



**HAL**  
open science

## Hub-and-spoke network design under congestion: A learning based metaheuristic

Maryam Karimi-Mamaghan, Mehrdad Mohammadi, Amir Pirayesh, Amir Mohammad Karimi-Mamaghan, Hassan Irani

► **To cite this version:**

Maryam Karimi-Mamaghan, Mehrdad Mohammadi, Amir Pirayesh, Amir Mohammad Karimi-Mamaghan, Hassan Irani. Hub-and-spoke network design under congestion: A learning based metaheuristic. *Transportation Research Part E: Logistics and Transportation Review*, 2020, 142, pp.102069. 10.1016/j.tre.2020.102069 . hal-03212162

**HAL Id: hal-03212162**

<https://imt-atlantique.hal.science/hal-03212162v1>

Submitted on 9 Sep 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

# Hub-and-Spoke Network Design under Congestion: A Learning based Metaheuristic

Maryam Karimi-Mamaghan <sup>a\*</sup>, Mehrdad Mohammadi <sup>a\*</sup>, Amir Pirayesh <sup>b</sup>, Amir Mohammad Karimi-  
Mamaghan <sup>c</sup>, Hassan Irani <sup>d</sup>

<sup>a</sup> *IMT Atlantique, Lab-STICC, UBL, F-29238 Brest, France*

<sup>b</sup> *Centre of Excellence in Supply Chain and Transportation (CESIT), KEDGE Business School, Bordeaux, France*

<sup>c</sup> *Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran*

<sup>d</sup> *Construction Management and Engineering, Civil and Architecture Faculty, Islamic Azad university, Science and Research  
Branch, Tehran, Iran*

## Abstract

This paper models a single allocation multi-commodity hub-and-spoke network problem through a bi-objective mathematical model, considering the congestion in both hubs and connection links. A novel aggregation model is developed based on a general GI/G/c queuing system to evaluate the congestion of the flow of the multiple products in the hubs. The proposed model is then solved using a novel learning-based metaheuristic based on NSGA-II, *k*-Means clustering method, and an Iterated Local Search (ILS) algorithm. The proposed model and solution algorithm are validated through a set of experiments against optimal solutions, and benchmarked against four existing well-known algorithms.

**Keywords:** Hub-and-spoke network design; Bi-objective optimization; Congestion; Queuing network; Machine Learning; *k*-Means clustering method; Learning-based Metaheuristics.

---

\* Corresponding authors:

Maryam Karimi Mamaghan, E-mail: [maryam.karimi@imt-atlantique.fr](mailto:maryam.karimi@imt-atlantique.fr)

Mehrdad Mohammadi, Tel: +33 2 29 00 10 30, E-mail: [mehrdad.mohammadi@imt-atlantique.fr](mailto:mehrdad.mohammadi@imt-atlantique.fr)

# Hub-and-Spoke Network Design under Congestion: A Learning-based Metaheuristic

## Abstract

This paper models a single allocation multi-commodity hub-and-spoke network problem through a bi-objective mathematical model, considering the congestion in both hubs and connection links. A novel aggregation model is developed based on a general GI/G/c queuing system to evaluate the congestion of the flow of the multiple products in the hubs. The proposed model is then solved using a novel learning-based metaheuristic based on NSGA-II, *k*-Means clustering method, and an Iterated Local Search (ILS) algorithm. The proposed model and solution algorithm are validated through a set of experiments against optimal solutions, and benchmarked against four existing well-known algorithms.

**Keywords:** Hub-and-spoke network design; Bi-objective optimization; Congestion; Queuing network; Machine Learning; *k*-Means clustering method; Learning-based Metaheuristics.

## 1. Introduction

Hub-and-spoke systems have been largely employed in many-to-many transportation systems, wherein commodities (i.e., passengers, cargo parcels, telecommunication packets, etc.) are transferred between a many pairs of origin-destination (OD) nodes. Instead of directly connecting each pair of OD nodes, in a hub-and-spoke system, commodities are transferred through a set of intermediate nodes called hubs (Correia et al., 2018; Mohammadi et al., 2019a). Hubs serve the network as switching, transshipment, sorting, and distribution facilities. In a path from an origin to a destination, commodities from different origin nodes are consolidated at these hub facilities prior to be routed to an intermediate hub or to be delivered to their final destinations (Zhalechian et al., 2018; Taleizadeh et al., 2018, Zheng et al., 2019). The aggregation of commodities in hub facilities allows the exploitation of economies of scale due to the utilization of more efficient carriers with higher capacities on hub-to-hub connection links.

Different versions of hub-and-spoke systems have been proposed and studied in the literature differing from one another in several aspects as follows: 1) the allocation of each spoke to one and only one hub (i.e., single allocation) or to multiple hubs (i.e., multiple allocation), 2) the number and type of the located hubs, 3) the presence of absence of hub capacities (i.e., capacitated vs. uncapacitated), and 4) full connectivity (i.e., complete graph) or partial connectivity of hubs (i.e., incomplete graph) (Alumur and Kara, 2008; Dukkanci et al., 2019).

Although the exploitation of economies of scale is a big advantage of hub networks, this exploitation may lead to commodity overload in a small number of hubs, or even result in heavy-utilization/congestion of some hub-to-hub connections. This congestion becomes more critical for transportation companies that employ the hub-and-spoke network for shipment delivery where high delivery performance (i.e., low transportation time) is desired. Congestions in hub nodes and hub-to-hub connections highly affect the delivery performance, and hence, it is essential to take congestion effects into account when designing a hub-and-spoke network (de Camargo et al., 2009; de Camargo and Miranda, 2012; Mohammadi et al., 2019a; Rahimi et al., 2016).

More specifically, one of the main real applications of this study can be found in the national/worldwide shipment transportation system. In this system, different shipments with different priorities are consolidated in

1 the origin hubs. These shipments are then processed (e.g., packaged, loaded, etc.) and transited in larger volumes  
2 to different destinations through different hubs. Once arrived to the destination hub, shipments are again  
3 processed (e.g., unloaded, unpackaged, etc.) and sent to their final destinations. Since the capacity of hubs are  
4 limited, arrival shipments to each hub are congested and they should wait until they receive services. Such  
5 waiting time in the hubs significantly affect the performance of the network in term of transportation time  
6 between the moment that the shipment demand takes place until the moment that the shipment arrives to its final  
7 destination.

8 Three ways of addressing congestion in hub-and-spoke systems are 1) imposing classical capacity  
9 constraints on hub nodes for limiting the amount of flow entering the hubs, 2) incorporating costs of congestion  
10 effects explicitly into the objective function using a convex cost function that increases exponentially as more  
11 flows go through the hubs, and 3) accounting for congestion in hub nodes using queuing theory, and calculating  
12 the waiting time of the flow in terms of congestion.

13 The third approach has recently attracted more attention (Ishfaq and Sox, 2012; Mohammadi et al., 2017,  
14 2019a) since it accounts for the congestion in the hubs in a more reliable and real way. In this regard, the most  
15 common way has been modelling the hubs as queuing system with arrival and/or service rates with Poisson  
16 distributions (i.e.,  $M/M/c$  and  $M/D/c$  queuing systems). However, the stochasticity nature of the demand between  
17 OD nodes imposes a high degree of uncertainty on the inter-arrival time of the products. On the other hand, high  
18 variety of demands results in varying service rate of the products. Accordingly, employing general  $GI/G/c$   
19 queuing system provides more reliable results (Ishfaq and Sox, 2012; Mohammadi et al., 2019b). Evaluating the  
20 congestion in the hubs when dealing with general  $GI/G/c$  queuing system become more complicated when  
21 dealing with multiple products with different characteristics (e.g., service rate, priority etc.).

22 In this regard, two real-world cases where the problem of this paper steps in are: 1) Shipment delivery  
23 system and 2) airway passenger transportation system. In the following, the first real-world application is  
24 explained in more detail. One of the most important shipment delivery systems is the mail or post system for  
25 physically transporting postcards, letters, and parcels. These shipments have different priorities (e.g., high  
26 priority, medium priority and low priority) and each shipment is processed based on its priority. In this system,  
27 different shipments with different priorities are first consolidated in the origin hub which is normally the center  
28 of each province in the country. These shipments are then processed (e.g., packaged, loaded, etc.) and transited  
29 in larger volumes to different destinations through different hubs and maybe with different transportation modes.  
30 Once arrived at the destination hub, shipments are again processed (e.g., unloaded, unpackaged, etc.) and sent to  
31 their final destinations. Since the capacity of hubs are limited, arrival shipments to each hub are congested and  
32 they should wait until they receive services. Such waiting time in the hubs significantly affects the performance  
33 of the network in terms of transportation time between the moments that the shipment demand takes place until  
34 the moment that the shipment arrives to its final destination. An efficient way to evaluate this congestion in the  
35 hubs is using queuing theory. On the other hand, the service time and inter-arrival time of shipments are  
36 stochastic and finding an exact distribution for them is hard (if not impossible). Accordingly, using typical  
37 queuing systems with exponential inter-arrival time and Poisson service rate are not practical enough. Therefore,  
38 general queuing systems (e.g.,  $GI/G/1$ ,  $GI/G/c$ ) are the most efficient way to cope with this congestion  
39 evaluation. At each hub, the processing of shipments (e.g., unloading, sorting, packaging and loading) are done  
40 through different parallel working centers. For example, once trucks that transport the shipments arrive at the

1 hubs, they are referred to different ports (servers) of unloading and the shipment processes (unloading and  
2 sorting) are done in parallel. Since a wide variety of shipments may exist in each truck, service time at each  
3 server does not follow a unique and specific distribution. On the other hand, the arrival time of shipments to the  
4 hubs does not follow a specific stochastic distribution. Therefore, the service time and inter-arrival time of  
5 shipments should follow general distribution and using general queuing systems is the most efficient way to take  
6 into account the congestion of shipments in the hubs.

7 This paper addresses a bi-objective multi-product single allocation hub-and-spoke network design  
8 problem, wherein congestion is considered in both hubs and connection links. A GI/G/c queuing system is  
9 developed to evaluate the waiting time of the hubs. Finally, an aggregation model is proposed to handle multiple  
10 products in the GI/G/c queuing system. The first objective function of the model minimizes the sum of  
11 installation, allocation, and transportation costs, and the second objective function of the model minimizes the  
12 maximum transportation time between each pair of OD nodes that is significantly affected by congestion in the  
13 hubs and connection links.

14 Finally to solve the proposed model for large-sized instances, this paper develops a new hybrid  
15 metaheuristic algorithm based on non-dominated sorting genetic algorithm-II (NSGA-II) and a learning-based  
16 Iterated Local Search (L-ILS) algorithm. The idea of this hybridization is to develop an algorithm that is  
17 powerful in terms of both diversification (global search) and intensification (local search) and intelligently learns  
18 information during the searching process.

19 The rest of the paper is organized as follows. The most relevant paper considering the congestion in the  
20 hub are reviewed in Section 2. A bi-objective mathematical model is proposed in Section 3. Section 4 develops a  
21 hybrid metaheuristic algorithm to solve the proposed bi-objective mathematical model. The performance of the  
22 proposed hybrid algorithm is validated in Section 5 through numerous experiments. Section 6 deals with  
23 comprehensive sensitivity analysis of the model. Finally, the paper is concluded in Section 7.

## 24 **2. Literature review**

25 This section reviews the most relevant papers addressing congestion in the hub location problem.

### 26 **2.1. Addressing congestion through capacity constraint**

27 Although numerous papers have addressed the congestion effects by restricting the amount of flow  
28 entering a hub using capacity constraints, few of them have had the purpose of addressing congestion in the hub  
29 network. Accordingly, this part reviews those papers that initially aim at addressing congestion through capacity  
30 constraint.

31 As the first efforts, Grove and O'Kelly (1986) addressed the impact of congestion on hub-and-spoke  
32 networks. They showed how the delays in the schedules of airline systems are affected by the amount of flow  
33 entering the hubs. Marianov and Serra (2003) are among the first researchers that accounted for congestion in  
34 hub-and-spoke networks using queuing theory. They modeled the hub network as an M/D/c queuing system and  
35 proposed capacity constraints based on the waiting probability of flows in the hubs. They have also proposed a  
36 model for allocating servers to each installed hub. A similar work has been proposed by Rodriguez et al. (2007),  
37 wherein each hub is modeled with simple M/M/1 queuing system. Costa et al. (2008) aim at controlling  
38 congestion by minimizing the total time required to process the flow entering each hub. For this aim, the authors  
39 propose a second objective function that minimizes the maximum processing time of flows at hubs. Mohammadi  
40 et al. (2011) have modeled each hub as M/M/c queuing system and proposed a probabilistic constraint to ensure

1 that the probability of entering flows to the hubs that wait in a queue is less than a threshold value for each hub.  
2 Rahimi et al. (2016) have accounted for congestion through a queuing system with finite queue capacity.  
3 Accordingly, the authors model each hub as  $M/M/1/K$  queue system, where the probability of the flow exceeding  
4 the capacity of the hub is tried to be minimized.

5 Yang and Chiu (2016) addressed the hub network design problem considering demand uncertainty and  
6 hub congestion effects. Authors formulated the problem as a two-stage stochastic program with recourse model.  
7 The proposed model provides a consistent set of hub locations, while adjusting network configuration in  
8 response to different demand realizations. Özgün-Kibiroğlu (2019) addressed the hub location problem in which  
9 capacity restrictions were introduced into the objective function as a penalty cost to represent their congestion  
10 effects on respective hubs. The authors proposed a model allocating hubs whose capacities are larger than a  
11 desired level and still congestion on hubs is considered as a penalty cost in the objective function. Therefore,  
12 hubs are chosen between nodes which have higher capacities in order to reduce the penalty costs arising from  
13 surplus flows on hubs.

14 These works attempt to control congestion via limiting the flow entering hubs; however, merely  
15 controlling the amount of flow does not guarantee the higher performance of the hub network in terms of  
16 delivery time. Indeed, once the amount of flow approaches the capacity, congestion would happen in hub,  
17 although the capacity is still respected.

## 18 **2.2. Addressing congestion through cost function**

19 Modeling congestion through capacity constraint on the flows does not reflect the exponential nature of  
20 congestion effects: the more the flow into the hub, the harder the handling process. Consequently, greater costs  
21 are imposed to the hub network. Usually these costs increase extremely rapid due to congestion.

22 In this regard, Elhedhli and Hu (2005) have explicitly considered the congestion effect of each located  
23 hub as a cost term in the objective function for the hub-and-spoke problems. Using a power-law function which  
24 is widely utilized to estimate delay costs in airport applications, the authors propose a non-linear convex cost  
25 function formulation that increases rapidly as more traffic flows through the located hubs. Afterward, Elhedhli  
26 and Wu (2009) have proposed the same approach while considering each hub as an  $M/M/1$  queue and used the  
27 Kleinrock's average delay function (Kleinrock, 2007) as a representation of the congestion effects.

28 Camargo and Miranda (2009) have studied a multiple allocation hub-and-spoke network design problem  
29 under hub congestion. They have proposed a non-linear mixed integer programming formulation, modeling the  
30 congestion as a convex cost function similar to Elhedhli and Hu (2005). In another work, Camargo and Miranda  
31 (2012) have addressed the hub-and-spoke network problem under congestion from two different network design  
32 perspectives; the network owner and the network user. The authors translate these two perspectives into  
33 mathematical programming models. The objective of the model is to minimize the installation, congestion, and  
34 routing costs over the network, wherein two different perspectives are analyzed: The network owner aims at  
35 designing network with the least cost, and the network user is willing to accept the minimum congestion effect at  
36 a reasonable cost. The proposed cost functions in the literature are disabled to evaluate the waiting time of the  
37 products, and they only account for the number of products accumulated in each hub. Evaluating the waiting  
38 time and the delay of products in the hubs are important when transport service providers aim at minimizing the  
39 total transportation time and attempt to offer the most competitive delivery services to their customers.

