billion of circular economy investment opportunities for Europe (SYSTEMIQ, 2017) and the Circular Economy Investment Fund by Zero Waste Scotland (Zero Waste Scotland, 2019) to find ways to incorporate circular economy concepts into supply chain processes. An efficient reverse logistics process has benefits for retailers such as enhanced supply chain transparency, improved visibility, increased profit and higher customer satisfaction levels. However, managing an efficient reverse logistics network imposes some unique challenges ranging from the difficulty of having accurate forecasts to finding the optimal network design.

Companies can opt for a full combination or a full separation of their forward and reverse logistic activities or take a position in between the two extremes (Hansen et al., 2018). Regardless of the stance a company takes, network design is the most indispensable decision to be made to enable a company meet its long-term strategic goals and be resilient against uncertainties in the market. Supply Chain Network Design (SCND) is the problem of creating a network that incorporates all the facilities and transportation vehicles, aiming at reducing the overall cost of the supply chain and increasing the availability of products (Kheirabadi et al., 2019). SCND provides fundamental and underlying support for other supply chain operations and activities (Zhang et al., 2016). One component of SCND which has been studied extensively in the literature is the vehicle routing problem (VRP) which seeks to find the optimal routes for a set of homogeneous/heterogeneous vehicles to serve customers with the minimal cost (or another objective). A practical version of VRP is the VRP with Simultaneous Pickup and Delivery (VRPSPD) with variants such as VRPSPD with time windows (Wang et al., 2015), multi-depot VRPSPD (Li et al., 2015), heterogeneous VRPSPD (Avci and Topaloglu, 2016) and two-echelon VRPSPD (Belgin et al., 2018). However, separating the decision of finding the optimal locations for depots and the set of routes for the vehicles is known to lead to sub-optimal solutions (Salhi and Rand, 1989; Prodhon, 2011). Location Routing Problem (LRP) handles these two problems simultaneously and streamlines the logistics management process even further. It has wide variety of applications in emergency logistics (Zhang et al., 2018) and supply chain management (Bagheri Hosseini et al., 2019) and has variants such as LRP with time windows (Ponboon et al., 2016), time-dependent LRP (Schmidt et al., 2019) and multi-echelon LRP (Vidović et al., 2016; Dai et al., 2019).

In this paper and in order to design the network of a retailer, we address a relatively new variant of LRP called Periodic Location Routing Problem (PLRP) which was first introduced by Prodhon (2008) and integrates simultaneous pickup and delivery with the classical LRP. We contribute to the literature by presenting an integer linear mathematical model for this problem and applying two efficient bi-objective solution algorithms, namely NSGA-II and NRGA, to compare their performance on a set of standard test problems. The problem seeks to find the optimal location for a set of depots, allocation of service days to the customers and to find routes for a set of homogeneous vehicles to serve customers. The goals are minimising the overall cost of network design and operation as the first and minimising violations from the times specified by the customers as the second. Two prime applications of our problem is in the beverage industry where firms not only distribute products over a number of days (periods), but also collect packaged materials for the sake of reuse or disposal as well as electronic and electrical equipment (personal computers, toasters, dishwashers and vacuum cleaners to name a few) for which the collection and disposal is part of the retailers/distributors responsibility in the UK (to either provide a free take back service or to set up an alternative free take back service (Gov.UK, n.d.)).

The paper commences by first providing an overview of the published literature on location-routing problem and reverse logistics in Section 2. This will be followed by presenting a mathematical model for our problem in Section 3 before the solution procedures are discussed in Section 4. We will present a set of computational experiments in Section 5 and conclude the paper in Section 6 providing some avenues for future research.

2 Related work

As far as we are concerned, Webb (1968) and Christofides and Eilon (1969) were the first scholars incorporating routing costs into location problems. Since a comprehensive review of the location routing problem is beyond the scope of our paper, interested readers can refer to Nagy and Salhi (2007); Prodhon and Prins (2014) and Drexl and Schneider (2015) and references therein for a better understanding of LRP and its variants. In this section, we review the most relevant publications to ours and position our study in the literature, explaining how it contributes to the field.

Periodic LRP (PLRP) is one of the widely-applicable variants of LRP addressing a problem where decisions are made over multiple periods, say days. Prodhon and Prins (2008) was the first to pay attention to this problem and used a memetic algorithm with population management to solve it. Later, Pirkwieser and Raidl (2010) applied a large variable neighbourhood search to solve PLRP and Prodhon (2011) developed a new mathematical model for PLRP utilising a hybrid evolutionary algorithm as the solution method based on an extended local search and the Clarke & Wright algorithm. Despite the significance of time windows and pickup/delivery inclusion in a periodic LRP context, none of the aforementioned studies included these two.

A mixed integer mathematical model for multi-period closed-loop supply chain was proposed by Demirel et al. (2014) assuming multiple periods and multiple parts. They investigated two policies, namely secondary market pricing and incremental incentive policy and proposed models for the crisp and fuzzy versions of the problem. Nasherahkami et al. (2015) proposed a model for periodic location routing problem where the predefined demand of customers in each period can decrease due to a violation of time windows in the previous periods. In their model of periodic location problem, the aggregate lost demand costs over multiple periods is minimised. Hemmelmayr et al. (2017) considered the collaborative recycling concept in the periodic location routing problem as a variant of PLRP in which one decides the set of depots to open, the capacity of depots to open and the visit frequency of the nodes in an effort to design networks for collaborative pickup activities. The flexible periodic location routing problem was another extension of the classical PLRP which was studied by Archetti et al. (2017). They assumed that each customer has a total demand to serve within a given time horizon and that there is a limit on the maximum quantity that can be delivered in each visit. Their problem has some similarities with the Inventory Routing Problem (IRP) where inventory levels are considered at each time period incurring additional cost in the objective function.

The LRP with time windows (LRPTW) constraint has been addressed in some publications. Nikbakhsh and Zegordi (2010) suggested a non-linear model with two layers and proposed an or-based heuristic to solve it. Gündüz (2011) developed a single-stage LRP with time windows, for which a Tabu search heuristic was proposed. Zarandi et al. (2011) introduced a model for a capacitated location routing problem (CLRP) with time windows and uncertainty on demands and travel times. They developed a simulated annealing procedure based on initial solutions generated using fuzzy c-means clustering method.

In some applications of LRP, there is a need for simultaneous pickup and delivery (LRPSPD) of customers' orders at the same time. It has numerous applications such as dial-a-ride problems, distribution and collection of products to and from central warehouses and delivery of express courier. To the best of our knowledge, Mosheiov (1994) was the first article published on a problem related to LRPSPD where pickup and delivery were considered for a travelling salesman problem with stochastic demand. However, the general form of the LRPSPD, called many-to-many LRP (MMLRP), originates from Nagy and Salhi (1998). A flow-based LRPSPD was introduced in Karaoglan et al. (2011) disregarding the number of vehicles and employing a branch and cut algorithm to solve it. Later, Karaoglan et al. (2012) developed a node-based model alongside the arc criterion for LRPSPD and proposed a heuristic approach inspired by simulated annealing to solve large-size LRPSPDs. In a last-mile delivery setting, a novel LRP with simultaneous home delivery and customer's pickup was investigated in Zhou et al. (2016). They presented a hybrid evolutionary algorithm by combining genetic algorithm and local search to solve the problem. Demircan-Yildiz et al. (2016) addressed the two-echelon LRPSPD (2E-LRPSPD) which deals with optimally locating primary and secondary facilities by collecting from customers and delivering goods from distribution centres. They presented flow-based and node-based mixed integer mathematical models and demonstrated the efficiency of the flow-based model through a set of numerical experiments. Yu and Lin (2016) addressed the location-routing problem with simultaneous pickup and delivery using simulated annealing. A low-carbon location routing problem with heterogeneous fleet, simultaneous pickup and delivery and time windows was studied in Wang and Li (2017) with a two-phase heuristic based on variable neighbourhood search and genetic algorithm presented. A variant of LRP was studied in Karimi (2018) where a capacitated hub covering location-routing problem for simultaneous pickup and delivery was modelled. A tabu-search based heuristic and valid inequalities were suggested as solution algorithms to determine the hub location and vehicle routes simultaneously. Recently, Nadizadeh and Kafash (2019) addressed a capacitated LRPSPD with fuzzy demand and put forward a greedy clustering as a solution method. A multi-objective mathematical model in the context of industrial hazardous waste management was investigated to address the integrated decisions of three levels with locating, vehicle routing, and inventory control by Rabbani et al. (2019) in presence of stochastic parameters. A sim-heuristic approach as an integration of Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) and Monte Carlo simulation was developed to solve the stochastic problem and the efficiency of the proposed model was verified. A bi-level multi-sized terminal location-routing problem (BL-MSTLRP) with simultaneous home delivery and customer's pickup services was proposed in Zhou et al. (2019) and a self-adaptive hybrid genetic algorithm with simulated annealing was used to solve the problem. A column generation algorithm was addressed to solve LRP with pickup and delivery problem in Capelle et al. (2019). Although their proposed algorithm performed better than Karaoglan et al. (2011) especially for large-size instances, time windows are not included in their study and customer satisfaction is neglected.

From what said, we conclude that in general, there is a paucity of research on bi-objective location-routing problems and there is no study combining PLRP with time windows and simultaneous pickup and delivery in a bi-objective setting. Table 1 juxtaposes our model with the published literature on PLRP for the sake of comparison. One can see from this table that none of these publications have addressed a bi-objective PLRP with simultaneous pickup and delivery in presence of time windows. Hence, our research contributes to the literature by putting forward the first periodic closed loop (forward and backward) model as a bi-objective mathematical optimisation problem with simultaneous pickup and delivery and time windows. We call this problem Periodic Closed Loop Network Design Problem (PCL-NDP) and compare the performance of two well-known algorithms to solve this problem. Since there is no existing benchmark in the literature for our problem, we adopted a set of test instances from the literature and analysed the results of the algorithms based on five metrics.

Table 1: Comparing our research with the published literature on PLRP

Paper	Periodic LRP	Bi-objective	Simultaneous P/D	Time windows	Solution algorithm
Prodhon and Prins (2008)	✓				Memetic
Prodhon (2009)	✓				Hybrid evolutionary
Pirkwieser and Raidl (2010)	✓				VNS
Prodhon (2011)	✓				Hybrid evolutionary
Hemmelmayr (2015)	✓				LNS
Nasherahkami et al. (2015)	✓			✓	LNS
Koç (2016)	✓			✓	LNS
Hemmelmayr et al. (2017)	✓				Heuristic
Archetti et al. (2017)	✓				Valid inequalities
Amiri et al. (2019)	✓			✓	Exact
Our model	✓	✓	✓	✓	Genetic Algorithm

Notes. VNS: Variable Neighbourhood Search; LNS: Large Neighbourhood Search

2.1 Contributions of the paper

Our paper contributes to the academic literature by filling major gaps and putting forward a mathematical model for a closed loop network design problem. It is the first attempt to address a periodic location routing problem with two conflicting objectives, simultaneous pickups and deliveries and time windows to serve customers. We explicitly incorporated these three assumptions into a mathematical model and investigated the performance of two bi-objective solution algorithms, namely NSGA-II and NRGA to solve a set of test problems. The outputs of our model is useful in industries ranging from healthcare to retail.