1 Kian and Kargar (2016) studied the hub location problem with a power-law congestion cost and propose  
2 an exact solution approach. Authors formulated this problem in a conic quadratic form and use a strengthening  
3 method which rests on valid inequalities of perspective cuts in mixed integer nonlinear programming. Alkaabneh  
4 et al. (2019) considered a hub-and-spoke network design problem with inter-hub economies-of-scale and hub  
5 congestion. They explicitly modeled the economies-of-scale as a concave piece-wise linear function and  
6 congestion as a convex function. The problem has been modeled as a nonlinear mixed integer program that is  
7 difficult to solve directly since the objective function has both convex and concave nonlinear terms and hence  
8 finding an optimal solution may not be easy. Finally, the authors proposed a Lagrangian approach to obtain tight  
9 upper and lower bounds.

### 10 **2.3. Addressing congestion through waiting time calculation**

11 A set of other works exist that address the congestion in the hub-and-spoke network, not via capacity  
12 constraint and congestion cost function, but by analyzing the waiting time resulted from the flow accumulation  
13 in the network. These works reflect the exponential nature of congestion effects in terms of the time spent by the  
14 products in the network.

15 In this regard, Ishfaq and Sox (2012) modeled the hubs as a GI/G/1 queuing systems and the shipments as  
16 multiple job classes with deterministic routings. By integrating the hub operation queuing model and the hub  
17 location-allocation model, the authors also investigated the effect of limited hub resources on the design of  
18 intermodal logistics networks under service time requirements. Mohammadi et al. (2017) have proposed a hub-  
19 and-spoke network for hazardous material (HAZMAT) transportation, wherein the congestion of HAZMAT in  
20 the hubs increases the risk of incidents. In addition, the risk increases more and more when the waiting time of  
21 the flows becomes longer and longer. Accordingly, the authors aim at minimizing the risk of HAZMAT  
22 congestion in the hubs by minimizing the waiting time of flows in the network. In their model, each hub is  
23 modeled as M/M/c queuing system where HAZMATs have different priorities, and HAZMATs with higher  
24 priority are served first.

25 As the most recent work, Mohammadi et al. (2019a) have designed a hub network by addressing  
26 congestion in both hubs and connection links. They consider each hub as M/M/1 queue system and model  
27 congestion at connection links via the Bureau roads link performance function (Lo and Tung, 2003). The authors  
28 try to minimize the maximum transportation time between each pair of OD nodes. This time is affected by the  
29 congestion in the hubs as well as at the connection links.

30 Although these studies are able to evaluate the products' waiting time in the hubs, the considered queuing  
31 systems with Poisson arrival and service rates distributions (i.e., M/M/c and M/D/c queuing systems) are not  
32 able to fully capture the high variety of products and consequently high variation of the service time in the hubs.  
33 Accordingly, a general queuing system should be required to control the congestion of the hub-and-spoke  
34 network.

### 35 **2.4. This paper's contributions**

36 Table 1 summarizes the relevant literature of HLP in terms of modeling and solution approaches.  
37 Regarding the modeling approach, the reviewed papers are evaluated if they deal with: 1) single or multiple  
38 objectives, 2) congestion in the hubs and connection links, 3) classical or general queuing systems, and 4)  
39 aggregation of the flow in the hubs. Regarding the solution approach, the reviewed papers are classified whether

1 the authors develop 1) exact methods or metaheuristic algorithms as local search, global search or hybrid search  
 2 algorithms and 2) learning-based search algorithms.

3  
 4  
 5  
 6  
 7

Table 1. Review of related work

Ref.	Year	Modeling approach						Solution Approach					
		Objective		Congestion		Queuing System		Aggregation model	Exact Method	Local Search	Global search	Hybrid	Learning-based
		Single	Multiple	Hub	Link	Classical queue	General queue						
O'Kelly	1986	✓		✓									
Marianov and Serra	2003	✓		✓		✓			✓				
Elhedhli and Hu	2005	✓		✓		✓		✓					
Rodriguez et al.	2007	✓		✓		✓			✓				
Costa et al.	2008		✓	✓		✓		✓					
Elhedhli and Wu	2009	✓		✓		✓		✓					
Camargo and Miranda	2009	✓		✓		✓		✓					
Camargo and Miranda	2012	✓		✓		✓		✓					
Ishfaq and Sox	2012	✓		✓			✓		✓				
Mohammadi et al.	2013	✓		✓		✓					✓		
Sedehzadeh et al.	2014	✓		✓		✓					✓		
Rahimi et al.	2016		✓	✓		✓					✓		
Yang et al.	2016	✓		✓							✓		
Kian and Kargar	2016	✓		✓				✓					
Mohammadi et al.	2017		✓	✓		✓					✓	✓	
Azizi et al.	2018	✓		✓							✓		
Hu et al.	2018	✓			✓		✓						
Mohammadi et al.	2019b		✓	✓	✓	✓					✓	✓	
Özgün-Kibiroğlu et al.	2019	✓		✓							✓		
Alkaabneh et al.	2019	✓		✓					✓	✓			
<b>This paper</b>			✓	✓	✓		✓	✓	✓	✓	✓	✓	✓

8

9

Based on Table 1, the main points that distinguish this work from the literature are listed below:

10

- Despite the literature, we study a multi-product version of the hub-and-spoke network design problem where multiple products are consolidated in the hubs and they need to be processed before being transferred toward their destinations.

11

12

13

- Almost all of the papers in the literature model the hubs as queuing system with arrival and/or service rates with Poisson distributions (i.e.,  $M/M/c$  and  $M/D/c$  queuing systems), while the stochasticity nature of the demand between OD nodes imposes a high degree of uncertainty on the arrival rate of the products. On the other hand, high variety of demands results in varying service rate of the products. Accordingly, this paper develops a general  $GI/G/c$  queuing system to model the hubs.

14

15

16

17

18

- Since each hub processes multiple products simultaneously, products experience the same mean waiting time in each hub. Therefore, an aggregation model is required that considers all products and calculates

19



1 the overall waiting time in each hub. In this paper we consider for the first time a queue-based  
 2 aggregation model for calculating the overall waiting time in a hub-and-spoke network.

- 3 • In addition to the contributions in the mathematical model, this paper proposes a new hybrid  
 4 metaheuristic algorithm based on non-dominated sorting genetic algorithm-II (NSGA-II) and a learning  
 5 based Iterated Local Search (L-ILS) algorithm. The idea of this hybridization is to develop an algorithm  
 6 that is powerful in terms of both diversification (global search) and intensification (local search) and  
 7 intelligently learns information during the searching process.
- 8 • Finally, sensitivity analyses are performed to investigate the performance of the network under stochastic  
 9 disruption of the queuing systems as well as under the prioritization of the products.

### 10 3. Mathematical Model

11 In this section the congested bi-objective multi-product single allocation hub-and-spoke network problem  
 12 under congestion is presented. Hereafter, this problem is simply called BiMSHC problem. In this problem, the  
 13 goal is to locate  $h$  hubs in the network from  $H$  potential candidate locations. Then, a set of  $N$  spokes are allocated  
 14 to located hubs. Each product  $p$  originating from a spoke is consolidated only in a single hub (i.e., single  
 15 allocation). The graph of hubs is complete, meaning that each pair of located hubs are directly connected.  
 16 Accordingly, the product flow between each pair of OD spokes, requires to pass through at least one hub (when  
 17 both OD spokes are allocated to the same hub) or at most two hubs (when OD spokes are allocated to different  
 18 hubs). Due to the limited capacity of the network, congestion happens in both 1) hubs that process the products  
 19 by unloading, sorting, packaging and loading of products, and 2) connection links.

20 The rest of this section is organized as follows. Necessary notations are first presented in Section 2.1.  
 21 Next, Section 2.2 presents the BiMSHC problem in terms of a mixed-integer non-linear programming (MINLP)  
 22 model. Afterwards, Section 2.3 and 2.4 model the congestion in the connection links and the hubs, respectively.  
 23 Finally, a piecewise function technique is proposed in Section 2.5 to linearize the proposed non-linear BiMSHC  
 24 model.

#### 25 3.1. Notations

26 Necessary notations are provided in this section.

##### Sets

$N$	Set of spokes
$H$	Set of potential hubs
$P$	Set of products

##### Indices

$i, j \in N$	Indices of spokes
$k, l \in H$	Indices of hubs
$p \in P$	Index of products

##### Parameters

$f_k^p$	Fixed cost of locating a hub at candidate node $k$ to serve product $p$ .
$c_{ij}^p$	Transportation cost of transferring product $p$ between spokes $i$ and $j$ .
$t_{ij}^p$	Mean transportation time of transferring product $p$ between spokes $i$ and $j$ .
$w_{ij}^p$	Flow of product $p$ between spokes $i$ and $j$ .
$\mu_k^p$	Service rate of processing product $p$ at hub $k$ .
$\tau_{S,k}^p$	Service time of processing product $p$ at hub $k$ (i.e., $\tau_{S,k}^p = 1/\mu_k^p$ ). “S” stands for service.
$c_{A,pi}^2$	Squared coefficient of variation (SCV) of the inter-arrival time of product $p$ from spoke $i$ . “A” stands for arrival.
$c_{A,pk}^2$	SCV of the inter-arrival time of product $p$ at hub $k$ .

$c_{S,pk}^2$	SCV of the service time of product $p$ at hub $k$ .
$Q_{kl}$	Capacity of link between hubs $k$ and $l$ for transferring the products. “S” stands for service.
$c_k$	Number of servers at hub $k$ .
$h$	Number of hubs to be located in the network.
$\alpha_C$	Cost discount factor of transferring flow between hubs (i.e., $0 < \alpha_C < 1$ ).
$\alpha_T$	Time discount factor of transferring flow between hubs ( $0 < \alpha_T < 1$ ).

### Decision variables

$X_{ik}^p$	1 if spoke $i$ is allocated to hub $k$ for product $p$ ; 0 otherwise.
$Z_k$	1 if a hub is located at candidate node $k$ ; 0 otherwise.
$X_{iklj}^p$	1 if the flow of product $p$ between spokes $i$ and $j$ passes first through hub $k$ then hub $l$ .
$T_{iklj}^p$	Variable transportation time of transferring product $p$ originated from spoke $i$ with destination to spoke $j$ passing first through hub $k$ then hub $l$ .
$T_{kl}^p$	Variable transportation time of transferring product $p$ from hub $k$ to hub $l$ .
$\lambda_{A,k}^p$	Flow of product $p$ arriving at hub $k$ .
$W_k^p$	Variable waiting time of processing product $p$ at hub $k$ .
$F_{kl}^p$	Amount of flow of product $p$ traversing the link between hub $k$ and $l$ in both directions.

1

### 2 3.2. Proposed MINLP Model

3 Based on the notations provided in Section 3.1, the proposed mixed-integer non-linear programming  
4 model, to present the BiMSHC problem, is proposed as follow. Consider  $c_{iklj}^p = c_{ik}^p + \alpha_C c_{kl}^p + c_{lj}^p$ .

$$\mathbb{Z}_1 = \min \sum_{p=1}^P \sum_{i=1}^N \sum_{k=1}^H \sum_{l=1}^H \sum_{j=1}^N w_{ij}^p c_{iklj}^p X_{iklj}^p + \sum_{p=1}^P \sum_{k=1}^H f_k^p Z_k \quad (1)$$

$$\mathbb{Z}_2 = \min \max_{i,j,k,l,p} \{T_{iklj}^p\} \quad (2)$$

s.t.

$$\sum_{k=1}^H Z_k = h \quad (3)$$

$$\sum_{k=1}^H \sum_{l=1}^H X_{iklj}^p = 1 \quad \forall i, j, p \quad (4)$$

$$X_{iklj}^p \leq Z_k \quad \forall i, j, k, l, p \quad (5)$$

$$X_{iklj}^p \leq Z_l \quad \forall i, j, k, l, p \quad (6)$$

$$X_{iklj}^p, Z_k \in \{0,1\} \quad \forall i, j, k, l, p \quad (7)$$

5

6 Objective function (1) minimizes the sum of total transportation cost and the fixed cost of locating the  
7 hubs. Objective function (2) minimizes the maximum transportation time between each pair of OD nodes;  
8 wherein  $T_{iklj}^p$  is the transportation time of transferring product  $p$  originated from spoke  $i$  with destination at spoke  
9  $j$  passing first through hub  $k$  then hub  $l$ . Constraint (3) determines the number of hubs to be located in the  
10 network. Constraint (4) ensures the single allocation of the spokes to the hub nodes for each product. Constraints  
11 (5) and (6) guarantee that the routes between OD nodes only traverse located hubs. Finally, constraint (7) is the  
12 integrality constraint for the decision variables.

13 Objective function (2) is not a linear function; therefore, its linear form is provided through a new  
14 objective function (8) and constraint (9), wherein  $\mathbb{T}$  is the maximum transportation time between each pair of  
15 OD nodes.

$$\min \mathbb{T} \quad (8)$$

$$T_{iklj}^p \leq \mathbb{T} \quad \forall i, j, k, l, p \quad (9)$$

1 In constraint (9), the variable  $T_{iklj}^p$  is the sum of the transportation time on the route from spoke  $i$  to spoke  
 2  $j$  as well as the waiting time of the products at the hubs  $k$  and  $l$ . Accordingly,  $T_{iklj}^p$  is calculated as Equation (10).

$$T_{iklj}^p = (t_{ik}^p + W_k^p + \alpha_T T_{kl}^p + W_l^p + t_{lj}^p) X_{iklj}^p \quad \forall i, j, k, l, p \quad (10)$$

3 The variable transportation time of product  $p$  between hubs  $k$  and  $l$ ,  $T_{kl}^p$ , and the variable waiting time of  
 4 product  $p$  at hub  $k$ ,  $W_k^p$ , are calculated in Sections 3.3 and 3.4, respectively.

### 5 3.3. Variable inter-hub transportation time

6 A typical consideration in the hub-and-spoke literature is that the transportation time over the hub  
 7 network is deterministic or stochastic but independent of the congestion of the connection links. In this section,  
 8 we model the transportation between each pair of hub nodes (as the most congested connection links) as a  
 9 function of a) capacity of the link and b) the amount of the flow traversing the link (Mohammadi et al., 2019a).  
 10 For this aim, the Bureau roads link performance function (Lo and Tung, 2003) is employed. It should be noticed  
 11 that the mean transportation time of product  $p$  at hub  $k$  to hub  $l$ ,  $T_{kl}^p$ , is equal for all products. Accordingly, the  
 12 mean transportation time at hub-to-hub link  $k$  to  $l$ ,  $T_{kl}$ , is formulated as Equation (11):

$$T_{kl}(F_{kl}, Q_{kl}) = t_{kl} \left[ 1 + \beta_{kl} \left( \frac{F_{kl}}{Q_{kl}} \right)^{\zeta_{kl}} \right] \quad (11)$$

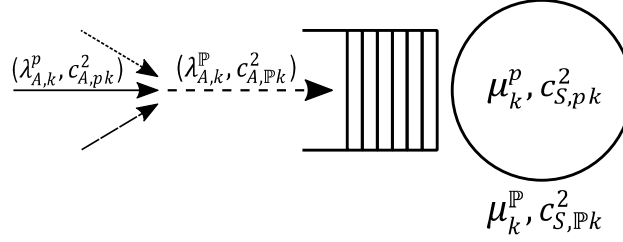
13 where  $t_{kl}$  is link free-flow travel time between hubs  $k$  and  $l$ , and  $F_{kl}$  is the total flow of products traversing the  
 14 link from hub  $k$  to hub  $l$  (i.e.,  $F_{kl} = \sum_p F_{kl}^p$ ). It is also considered that the transportation time is the same for all  
 15 types of the products. As the congestion ratio  $F_{kl}/Q_{kl}$  increases, the transportation time of the link increases. The  
 16 overload happens in the link if  $\frac{F_{kl}}{Q_{kl}} > 1$  and no overload happens for cases where  $0 < \frac{F_{kl}}{Q_{kl}} \leq 1$ . In Equation (11),  
 17 the constant and deterministic parameters  $\beta_{kl}$  and  $\zeta_{kl}$  determine how the congestion ratio affects the  
 18 transportation time. The parameter  $\beta_{kl}$  determines how the congestion ratio ‘directly’ affects the transportation  
 19 time and the parameter  $\zeta_{kl}$  is the shape of this effect that could be linear ( $\zeta_{kl}=1$ ), polynomial ( $\zeta_{kl}=2$ ) or with  
 20 higher degrees ( $\zeta_{kl} > 2$ ). For instance, if  $\beta_{kl}=1$  and  $\zeta_{kl}=2$ , the congestion ratio affects the transportation time  
 21 over the link between hubs  $k$  and  $l$  in a polynomial manner (i.e.,  $T_{kl}(F_{kl}, Q_{kl}) = t_{kl} \left[ 1 + \left( \frac{F_{kl}}{Q_{kl}} \right)^2 \right]$ ).

### 22 3.4. Variable waiting time at hub

23 In general, the products arriving in a hub have to wait in a queue if the hub is busy (Sedehzadeh et al.,  
 24 2014). This waiting mainly depends on the rate at which the products arrive at the hub and the rate at which the  
 25 hub processes the products. Since the inter-arrival time between the two arrivals and the service time of the hub  
 26 are not deterministic and distributed around a mean value, it is essential to include the variability in time and its  
 27 distribution while evaluating the performance indicators that account for the congestion in the hub (i.e., waiting  
 28 time, queue length, throughput rate). Although most of the papers in the literature model the hubs as queuing  
 29 system with arrival and/or service rates with Poisson distributions (i.e., M/D/c and M/M/c queuing systems), the  
 30 stochasticity nature of the demand between OD nodes imposes a high degree of uncertainty on the arrival rate of  
 31 the products. In addition, high variety of demands results in varying service rate of the products. Accordingly,  
 32 we model each hub as a multi-server GI/G/c queue, wherein the inter-arrival times between flow units of each  
 33 product  $p$  are given by a random variable with general distribution and mean  $1/\lambda_{A,k}^p$ . The service times are  
 34 random variables with general distribution and mean  $1/\mu_k^p$ . At each hub, there exists a queue with infinite size,  
 35 wherein the products are served in a first-come-first-served (FCFS) rule. At hub  $k$ ,  $c_k$  parallel servers process the

1 products and each server serves only one unit of each product at a time and devotes all of its resources to  
 2 complete the service. The throughput rate of product  $p$  at hub  $k$  is  $\rho_k^p = \lambda_{A,k}^p / (c_k \mu_k^p)$  (i.e.,  $\rho_k^p = \tau_{S,k}^p / (c_k \tau_{A,k}^p)$ ).

3 Since obtaining an exact value for a performance indicator for hubs modeled by GI/G/c queues is difficult  
 4 if not impossible, an approximation technique is required. As the first time in the hub location problem, the fork-  
 5 join analysis (Satyam and Krishnamurthy, 2008) is adopted to evaluate the performance of each hub separately.  
 6 In this analysis, the arrival products are aggregated into a single product (called aggregated product  $\mathbb{P}$ ) and the  
 7 performance of the hub is evaluated for the aggregated product. The parameters for the aggregated product are  
 8 approximated based on the parameters of the original products at each hub. Figure 1 illustrates hub  $k$  that  
 9 processes a given set of products.



10  
 11 Figure 1. Hub  $k$  and aggregated product (Mohammadi et al., 2019b)  
 12

13 Based on the notations in Section 3.1, Equations (12) to (17) are proposed to estimate the parameters of  
 14 the aggregated product (Mohammadi et al., 2018, 2019b).

$$\lambda_{A,k}^p = \sum_{i=1}^N \sum_{j=1}^N \sum_{l=1}^H w_{ij}^p X_{iklj}^p + \sum_{i=1}^N \sum_{j=1}^N \sum_{l=1}^H w_{ji}^p X_{jlki}^p \quad \forall k, p \quad (12)$$

$$\lambda_{A,k}^{\mathbb{P}} = \sum_{p=1}^P \lambda_{A,k}^p \quad \forall k \quad (13)$$

$$\tau_{S,k}^{\mathbb{P}} = \sum_{p=1}^P \left( \frac{\lambda_{A,k}^p}{\lambda_{A,k}^{\mathbb{P}}} \right) \tau_{S,k}^p \quad \forall k \quad (14)$$

$$c_{A,pk}^2 = \vartheta_{pk} \gamma_{pk}^2 + (1 - \vartheta_{pk}) \quad \forall k, p \quad (15)$$

$$c_{A,\mathbb{P}k}^2 = \theta_k \delta_k^2 + (1 - \theta_k) \quad \forall i \quad (16)$$

$$c_{S,\mathbb{P}k}^2 = \frac{\left\{ \sum_{p=1}^P \left( \frac{\lambda_{A,k}^p}{\lambda_{A,k}^{\mathbb{P}}} \right) (c_{S,pk}^2 + 1) (\tau_{S,k}^p)^2 \right\} - (\tau_{S,k}^{\mathbb{P}})^2}{(\tau_{S,k}^{\mathbb{P}})^2} \quad \forall k \quad (17)$$

15

16 where  $\vartheta_{pk} = [1 + 4(1 - \rho_{pk})^2 (v_{pk} - 1)]^{-1}$ ,  $\rho_{pk} = \sum_i \left( \frac{\sum_{j=1}^N \sum_{l=1}^H w_{ij}^p X_{iklj}^p + \sum_{j=1}^N \sum_{l=1}^H w_{ji}^p X_{jlki}^p}{\mu_k^p} \right)$ ,  $\gamma_{pk}^2 =$

17  $\sum_i \left( \frac{\sum_{j=1}^N \sum_{l=1}^H w_{ij}^p X_{iklj}^p + \sum_{j=1}^N \sum_{l=1}^H w_{ji}^p X_{jlki}^p}{\lambda_{A,k}^p} \right) c_{A,pi}^2$  and  $v_{pk}^{-1} = \sum_i \left( \frac{\sum_{j=1}^N \sum_{l=1}^H w_{ij}^p X_{iklj}^p + \sum_{j=1}^N \sum_{l=1}^H w_{ji}^p X_{jlki}^p}{\lambda_{A,k}^p} \right)^2$ . In addition,  $\theta_k =$

18  $[1 + 4(1 - \rho_{\mathbb{P}k})^2 (v_k - 1)]^{-1}$ ,  $\delta_k^2 = \sum_p \left( \frac{\lambda_{A,k}^p}{\lambda_{A,k}^{\mathbb{P}}} \right) c_{A,pk}^2$  and  $v_k^{-1} = \sum_p \left( \frac{\lambda_{A,k}^p}{\lambda_{A,k}^{\mathbb{P}}} \right)^2$ . After calculating the parameters of

19 the aggregated product  $\mathbb{P}$  at hub  $k$ , the waiting time of product  $p$  in the queue at hub  $k$ ,  $W_k^p$ , is provided as  
 20 Equation (18), where in  $\rho_k^{\mathbb{P}} = \lambda_{A,k}^{\mathbb{P}} \tau_{S,k}^{\mathbb{P}} / c_k$ . The main point is that all the products entering a particular hub,  
 21 where there is no priority between them, experience the same waiting time in that hub.

$$W_k^p \approx W_k^{\mathbb{P}} \approx \frac{\phi_k^{\mathbb{P}} \tau_{S,k}^{\mathbb{P}}}{1 - \rho_k^{\mathbb{P}}} \times \frac{c_{A,\mathbb{P}k}^2 + c_{S,\mathbb{P}k}^2}{2} \times G_k^{\mathbb{P}} \quad \forall k, p \quad (18)$$

1 where  $\phi_k^{\mathbb{P}}$  and  $G_k^{\mathbb{P}}$  are calculated as Equations (19) and (20):

$$\phi_k^{\mathbb{P}} = \frac{(c_k \rho_k^{\mathbb{P}})^{c_k}}{c_k! (1 - \rho_k^{\mathbb{P}})} \left[ \sum_{n=0}^{c_k-1} \frac{(c_k \rho_k^{\mathbb{P}})^n}{n!} + \frac{(c_k \rho_k^{\mathbb{P}})^{c_k}}{c_k!} \frac{1}{1 - \rho_k^{\mathbb{P}}} \right]^{-1} \quad \forall k \quad (19)$$

$$G_k^{\mathbb{P}} = \begin{cases} \exp\left(-\frac{2}{3} \times \frac{1 - \rho_k^{\mathbb{P}}}{\rho_k^{\mathbb{P}}} \times \frac{(1 - c_{A,\mathbb{P}k}^2)^2}{c_{A,\mathbb{P}k}^2 + c_{S,\mathbb{P}k}^2}\right), & 0 \leq c_{A,\mathbb{P}k} \leq 1 \\ \exp\left(-\frac{1 - \rho_k^{\mathbb{P}}}{c_{A,\mathbb{P}k}^2 + c_{S,\mathbb{P}k}^2}\right), & c_{A,\mathbb{P}k} > 1 \end{cases} \quad \forall k \quad (20)$$

2

3

### 4 3.5. Linearization of the MINLP model: A piecewise function

5 The proposed model in Section 3.2 along with the variable transportation time and approximated waiting  
6 time of the Sections 3.3 and 3.4 is non-linear. Accordingly, finding an optimal solution even for some small-size  
7 instances of the problem might be impossible or very time-consuming. In this section, we employ a piecewise  
8 linear approximation technique (Mohammadi et al., 2019a) to approximately linearize the non-linear terms  $T_{kl}$   
9 and  $W_k^{\mathbb{P}}$ .

10 Let  $f(a)$  be the piecewise linear approximation of a single variable by considering  $S$  number of sampling  
11 coordinates  $a_1, \dots, a_S$  on the  $s$  axis (*breakpoints*), on which the function is evaluated. The function is then  
12 approximated by the linear term  $[(a_s, f(a_s)), (a_{s+1}, f(a_{s+1}))]$  ( $s = 1, \dots, S - 1$ ). Therefore, for any given  $a$   
13 value, where  $a_s \leq \bar{a} \leq a_{s+1}$ , the function value  $f^{\approx}(\bar{a})$  is approximated as Equations (21) and (22).

$$\bar{a} = \omega a_s + (1 - \omega) a_{s+1} \quad (21)$$

$$f^{\approx}(\bar{a}) = \pi f(a_s) + (1 - \pi) f(a_{s+1}) \quad (22)$$

14 where  $\pi$  is a (unique) value in  $[0, 1]$ .

15 To use this linearization technique, a set of new variables and constraints should be defined to associate  
16 any  $a$  value to the proper pair of consecutive breakpoints. Since  $T_{kl}^{\mathbb{P}}$  and  $W_k^{\mathbb{P}}$  are non-linear variables, two  
17 piecewise linear approximations are needed. Consider a continuous variable  $v_s$  for each breakpoint  $s$ , where  $v_s \in$   
18  $[0, 1]$ ; ( $s = 1, \dots, S$ ). In addition,  $u_s$  is a binary variable associated with the  $s$ th interval  $[a_s, a_{s+1}]$  ( $s = 1, \dots, S -$   
19  $1$ ), where  $u_0 = u_S = 1$  at the extremes. Finally, the approximate value  $f^{\approx}$  can be obtained by imposing the  
20 following constraints and introducing special  $v_s$  and  $u_s$  corresponding to each nonlinear term. Required  
21 notations are first presented and additional constraints are then presented to linearize the non-linear terms of  
22 objective function (2).

#### Parameters:

$\lambda_{A,k,\pi}^{\mathbb{P}}$	$\pi$ th breakpoint of the total flow arriving at hub $k$ ( $\lambda_{A,k}^{\mathbb{P}}$ ).
$F_{kl,\pi}$	$\pi$ th breakpoint of the total flow traversing the link $k$ to $l$ ( $F_{kl}$ ).
$W(\lambda_{A,k,\pi}^{\mathbb{P}})$	Waiting time corresponding to the total flow $\lambda_{A,k,\pi}^{\mathbb{P}}$ arriving at hub $k$ .
$T(F_{kl,\pi})$	Transportation time corresponding to the total flow $F_{kl,\pi}$ traversing the link $k$ to $l$ .
$G$	An arbitrary large number.

#### Decision variables:

$v_s^{W,k}$	Continuous variable for breakpoint $\pi$ associated with $W(\lambda_{A,k,\pi}^{\mathbb{P}})$ .
$v_s^{T,kl}$	Continuous variable for breakpoint $\pi$ associated with $T(F_{kl,\pi})$ .
$u_s^{W,k}$	Binary variable for breakpoint $\pi$ associated with $W(\lambda_{A,k,\pi}^{\mathbb{P}})$ .
$u_s^{T,kl}$	Binary variable for breakpoint $\pi$ associated with $T(F_{kl,\pi})$ .

23

1 Constraints (23) to (34) are provided to linearly approximate the non-linear terms  $T_{kl}$  and  $W_k^p$  in the  
 2 proposed model.

$$\sum_{s=1}^{S-1} u_s^{W,k} = 1 \quad \forall k \quad (23)$$

$$v_s^{W,k} \leq u_{s-1}^{W,k} + u_s^{W,k} \quad \forall k, s \in S \quad (24)$$

$$\sum_{s=1}^S v_s^{W,k} = 1 \quad \forall k \quad (25)$$

$$\lambda_{A,k}^p = \sum_{s=1}^S v_s^{W,k} \lambda_{A,k,s}^p \quad \forall k \quad (26)$$

$$W_k^p = \sum_{s=1}^S v_s^{W,k} W(\lambda_{A,k,s}^p) \quad \forall k \quad (27)$$

$$\sum_{s=1}^{S-1} u_s^{T,kl} = 1 \quad \forall k, l \quad (28)$$

$$v_s^{T,kl} \leq u_{s-1}^{T,kl} + u_s^{T,kl} \quad \forall k, l, s \in S \quad (29)$$

$$\sum_{s=1}^S v_s^{T,kl} = 1 \quad \forall k, l \quad (30)$$

$$F_{kl} = \sum_{s=1}^S v_s^{T,kl} F_{kl,s} \quad \forall k, l \quad (31)$$

$$T_{kl} = \sum_{s=1}^S v_s^{T,kl} T(F_{kl,s}) \quad \forall k, l \quad (32)$$

$$v_s^{W,k}, v_s^{T,kl} \in [0,1] \quad \forall k, l, s \in S \quad (33)$$

$$u_s^{W,k}, u_s^{T,kl} \in \{0,1\} \quad \forall k, l, s \in S \quad (34)$$

3

4 Finally, the final proposed mixed-integer linear programming model is presented as follow:

$$\mathbb{Z}_1 = \min \sum_{p=1}^P \sum_{i=1}^N \sum_{k=1}^H \sum_{l=1}^H \sum_{j=1}^N w_{ij}^p c_{iklj}^p X_{iklj}^p + \sum_{p=1}^P \sum_{k=1}^H f_k^p Z_k \quad (1)$$

$$\mathbb{Z}_2 = \min \mathbb{T} \quad (8)$$

$$(t_{ik}^p + W_k^p + \alpha_T T_{kl}^p + W_l^p + t_{lj}^p) X_{iklj}^p \leq \mathbb{T} \quad \forall i, j, k, l, p$$

s.t.: Constraints (3)-(7), (23)-(34)

5

#### 6 4. Hybrid metaheuristic solution algorithm

7 Due to the complexity of the hub-and-spoke network design problem to find the optimal solution (Alumur  
 8 and Kara, 2008), metaheuristic algorithms have been widely proposed to solve this problem (Rodríguez et al.,  
 9 2007; Mohammadi et al., 2016; Rahimi et al., 2016; Mohammadi et al., 2017; Zhalechian et al., 2018;  
 10 Mohammadi et al., 2019a). Among them, hybrid metaheuristic algorithms, as the combination of evolutionary  
 11 algorithms (EAs) and local search (LS) algorithms, have attracted higher attention for their outperformance in  
 12 terms of both diversification and intensification of the solution space (Mohammadi et al., 2019a).

13 In recent years, the combination between evolutionary algorithms (EAs) and machine learning (ML) and  
 14 has received considerable attention from the research community. Interested readers are referred to (Jourdan et  
 15 al., 2006; Zhang et al., 2011; Calvet et al., 2017) and references therein. In this regard, this section develops a  
 16 hybrid metaheuristic solution algorithm to solve the proposed BiMSHC model based on Non-dominated Sorting  
 17 Genetic Algorithm-II (NSGA-II) (Deb et al., 2000) and a Learning-based Iterated Local Search (L-ILS)

1 algorithm. The well-known k-means clustering algorithm (Likas et al., 2003) is used as the machine learning  
2 technique. For simplicity, the proposed hybrid algorithm is then called GAMLS.