3 Mathematical formulation of PCL-NDP

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{C})$ be a complete, weighted and undirected graph in which $\mathcal{V}$ and $\mathcal{E}$ represent nodes and edges respectively and $\mathcal{C}$ is the set of travelling costs associated with the set of edges $\mathcal{E}$. The set $\mathcal{V}$ is composed of two subsets $\mathcal{I}$ for customers and $\mathcal{J}$ for depots (not necessarily disjoint). Each customer has a combination of pickup (denoted as p) and delivery (denoted as d) orders to meet. We consider the case of homogeneous vehicles which means that each vehicle has a pre-determined capacity with a fixed operational cost which is identical for the vehicles. Moreover, there are fixed costs for opening depots and a capacity for pickup and delivery orders. Each customer can choose a combination of days to be served (for instance, one might choose Monday and Thursday while another customer might be more flexible choosing any day during the week but Tuesdays). The demand of each customer must be served on each day of exactly one combination of days and by one vehicle. All the parameters in the model are assumed to be deterministic and the deliveries are positive integers. Similar to the majority of the LRP applications, each route must begin from and end at the same depot on the same day while the capacity of the vehicle must be respected throughout the day. The goal is to find the set of depots to open among a set of potential locations, combinations of service days to allocate to each customer and the routes to take from each depot in each period, so two conflicting objectives are optimised. Firstly, the total cost of the system should be minimised and then, the total violation from the pre-defined times by the customers should be minimised. These two objectives are clearly conflicting in urban areas, since on-time delivery calls for investing in opening new depots and deploying vehicles. The overall cost of the network is composed of the costs for opening depots, the aggregate cost of deploying the set of vehicles and the operational costs of distributing and collecting products to/from customers. We have modelled the problem as a bi-objective integer linear programming model. In Tables 2-4, the notations are introduced and the mathematical formulation of the problem is given in the following.

Table 2: Sets

Symbol	Definition
$\mathcal{I}$	Set of customers
$\mathcal{J}$	Set of depots
$\mathcal{V} = \mathcal{I} \cup \mathcal{J}$	Set of all nodes
$\mathcal{T}$	Periods
$\mathcal{R}_i$	Set of combinations of service days to node $i \in \mathcal{V}$

Table 3: Input parameters used in the model

Symbol	Definition
a_{rt}	If day $t \in \mathcal{T}$ is in combination $r \in \mathcal{R}_i$
c_{ij}	Travel cost between nodes $i \in \mathcal{V}$ and $j \in \mathcal{V}$
d_{irt}	Delivery quantity for customer $i \in \mathcal{I}$ in day $t \in \mathcal{T}$ and $r \in \mathcal{R}_i$
p_{irt}	Pickup quantity for customer $i \in \mathcal{I}$ in day $t \in \mathcal{T}$ and $r \in \mathcal{R}_i$
s_i	Service time of customer $i \in \mathcal{I}$
σ_i^-	Lower bound of the time window for customer $i \in \mathcal{I}$
σ_i^+	Upper bound of the time window for customer $i \in \mathcal{I}$
τ_{ij}	Travel time between nodes $i \in \mathcal{V}$ and $j \in \mathcal{V}$
ϕ_j	Fixed cost of constructing depot $j \in \mathcal{J}$
Φ	Fixed cost of using a vehicle
ψ_j	Capacity of depot $j \in \mathcal{J}$
Ψ	Capacity of a vehicle

Table 4: Decision variables used in the model

Symbol	Definition
b_{ir}	Binary variable which equals one if combination $r \in \mathcal{R}_i$ is assigned to customer $i \in \mathcal{I}$
$x_{ijt} = 1$	If a vehicle travels directly from node $i \in \mathcal{V}$ to node $j \in \mathcal{V}$ in the period $t \in \mathcal{T}$ and zero otherwise
$y_j = 1$	If depot $j \in \mathcal{J}$ is open and zero otherwise
$z_{ij} = 1$	If customer $i \in \mathcal{I}$ is assigned to depot $j \in \mathcal{J}$ and zero otherwise
u_{it}	Arriving time for a vehicle to customer $i \in \mathcal{I}$ in period $t \in \mathcal{T}$
w_{ikrst}	Auxiliary binary variable
α_{it}	Delivery load of a vehicle before having served customer $i \in \mathcal{I}$ in period $t \in \mathcal{T}$
β_{it}	Pickup load of a vehicle after having served customer $i \in \mathcal{I}$ in period $t \in \mathcal{T}$
θ_{it}^-	Overall violation of the lower time bound for $i \in \mathcal{I}$ in the period $t \in \mathcal{T}$
θ_{it}^+	Overall violation of the upper time bound for $i \in \mathcal{I}$ in period $t \in \mathcal{T}$

Using the above-defined parameters and variables, we present the formulation of a bi-objective integer linear mathematical model.

$$\min \left[\sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} \sum_{t \in \mathcal{T}} c_{ij} x_{ijt} + \sum_{j \in \mathcal{J}} \phi_j y_j + \sum_{i \in \mathcal{I}} \sum_{j \in \mathcal{J}} \sum_{t \in \mathcal{T}} \Phi x_{jit} \right] \tag{1}$$

$$\min \left[\sum_{i \in \mathcal{I}} \sum_{t \in \mathcal{T}} (\theta_{it}^- + \theta_{it}^+) \right] \tag{2}$$

$$\sum_{i\in\mathcal{V}} x_{ijt} \leq 1 \qquad \forall j \in \mathcal{V}; \forall t \in \mathcal{T}; i \neq j \tag{3}$$

$$\sum_{\substack{i\in\mathcal{V}\\ i\neq j}} x_{ijt} - \sum_{\substack{i\in\mathcal{V}\\ i\neq j}} x_{jit} = 0 \qquad \forall j \in \mathcal{V}; \forall t \in \mathcal{T} \tag{4}$$

$$\sum_{j\in\mathcal{J}} z_{ij} = 1 \qquad \forall i \in \mathcal{I} \tag{5}$$

$$x_{ijt} \leq z_{ij} \qquad \forall i \in \mathcal{I}; \forall j \in \mathcal{J}; \forall t \in \mathcal{T} \tag{6}$$

$$x_{jit} \leq z_{ij} \qquad \forall i \in \mathcal{I}; \forall j \in \mathcal{J}; \forall t \in \mathcal{T} \tag{7}$$

$$x_{ikt} + z_{ij} + \sum_{\substack{h\in\mathcal{J}\\ j\neq h}} z_{kh} \leq 2 \qquad \forall i,k \in \mathcal{I} : i \neq k; \forall j \in \mathcal{J}; \forall t \in \mathcal{T} \tag{8}$$

$$\sum_{i\in\mathcal{I}} d_{irt} z_{ij} \leq \psi_j y_j \qquad \forall j \in \mathcal{J}; \forall r \in \mathcal{R}_i; \forall t \in \mathcal{T} \tag{9}$$

$$\sum_{i\in\mathcal{I}} p_{irt} z_{ij} \leq \psi_j y_j \qquad \forall j \in \mathcal{J}; \forall r \in \mathcal{R}_i; \forall t \in \mathcal{T} \tag{10}$$

$$w_{ikrst} \geq b_{ir} + b_{ks} + x_{ikt} - 2 \qquad \forall (i \neq k) \in \mathcal{V}; \forall r \in \mathcal{R}_i; \forall s \in \mathcal{R}_k; \forall t \in \mathcal{T} \tag{11}$$

$$3w_{ikrst} \leq b_{ir} + b_{ks} + x_{ikt} \qquad \forall (i \neq k) \in \mathcal{V}; \forall r \in \mathcal{R}_i; \forall s \in \mathcal{R}_k; \forall t \in \mathcal{T} \tag{12}$$

$$\alpha_{kt} - \alpha_{it} + \Psi w_{ikrst} + (\Psi - d_{irt} - d_{kst}) w_{kisrt} \leq \Psi - d_{irt} \qquad \forall i,k \in \mathcal{I} : i \neq k, \forall t \in \mathcal{T}, \forall r \in \mathcal{R}_i; \forall s \in \mathcal{R}_k \tag{13}$$

$$\beta_{it} - \beta_{kt} + \Psi w_{ikrst} + (\Psi - p_{irt} - p_{kst}) w_{kisrt} \leq \Psi - p_{kst} \qquad \forall i; k \in \mathcal{I} : i \neq k; \forall t \in \mathcal{T}; \forall r \in \mathcal{R}_i; \forall s \in \mathcal{R}_k \tag{14}$$

$$\alpha_{it} - d_{irt} + p_{irt} \leq \Psi \qquad \forall i \in \mathcal{I}; \forall r \in \mathcal{R}_i; \forall t \in \mathcal{T} \tag{15}$$

$$\alpha_{it} \geq d_{irt} + \sum_{\substack{k\in\mathcal{I}\\ i\neq k}} \sum_{s\in\mathcal{R}_k} d_{krt} w_{ikrst} \qquad \forall i \in \mathcal{I}; \forall r \in \mathcal{R}_i; \forall t \in \mathcal{T} \tag{16}$$

$$\beta_{it} \geq p_{irt} + \sum_{\substack{k\in\mathcal{I}\\ i\neq k}} \sum_{s\in\mathcal{R}_k} p_{krt} w_{kisrt} \qquad \forall i \in \mathcal{I}; \forall r \in \mathcal{R}_i; \forall t \in \mathcal{T} \tag{17}$$

$$\alpha_{it} \leq \Psi - (\Psi - d_{irt}) \sum_{j\in\mathcal{J}} \sum_{s\in\mathcal{R}_j} w_{ijrst} \qquad \forall i \in \mathcal{I}; \forall t \in \mathcal{T}; \forall r \in \mathcal{R}_i \tag{18}$$

$$\beta_{it} \leq \Psi - (\Psi - p_{irt}) \sum_{j\in\mathcal{J}} \sum_{r\in\mathcal{R}_j} w_{jirst} \qquad \forall i \in \mathcal{I}; \forall t \in \mathcal{T}; \forall s \in \mathcal{R}_i \tag{19}$$

$$\sum_{r\in\mathcal{R}_i} b_{ir} = 1 \qquad \forall i \in \mathcal{I} \tag{20}$$