3 As a population-based EA, NSGA-II can present a set of non-dominated Pareto (NDP) optimal solutions  
4 of the multi-objective optimization problems (MOOPs) with a set of conflicting objective functions (Deb et al.,  
5 2000). NSGA-II evolves a population of randomly generated solutions with size  $P_{size}$  using elitism, crossover and  
6 mutation operators. Though elitism, a percent  $E_R$  (elitism rate) of the best solutions are directly transferred to the  
7 next generation. The rest of the population of the new generations are filled by crossover and mutation operators  
8 with rates  $C_R$  and  $M_R$ , respectively, where  $E_R + C_R + M_R = 1$ . In crossover, a pair of selected parents (using  
9 tournament selection mechanism) are crossed with the hope of creating better children. Despite crossover,  
10 mutation operator is employed on an individual solution in order to diversify the solution space and escape from  
11 the local optimum. These operators are employed to generate new solutions as a next offspring so that the current  
12 population and generated offspring are combined together for creating next generation according to non-  
13 dominance and crowding distance (CD) techniques. The CD is a measure of the density of solutions. The value  
14 of the CD presents an estimate of the density of solutions surrounding a particular solution. Accordingly, a  
15 solution with higher CD value is preferred. The idea behind the CD is that isolated solutions with high CD value  
16 belong to undiscovered regions and keeping these solutions in the population increases the diversification  
17 capability of the algorithm. This evolution process continues until a maximum iteration  $Itr_{max}^{GA}$  is reached.

18 ILS is a well-known metaheuristic algorithm for solving NP-hard optimization problems due to its  
19 effectiveness in both intensification and diversification as well as its simplicity in practice. When a search is  
20 trapped in a local optimum, ILS helps the search to escape the trap without losing many of the good properties of  
21 the current solution. The main iteration loop of the ILS algorithm does three main following steps (Karimi-  
22 Mamaghan et al., 2020): 1) ILS algorithm performs a *Perturbation(.)* function over the current local optimum  
23 solution  $s_i^*$  to help the search to escape the local minimum; whereby an intermediate solution  $s_i'$  is generated.  
24 Through this random perturbation, the local optimum solution  $s_i^*$  can be transferred to another place in the  
25 solution space, 2) After applying the *Perturbation(.)* function, the *LocalSearch(.)* function is performed on the  
26 intermediate solution  $s_i'$  to obtain a new local optimal solution  $s_i^{*'}$ , and 3) After applying the *LocalSearch(s\_i')*  
27 function, *AcceptanceCriterion(s\_i^\*, s\_i^{\*'}, s\_{best})* is employed to check whether to accept the current local optimal  
28 solution  $s_i^*$  comparing to the  $s_i^*$ . The *AcceptanceCriterion(.)* function can only accept better solution or it can  
29 even accept worse solution with a small gap (e.g., 5% gap from  $s_i^*$ ). It is furtherly checked if  $s_{best}$  can be also  
30 updated. These three steps continue until a maximum iteration  $Itr_{max}^{ILS}$  is reached.

#### 31 **4.1. Proposed hybrid GAMLS algorithm**

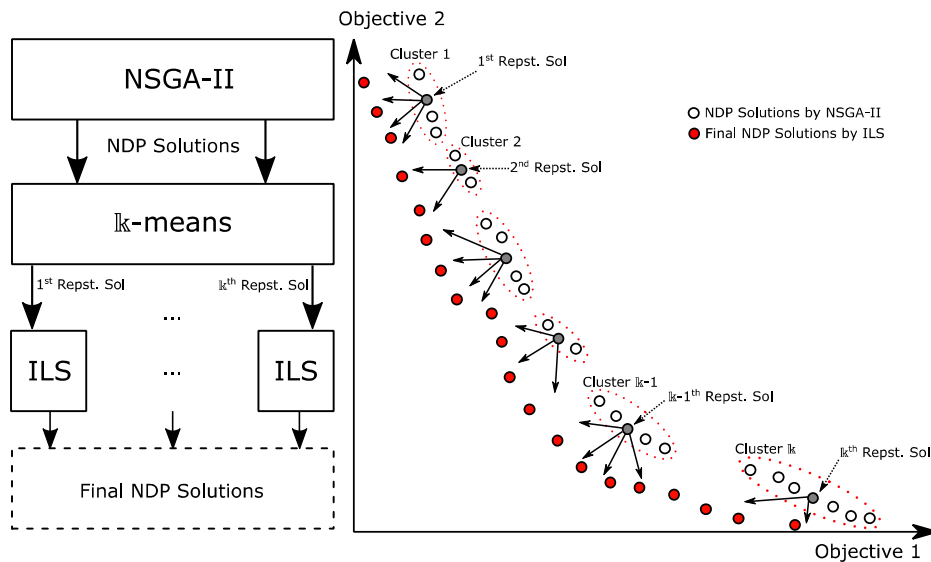
32 In hybrid algorithms for solving MOOPs, three main issues need to be resolved concerning the local  
33 search mechanism: 1) which solutions to initiate the local search, 2) how to select solutions from a set of  
34 candidate solutions, and 3) how to perform the local search on the selected solutions in order to escape from the  
35 local optimum and find better solutions. Regarding the first issue, the set of NDP solutions are more preferable  
36 to be the candidate to perform local search (Chiang et al., 2011; Wang and Tang, 2017). Sometimes, the number  
37 of NDP solutions are too high and performing a local search over each of them is computationally inefficient (if  
38 not impossible). But it should be noted that the number of initial solutions should be also enough to have a good  
39 diversification. So a set of representative solutions among the NDP solutions to initiate the local searches are  
40 necessary. For the third issue, the local search algorithm should be not only powerful in intensification, but also

1 performant enough in terms of diversification. In addition, an efficient mechanism should exist to escape from  
 2 the local optimum.

3 Based on this motivation, the hybrid GAMLS algorithm is proposed to handle these three issues. The  
 4 NSGA-II, as a powerful population-based EA, is responsible to obtain NDP solution to initiate the ILS. In the  
 5 meantime,  $k$ -means clustering algorithm is used as the link between NSGA-II and ILS to select the  
 6 representative solutions from the results of NSGA-II and to feed them to ILS algorithm.

7 Once the NSGA-II obtains the NDP solutions, the  $k$ -means algorithm is employed to find  $k$   
 8 representative solutions (Repst. Sol) from the NDP set. Actually, the  $k$ -means clustering algorithm is employed  
 9 over the solution representation of the solutions. The  $k$ -means algorithm creates  $k$  separate solution clusters,  
 10 wherein the solutions in each cluster are closed together and probably from the same region of the solution space  
 11 (i.e., same local optimum); while the solutions of a cluster are probably far from the other clusters' solutions  
 12 (i.e., different local optimums). One may create clusters based on the objective function values of the solutions  
 13 but it does not guarantee that the solutions in one cluster are closed together. Indeed, two solutions may have the  
 14 same objective function but they belong two different regions of the solution space. Afterward, the solution at  
 15 the center of each cluster or the closest solution to the center of the cluster will represent the whole cluster. The  
 16 closeness of two solutions is defined as the minimum difference between the solution representations of those  
 17 two solutions. Finally, these  $k$  representative solutions are fed to the ILS algorithm and their neighborhood is  
 18 intensified. At the end, the new found NDP solutions via ILS executions are combined with those of NSGA-II  
 19 and the final NDP solutions are returned. Figure 2 depicts the overall mechanism of the hybrid GAMLS  
 20 algorithm.

21



22  
 23  
 24

Figure 2. Mechanism of the hybrid GAMLS algorithm

#### 25 4.2. Solution representation

26 The solution vector representation in every EA should be as compact as possible but should contain  
 27 enough information to represent any solution of the problem. The way of representing solutions significantly  
 28 affect the choice of searching operators. Accordingly, efficient representation of the solutions helps to use well-  
 29 known operators in the literature that their high performance has been proved. An important recommendation is  
 30 that represent the solution in a way that applying operators do not possibly lead to infeasible solutions. But if a

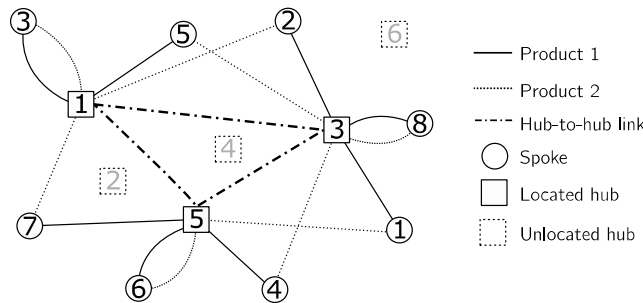


1 solution vector is infeasible and violates one or certain number of constraints, a penalty value can be added to its  
 2 corresponding objective function values. This penalty value decreases the chance of that solution to participate in  
 3 the reproduction procedures. Generally, it is much better to propose a representation that only includes feasible  
 4 solutions.

5 In this paper, a solution is represented as a  $(P \times N)$  matrix where the arrays are the  $h$  values from  $\{1, \dots, H\}$ .  
 6 Figure 3 depicts a hub-and-spoke network with  $P=2$  products,  $N=8$  spokes, and  $h=3$  hubs among a set of  $H=6$   
 7 possible candidate locations. It can be seen that spokes 1 to 8, transfer product 1 to hubs 3, 3, 1, 5, 1, 5, 5, and 3,  
 8 respectively. For product 2, the assignment of spokes is to hubs 5, 1, 1, 3, 3, 5, 1, and 3, respectively. Figure 4  
 9 depicts the corresponding hub-and-spoke network of Figure 3.

	Spokes							
	1	2	3	4	5	6	7	8
Product 1	3	3	1	5	1	5	5	3
Product 2	5	1	1	3	3	5	1	3

10 Figure 3. Solution representation of a multi-commodity hub-and-spoke network  
 11



12 Figure 4. Hub-and-spoke network design of Figure 3  
 13  
 14

15 Having the structure of the hub-and-spoke network (e.g., Figure 3 and 4), the objective functions of the  
 16 proposed model can be easily calculated. The first objective function is calculated based on the decision  
 17 variables that shows which spokes have been allocated to which hubs and which hubs have been located in the  
 18 network. These decision variables can be calculated through the structure of the network determined by the  
 19 solution representation. For the second objective function, two terms should be calculated, the mean  
 20 transportation time at hub-to-hub link  $k$  to  $l$ ,  $T_{kl}$ , and the waiting time in hub  $k$  for product  $p$ ,  $W_k^p$ . In these terms  
 21 of the second objective function, the amount of flow of products traversing the spoke-to-hub and hub-to-hub  
 22 links should be first calculated. To calculate these amounts of flow, it should be noted first that each spoke  $i$   
 23 is allocated to only a single hub  $k$  (single allocation) for each product  $p$ . Next, the flow of product  $p$  over the  
 24 connection link from spoke  $i$  to hub  $k$  is equal to the total flow originating from spoke  $i$  ( $\sum_j w_{ij}^p$ ). The amount of  
 25 flow between each pair of hubs  $k$  and  $l$  ( $F_{kl}$ ) can be simply calculated based on the spokes allocated to these two  
 26 hubs  $k$  and  $l$  and the corresponding flow between each pair of the spokes. Having the amounts flow traversing  
 27 the hub network, the mean transportation time at hub-to-hub link  $k$  to  $l$ ,  $T_{kl}$ , and the waiting time in hub  $k$  for  
 28 product  $p$ ,  $W_k^p$  are calculated by equations (11) and (18)-(20), respectively.

### 29 4.3. Crossover, mutation & local search operators

30 For doing crossover on two selected parents, two types of crossover operators are used as *one-point*  
 31 crossover and *two-point* crossover (Mohammadi et al., 2013). In *one-point* crossover, two parents are crossed  
 32 from one cutting point across the columns of the solution matrix. In *two-point* crossover, two parents are crossed

1 from two cutting point across the columns of the solution matrix. At each call of crossover function, one operator  
2 is employed randomly. For employing mutation on a single solution, two types of mutation operators are used as  
3 *reversion* and *random generation*. In the *reversion* operator, two columns far enough each other are selected and  
4 the permutation of the in-between columns are reversed. In the *random generation* mutation, a new solution is  
5 generated randomly. Similar to crossover, mutation function uses one operator at each call of doing mutation.

6 In the ILS algorithm, local search operators are *swap*, *re-allocation* and *hub-removal* operators. In a *swap*  
7 operator, two adjacent arrays at each row of the solution matrix are swapped. In *re-allocation*, a spoke is selected  
8 at each row of the solution matrix and it is re-allocated to an existing hub. Finally, in *hub-removal* operator, a  
9 hub is selected from the network and is replaced by a non-existing hub. The perturbation operator of ILS is a  
10 consecutive employment of *reversion* and *hub-removal* operators.

11 Indeed, the crossover operator may create infeasible solutions that violate Constraint (3). In this situation,  
12 two cases happen: 1) the number of located hub is less than  $h$  or 2) the number of located hubs is greater than  $h$ .  
13 For the first case, a penalty is added to the objective value of that solution. In the second case, the redundant  
14 hubs are eliminated from the solution and their spokes are allocated to their closest hubs. The redundant hubs are  
15 the hubs with higher fixed cost. Finally, the Pseudo code of the proposed GAMLS algorithm is as Algorithm 2.  
16

---

Algorithm 2. GAMLS algorithm

---

Set parameters:  $Itr_{max}^{GA}$ ,  $P_{size}$ ,  $ER$ ,  $CR$ ,  $MR$ ,  $Itr_{max}^{ILS}$ ,  $k$  //  $k$  is the number of clusters in the  $k$ -means algorithm  
 $\$ = \{ \}$  // Archive of NDP solutions

Generate initial random population of size  $P_{size}$   
Evaluate the objective functions of each solution  
Rank the solutions based on the non-dominance sorting // 1<sup>st</sup> ranked solutions are NDP solutions  
Calculate CD for each solution // CD is calculated among the same-ranking solutions  
 $\$ :=$  Update archive of NDP solutions via 1<sup>st</sup> ranked solutions  
**While**  $Itr_{max}^{GA}$  not reached **Do**  
     $NG = \{ \}$  // NG: Next Generation  
     $NG = NG \cup \{ \text{Best } ER \text{ NDP solutions with high CD from } \$ \}$  // Elitism  
    **For**  $CR/2$  times **Do**  
         $s_1, s_2 :=$  Select two solutions via TS // TS: Tournament Selection  
         $o_1, o_2 :=$  Perform crossover on  $s_1, s_2$  // Creating offspring using crossover operator  
         $NG = NG \cup \{ o_1, o_2 \}$   
    **EndFor**  
    **For**  $CM$  times **Do**  
         $s_3 :=$  Select a solution randomly // Diversifying solutions using mutation operator  
         $s_M :=$  Perform mutation on  $s_3$   
         $NG = NG \cup \{ s_M \}$   
    **EndFor**  
     $\$ :=$  Update archive of NDP solutions  
    Evaluate the objective functions of each solution  
    Rank the solutions based on the non-dominance sorting  
    Calculate CD for each solution  
**EndWhile**

NSGA-II Global Search Algorithm

// 1<sup>st</sup> ranked solutions are NDP solutions  
// CD is calculated among the same-ranking solutions  
// NG: Next Generation  
// Elitism  
// TS: Tournament Selection  
// Creating offspring using crossover operator  
// Diversifying solutions using mutation operator

$[NDP_1, \dots, NDP_k] :=$  Select  $k$  NDP solutions by  $k$ -means

Machine Learning Algorithm

**For**  $s_i$  in  $[NDP_1, \dots, NDP_k]$  **Do**  
     $s_i^* := LocalSearch(s_i)$   
     $s_{best} := s_i^*$   
    **While** stopping criterion not reached **Do**  
         $s_i^t := Perturbation(s_i^*)$   
         $s_i^{*t} := LocalSearch(s_i^t)$   
         $s_i^* := AcceptanceCriterion(s_i^*, s_i^{*t}, s_{best})$   
    **EndWhile**  
     $\$ :=$  Update archive of NDP solutions  
**EndFor**

Iterated Local Search Algorithm

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43

#### 4.4. Time complexity of the proposed GAMLS algorithm

The time complexity of an algorithm is concerned about how fast or slow that algorithm performs and describes the amount of time it takes to run that algorithm. Time complexity is usually calculated by counting the number of elementary operations performed by the algorithm, supposing that each elementary operation takes a fixed amount of time to perform. Therefore, the amount of time taken by an algorithm is the number of elementary operations multiplied by a constant factor (Oliveto and Witt, 2015). Since the execution time of an algorithm may vary among different inputs of the same size, one usually considers the worst-case time complexity, which is the maximum amount of time required for inputs of a given size. The complexity of an algorithm is commonly shown as  $O(N)$ , where  $N$  is the size of the problem.