$$\sum_{j\in\mathcal{V}} x_{jit} - \sum_{r\in\mathcal{R}_i} b_{ir} a_{rt} = 0 \qquad \forall i \in \mathcal{I}; \forall t \in \mathcal{T} \tag{21}$$

$$u_{it} + s_i + \tau_{ij} - u_{jt} \leq M(1 - x_{ijt}) \qquad \forall i \in \mathcal{V}; \forall j \in \mathcal{J} : i \neq j; \forall t \in \mathcal{T} \tag{22}$$

$$u_{it} \leq M \sum_{j\in\mathcal{V}} x_{jit} \qquad \forall i \in \mathcal{I}; \forall t \in \mathcal{T} \tag{23}$$

$$\theta^+_{it} \geq u_{it} - \sigma^+_i \sum_{j\in\mathcal{V}} x_{jit} \qquad \forall i \in \mathcal{I}; \forall t \in \mathcal{T} \tag{24}$$

$$\theta^-_{it} \geq \sigma^-_i \sum_{j\in\mathcal{V}} x_{jit} - u_{it} \qquad \forall i \in \mathcal{I}; \forall t \in \mathcal{T} \tag{25}$$

$$\alpha_{it}, \beta_{it}, \theta^+_{it}, \theta^-_{it}, u_{it} \geq 0 \qquad \forall i \in \mathcal{V}; \forall t \in \mathcal{T} \tag{26}$$

$$x_{ijt} \in \{0,1\} \qquad \forall i; j \in \mathcal{V}; \forall t \in \mathcal{T} \tag{27}$$

$$z_{ij} \in \{0,1\} \qquad \forall i \in \mathcal{J}; \forall j \in \mathcal{J} \tag{28}$$

$$b_{jr} \in \{0,1\} \qquad \forall j \in \mathcal{V}; \forall r \in \mathcal{R}_j \tag{29}$$

$$w_{ikrst} \in \{0,1\} \qquad \forall i,k \in \mathcal{V}; \forall r \in \mathcal{R}_i; \forall t \in \mathcal{T}; \forall s \in \mathcal{R}_k \tag{30}$$

$$y_j \in \{0,1\} \qquad \forall j \in \mathcal{J} \tag{31}$$

While the objective function 1 minimises the sum of transportation costs, fixed costs of constructing depots and costs for using vehicles, the second objective maximises the customers satisfactions by minimising the violation from the specified time windows. The descriptions of the model constraints are given in Table 5.

Table 5: The descriptions of the constraints

Symbol	Definition
(3)	A degree constraint ensuring that each customer is visited at most once
(4)	A degree constraint ensuring that the number of arcs entering a node equals those leaving it
(5)	Each customer is assigned to one and only one depot
(6)-(7)	Forbids allocation of a node to a non-functional depot
(8)	If an arc exists between two nodes on a day, both must be served by the same depot
(9)-(10)	The capacity constraints for pickup and delivery loads of each depot
(11)-(12)	Ensuring that $w_{ikrst} = 1$ if and only if $b_{ir} = b_{ks} = x_{ikt} = 1$
(13)-(14)	Flow inequalities for delivery and pickup demands respectively, besides serving as sub-tour elimination constraints They also compute the amount of delivery and pickup demands of each customer
(15)	The total load on each arc must not be larger than the capacity of vehicles
(16)-(17)	Ensuring that the load of a vehicle is not violated before or after visiting a node
(18)-(19)	Computes values of α and β variables on the feasible routes. Furthermore, these constraints besides (13) and (14) are bounding constraints for α and β
(20)	Each customer is allocated to one and only one combination of days
(21)	If a node is served on a specific day, it must be allocated to a combination the day is part of
(22)	The relationship between the arrival times to a customer's location and its immediate successor
(23)	The arrival time to a node is zero if there is no arc entering it
(24)-(25)	Soft time window constraints
(26)-(31)	Definition of variables

4 Solution procedure

Over the last decades, different techniques have been developed to solve multi-objective optimisation problems with the aim of striking a balance between convergence and diversity. These algorithms can be classified into meta-heuristic, decision-aided, interactive, fuzzy and scalar ones (Collette and Siarry, 2013). In the following, two popular meta-heuristic algorithms, NSGA-II (Non-dominated Sorting Genetic Algorithm-II) and NRGA (Non-dominated Ranked Genetic Algorithm) are utilised to solve PCL-NDP. We used the Matlab implementation of both algorithms and to ensure a fair comparison between the two, we ran all the experiments with identical hardware.

NSGA-II was proposed on the basis of NSGA with improvements to decrease its complexity from $O(mN^3)$ (m is the number of objective functions and N is the population size) to $O(mN^2)$ (Li et al., 2016). It has less computational complexity, considers elitism, systematically preserves the diversity of Pareto-optimal solutions and adaptively handles the problem constraints (Deb and Jain, 2012). These features have made NSGA-II one of the most popular multi-objective optimisation algorithms in the literature with applications ranging from scheduling (Wang et al., 2017) to diabetes diagnosis (Alirezaei et al., 2019). Different test problems from previous studies applying NSGA-II were compared in Deb et al. (2002), showing that NSGA-II outperforms algorithms such as Pareto Archived Evolution Strategy (PAES) and Strength Pareto Evolutionary Algorithm (SPEA) in obtaining a more diverse set of solutions.

NRGA is a modification to the NSGA-II algorithm by exchanging its selection strategy. It was first presented by Al Jadaan et al. (2008) and operates with two tiers of rank-based roulette wheel selection strategies. The probability of selecting a front (P_f) and the probability of selecting a solution in a front (P_{fs}) are found using equations (32) and (33) respectively where N_F

and NS_f are the number of fronts and the number of solutions in the front f respectively.

$$P_f = \frac{2 \times rank_f}{NF \times (NF + 1)} \qquad f = 1, ..., NF \tag{32}$$

$$P_{fs} = \frac{2 \times rank_{fs}}{NS_f \times (NS_f + 1)} \qquad f = 1, ..., NF; s = 1, .., NS \tag{33}$$

NRGA works based on a ranking of the individuals in a front and then employs roulette wheel selection to choose individuals for the next iteration. It shares a fundamental feature with NSGA-II where both algorithms penalise infeasible solutions during their iterations. It is also used in several publications to date and its performance has been compared against other multi-objective optimisation heuristics such as NSGA-II (Sadeghi et al., 2014) and MOPSO (Alikar et al., 2017).

4.1 Non-dominated sorting and crowding distance

In sorting the non-dominated solutions, a population is ranked by using the concept of predominance. In general, to sort a population by size on the basis of non-dominated levels, each solution is compared with all the other solutions in the population to determine whether or not the solution is dominated. This leads to generation of a set of solutions that neither dominate nor defeat each other. This process is repeated until all the remaining solutions are in the non-dominated front. To estimate the solution around a particular solution in the population, the average distance from both adjacent solutions is calculated based on the values of the objective functions (crowding distance). The notion of crowding distance is one of the major proxies for an algorithm which estimates the density of solutions surrounding a particular solution. To calculate the crowding distance, we used an approach similar to Deb et al. (2002) by estimating the perimeter of the cuboid formed by using the nearest neighbours of a solution as the vertices. Algorithm 1 presents the computational procedure of the crowding distance where $L[i]_m$ refers to the value of the mth objective function of the ith individual in set L. Figure 1 depicts a sample efficient frontier in a bi-objective problem where the crowding distance for individual X_1 is larger than X_2, hence, individual X_1 has more probability to be chosen as a parent.

Algorithm 1 Cuboid along locally non-dominated frontier (Moradi et al., 2011)

```
procedure
    Crowding-distance-assignment (L)
    l = |L|
    for each i, set L[i]_distance=0
    for each objective m:
        L = sort(L, m)                        sort using each objective value
        L[1]_distance = L[l]_distance = ∞     boundary points are always selected for all other points
        for i = 2 to (l = 1)
            L[i]_distance=L[i]_distance+(L[i+1]_.m+L[i-1]_.m)
```

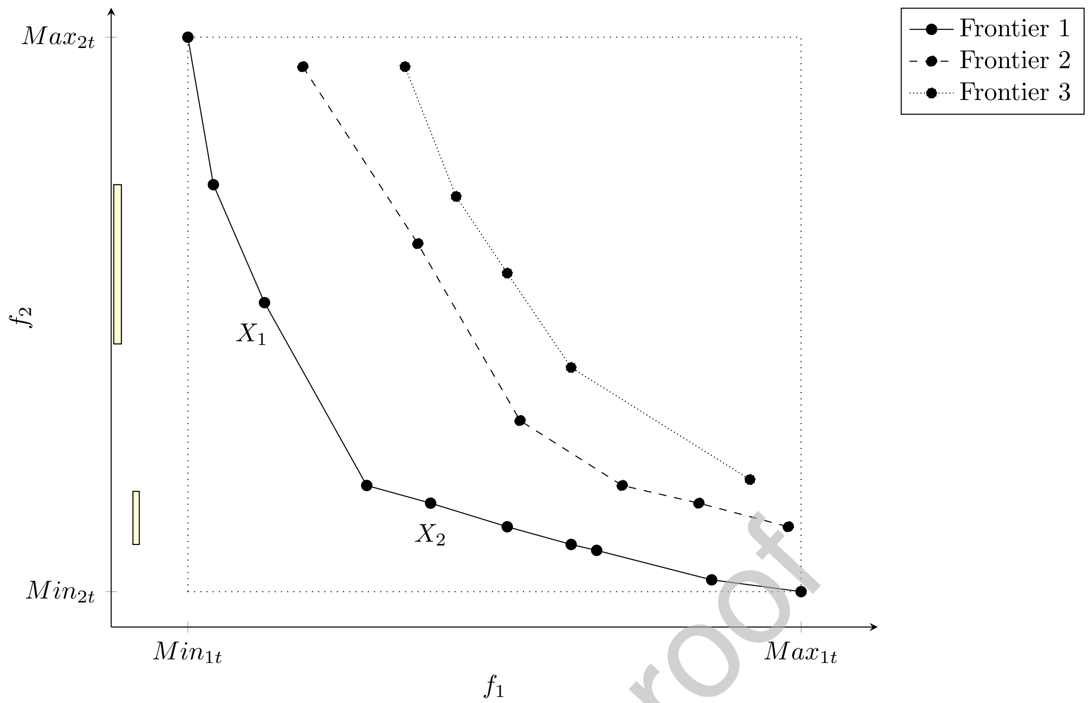


Figure 1: Cuboid along locally non-dominated frontier

4.2 Solution representation

Solution representation is a key factor influencing the performance of any heuristic algorithm. In our study, a chromosome is composed of two vectors S_1, S_3, and a matrix S_2. The size of vector S_1 equals the number of customers, and the value of each element is the index of the depot the customer is allocated to. Matrix S_2 indicates the service priority of each customer where its number of rows equals the number of periods and the number of its columns equals the number of customers. The size of vector S_3 is similar to vector S_1 and its elements denote the possible combination of days to serve a customer. For instance, if a customer can be served in two days and there are two different combinations of days for its service such as (1, 2) and (2, 3) the index of the first combination is one and the second combination has an index of two. Each element of this vector specifies the index of customer's combination to serve in the appropriate period (day). In a nutshell, the allocation of customers to depots is determined using S_1, the priority of customers in each service day is represented with S_2 and the service day combinations are defined with S_3. Figure 2 demonstrates a sample solution with four customers and two periods to clarify the encoding procedure further.

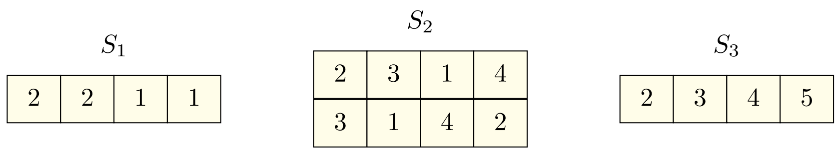


Figure 2: Solution representation

4.3 Genetic operators

Using the chromosome structure defined for the bi-objective PCL-NDP, we propose two crossover and two mutation operators which are explained in the following sections.

4.3.1 Crossover operator on vectors S_1 and S_3

For the two one-dimensional vectors of the solution representation (S_1 and S_3), we applied a uniform crossover. To this end and for each solution, we first generated a binary mask vector of the same size as the chromosome. The offspring is then generated from one of the two parents depending on the corresponding value of the mask vector. To put it in simple terms, a gene of the first parent is transferred to the first offspring if the corresponding gene of the mask vector is zero and from the second otherwise. Figures 3 and 4 are illustrative examples of how these two operators work. For the sake of consistency, we kept the example of Figure 2 as the first parent in both figures

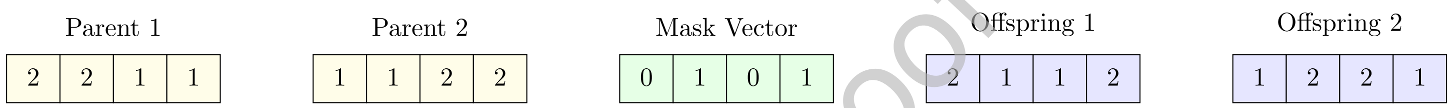


Figure 3: Applying uniform Crossover on S_1 for a sample solution with four nodes

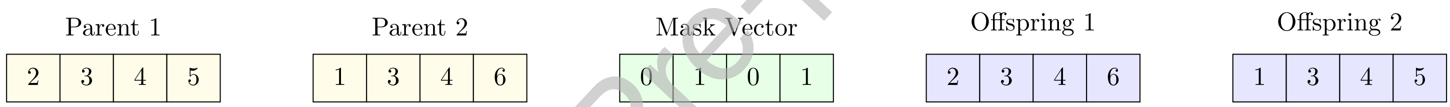


Figure 4: Applying uniform Crossover on S_3 for a sample solution with four nodes

4.3.2 Crossover operator on matrix S_2

To apply a crossover operator on S_2, we used a single-point crossover owing to its permutation structure. In each iteration of the algorithm, each row of S_2, which indicates a period, undergoes a single-point crossover guaranteeing the feasibility of the offspring and preventing a repetition of genes. To operate this, two parent chromosomes are selected from the current population by applying a roulette wheel approach, while the crossover point is randomly chosen among the genes. The crossover point splits chromosomes into two parts (not necessarily identically-sized) which are used to build the offspring. To generate the first offspring, all genes of the first parent to the left of the crossover point are transferred respectively to produce the first part of the first offspring before it is combined with a second part which is generated by comparing the genes of the second parent with the genes of the first offspring already created. After ignoring duplicate genes, all non-repetitive genes are transferred to create the second part after the crossover point. A similar approach is adopted to generate the second offspring. Figure 5 demonstrates this process on two sample parent chromosomes. For the sake of consistency, we kept the example of Figure 2 as the first parent in Figure 5.

Parent 1			
2	3	1	4
3	1	4	2

Parent 2			
1	4	3	2
4	2	3	1

Offspring 1			
2	1	4	3
3	4	2	1

Offspring 2			
1	2	3	4
4	3	1	2

Figure 5: Applying single point Crossover on S_2 for a sample solution with four nodes

4.3.3 Mutation on vectors S_1 and S_3

For mutation of the first and third parts of the solution representation, we implemented a procedure which can be summarised as follows.

Algorithm 2 The mutation steps

1: **procedure** Mutation
2: Randomly determine the number of genes to mutate. This number, denoted by l, is obtained by the following procedure.
3: Randomly generate an integer number h, between 1 and $\dim(S_1/S_3)$, where $\dim(S_1/S_3)$ is the dimension of vector
4: S_1/S_3.
5: Calculate hr by multiplying h with a mutation rate r.
6: Round hr to the least integer number larger than or equal to it.
7: Choose randomly l genes of vector S_1/S_3.
8: Generate randomly l integer numbers between 1 and the number of all depots.
9: Replace the l genes of vector S_1/S_3 obtained in Step 2 by the l integer numbers obtained in Step 3.

4.3.4 Mutation on Matrix S_2

For the mutation of the second part of a solution, two genes of a vector are randomly chosen first and then, one of the three operators (insertion, swap or reversion) is applied to the solution. We did not use all the three operators simultaneously to avoid an over-complication of the solution procedure. Instead, in each run, we used a random selection rule to choose one of these three operators. While the swap operator basically exchanges the position of two randomly chosen bits in a solution, a reversion operator is performed in order to get a more diversified solution by taking a random section of a solution and reversing it. The inversion operator also randomly chooses a bit and replaces its value with a new random value.

5 Computational study

To the best of the authors' knowledge, no standard dataset exists in the literature to investigate the performance of the solution algorithms in solving the problem. Hence, we adopted 45 problems from Karaoglan et al. (2012) to examine the performance of the solution procedures. Moreover, to incorporate pickup and delivery in the test problems, we used a similar approach to Angelelli and Mansini (2002). To do so, the demand of each node (delivery) is set as q_i and the demand of a pickup node is calculated by $(1-b)q_i$ if i is even and $(1+b)q_i$ if i is odd, with the value of b set as 0.8. Table 6 summarises these test problems and their specifications including a code assigned to each instance, the number of customers in each instance, the capacity of vehicles and the number of potential depots. All the experiments were implemented in MatLab 2013 on a laptop with an Intel core i5 CPU and 4.00 GB RAM on Windows 7.0.

Table 6: The specifications of our test problems

Instances	# Customers	Capacity of vehicles	# Potential depots
A1-A11	50	200	9
B1-B11	100	1,000	9
C1-C8	100	200	16
D1-D7	100	1,000	16
E1-E8	125	1,000	16

5.1 Comparison metrics for multi-objective optimisation algorithms

In order to test the performance of the algorithms, five criteria were used including spacing, diversification measure, number of solutions, runtime and Main Ideal Distance (MID).

- **Spacing**

  Spacing which can be defined as (34), is defined as the variance of the distance of each member of an efficient Pareto frontier to its closest neighbour and was first proposed by Scott (1995). It is preferred to be as low as possible with an ideal value of zero showing that all members of the efficient frontier are equally spaced.

$$\text{Spacing} = \sqrt{\frac{\sum_{i=1}^{n}(d_i - \bar{d})^2}{n-1}} \tag{34}$$

  where $d_i = \min_{\substack{k \in n \\ k \neq 1}} \sum_{m=1}^{M} |f_m^i - f_m^k|$, f_m^i is the i^{th} objective function value of the m^{th} solution in the Pareto front, and $\bar{d}$ is the mean value of these distances as $\bar{d} = \sum_{i=1}^{|n|} \frac{d_i}{|n|}$.

- **Diversification measure**

  The diversification metric measures the spread of the solutions found (Govindan et al., 2015) and is computed as follows where n is number of Pareto front solutions and $||\kappa_i^1 - \kappa_i^2||$ is the Euclidean distance between the best front of non-dominated solutions, κ_i^1 and κ_i^2.

$$\text{DM} = \sqrt{\sum_{i=1}^{n} \max(||\kappa_i^1 - \kappa_i^2||)} \tag{35}$$

- **The number of Pareto solutions (NOS)**

  This metric enumerates the number of Pareto solutions in the optimal front. One issue with some multi-objective optimisation algorithms is the generation of far too many non-dominated solutions which renders the outputs impractical for a decision maker and leads to confusion. Hence, finding a limited number of Pareto solutions is preferred in many real-world instances. Various strategies have been used in the literature to this end such as Subtractive clustering by Zio and Bazzo (2012) and fuzzy preference assignment by Abido (2003) and level diagram analysis by Blasco et al. (2008).

- **Main Ideal Distance (MID)**

  This criterion aims at finding the average distance of the Pareto solutions to the ideal solution which equals (0, 0) in our problem and is computed as equation (36) where f_{ji} is the value of the jth objective function in the ith solution of optimal front. One should note that a lower value for the MID index is more desirable.

$$\text{MID} = \frac{1}{NOS} \sum_{i=1}^{NOS} C_i \text{ where } C_i = \sqrt{\sum_{j=1}^{2} f_{ji}^2} \tag{36}$$

5.2 Algorithm parameter tuning

We expected the performance of both algorithms to be influenced by their parameter settings. Hence, we used the Taguchi method as the preferred technique in finding the best combination of parameters for each algorithm owing to its strength in providing results with fewer experiments. We performed 18 independent runs for each algorithm. To compare the two proposed multi-objective optimisation algorithms, we considered computational time and mean ideal instance (MID) as two metrics to evaluate the convergence of the algorithms. Diversity and MID can be aggregated as a single metric called Multi-objective Coefficient of Variation, as set out in equation (37) and as a measure for the Taguchi method.

$$\text{MOCV} = \frac{\text{MID}}{\text{Diversity}} \tag{37}$$

To apply the Taguchi method, three levels were considered for each factor as given in Table 7 and based on the existing literature as well as a set of preliminary experiments. In this table, n_{pop} denotes the population size, n_{It} is the number of iterations in each run, P_c and P_m are the probabilities of crossover and mutation respectively and U_m represents mutation rate. For each algorithm, the effect plots for Signal to Noise (S/N) ratio are presented in Figures 6 and 7 where the horizontal axis indicates the index of the setting for the parameter and the vertical axis represents the S/N ratio. Our experiments have led to selection of optimal conditions to use for each algorithm as given in Table 8. These results were then adopted to run the numerical experiments.