The complexity of the proposed GAMLS of Algorithm 2 is the sum of the time complexity of three NSGA-II,  $\mathbb{k}$ -Means and ILS algorithms. The time complexity of each part is described as follows:

- NSGA-II: The complexity of the NSGA-II is the sum of different parts as
  - 1) Population sorting of size  $P_{size}$ . Time complexity is  $O(P_{size} \log P_{size})$ ,
  - 2) Fitness function evaluation that depends on the population size ( $P_{size}$ ), number of products ( $P$ ) and number of spokes ( $N$ ). Time complexity is  $O(P_{size} \times PN^2)$ ,
  - 3) Solution generating using crossover that includes both tournament selection (TS) and crossover (XO) operator with rate  $C_R$ . Time complexity is  $O\left(\frac{XO}{C_R \times P_{size}} \times \overbrace{P_{size}}^{TS}\right)$ ,
  - 4) Solution generating using mutation that includes the mutation (MO) operator with rate  $M_R$ . Time complexity is  $O(M_R \times P_{size})$ ,
  - 5) The main loop of the NSGA-II with  $Itr_{max}^{GA}$  number of iterations.Finally, the complexity of the NSGA-II is equal to.
- $\mathbb{k}$ -Means: The complexity of the  $\mathbb{k}$ -Means clustering algorithm in the proposed GAMLS algorithms is equal to  $O(|\mathbb{S}| \times N \times \mathbb{k})$ , where  $|\mathbb{S}|$  is the number of the NDP solutions obtained by the NSGA-II.
- ILS: The complexity of the ILS algorithm depends on the number of the NDP solutions ( $|\mathbb{S}|$ ), local search operators' complexity and the  $Itr_{max}^{ILS}$  number of iterations. Time complexity of the ILS algorithm is equal to  $O(Itr_{max}^{ILS} \times |\mathbb{S}| \times N)$ .

#### 5. Computational experiment

The performance of the proposed GAMLS algorithm is validated through a benchmark with three metaheuristic algorithms in the literature, classical NSGA-II (Deb et al., 2000) and two new recently developed hybrid algorithms that employ machine learning to improve the performance of the metaheuristic algorithms (Sun et al., 2019; Zhang et al., 2016). The classical NSGA-II algorithm is similar to GAMLS without performing  $\mathbb{k}$ -means and ILS algorithms. The aim of this comparison is to discover how hybridizing the NSGA-II with machine learning and ILS algorithm improves the quality of the NDP solutions.

Sun et al. (2019) propose an adaptive MOEA for MOOPs. In their algorithm, a  $\mathbb{k}$ -means clustering method is employed to learn the Pareto optimal set's manifold structure adaptively, in accordance with the regularity property of MOOPs, along the evolution. A new offspring is generated in two ways: either 1) a solution in each cluster is Gaussian-perturbed using the variance-covariance matrix within its cluster, or 2) a solution is generated using differential evolution (DE) operator from two parents and a global solution from different clusters. These two strategies make balance between diversification and intensification. Zhang et al. (2016) propose a self-organizing multi-objective EA (MOEA), wherein a self-organizing mapping (SOM) method with  $(m-1)$  latent variables is applied to establish the neighborhood relationship among current

1 solutions. Their method only allows the mate between neighboring solutions to generate a new solution. To  
2 reduce the computational overhead, the self-organizing training step and the evolution step are conducted in an  
3 alternative manner.

4 Although these algorithms have been proposed for optimization problems with continuous variables and  
5 differentiable objective functions, we adapt their idea and develop two new algorithms: one based on NSGA-II  
6 and  $\mathbb{k}$ -means (Sun et al., 2019) and second one based on NSGA-II and self-organizing map (SOM) method  
7 (Zhang et al., 2016). These algorithms are, respectively, called *HEU1* and *HEU2* hereafter. The  $\mathbb{k}$ -means and  
8 SOM are used to make clusters and the clusters are updated at each iteration of the algorithms if significant  
9 difference happens between the new generated solutions and those of the previous generation; otherwise, the  
10 most recent clusters are used in the new iteration of the algorithm. Once clusters are generated, new solutions are  
11 generated to search the solution space. When generating new solutions using crossover, two intra-cluster and  
12 inter-cluster search mechanisms are adapted. For intra-cluster mechanism, inside each cluster, two solutions are  
13 randomly selected using a binary tournament selection and one-point crossover operator (Section 4.3) is applied  
14 to create two new offspring. For inter-cluster mechanism, three solutions  $s_1$ ,  $s_2$  and  $s_3$  are selected in such a way  
15 that  $s_2$  and  $s_3$  are from the same cluster and  $s_1$ , is the best solution in the population (in terms of dominance and  
16 crowding distance) but not in the same cluster as  $s_2$  and  $s_3$ . Solution  $s_1$  is named as the target solution. Afterward,  
17 solutions  $s_2$  and  $s_3$  are first crossed using one-point crossover (Section 4.3). Then, new created solution from  $s_2$   
18 and  $s_3$  is crossed by the target solution  $s_1$  again using one-point crossover. The idea of inter-cluster mechanism is  
19 not only to keep the similarity of solutions in a cluster but also to diversify the solutions between clusters. Inter-  
20 cluster mechanism is applied as many as the number of solutions in the population. For doing mutation, at each  
21 iteration, a solution is randomly selected from each cluster and reversion operator (Section 4.3) is employed.

22 All algorithms are compiled in Python 3 programming language executed on a Pentium 8 CPU with 3.4  
23 GHz processor and 32 GB of RAM.

## 24 5.1. Design of experiments

25 This section design a set of 20 instances with three size categories to compare the performance of the  
26 proposed GAMLs algorithm with those of NSGA-II, HEU1 and HEU2. The three categories are created based  
27 on the computational time required to find the optimal Pareto solutions. Categories A and B consist of the  
28 instances that obtaining optimal Pareto solutions for them takes [82s, 2h] and [3h, 5h], respectively. Finally,  
29 category C includes the instances that cannot be solved to optimality in reasonable time (i.e., the augmented  $\epsilon$ -  
30 constraint method cannot find even a single Pareto optimal solution in a reasonable time). Table 2 shows the  
31 property of the instances in terms of number of products  $P$ , number of spokes  $N$ , and number of hubs  $h$ . It is  
32 worth mentioning that the generation of these data has been inspired from classical instances in hub-and-spoke  
33 network design problem (i.e., CAB, AP, USA423, Turkish network). Actually, this inspiration has been only  
34 to get the idea on the range of input parameters such as flow and cost. In addition, the transportation  
35 time between each pair of nodes has been calculated based on the distance between cities. On the other  
36 hand, since the CAB, AP and other data sets do not contain other parameter that our model requires,  
37 these parameters have been generated randomly but based on real assumption. For example, hubs with  
38 higher fixed cost have higher performance when serving the arrival flow. Therefore, the CAB, AP and  
39 other data sets could not be used directly in the proposed model. Finally, we consider that  $\alpha_C = 0.75$ ,  $\alpha_T =$   
40  $0.70$ ,  $\beta_{kl}=1$  and  $\zeta_{kl}=2$ .

1 It is obvious the performance of an algorithm is significantly affected by the value of its parameters.  
2 Higher performance of each algorithm can be obtained by appropriate tuning of its parameters using response  
3 surface methodology (RSM) (Azizmohammadi et al., 2013; Mohammadi et al., 2014). RSM is defined as a  
4 collection of mathematical and statistical method-based experiments, which can be used to optimize processes.  
5 Regression equation analysis is used to evaluate the response surface model. To tune the parameters, two levels  
6 for each parameter are considered. Each factor is measured at two levels, which can be coded as  $-1$  when the  
7 factor is at its low level ( $\mathbb{L}$ ) and  $+1$  when the factor is at its high level ( $\mathbb{H}$ ). The coded variable can be then  
8 defined as  $x_i = (r_i + 0.5(\mathbb{h} + \mathbb{l}))/0.5(\mathbb{h} - \mathbb{l})$ , where  $x_i$  and  $r_i$  are coded as variable and real variable,  
9 respectively.  $\mathbb{h}$  and  $\mathbb{l}$  represent high level and low level of the factor. Using RSM, tuned parameters of the  
10 GAMLs, NSGA-II, HEU1 and HEU2 algorithms are considered as follows:

11 Table 2. Problem instances to evaluate the performance of GAMLs algorithm

Category	Instance	Parameters		$h$	$w^*$ (unit/h)		$c^*$	$\mu^*$ (unit/h)	
		$P$	$N$						
					mean	SCV		mean	SCV
Cat. A	I1	2	6	2	[10,30]	[1,5]	2	[40,80]	[1,5]
	I2	2	8	3	[30,50]	[4,8]	2	[60,100]	[4,8]
	I3	3	8	2	[50, 80]	[6,10]	3	[100,180]	[6,10]
	I4	3	10	4	[80,120]	[8,12]	3	[150,250]	[8,12]
	I5	4	10	3	[10,30]	[1,5]	2	[100,180]	[1,5]
	I6	5	10	4	[30,50]	[4,8]	3	[60,120]	[4,8]
Cat. B	I7	5	12	4	[50, 80]	[6,10]	3	[200,240]	[6,10]
	I8	6	12	5	[80,120]	[8,12]	4	[220,300]	[8,12]
	I9	6	14	3	[10,30]	[1,5]	4	[180,260]	[1,5]
	I10	7	14	5	[30,50]	[4,8]	4	[200,300]	[4,8]
	I11	7	16	4	[50, 80]	[6,10]	5	[260,340]	[6,10]
	I12	7	18	6	[80,120]	[8,12]	5	[280,380]	[8,12]
	I13	8	22	6	[10,30]	[1,5]	5	[200,300]	[1,5]
Cat. C	I14	8	28	6	[30,50]	[4,8]	6	[300,380]	[4,8]
	I15	8	36	8	[50, 80]	[6,10]	6	[340,420]	[6,10]
	I16	9	40	8	[80,120]	[8,12]	8	[400,500]	[8,12]
	I17	9	60	10	[10,30]	[1,5]	8	[200,320]	[1,5]
	I18	10	100	12	[30,50]	[4,8]	10	[280,380]	[4,8]
	I19	12	150	16	[50, 80]	[6,10]	12	[400,560]	[6,10]
	I20	15	200	20	[80,120]	[8,12]	12	[440,600]	[8,12]

\*  $w$ ,  $c$  and  $\mu$  stand for the flow of products, number of servers and service rate in the hubs, respectively

- 13
- 14 • **GAMLs**:  $P_{size} = 100$ ;  $Itr_{max}^{GA} = 100$ ; Selection operator = “Binary tournament selection” (Mohammadi et  
15 al., 2015); Crossover & Mutation operators = see Section 4.5.2;  $E_R = 10\%$ ;  $C_R = 75\%$ ;  $M_R = 15\%$ ;  $\mathbb{k} = 10$ ;  
16  $Itr_{max}^{LS} = 80$ ; Local search & Perturbation operations = see Section 4.5.2; Acceptance criterion = 5-10% of  
17 gap.
  - 18 • **NSGA-II**:  $P_{size} = 200$ ;  $Itr_{max}^{GA} = 200$ ; Selection operator = “Binary tournament selection”; Crossover &  
19 Mutation operators = see Section 4.5.2;  $E_R = 10\%$ ;  $C_R = 75\%$ ;  $M_R = 15\%$ .
  - 20 • **HEU1**:  $P_{size} = 120$ ;  $Itr_{max}^{ALG1} = 150$ ;  $\mathbb{k} = 10$ ; Selection operator = based on clusters from  $\mathbb{k}$ -means;  
21 Mutation & Crossover operators = see beginning of Section 5.
  - 22 • **HEU2**:  $P_{size} = 120$ ;  $Itr_{max}^{ALG2} = 150$ ;  $\mathbb{k} = 10$ ; Selection operator = based on clusters from SOM; Mutation &  
23 Crossover operators = see beginning of Section 5.

24

## 25 5.2. Numerical results

26 This section compares the performance of the proposed GAMLs algorithm with NSGA-II, HEU1 and  
27 HEU2 algorithms through generated instances of Table 2. The performance of the algorithms in terms of  
28 obtaining (near)-optimal NDP solutions is compared with optimal NDP solutions obtained, over instances of Cat.

1 A and some of instances of Cat. B, by a well-known augmented  $\epsilon$ -constraint method (Mohammadi et al., 2019a)  
 2 programmed using Cplex solver.

3 The comparison is carried out based on five common metrics used to compare the multi-objective  
 4 algorithms as (Deb et al., 2000):

- 5 • *Convergence Metric (CM)* that evaluates the tightness between the NDP solutions obtained by the  
 6 algorithms and the optimal NDP solutions (for small- and some of medium-sized instances),
- 7 • *Quality Metric (QM)* that accounts for the number of NDP solutions obtained by an algorithm in comparison  
 8 to other benchmarking algorithms,
- 9 • *Divergence Metric (DM)* that reports how large the non-dominated frontier of an algorithm is,
- 10 • *Spacing Metric (SM)* that shows that how the NDP solutions have dispersed across the non-dominated  
 11 frontier, and finally
- 12 • *Mean Ideal Distance (MID)* metric that shows how closed the NDP solutions are from the ideal point  
 13  $(f_1^{min}, f_2^{min})$ , where  $f_1^{min}$  and  $f_2^{min}$  are the minimum value of the first and the second objective functions,  
 14 respectively.

15 They represent both quantitative and qualitative comparison metric and directly calculated from the NDP  
 16 solutions. An algorithm with higher values of QM and DM and lower values of CM, SM and MID is a better  
 17 algorithm.

18 Tables 3 and 4 report the comparison between the metaheuristic algorithms and the ‘Optimal’ solutions  
 19 obtained by the augmented  $\epsilon$ -constraint method. The reported values for CM, DM, SM and MID metrics are the  
 20 mean values for problem instances with different sizes. The user CPU-time contains min and max ([min, max])  
 21 computational time for solving corresponding instances. Detail results can be found in Appendix A.

22 According to the mean value of CM in Table 3, it is concluded that the mean tightness of the proposed  
 23 GAMLs algorithm comparing to the optimal solutions is equal to 1.03% (3% of gap) and 1.11% (11% of gap)  
 24 for the instances of Cat. A and Cat. B, respectively. For instances of Cat. A and Cat. B, the effectiveness (quality  
 25 of solutions) of the proposed GAMLs algorithm can be shown in terms of low CM, and its efficiency can be  
 26 demonstrated by low CPU-time comparing to Cplex solver and other algorithms. Considering the CM for other  
 27 algorithms from Tables 3 and 4, it is observed that the mean gaps of NSGA-II, HEU1, HEU2 in comparison with  
 28 the optimal results are, respectively, equal to 11%, 7% and 9% for instances of Cat. A. These values are 32%,  
 29 23% and 27% for instances of Cat. B.

30  
 31 Table 3. Comparison with Classical Metaheuristics

Metric	Instance & Algorithm									
	Cat. A			Cat. B			Cat. C			
	Optimal	GAMLs	NSGA-II	Optimal	GAMLs	NSGA-II	Optimal	GAMLs	NSGA-II	
CM	-	1.03	1.11	-	1.11	1.32	-	-	-	
QM	-	0.76	0.00	-	0.99	0.00	-	1.00	0.00	
DM	1.27	1,39	1,14	1.17	1,43	1,01	-	1,47	1,07	
SM	0.63	0.37	0.53	0.70	0.32	0.53	-	0.36	0.63	
MID	0.71	0.62	1.04	0.66	0.46	0.91	-	0.60	0.97	
Time (s)	[124,8976]	[92,177]	[129,219]	[12587,-]	[190,312]	[231,452]	-	[338,752]	[537,1112]	

32  
 33 Table 4. Comparison with Learning-based Metaheuristics

Metric	Instance & Algorithm											
	Cat. A				Cat. B				Cat. C			
	Optimal	GAMLs	HEU1	HEU2	Optimal	GAMLs	HEU1	HEU2	Optimal	GAMLs	HEU1	HEU2
CM	-	1.03	1.07	1.09	-	1.11	1.23	1.27	-	-	-	-

QM	-	0.76	0.34	0.00	-	0.99	0.01	0.00	-	1.00	0.00	0.00
DM	1.27	1.39	1.26	1.22	1.17	1.43	1.30	1.25	-	1.47	1.27	1.18
SM	0.63	0.37	0.37	0.41	0.70	0.32	0.37	0.47	-	0.36	0.36	0.44
MID	0.71	0.62	0.82	0.85	0.66	0.46	0.81	0.94	-	0.60	0.79	0.84
Time (s)	[124, 976]	[92, 177]	[112, 192]	[121, 210]	[12587, -]	[190, 312]	[214, 386]	[227, 439]	-	[338, 752]	[419, 911]	[512, 994]

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

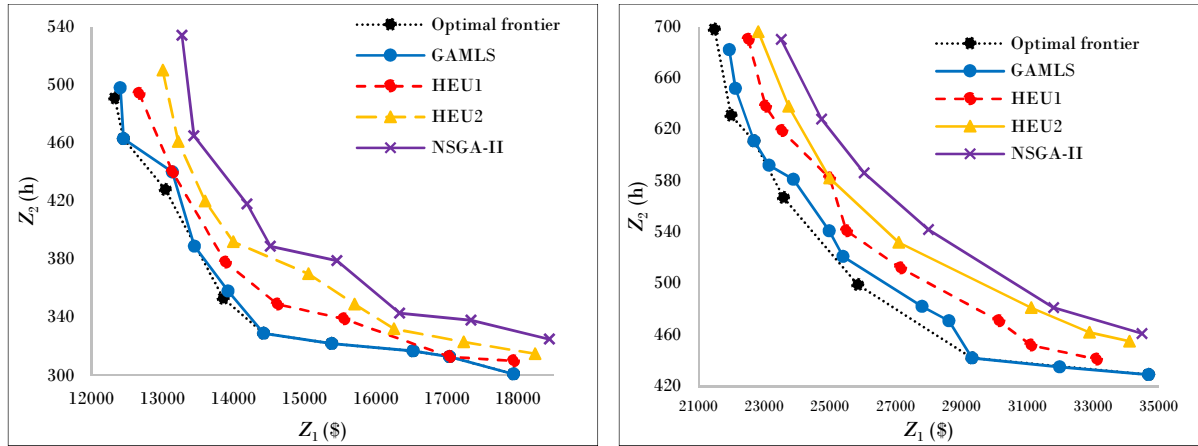
30

Although HEU1 and HEU2 use machine learning methods to enhance their performance, the proposed GAMLS algorithm outperforms these algorithms thank to high performance of ILS algorithm in exploiting the neighborhood of NDP solutions. On the other hand, it is also discovered that HEU1 and HEU2 outperforms classical NSGA-II in almost all instances due to their higher performance thank to clustering method and specific search mechanism that keeps a good balance between diversification and intensification. For the values of QM of each algorithm at each category of the problem, please look at both Tables 3 and 4 at the same time since the total share between all four algorithm becomes 100%.

Considering instances of Cat. A and Cat. B, it is observed that the computational time for obtaining optimal NDP by Cplex solver exponentially increases from 124 to 12587 seconds when the size of the instances slightly increases. However, the proposed GAMLS obtains high quality NDP solutions less than 752 seconds even for large-enough instances of Cat. C. The results of Tables 3 and 4 particularly show the superior performance of the proposed GAMLS algorithm in comparison with NSGA-II, HEU1 and HEU2. The outperformance of the proposed GAMLS is supported by lower values of the CM, SM, and MID metric and higher values of DM.

The quality of the proposed GAMLS is further demonstrated by QM, where the NDP solutions of the GAMLS algorithm partially or even completely dominate the NDP solutions obtained by other algorithms in almost all instances. Comparing the performance of the GAMLS algorithm and the NSGA-II reveals that hybridizing the classical NSGA-II with a learning-based ILS significantly improves the performance of the search algorithm in terms of both diversification and intensification.

As another experiment, the learning part of the proposed GAMLS algorithm was removed. Removing the learning part implies that the ILS algorithm be applied on the all the NDP solutions obtained by the NSGA-II. After solving some of the large-sized instances, up to 100 NDP solutions are obtained by NSGA-II. Therefore, the ILS algorithm should be executed 100 times. However, in the proposed GAMLS algorithm, the ILS algorithm is executed only 10 times. On the other hand, we also investigated that removing the learning part does not highly affect the quality of the final non-dominated solutions. Although, higher number of non-dominated solutions were obtained but the majority of the solutions had been already obtained via the GAMLS algorithm. This explanation was added to the revised manuscript without adding the results since they do not provide useful information.



a) Instance I4

b) Instance I8

Figure 5. Optimal vs. metaheuristics: Pareto frontier

1  
2

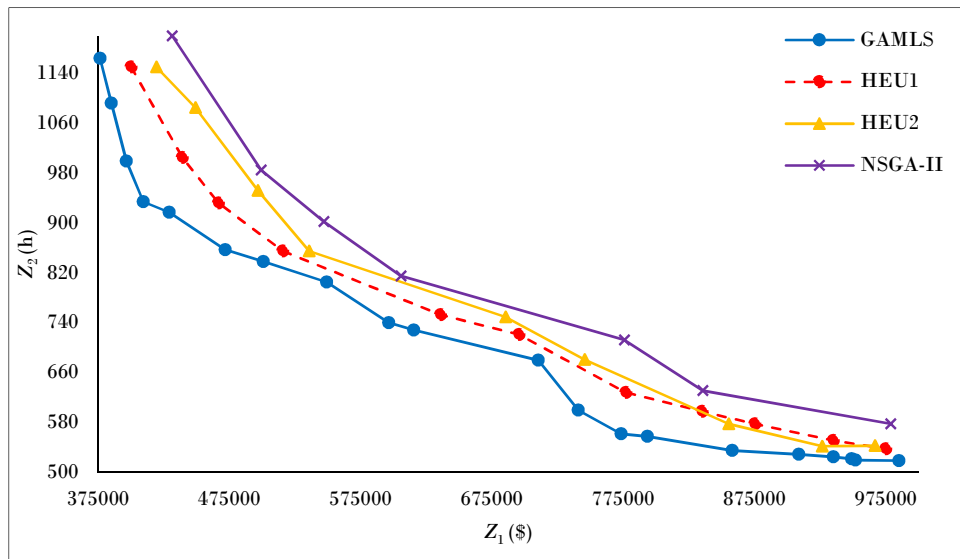


Figure 6. GAMLs vs. NSGA-II, HEU1 & HEU2: Pareto frontier of instance I20

3  
4  
5

6 In order to visually show the performance of the proposed GAMLs algorithm, Figures 5a and 5b  
7 illustrate the Pareto frontier of the metaheuristic algorithms, respectively for the instances I4 and I8 in  
8 comparison with the optimal Pareto frontier obtained by the augmented  $\epsilon$ -constraint method. In addition, Figure  
9 6 compares the Pareto frontier obtained by the proposed GAMLs and other metaheuristic algorithms for the  
10 instance I20. Indeed, we investigated that for the solutions with higher cost, the hubs with higher service level  
11 have been located that leads to the lower waiting time in the hubs and consequently lower total transportation  
12 time between each pair of OD nodes.

### 13 6. Sensitivity Analysis

14 In this section, the aim is to analyze the sensitivity of the proposed model regarding to the input  
15 parameters as well as some other non-parametric factors. The input parameters include flow of products,  
16 transportation cost, free-flow transportation time, service rate and number of servers at hubs. For the non-  
17 parametric factors, we consider two main sources of variation that affect the performance of the hub network and  
18 even the structure of the network. These are 1) Stochastic disruption of the hubs, wherein the hubs are subject to  
19 random failures and these failures directly affect the performance of the hub during the process of the products;  
20 and 2) Product prioritization, wherein the products are processed based on their priority and the products with



1 lower priority (i.e., products with lower importance) should wait until the products with higher priority are  
2 processed.

3 Another important issue in any sensitivity analysis, which has been mostly ignored in the literature, is that  
4 analyzing the impact of parameters (or non-parametric factor) only on the objective function value could be  
5 misleading. Actually, the uncertainty in a parameter is more important if the variation of that parameter affects  
6 not only the objective function value, but also the structure of the solution (e.g., the hub-and-spoke network  
7 structure in this paper). These types of parameters/factors would be more important to be controlled by the  
8 decision makers comparing to those that only affect the objective function values. This sensitivity is then called  
9 ‘*model sensitivity*’. Accordingly, this section investigates both *objective sensitivity* and *model sensitivity* of the  
10 input parameters and non-parametric factors. In the following, these sensitivity analyses are performed  
11 separately over instance I10.

### 12 13 14 15 **6.1. Sensitivity to the input parameters**

16 In this section the aim is to investigate the *objective* and *model* sensitivities to the variation of the  
17 transportation cost ( $c_{ij}^p$ ), the free-flow transportation time ( $t_{ij}^p$ ), the products’ flow ( $w_{ij}^p$ ) and the service rate at  
18 hubs ( $\mu_k^p$ ). The sensitivity results are shown based on changes according to the basis scenario (i.e., instance I10).  
19 For quantifying the objective sensitivity, the percentage of changes is simply calculated by comparing the  
20 objective function values. To quantify the model sensitivity, we calculate a ratio as the number of elements  
21 changed in the solution after the parameter variation over the total number of changeable elements in the  
22 solution. The elements that construct a solution contains  $h$  number of hubs and  $|N|\times|P|$  number of connection  
23 links from spokes to the hubs. Therefore, the total number of elements in a hub network that may change due to  
24 parameter variations is  $(|N|\times|P|)+h$ . Table 5 shows the objective and model sensitivities to the variation of the  
25 transportation cost and free-flow transportation time. In addition, Table 6 shows the sensitivities to the products’  
26 flow and hub’s service rate. In Tables 5 and 6, column “Sensitivity (%)” represents the mean increase imposed to  
27 the corresponding parameters. Negative values in these tables show the reduction in the corresponding values.

28 From Table 5, it can be observed (as expected) that transportation cost mostly imposes objective  
29 sensitivity over first objective function  $Z_1$  that is the sum of the total cost. More interesting, variation of the  
30 transportation does not highly affects the model structure such that in the worst case (i.e., 100% increase in the  
31 cost), only 8% of the network is changed. In addition, cost increase slightly affects the second objective function  
32  $Z_2$  that is the maximum transportation time between each pair of OD spokes. It can be also seen that increasing  
33 the cost up to 15% does not affect the second objective as well as the network structure. It means that the  
34 obtained solutions are robust to the variation of the cost up to 15%. Looking at the changes in the free-flow  
35 transportation time, inverse results observed comparing to the transportation cost. Changes in the free-flow  
36 transportation time highly affects the structure of the network (i.e., high model sensitivity) but has less effect on  
37 the first objective function.

38 It can be then concluded that the second objective function is highly influenced by the structure of the  
39 network. Accordingly, companies whose most important objective is the transportation time should be more  
40 careful about the variation of those parameters/factors that affect the network structure.

1

Table 5. Objective & Model sensitivities to input parameters – Part I

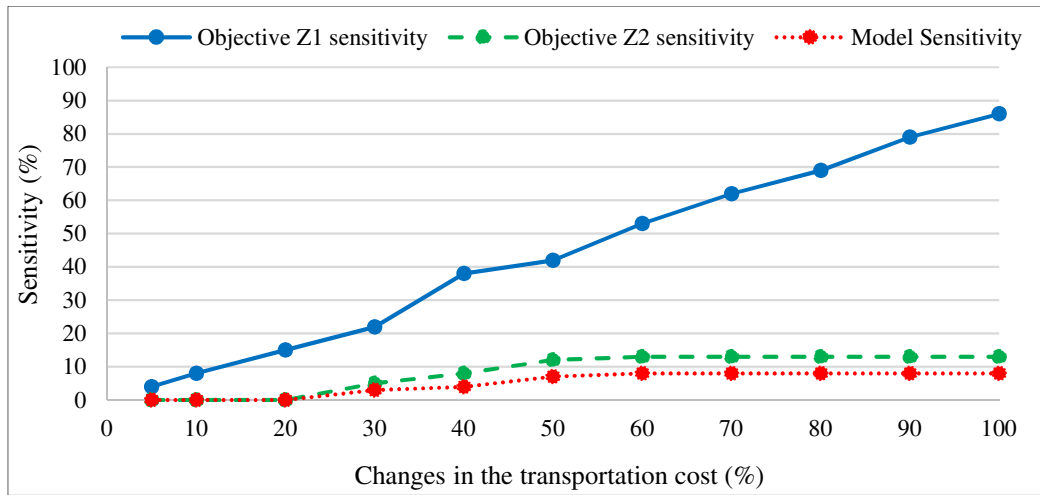
Parameter	Change (%)	Sensitivity (%)			Parameter	% of change	Sensitivity (%)		
		Objective increase		Model			Objective increase		Model
		Z <sub>1</sub>	Z <sub>2</sub>				Z <sub>1</sub>	Z <sub>2</sub>	
$c_{ij}^p$	5	4	0	0	$t_{ij}^p$	5	4	3	2
	10	8	0	0		10	5	9	5
	20	15	0	0		20	8	11	14
	30	22	5	3		30	11	12	19
	40	38	8	4		40	11	16	19
	50	42	12	7		50	19	19	22
	60	53	13	8		60	19	21	22
	70	62	13	8		70	19	23	22
	80	69	13	8		80	23	24	27
	90	79	13	8		90	25	25	29
100	86	13	8	100	29	25	29		

2

3

Figures 7 and 8 illustrate how the sensitivity in transportation cost and transportation time affect the objective functions Z<sub>1</sub> and Z<sub>2</sub> as well as the model structure. Comparing Figures 7 and 8 it can be figured out that transportation time has more impact on the objective and model sensitivities.

6



7

8

9

Figure 7. Impact of the transportation cost increase on the objective and model sensitivities

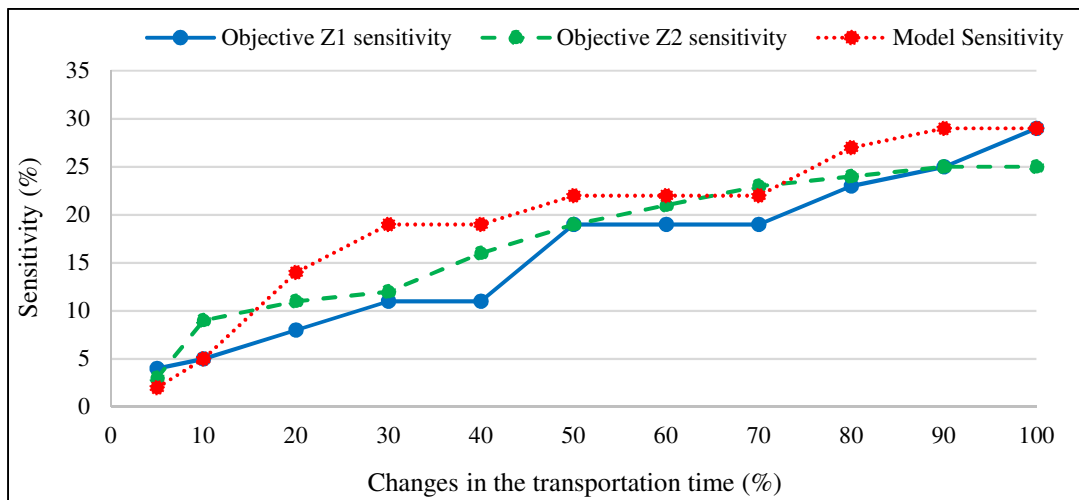


Figure 8. Impact of the transportation time increase on the objective and model sensitivities

10

11

12

From Table 6, more objective and model sensitivities are observed comparing to Table 5. Even small increases in the products' flow highly affects both objective functions as well as the network structure. Once the quantity of the flow increases, the total transportation cost increases somehow linearly. However, the second objective function increases polynomially by the increase of flow. Such significant increase in the second objective function directly relates to the increased congestion of flow in the hubs and the connection links. On the other hand, when the flow increases, the structure of the network is affected significantly. It can be interpreted in such a way that the network tries to adapt itself to the changes in order to absorb these changes as high as possible.