Table 7: The domain of candidate parameters of NSGA-II and NRGA for calibration

Methodology	Parameter	Range	Low (1)	Medium (2)	High (3)
NSGA-II	n_{pop}	50-200	50	100	200
	n_{It}	100-300	100	200	300
	P_c	0.6-0.8	0.6	0.7	0.8
	P_m	0.2-0.6	0.2	0.4	0.6
	U_m	0.1-0.4	0.1	0.2	0.4
NRGA	n_{pop}	50-200	50	100	200
	n_{It}	100-300	100	200	300
	P_c	0.6-0.85	0.6	0.75	0.85
	P_m	0.1-0.5	0.1	0.3	0.5
	U_m	0.1-0.4	0.1	0.2	0.4

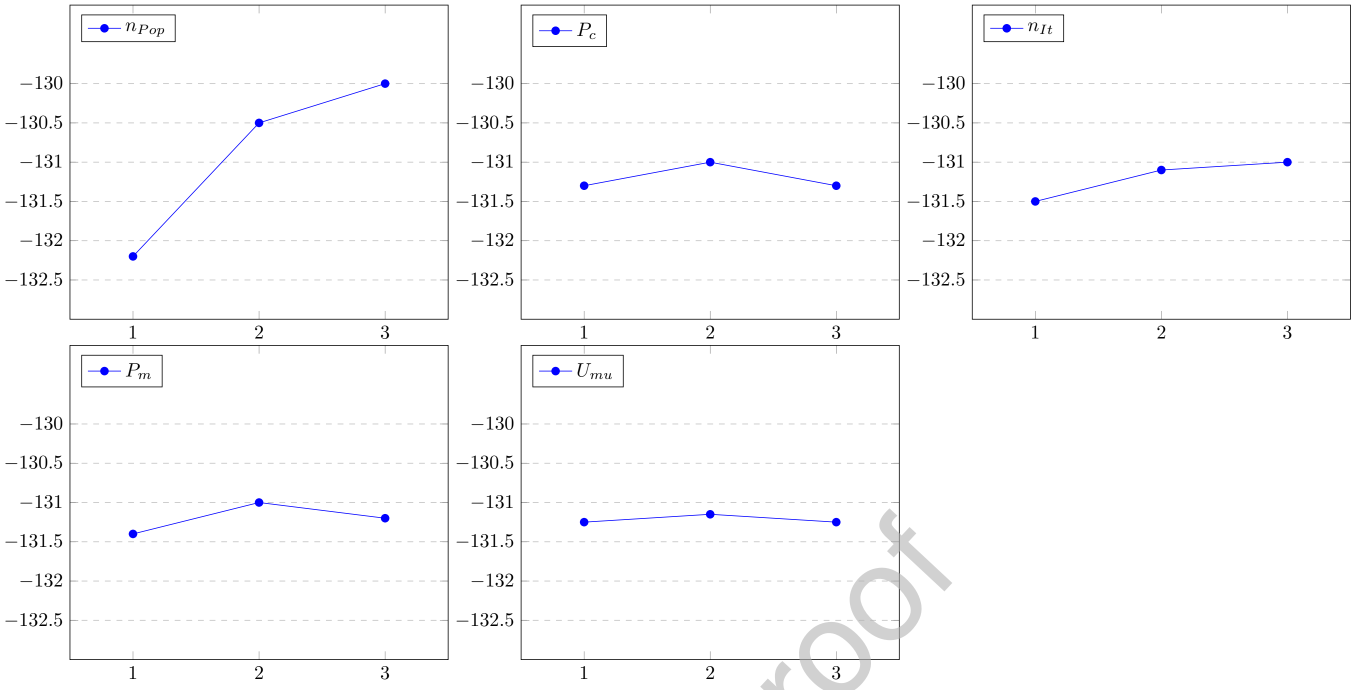


Figure 6: Outputs of the Taguchi ratio for NSGA-II

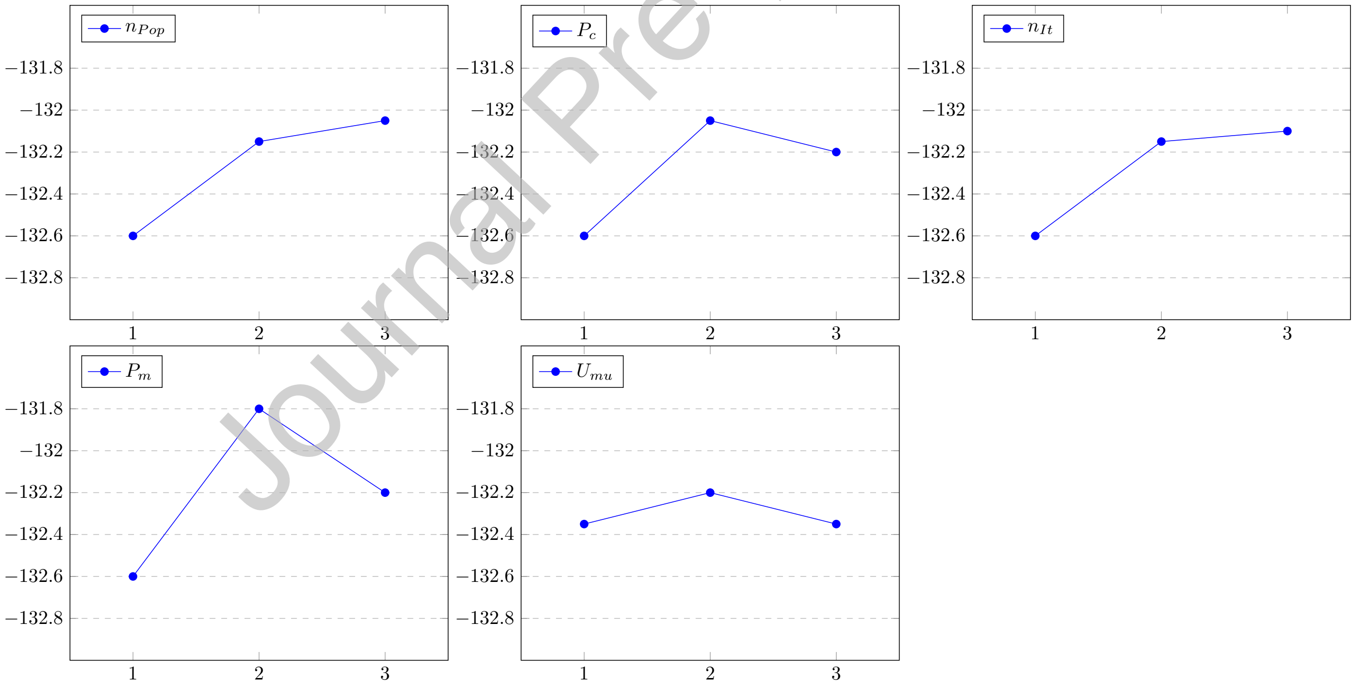


Figure 7: Outputs of the Taguchi ratio for NRGA

Table 8: Calibrated parameters of the algorithms

	NSGA-II	NRGA
u_m	0.2	0.2
P_m	0.4	0.3
n_{It}	300	300
P_c	0.7	0.75
n_{pop}	200	200

5.3 Results and discussion

Using the fine-tuned parameters identified in an earlier stage, we turned our attention to compare the outputs of the two algorithms. To do so, we have operated each algorithm three times and removed the effect of problem size by utilising the RPD index as given in Equation (38) where $Criterion_{Best}$ and $Criterion_{Alg}$ indicate the best value of a criterion obtained and the best value of the same criterion achieved by the algorithm respectively.

$$\text{RPD} = \frac{|Criterion_{Best} \text{ - } Criterion_{Alg}|}{Criterion_{Best}} \times 100 \tag{38}$$

One should note that in calculation of RPD in terms of NOS and diversity, the algorithm with the large value of NOS and Diversity has better performance whereas in comparison of the algorithms in terms of MID, spacing, and runtime the algorithm with small values of have better performance. However, regardless of the metric to use, a lower RPD is preferred as it shows a lower distance to the ideal point.

Figure 8 shows the non-dominated solutions obtained from each algorithm for one sample problem with 100 customers (thicker line denotes NSGA-II) showing the superiority of NSGA-II in terms of the number of solutions. However, to quantitatively measure the performance of each algorithm and to have a better understanding of how significant possible differences are, we carried out a set of additional experiments.

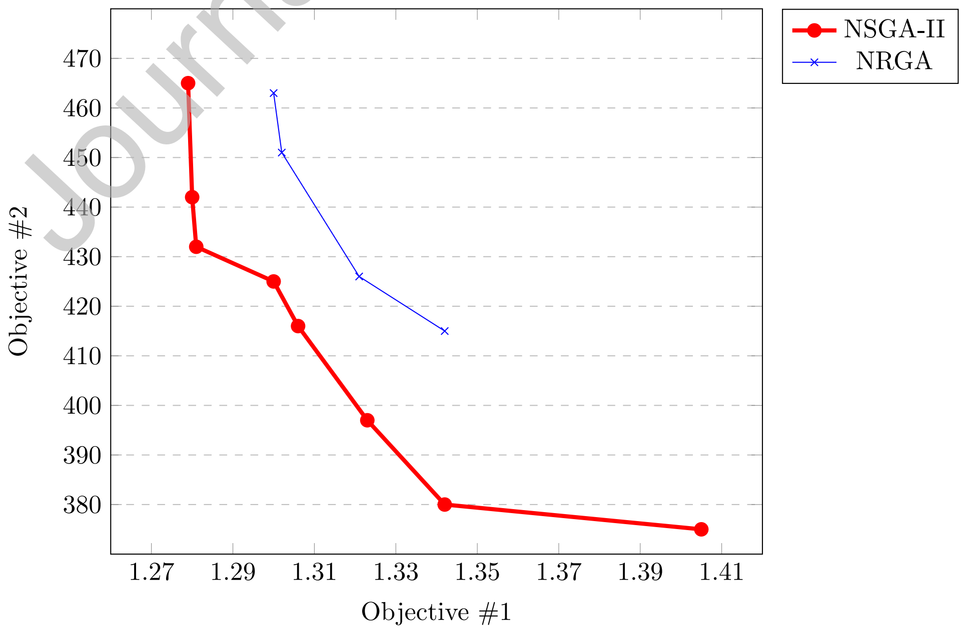


Figure 8: A comparison of the Pareto frontiers for NRGA and NSGA-II for a case with 100 customers

To shed light on the performance of NSGA-II and NRGA, we compared the results from different perspectives including diversity, MID, number of solutions (NOS), spacing and runtime. As discussed earlier, time and Main Ideal Distance (MID) are metrics used to show the convergence of an algorithm whereas spacing, number of solution and diversity indicate the diversification of a algorithm. Figures 9-13 illustrate the performance of each algorithm in each metric which are quite revealing in several ways.