On the other hand, high sensitivities are discovered to the hubs service rate. When the service rate increases, products are processed with higher rate and consequently the waiting time of the products in the hubs is decreased. By accelerating the service rate up to 100%, the transportation time decreases up to 50%. With a simple comparison between the effects of transportation time ( $t_{ij}^p$ ) and service time ( $\mu_k^p$ ) variations, it is seen that service rate has higher effect on the second objective function and it means that the most of the total transportation time is spent in the hubs and consequently the congestion in the hubs are very important to be taken into account in the design of hub-and-spoke network.

Table 6. Objective & Model sensitivities to input parameters – Part II

Parameter	Change (%)	Sensitivity (%)			Parameter	% of change	Sensitivity (%)		
		Objective increase		Model			Objective increase		Model
		Z <sub>1</sub>	Z <sub>2</sub>				Z <sub>1</sub>	Z <sub>2</sub>	
$w_{ij}^p$	5	3	4	3	5	0	-2	2	
	10	9	11	7	10	0	-4	5	
	20	18	19	12	20	2	-10	7	
	30	27	24	18	30	3	-12	11	
	40	37	32	23	40	-5	-18	15	
	50	47	42	25	50	-11	-22	19	
	60	56	76	31	60	-11	-29	22	
	70	68	105	39	70	-15	-35	24	
	80	72	249	42	80	-15	-39	27	
	90	83	387	45	90	-15	-42	32	
100	95	551	45	100	-15	-47	35		
$\mu_k^p$	5	3	4	3	5	0	-2	2	
	10	9	11	7	10	0	-4	5	
	20	18	19	12	20	2	-10	7	
	30	27	24	18	30	3	-12	11	
	40	37	32	23	40	-5	-18	15	
	50	47	42	25	50	-11	-22	19	
	60	56	76	31	60	-11	-29	22	
	70	68	105	39	70	-15	-35	24	
	80	72	249	42	80	-15	-39	27	
	90	83	387	45	90	-15	-42	32	
100	95	551	45	100	-15	-47	35		

Figures 9 and 10 illustrate how the sensitivity in transportation cost and transportation time affect the objective functions Z<sub>1</sub> and Z<sub>2</sub> as well as the model structure. Comparing Figures 9 and 10 it can be figured out that transportation time has more impact on the objective and model sensitivities.

These sensitivity analyses figure out that which input parameters are more important to be controlled since their variation in the real setting is unavoidable. Therefore, managers should have higher control on and better estimation of these parameters. In this study, products' flow seems to be the most important parameters that affects the robustness of the solution. On the other hand, service rate of the hubs plays an important role to decrease the second objective function.

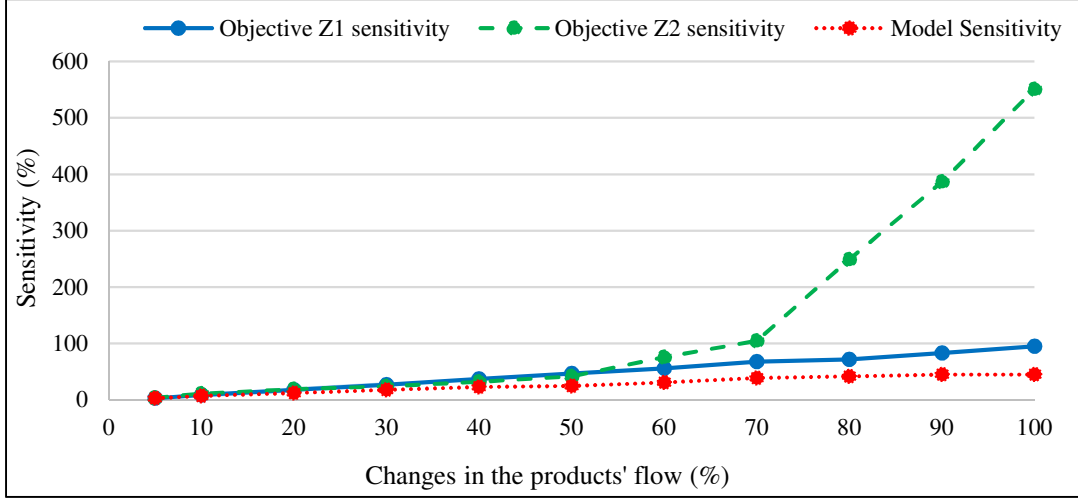


Figure 9. Impact of the products' flow increase on the objective and model sensitivities

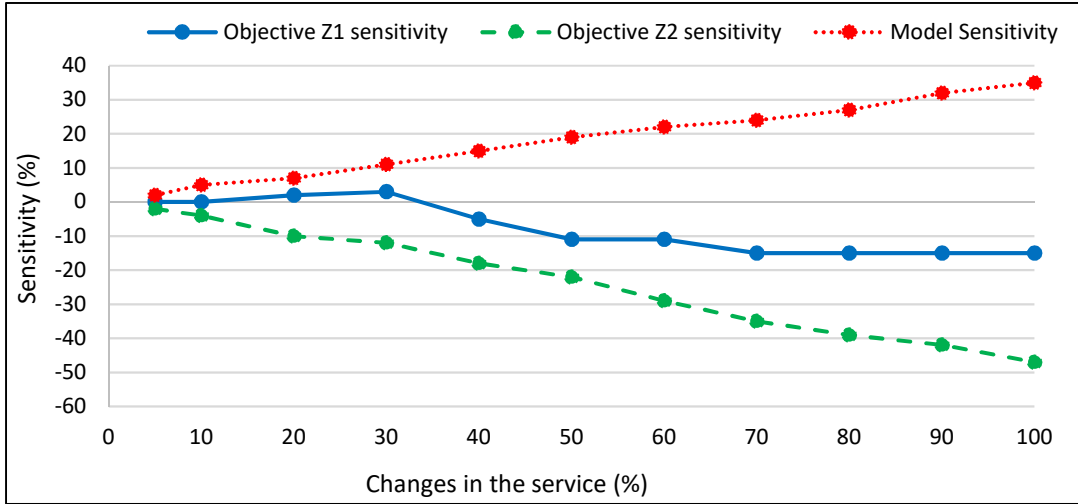


Figure 10. Impact of the products' flow increase on the objective and model sensitivities

## 6.2. Sensitivity to the stochastic disruption at hubs

We consider that hubs are subject to random failures (Mohammadi and Tavakkoli-Moghaddam, 2016). Accordingly, hub  $k$  stochastically fails with rate  $f_k$  and consequently the inter-arrival time of failures in hub  $k$  is exponentially distributed with mean  $1/f_k$ . Once a failure happens, the hub becomes completely unavailable and is stochastically retrieved with rate  $r_k$ . Accordingly, the time until the hub is retrieved is generally distributed with mean  $1/r_k$ , standard deviation  $\sigma_{R,k}$  and squared coefficient of variation (SCV) of  $c_{R,k}^2$  (i.e.,  $c_{R,k}^2 = r_k \sigma_{R,k}$ ). Therefore, the mean availability of hub  $k$  is  $AV_k = \frac{r_k}{r_k + f_k}$  (Morrison and Martin, 2007, Mohammadi et al., 2019b). In addition, the new service rate ( $\mu_k^{p*}$ ) and the new SCV of the service time ( $c_{S,pk}^{2*}$ ) of the hub  $k$  are updated as Equations (35) and (36). Equations (12) to (19) are modified accordingly (if necessary).

$$\mu_k^{p*} \equiv AV_k \mu_k^p = \frac{r_k \mu_k^p}{r_k + f_k} \quad (35)$$

$$c_{S,pk}^{2*} = c_{S,pk}^2 + \left(1 + c_{S,pk}^2\right) \left(1 - \frac{r_k}{r_k + f_k}\right) \frac{\mu_k^p}{(r_k + f_k)} \quad (36)$$

16

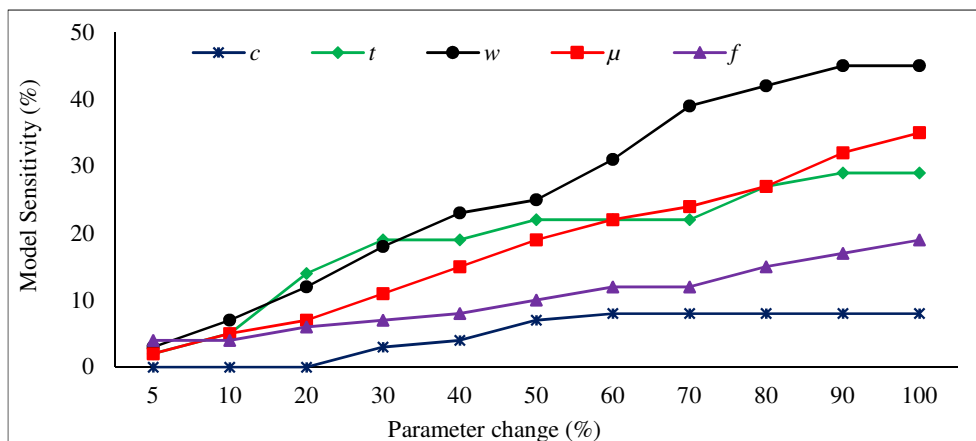
1 With the same procedure as Section 6.2, the objective and the model sensitivities to the failure rate  $f_k$  and  
 2 retrieve rate  $r_k$  are quantified and presented in Table 7. In this experiment,  $f_k = 0.2$  and  $r_k = 1$ .

3 Based on Equation (35), it is concluded that disruption of the hub services directly affects the hub's  
 4 service rate that its impact was investigated in Table 6. In Table 7, when the failure rate increases, hubs become  
 5 less available and consequently the service rate decreases. This decrease leads to the changes in the network  
 6 structure and consequently the increase of the second objective function. In the other word, once a hub is  
 7 disrupted, the process of the products is interrupted or done with lower speed; then the congestion is augmented  
 8 in the hubs. This augmented congestion also increases the waiting time and the value of the second objective  
 9 function.

10 On the other hand, increase of the retrieve rate has inverse effect on the augmented congestion. Once the  
 11 retrieve rate increases, the hub comes back to its designed processing condition and consequently the network  
 12 structure is not exposed to any changes. Accordingly, managers should take the disruption of the hubs into  
 13 account as an important non-parametric factor that affects the robustness of the network and try to control the  
 14 failure rate and/or increase the retrieve rate of the hubs. Figure 11 summarizes the effect of input parameters in  
 15 the model sensitivity and it can be figured out that through small or big variation, which parameters jeopardize  
 16 the robustness of the solutions.

17 Table 7. Objective & Model sensitivities to network disruption

Parameter	Change (%)	Sensitivity (%)			Parameter	% of change	Sensitivity (%)		
		Objective increase		Model			Objective increase		Model
		Z <sub>1</sub>	Z <sub>2</sub>				Z <sub>1</sub>	Z <sub>2</sub>	
$f_k$	5	0	5	4	5	11	15	12	
	10	0	6	4	10	10	14	12	
	20	2	8	6	20	9	12	10	
	30	4	9	7	30	9	12	9	
	40	4	11	8	40	8	12	9	
	50	5	13	10	50	8	8	7	
	60	7	14	12	60	8	8	6	
	70	7	17	12	70	5	5	6	
	80	9	18	15	80	3	4	5	
	90	10	20	17	90	0	3	4	
	100	11	22	19	100	0	3	2	



18 Figure 11. Model sensitivity vs. Parameter changes

19  
 20  
 21  
 22 **6.3. Sensitivity to the products' prioritization**

23 In order to keep competitiveness and the customer satisfaction with significant variety and high  
 24 variability of their demands, hub-and-spoke system providers usually require to serve the shipment orders

1 differently, and thus products need to have different priorities. Priorities is usually divided into hot, rush and  
 2 normal levels (Lee et al., 2008). A product with a higher priority spends shorter waiting time at hubs and should  
 3 be processed before the lower priority products. In contrary, the products with lower priority need to wait until  
 4 higher priority products finish their service and the servers become available.

5 In this paper, we consider the products with multiple priorities in the hub-and-spoke network and figure  
 6 out the impact product's prioritization on the waiting time of products and particularly the second objective  
 7 function. For this aim, a priority queuing system is developed for the hub network. All products keep their  
 8 priority on all visiting hubs. In such a priority queueing system, we assume that an arriving product belongs to a  
 9 priority class  $r$  ( $r = 1, 2, \dots, R$ ). The next product to be served is the customer with the highest priority  $r$  (e.g., a  
 10 product with priority of  $r$  has higher priority than a product with priority  $r'$  where  $r' > r$ ). The products with the  
 11 same priority are served upon their arrival to the hub. In addition, no preemption is allowed in the queuing  
 12 system (i.e., the process of a product already in service is not preempted by an arriving product with higher  
 13 priority). The mean waiting time for an arriving product  $p$  with priority level  $r$  at each hub consists of three  
 14 components as (Bolch et al., 2006): 1) the mean remaining service time at hub  $k$ ,  $\bar{\tau}_k$ , of the products in service (if  
 15 any), 2) the mean service time of products in the queue with the same or higher priority as the tagged product,  
 16 and 3) the mean service time of products with higher priority that arrive at the queue while the tagged product is  
 17 in the queue and are served before it.

18 For simplicity, we consider that each product has a unique priority and several products do not have the  
 19 same priority (Zahiri et al., 2014; Vahdani and Mohammadi, 2015). Therefore, having  $P$  products in the system  
 20 implies that we have also totally  $P$  classes of priority (i.e.,  $R = P$ ). We also consider that products are numbered  
 21 based on their priority. For example, the product with the highest priority is  $p = 1$ , the product with the second  
 22 highest priority is  $p = 2$ , and so on for other products. The mean waiting time of product  $p$  (with priority  $p$ ) in  
 23 hub  $k$ ,  $W_k^p$ , can be then calculated as Equation (37).

$$W_k^p = \frac{\bar{\tau}_k}{(1 - \delta_k^p)(1 - \delta_k^{p+1})} \quad \forall k, p \quad (37)$$

24 where  $\delta_k^p$  and  $\bar{\tau}_k$  are calculated as Equations (38) and (39).

$$\delta_k^p = \sum_{i=p}^P \rho_k^i \quad \forall k, p \quad (38)$$

$$\bar{\tau}_k \approx \frac{\phi_k^p}{2c_k} \sum_{i=1}^P \rho_k^i \frac{c_{A,ik}^2 + c_{S,ik}^2}{\mu_k^p} G_k^p \quad \forall k \quad (39)$$

25  
 26 This experiment is done on instance I4 containing three products and let's name them as  $P_1$ ,  $P_2$  and  $P_3$ .  
 27 We consider that  $P_1$  has more priority than  $P_2$  and  $P_2$  has more priority than  $P_3$  and consequently  $P_1$  has higher  
 28 priority comparing to  $P_3$ . In Table 8, we investigate the impact of product prioritization on the product's waiting  
 29 time at hubs. For this aim, we increases the flow of products simultaneously to see how the increase of flow of  
 30 the products with high priority affects the waiting time of the products with less priority.

31 It can be seen from Table 8 that product prioritization in parallel with the flow increase catastrophically  
 32 affects the second objective function and particularly the waiting time  $P_2$  and  $P_3$ . The flow increase of 0% means  
 33 that the flow of products is kept as the original generated flow of instance I4 and only the prioritization of the  
 34 product is imposed to the model that leads to decrease in the waiting time of product  $P_1$  but increase in the

1 waiting time of products  $P_2$  and  $P_3$ . These increase and decrease are proportional to the mean waiting time of  
 2 each products before prioritization. In case of 5%, the same situation happens as the case of 0% but the value of  
 3 increases are less than the case of 0% because the flow has been increased and this increase itself leads to higher  
 4 congestion in the hubs and consequently higher mean waiting time in the hubs.