In terms of spacing (Figure 9), we observed that NSGA-II performs relatively better with a lower spacing for the majority of the test problems. The results of comparing the two algorithms with regards to MID does not lend itself to a meaningful comparison as shown in Figure 10. Both algorithms performed well in this criterion with average RPDs around 8%. NRGA showed a more consistent performance though as NSGA-II has led to pretty high RPDs for two large-scale test instances (C1 and D3). In terms of diversity and looking at Figure 11, we observed that the performance of NRGA is relatively better for the larger instances D and E while for the other three groups of problems, there is not a specific dominance by any of the two algorithms. When it comes to the number of solutions generated by each algorithm, Figure 12 illustrates that although NSGA-II seems to generate less Pareto-efficient solutions for some of the small-scale cases, the difference between the two algorithms is not significant for larger instances and hence, the results using this metric are not conclusive. This is in tandem with findings of Rahmati et al. (2013) (although on a different problem) with regards to the number of solutions generated by NSGA-II and NRGA. Nevertheless, this figure is revealing in another way as the performance of NRGA is relatively better for those instances with a smaller capacity of vehicles (A and C). Moreover, Figure 13 reveals that on average, the runtime for the NRGA algorithm is up to ten times more than NSGA-II for the small-scale problems (instances of groups A and B) and for the one third of large-scale ones (instances of group D) while the difference is not considerable for two other large-scale instances of groups C and E.

Taken together, we conclude that the performance of NSGA-II is relatively better than NRGA in terms of spacing and runtime; however, NRGA demonstrates a better performance in terms of diversity. In order to investigate this further, we carried out an additional examination by conducting an Analysis of Variance (ANOVA) test to investigate the potential difference between the two algorithms and if these differences are statistically significant at s 95% confidence level. According to the results in Table 9, the null hypothesis of no difference between the algorithms is rejected for spacing and runtime, showing that the two algorithms are significantly different; however, the differences of the two algorithms are not statistically significant for the other three measures.

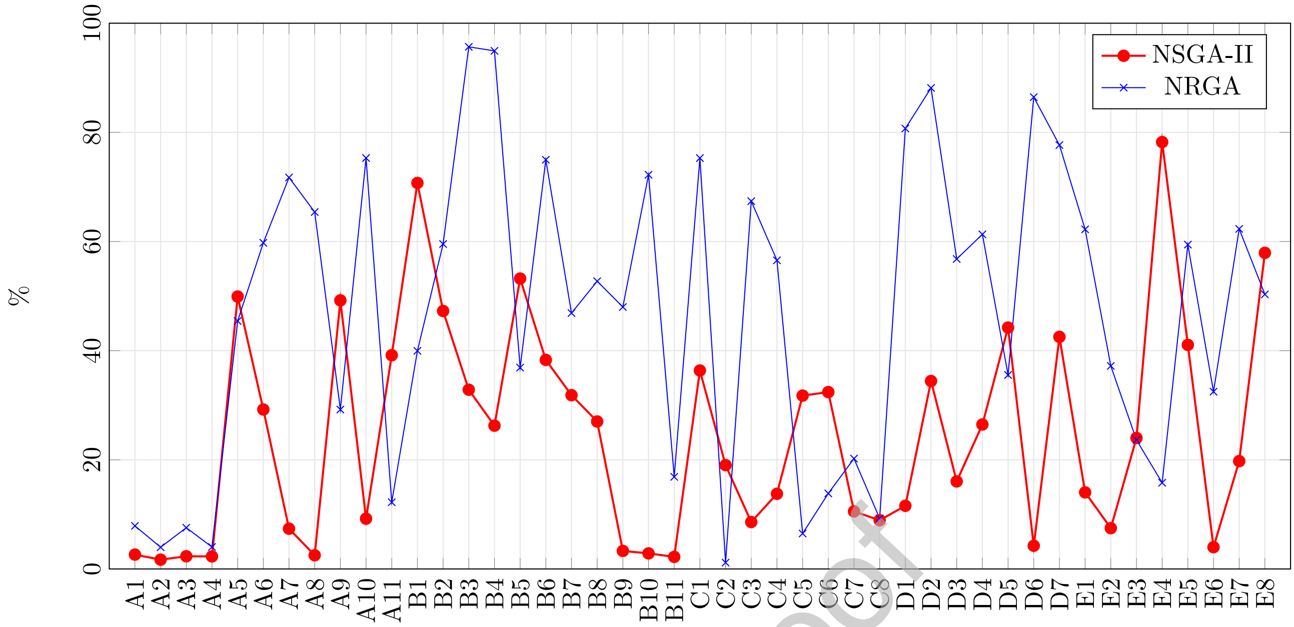


Figure 9: Graphical comparisons of Spacing metric for NSGA-II and NRGA algorithms on all test problems

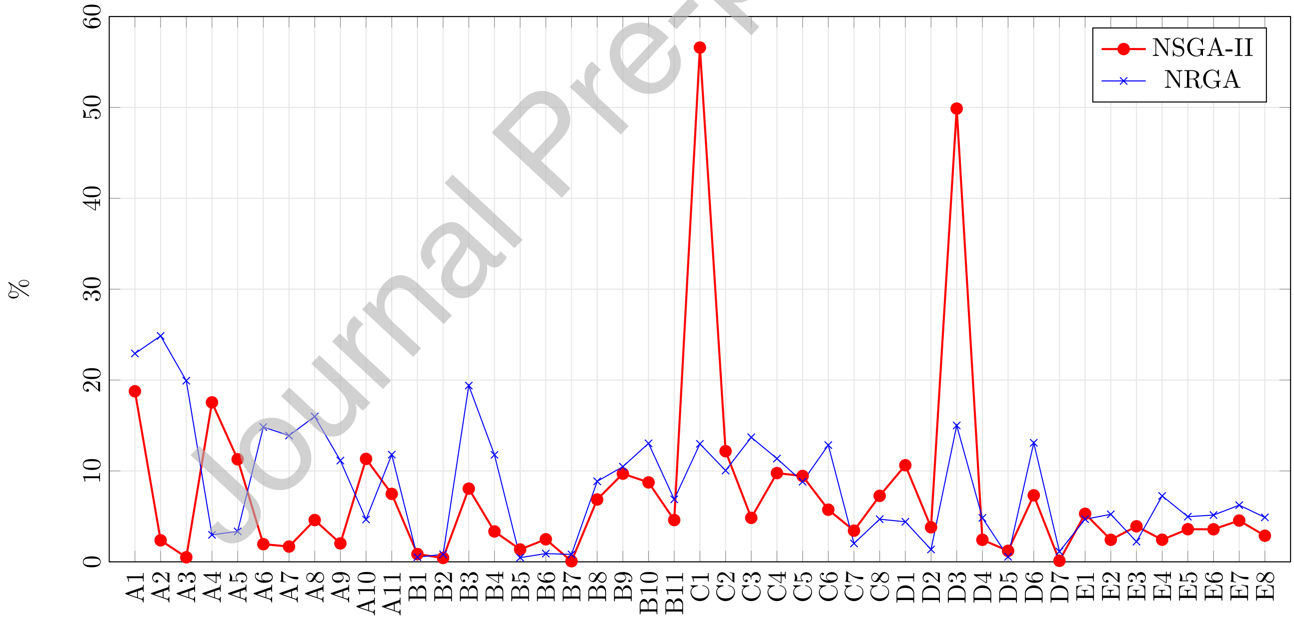


Figure 10: Graphical comparisons of MID for NSGA-II and NRGA algorithms on all test problems

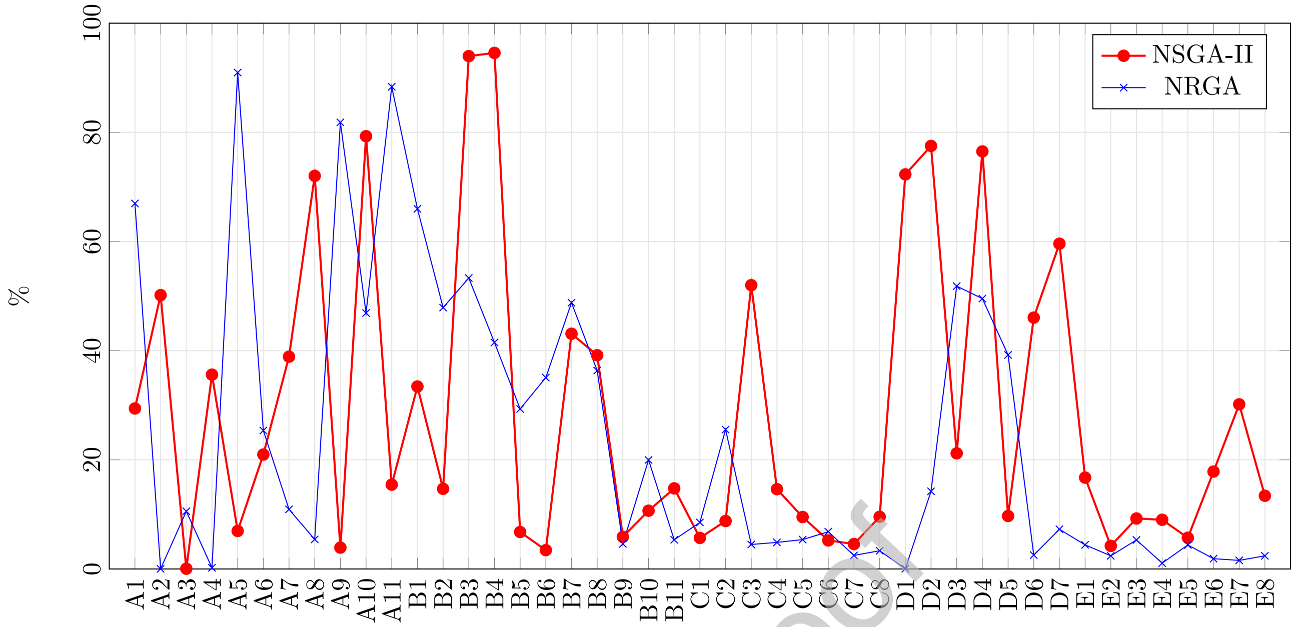


Figure 11: Graphical comparisons of diversity for NSGA-II and NRGA algorithms on all test problems

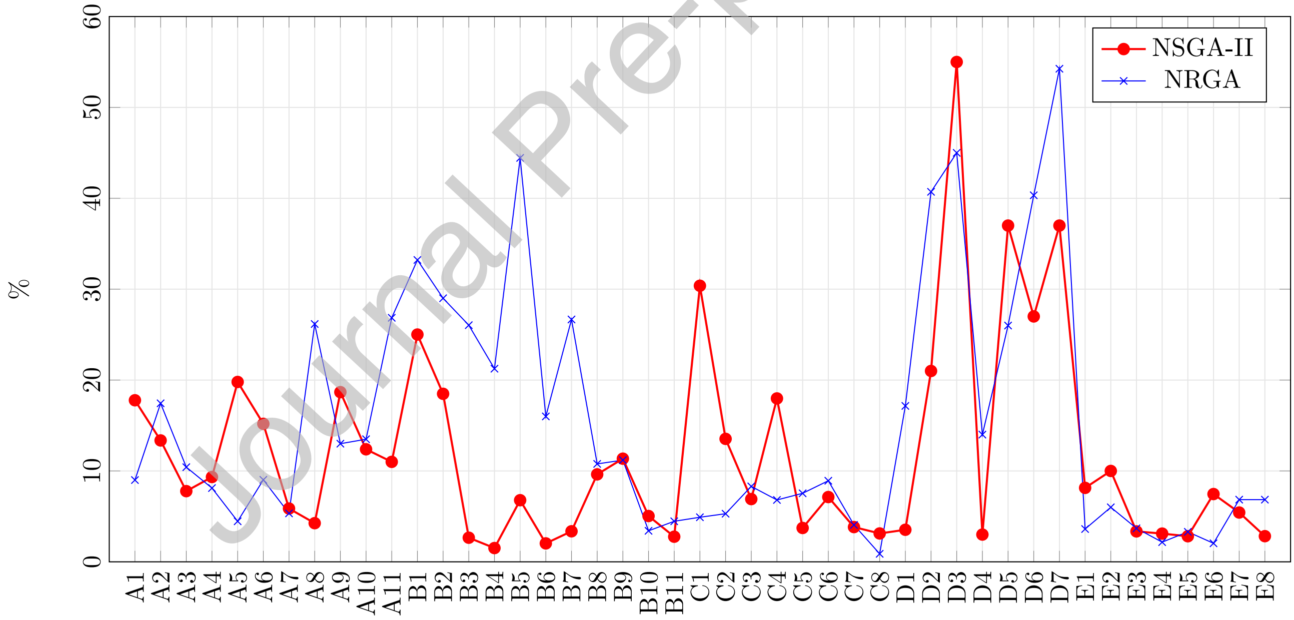


Figure 12: Graphical comparisons of NOS for NSGA-II and NRGA algorithms on all test problems

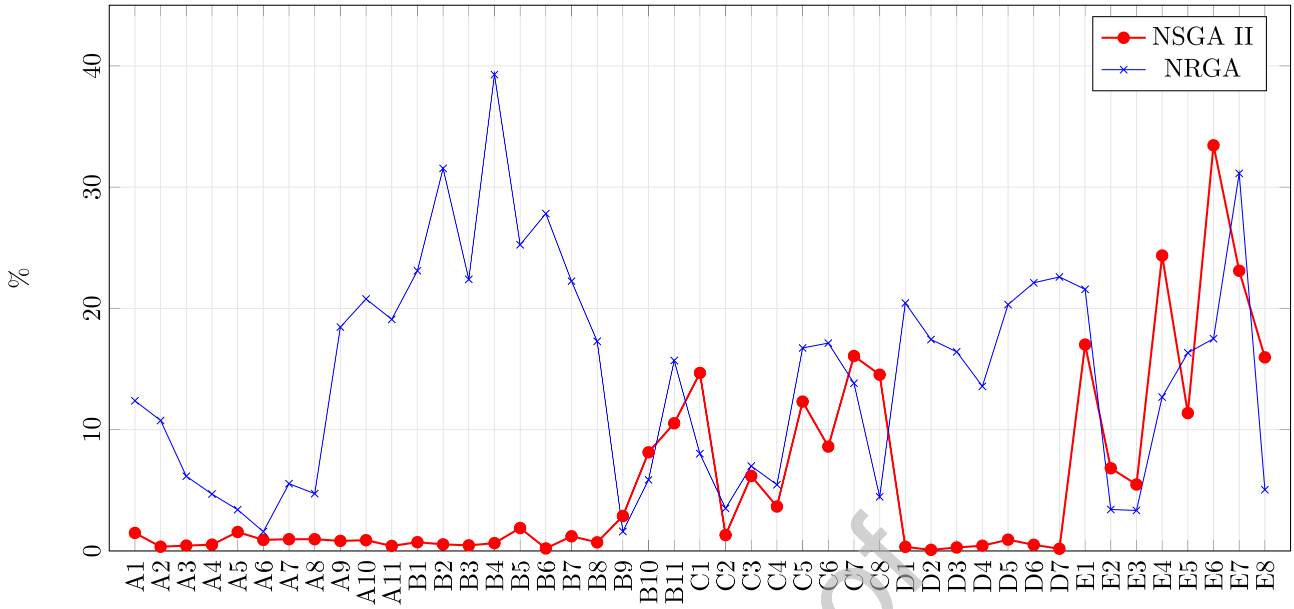


Figure 13: Graphical comparisons of time for NSGA-II and NRGA algorithms on all test problems

Table 9: The results of ANOVA test over the studied bi-objective mathematical model

Metrics	Source	*df*	SS	MS	F-test	p-value	Results
Diversity	Algorithm	1	681.0	681.0	0.95	0.332	Null hypothesis is not rejected
	Error	90	64,292.3	714.4			
	Total	91	64,973.3				
Time	Algorithm	1	1,827	1,827.44	25.70	0.000	**Null hypothesis is rejected**
	Error	90	6,399	71.10			
	Total	91	8,226				
NOS	Algorithm	1	199.4	199.4	1.29	0.260	Null hypothesis is not rejected
	Error	90	13,940.3	154.9			
	Total	91	14,139.7				
Spacing	Algorithm	1	1,827	1,827.44	25.70	0.000	**Null hypothesis is rejected**
	Error	90	6,399	71.10			
	Total	91	8,226				
MID	Algorithm	1	15.67	15.67	0.20	0.655	Null hypothesis is not rejected
	Error	90	7,001.97	77.80			
	Total	91	7,017.65				

6 Conclusion

In this paper, we have focused on a new variant of PLRP in which simultaneous pickups and deliveries must be made to a set of customers who have their time windows of being served and with two conflicting objectives of minimising cost and maximising customer satisfaction. We have presented a novel mathematical model for the problem and solved it using two meta-heuristic algorithms on a set of standard test problems. We then compared the two algorithms according to well-known metrics and the performance of each is reported on the set of test problems. Our results revealed that while NRGA outperforms NSGA-II with regards to diversity of solutions (although not statistically significant), NSGA-II performs better when it comes to spacing and runtime. In terms of number of solutions generated and MID, the results are inconclusive and the difference between the two algorithms is not statistically significant.

No model is perfect and we are aware that ours is not an exception. Therefore, there are several ways this work can be improved in future. Firstly, we assumed the travel times to be known and deterministic, while in real-world applications, they can be uncertain due to unforeseen events and this can be integrated into the model. This will make the problem stochastic and more difficult to solve though. Although the set of vehicles in our study was homogeneous, one can consider the case that vehicles are heterogeneous with varying capacities. Additionally, inter-related routes among depots and transferring loads can be added to our model.. Furthermore, a comparison of other multi-objective optimisation methods with the two addressed in this paper can be a potential area for further research. Including CO_2 emissions in the model as a variant of pollution-routing problem will also add to the value of our research. Last but not least, it would be interesting to solve a real-world instance of our model with potentially another objective.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abido, M. A. (2003), Environmental/economic power dispatch using multiobjective evolutionary algorithms, *in* '2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No. 03CH37491)', Vol. 2, IEEE, pp. 920–925.

Al Jadaan, O., Rajamani, L. and Rao, C. (2008), Non-dominated ranked genetic algorithm for solving multiobjective optimization problems, *in* 'NRGA", Journal of Theoretical and Applied Information Technology', Citeseer.

Alikar, N., Mousavi, S. M., Ghazilla, R. A. R., Tavana, M. and Olugu, E. U. (2017), 'A bi-objective multi-period series-parallel inventory-redundancy allocation problem with time value of money and inflation considerations', *Computers & Industrial Engineering* **104**, 51–67.

Alirezaei, M., Niaki, S. T. A. and Niaki, S. A. A. (2019), 'A bi-objective hybrid optimization algorithm to reduce noise and data dimension in diabetes diagnosis using support vector machines', *Expert Systems with Applications* **127**, 47–57.

Amiri, M., Sadjadi, S., Tavakkoli-Moghaddam, R. and Jabbarzadeh, A. (2019), 'Optimal fleet composition and mix periodic location-routing problem with time windows in an offshore oil and gas industry: A case study of national Iranian oil company', *Scientia Iranica* **26**(1E), 522–537.

Angelelli, E. and Mansini, R. (2002), The vehicle routing problem with time windows and simultaneous pick-up and delivery, *in* 'Quantitative approaches to distribution logistics and supply chain management', Springer, pp. 249–267.

Archetti, C., Fernández, E. and Huerta-Muñoz, D. L. (2017), 'The flexible periodic vehicle routing problem', *Computers & Operations Research* **85**, 58–70.

Avci, M. and Topaloglu, S. (2016), 'A hybrid metaheuristic algorithm for heterogeneous vehicle routing problem with simultaneous pickup and delivery', *Expert Systems with Applications* **53**, 160–171.

Bagheri Hosseini, M., Dehghanian, F. and Salari, M. (2019), 'Selective capacitated location-routing problem with incentive-dependent returns in designing used products collection network', *European Journal of Operational Research* **272**(2), 655–673.

Belgin, O., Karaoglan, I. and Altiparmak, F. (2018), 'Two-echelon vehicle routing problem with simultaneous pickup and delivery: Mathematical model and heuristic approach', *Computers and Industrial Engineering* **115**, 1–16.

Blasco, X., Herrero, J. M., Sanchis, J. and Martínez, M. (2008), 'A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization', *Information Sciences* **178**(20), 3908–3924.

Capelle, T., Cortés, C., Gendreau, M., Rey, P. and Rousseau, L.-M. (2019), 'A column generation approach for location-routing problems with pickup and delivery', *European Journal of Operational Research* **272**(1), 121–131.

Christofides, N. and Eilon, S. (1969), 'Expected distances in distribution problems', *Journal of the Operational Research Society* **20**(4), 437–443.

CNBC (2016), 'A $260 billion 'ticking time bomb': The costly business of retail returns'.
URL: *https://tinyurl.com/yxmoskut*

Collette, Y. and Siarry, P. (2013), *Multiobjective optimization: Principles and case studies*, Springer Science & Business Media.

Dai, Z., Aqlan, F., Gao, K. and Zhou, Y. (2019), 'A two-phase method for multi-echelon location-routing problems in supply chains', *Expert Systems with Applications* **115**, 618–634.

Deb, K. and Jain, H. (2012), Handling many-objective problems using an improved NSGA-II procedure, *in* '2012 IEEE Congress on Evolutionary Computation', IEEE, pp. 1–8.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002), 'A fast and elitist multiobjective genetic algorithm: NSGA-II', *IEEE Transactions on Evolutionary Computation* **6**(2), 182–197.

Demircan-Yildiz, E., Karaoglan, I. and Altiparmak, F. (2016), 'Two echelon location routing problem with simultaneous pickup and delivery: Mixed integer programming formulations and comparative analysis', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **9855 LNCS**, 275–289.

Demirel, N., Özceylan, E., Paksoy, T. and Gökçen, H. (2014), 'A genetic algorithm approach for optimising a closed-loop supply chain network with crisp and fuzzy objectives', *International Journal of Production Research* **52**(12), 3637–3664.

Drexl, M. and Schneider, M. (2015), 'A survey of variants and extensions of the location-routing problem', *European Journal of Operational Research* **241**(2), 283–308.

Financial Times (2016), 'UK retailers count the cost of returns'.
URL: *https://tinyurl.com/y23ab7vp*

Govindan, K., Jafarian, A. and Nourbakhsh, V. (2015), 'Bi-objective integrating sustainable order allocation and sustainable supply chain network strategic design with stochastic demand using a novel robust hybrid multi-objective metaheuristic', *Computers & Operations Research* **62**, 112–130.

Gov.UK (n.d.), 'Electrical waste: Retailer and distributor responsibilities'.
URL: *https://tinyurl.com/yywul6gf*

Gündüz, H. I. (2011), The single-stage location-routing problem with time windows, *in* 'International Conference on Computational Logistics', Springer, pp. 44–58.

Hansen, Z. N. L., Larsen, S. B., Nielsen, A. P., Groth, A., Gregersen, N. G. and Ghosh, A. (2018), 'Combining or separating forward and reverse logistics', *The International Journal of Logistics Management* **29**(1), 216–236.

Hemmelmayr, V. (2015), 'Sequential and parallel large neighborhood search algorithms for the periodic location routing problem', *European Journal of Operational Research* **243**(1), 52–60.

Hemmelmayr, V., Smilowitz, K. and de la Torre, L. (2017), 'A periodic location routing problem for collaborative recycling', *IISE Transactions* **49**(4), 414–428.

Karaoglan, I., Altiparmak, F., Kara, I. and Dengiz, B. (2011), 'A branch and cut algorithm for the location-routing problem with simultaneous pickup and delivery', *European Journal of Operational Research* **211**(2), 318–332.

Karaoglan, I., Altiparmak, F., Kara, I. and Dengiz, B. (2012), 'The location-routing problem with simultaneous pickup and delivery: Formulations and a heuristic approach', *Omega* **40**(4), 465–477.

Karimi, H. (2018), 'The capacitated hub covering location-routing problem for simultaneous pickup and delivery systems', *Computers and Industrial Engineering* **116**, 47–58.

Kheirabadi, M., Naderi, B., Arshadikhamseh, A. and Roshanaei, V. (2019), 'A mixed-integer program and a Lagrangian-based decomposition algorithm for the supply chain network design with quantity discount and transportation modes', *Expert Systems with Applications* .

Koç, Ç. (2016), 'A unified-adaptive large neighborhood search metaheuristic for periodic location-routing problems', *Transportation Research Part C: Emerging Technologies* **68**, 265–284.

Li, J., Pardalos, P., Sun, H., Pei, J. and Zhang, Y. (2015), 'Iterated local search embedded adaptive neighborhood selection approach for the multi-depot vehicle routing problem with simultaneous deliveries and pickups', *Expert Systems with Applications* **42**(7), 3551–3561.

Li, K., Zhou, C., Leung, J. Y. and Ma, Y. (2016), 'Integrated production and delivery with single machine and multiple vehicles', *Expert Systems with Applications* **57**, 12–20.

Moradi, H., Zandieh, M. and Mahdavi, I. (2011), 'Non-dominated ranked genetic algorithm for a multi-objective mixed-model assembly line sequencing problem', *International Journal of Production Research* **49**(12), 3479–3499.

Mosheiov, G. (1994), 'The travelling salesman problem with pick-up and delivery', *European Journal of Operational Research* **79**(2), 299–310.

Nadizadeh, A. and Kafash, B. (2019), 'Fuzzy capacitated location-routing problem with simultaneous pickup and delivery demands', *Transportation Letters* **11**(1), 1–19.

Nagy, G. and Salhi, S. (1998), 'The many-to-many location-routing problem', *Top* **6**(2), 261–275.

Nagy, G. and Salhi, S. (2007), 'Location-routing: Issues, models and methods', *European Journal of Operational Research* **177**(2), 649–672.

Nasherahkami, N., Bashiri, M. and Bagherinejad, J. (2015), Modeling of periodic location routing problem with time window and satisfaction dependent demands, *in* '2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)', IEEE, pp. 1457–1461.

Nikbakhsh, E. and Zegordi, S. (2010), 'A heuristic algorithm and a lower bound for the two-echelon location-routing problem with soft time window constraints', *Scientia Iranica. Transaction E, Industrial Engineering* **17**(1), 36.

Pirkwieser, S. and Raidl, G. R. (2010), Variable neighborhood search coupled with ILP-based very large neighborhood searches for the (periodic) location-routing problem, *in* 'International Workshop on Hybrid Metaheuristics', Springer, pp. 174–189.

Ponboon, S., Qureshi, A. G. and Taniguchi, E. (2016), 'Branch-and-price algorithm for the location-routing problem with time windows', *Transportation Research Part E: Logistics and Transportation Review* **86**, 1 – 19.

Prodhon, C. (2008), A metaheuristic for the periodic location-routing problem, *in* 'Operations Research Proceedings 2007', Springer, pp. 159–164.

Prodhon, C. (2009), 'An ELSxPath relinking hybrid for the periodic location-routing problem', *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **5818 LNCS**, 15–29.

Prodhon, C. (2011), 'A hybrid evolutionary algorithm for the periodic location-routing problem', *European Journal of Operational Research* **210**(2), 204–212.

Prodhon, C. and Prins, C. (2008), A memetic algorithm with population management (MA—PM) for the periodic location-routing problem, *in* 'International Workshop on Hybrid Metaheuristics', Springer, pp. 43–57.

Prodhon, C. and Prins, C. (2014), 'A survey of recent research on location-routing problems', *European Journal of Operational Research* **238**(1), 1–17.

Rabbani, M., Heidari, R. and Yazdanparast, R. (2019), 'A stochastic multi-period industrial hazardous waste location-routing problem: Integrating NSGA-II and Monte Carlo simulation', *European Journal of Operational Research* **272**(3), 945–961.

Rahmati, S. H. A., Hajipour, V. and Niaki, S. T. A. (2013), 'A soft-computing Pareto-based meta-heuristic algorithm for a multi-objective multi-server facility location problem', *Applied Soft Computing* **13**(4), 1728–1740.

Sadeghi, J., Sadeghi, S. and Niaki, S. T. A. (2014), 'A hybrid vendor managed inventory and redundancy allocation optimization problem in supply chain management: An NSGA-II with tuned parameters', *Computers & Operations Research* **41**, 53–64.

Salhi, S. and Rand, G. K. (1989), 'The effect of ignoring routes when locating depots', *European Journal of Operational Research* **39**(2), 150–156.

Schmidt, C. E., Silva, A. C., Darvish, M. and Coelho, L. C. (2019), 'The time-dependent location-routing problem', *Transportation Research Part E: Logistics and Transportation Review* **128**, 293 – 315.

Scott, J. R. (1995), Fault tolerant design using single and multicriteria genetic algorithm optimization, Technical report, Air Force Institute of Tech Wright-Patterson AFB OH.

Statista (2019), 'Retail e-commerce sales worldwide from 2014 to 2021 (in billion U.S. Dollars)'.
URL: *https://tinyurl.com/gqdqyba*

SYSTEMIQ (2017), 'Achieving 'growth within''.
URL: *https://tinyurl.com/y5cw9ksf*

Vidović, M., Ratković, B., Bjelić, N. and Popović, D. (2016), 'A two-echelon location-routing model for designing recycling logistics networks with profit: MILP and heuristic approach', *Expert Systems with Applications* **51**, 34–48.

Wang, C., Mu, D., Zhao, F. and Sutherland, J. (2015), 'A parallel simulated annealing method for the vehicle routing problem with simultaneous pickup-delivery and time windows', *Computers and Industrial Engineering* **83**, 111–122.

Wang, D.-J., Liu, F. and Jin, Y. (2017), 'A multi-objective evolutionary algorithm guided by directed search for dynamic scheduling', *Computers & Operations Research* **79**, 279–290.

Wang, X. and Li, X. (2017), 'Carbon reduction in the location routing problem with heterogeneous fleet, simultaneous pickup-delivery and time windows', *Procedia Computer Science* **112**, 1131–1140.

Webb, M. (1968), 'Cost functions in the location of depots for multiple-delivery journeys', *Journal of the Operational Research Society* **19**(3), 311–320.

Yu, V. F. and Lin, S.-Y. (2016), 'Solving the location-routing problem with simultaneous pickup and delivery by simulated annealing', *International Journal of Production Research* **54**(2), 526–549.

Zarandi, M. H. F., Hemmati, A. and Davari, S. (2011), 'The multi-depot capacitated location-routing problem with fuzzy travel times', *Expert Systems with Applications* **38**(8), 10075–10084.

Zero Waste Scotland (2019), 'Electrical waste: Retailer and distributor responsibilities'.
URL: *https://tinyurl.com/yy6drbr4*

Zhang, B., Li, H., Li, S. and Peng, J. (2018), 'Sustainable multi-depot emergency facilities location-routing problem with uncertain information', *Applied Mathematics and Computation* **333**, 506–520.

Zhang, S., Lee, C. K. M., Wu, K. and Choy, K. L. (2016), 'Multi-objective optimization for sustainable supply chain network design considering multiple distribution channels', *Expert Systems with Applications* **65**, 87–99.

Zhou, L., Lin, Y., Wang, X. and Zhou, F. (2019), 'Model and algorithm for bilevel multisized terminal location-routing problem for the last mile delivery', *International Transactions in Operational Research* **26**(1), 131–156.

Zhou, L., Wang, X., Ni, L. and Lin, Y. (2016), 'Location-routing problem with simultaneous home delivery and customer's pickup for city distribution of online shopping purchases', *Sustainability* **8**(8), 828.

Zio, E. and Bazzo, R. (2012), A comparison of methods for selecting preferred solutions in multiobjective decision making, *in* 'Computational Intelligence Systems in Industrial Engineering', Springer, pp. 23–43.

CRediT author statement

Elham Jelodari Mamaghani: Conceptualization, Methodology, Software, Formal analysis, Writing - Original Draft, Writing - Review & Editing.

Soheil Davari: Conceptualization, Methodology, Formal analysis, Writing - Original Draft, Writing - Review & Editing.