5 By increasing the flow of  $P_1$  to 80%, the processing of  $P_3$  is blocked since there would be always a unit of  
 6  $P_1$  to be processed in a hub. This event happens for  $P_2$  when the flow of  $P_1$  increases up to 100%. On the other  
 7 hand, the second objective function that attempts to minimize the maximum transportation time between each  
 8 pair of OD spokes, increases exponentially even to infinity because of the infinite waiting time of  $P_3$ . An  
 9 interesting result is that prioritizing the products lead to decrease of the waiting time of  $P_1$  since it does not need  
 10 to wait in the hubs in presence of other products. However, it should be noted that prioritizing in parallel with the  
 11 products' flow augmentation significantly decreases the performance of the hub-and-spoke network in terms of  
 12 delivery time.

13  
 14  
 15 **Table 8. Objective & Model sensitivities to flow increase with product's priority**

Flow increase (%)	Waiting time increase (%)			Sensitivity (%)		Model
	$P_1$	$P_2$	$P_3$	Objective increase		
				$Z_1$	$Z_2$	
0	-20	10	16	3	8	4
5	-17	13	19	6	11	7
10	-10	20	29	11	23	11
20	-3	28	75	14	31	19
30	1	41	142	20	84	21
40	9	89	458	28	324	49
50	14	148	1258	32	895	63
60	28	215	8475	41	6794	74
70	32	428	92187	41	85974	74
80	38	782	Inf	41	Inf	74
90	45	6587	Inf	41	Inf	74
100	49	Inf	Inf	41	Inf	74

16  
 17 **7. Conclusion**

18 In a hub-and-spoke network, commodities from different origin spokes are consolidated at hub facilities  
 19 prior to be routed to an intermediate hub or to be delivered to their final destinations. The aggregation of  
 20 commodities in the hub facilities allows the exploitation of scale economies due to the utilization of more  
 21 efficient carriers with higher capacities on hub-to-hub connection links. Despite all advantages of economic  
 22 scales, this exploitation may lead to commodity overload in a small number of hubs, or even result in heavy-  
 23 utilization of some hub-to-hub connections. This congestion becomes more critical for transportation companies  
 24 that employs the hub-and-spoke network for shipment delivery.

25 This paper addresses the single allocation multi-commodity hub-and-spoke network design under hub  
 26 congestion. The network is modeled through a bi-objective non-linear mixed integer programming model with  
 27 congestion in both hubs and hub-to-hub connection links. The proposed bi-objective model minimizes: 1) the  
 28 total transportation cost and 2) the maximum transportation time between each pair of spokes. A novel  
 29 aggregation model was developed based on a general GI/G/c queuing system to evaluate the congestion of the  
 30 flow in the hubs. In addition, a stochastic traffic model was used to account for the congestion of flow traversing  
 31 the hub-to-hub connection links.

1 For solving the model, a new hybrid metaheuristic algorithm was developed based on non-dominated  
2 sorting genetic algorithm-II (NSGA-II) and a learning-based Iterated Local Search (ILS). A k-means clustering  
3 method is used to link the NSGA-II and ILS to have a powerful search mechanism in terms of both  
4 diversification and intensification. The performance of the proposed hybrid algorithm was validated through a  
5 benchmark against classical NSGA-II and two new recently developed hybrid algorithms that employ machine  
6 learning to improve the performance of the metaheuristic algorithms to show how hybridization of a global  
7 search algorithm with a learning-based local search algorithm improves the performance of the searching  
8 process. It was observed that adding intelligence of machine learning methods to the metaheuristic algorithm and  
9 combining global search and local search algorithms result in high quality solutions even in reasonably low  
10 computational time.

11 Finally, a comprehensive sensitivity analysis was conducted to test not only the sensitivity of the  
12 objective functions, but also the robustness of the solutions in terms of their structure. It was concluded that the  
13 second objective function is highly influenced by the structure of the network. Accordingly, companies whose  
14 most important objective is the transportation time should be more careful about the variation of those  
15 parameters/factors (i.e., transportation time, amount of flow and hub's service rate) that affect the network  
16 structure.

17 In addition, the sensitivity of the objective functions and the structure of the solutions was tested to hub  
18 disruption and products' priority. Regarding the hub disruption, it was concluded that managers should take the  
19 disruption of the hubs into account as an important non-parametric factor that affects the robustness of the  
20 network and try to control the failure rate and/or increase the retrieve rate of the hubs. Regarding the products'  
21 priority, it was observed that prioritizing of products in parallel with the products' flow augmentation  
22 significantly decreases the performance of the hub-and-spoke network in terms of delivery time. Therefore,  
23 managers should be careful when prioritizing the products in a network where the flow has the possibility to be  
24 augmented.

25 One of the main further research direction could be studying multiple allocation version of the hub-and-  
26 spoke network and investigate if splitting the products and sending them to several hubs can reduce the  
27 congestion of the flow in the hubs. Regarding the solution approach, future research effort could be developing  
28 new hybrid algorithms that employs machine learning and powerful local search algorithms (e.g., ILS) not only  
29 as a posteriori method (e.g., proposed GAMLIS algorithm), but also as an interactive way (Sun et al., 2019;  
30 Zhang et al., 2016). Such an algorithm is expected to be both intelligent in searching the solution space and  
31 powerful in diversification and intensification.



1 **Appendix A**

2

3

Table A.1. Metaheuristics performance: CM, QM and Time

Instance	Comparison metric												
	CM				QM				Time (s)				
	GAMLS	HEU1	HEU2	NSGA-II	GAMLS	HEU1	HEU2	NSGA-II	Optimal	GAMLS	HEU1	HEU2	NSGA-II
I1	1.00	1.00	1.00	1.00	0.50	0.50	0.00	0.00	124	92	112	121	129
I2	1.00	1.00	1.00	1.02	0.60	0.40	0.00	0.00	295	111	119	129	136
I3	1.00	1.04	1.05	1.09	0.80	0.20	0.00	0.00	484	125	125	141	152
I4	1.03	1.09	1.11	1.12	0.80	0.20	0.00	0.00	1286	132	142	166	174
I5	1.05	1.12	1.14	1.18	0.90	0.10	0.00	0.00	3485	154	183	193	198
I6	1.09	1.17	1.19	1.22	1.00	0.00	0.00	0.00	8976	177	192	210	219
I7	1.10	1.20	1.24	1.29	1.00	0.00	0.00	0.00	12587	190	214	227	231
I8	1.13	1.25	1.30	1.34	1.00	0.00	0.00	0.00	18654	219	234	239	242
I9	-	-	-	-	1.00	0.00	0.00	0.00	> 10h	228	261	279	289
I10	-	-	-	-	1.00	0.00	0.00	0.00	-	247	289	317	329
I11	-	-	-	-	1.00	0.00	0.00	0.00	-	273	316	359	364
I12	-	-	-	-	1.00	0.00	0.00	0.00	-	294	338	412	423
I13	-	-	-	-	1.00	0.00	0.00	0.00	-	312	386	439	452
I14	-	-	-	-	1.00	0.00	0.00	0.00	-	338	419	512	537
I15	-	-	-	-	1.00	0.00	0.00	0.00	-	384	459	549	589
I16	-	-	-	-	1.00	0.00	0.00	0.00	-	398	483	589	612
I17	-	-	-	-	1.00	0.00	0.00	0.00	-	419	528	629	699
I18	-	-	-	-	1.00	0.00	0.00	0.00	-	556	642	732	786
I19	-	-	-	-	1.00	0.00	0.00	0.00	-	649	829	896	923
I20	-	-	-	-	1.00	0.00	0.00	0.00	-	752	911	994	1112

4

5

Table A.2. Optimal vs. GAMLS & NSGA-II: DM, SM and MID

Instance	Comparison metric									
	DM			SM			MID			
	Optimal	GAMLS	NSGA-II	Optimal	GAMLS	NSGA-II	Optimal	GAMLS	NSGA-II	NSGA-II
I1	1.41	1.52	1.19	0.54	0.44	0.61	0.72	0.59	1.05	
I2	1.22	1.35	1.29	0.64	0.27	0.53	0.71	0.68	0.82	
I3	1.09	1.25	1.20	0.72	0.41	0.52	0.81	0.61	1.19	
I4	1.26	1.29	1.09	0.59	0.24	0.45	0.67	0.67	1.18	
I5	1.23	1.45	1.13	0.78	0.35	0.62	0.62	0.49	0.86	
I6	1.41	1.47	0.95	0.53	0.48	0.46	0.73	0.67	1.11	
I7	1.03	1.51	1.08	0.67	0.38	0.45	0.68	0.48	0.61	
I8	1.31	1.48	1.13	0.73	0.24	0.48	0.63	0.47	0.93	
I9	-	1.34	1.25	-	0.48	0.46	-	0.45	0.78	
I10	-	1.51	0.91	-	0.23	0.62	-	0.42	1.12	
I11	-	1.14	0.86	-	0.36	0.64	-	0.44	0.85	
I12	-	1.52	0.89	-	0.33	0.54	-	0.43	0.92	
I13	-	1.51	0.84	-	0.23	0.51	-	0.51	1.19	
I14	-	1.42	1.15	-	0.33	0.61	-	0.47	0.64	
I15	-	1.48	1.16	-	0.38	0.67	-	0.51	1.04	
I16	-	1.58	1.28	-	0.36	0.65	-	0.69	1.15	
I17	-	1.55	0.88	-	0.41	0.46	-	0.68	1.02	
I18	-	1.43	1.03	-	0.39	0.77	-	0.56	1.09	
I19	-	1.47	0.86	-	0.29	0.67	-	0.61	0.89	
I20	-	1.33	1.19	-	0.33	0.58	-	0.70	0.99	

6

1

Table A.3. GAMLS vs. HEU1 &amp; HEU2: DM, SM and MID

Instance	Comparison metric								
	DM			SM			MID		
	GAMLS	HEU1	HEU2	GAMLS	HEU1	HEU2	GAMLS	HEU1	HEU2
I1	1.52	1.24	1.22	0.44	0.45	0.43	0.59	0.98	1.02
I2	1.35	1.23	1.11	0.27	0.37	0.36	0.68	0.56	0.93
I3	1.25	1.25	1.15	0.41	0.41	0.42	0.61	0.86	0.67
I4	1.29	1.25	1.17	0.24	0.32	0.45	0.67	0.96	0.73
I5	1.45	1.37	1.29	0.35	0.31	0.38	0.49	0.88	0.87
I6	1.47	1.24	1.35	0.48	0.34	0.39	0.67	0.65	0.89
I7	1.51	1.35	1.32	0.38	0.39	0.48	0.48	0.94	1.08
I8	1.48	1.22	1.19	0.24	0.33	0.44	0.47	0.78	0.97
I9	1.34	1.28	1.23	0.48	0.27	0.45	0.45	0.98	1.06
I10	1.51	1.22	1.18	0.23	0.44	0.42	0.42	0.79	1.03
I11	1.14	1.36	1.32	0.36	0.34	0.49	0.44	0.59	1.07
I12	1.52	1.33	1.31	0.33	0.38	0.54	0.43	0.66	0.69
I13	1.51	1.36	1.23	0.23	0.41	0.48	0.51	0.94	0.71
I14	1.42	1.26	1.23	0.33	0.25	0.41	0.47	0.56	0.94
I15	1.48	1.12	1.05	0.38	0.37	0.57	0.51	0.85	0.86
I16	1.58	1.28	1.14	0.36	0.45	0.49	0.69	0.78	0.71
I17	1.55	1.34	1.25	0.41	0.43	0.45	0.68	0.92	1.05
I18	1.43	1.42	1.24	0.39	0.34	0.45	0.56	0.87	0.67
I19	1.47	1.15	1.11	0.29	0.31	0.37	0.61	0.61	0.83
I20	1.33	1.29	1.27	0.33	0.35	0.36	0.70	0.96	0.85

2

3

### References

- 4 Alkaabneh, F., Diabat, A., & Elhedhli, S. (2019). A Lagrangian heuristic and GRASP for the hub-and-spoke network system  
5 with economies-of-scale and congestion. *Transportation Research Part C: Emerging Technologies*, 102, 249-273.
- 6 Alumur, S., Kara, B.Y., 2008. Network hub location problems: The state of the art. *Eur. J. Oper. Res.* 190, 1–21.  
7 <https://doi.org/10.1016/j.ejor.2007.06.008>
- 8 Azizi, N., Vidyarthi, N., & Chauhan, S. S. (2018). Modelling and analysis of hub-and-spoke networks under stochastic  
9 demand and congestion. *Annals of Operations Research*, 264(1-2), 1-40.
- 10 Azizmohammadi, R., Amiri, M., Tavakkoli-Moghaddam, R., Mohammadi, M., 2013. Solving a Redundancy Allocation  
11 Problem by a Hybrid Multi-objective Imperialist Competitive Algorithm. *Int. J. Eng. - Trans. C Asp.* 26, 1031–1042.
- 12 Bolch, G., Greiner, S., Meer, H. de, Trivedi, K.S., 2006. *Queueing Networks and Markov Chains: Modeling and Performance  
13 Evaluation with Computer Science Applications*. John Wiley & Sons.
- 14 Calvet, L., de Armas, J., Masip, D., & Juan, A. A. (2017). Learnheuristics: hybridizing metaheuristics with machine learning  
15 for optimization with dynamic inputs. *Open Mathematics*, 15(1), 261-280.
- 16 Chiang, T.-C., Cheng, H.-C., Fu, L.-C., 2011. NNMA: An effective memetic algorithm for solving multiobjective  
17 permutation flow shop scheduling problems. *Expert Syst. Appl.* 38, 5986–5999.  
18 <https://doi.org/10.1016/j.eswa.2010.11.022>
- 19 Correia, I., Nickel, S., & Saldanha-da-Gama, F., 2018. A stochastic multi-period capacitated multiple allocation hub location  
20 problem: Formulation and inequalities. *Omega*, 74, 122-134.
- 21 da Graça Costa, M., Captivo, M.E., Clímaco, J., 2008. Capacitated single allocation hub location problem—A bi-criteria  
22 approach. *Comput. Oper. Res., Part Special Issue: Topics in Real-time Supply Chain Management* 35, 3671–3695.  
23 <https://doi.org/10.1016/j.cor.2007.04.005>
- 24 de Camargo, R.S., Miranda, G., 2012. Single allocation hub location problem under congestion: Network owner and user  
25 perspectives. *Expert Syst. Appl.* 39, 3385–3391. <https://doi.org/10.1016/j.eswa.2011.09.026>
- 26 de Camargo, R.S., Miranda Jr., G., Ferreira, R.P.M., Luna, H.P., 2009. Multiple allocation hub-and-spoke network design  
27 under hub congestion. *Comput. Oper. Res., New developments on hub location* 36, 3097–3106.  
28 <https://doi.org/10.1016/j.cor.2008.10.004>
- 29 Deb, K., Agrawal, S., Pratap, A., Meyarivan, T., 2000. A Fast Elitist Non-dominated Sorting Genetic Algorithm for Multi-  
30 objective Optimization: NSGA-II, in: Schoenauer, M., Deb, K., Rudolph, G., Yao, X., Lutton, E., Merelo, J.J., Schwefel,  
31 H.-P. (Eds.), *Parallel Problem Solving from Nature PPSN VI, Lecture Notes in Computer Science*. Springer Berlin  
32 Heidelberg, pp. 849–858.
- 33 Dukkanci, O., Peker, M., & Kara, B. Y. (2019). Green hub location problem. *Transportation Research Part E: Logistics and  
34 Transportation Review*, 125, 116-139.
- 35 Elhedhli, S., Hu, F.X., 2005. Hub-and-spoke network design with congestion. *Comput. Oper. Res.* 32, 1615–1632.  
36 <https://doi.org/10.1016/j.cor.2003.11.016>
- 37 Elhedhli, S., Wu, H., 2009. A Lagrangean Heuristic for Hub-and-Spoke System Design with Capacity Selection and  
38 Congestion. *Inf. J. Comput.* 22, 282–296. <https://doi.org/10.1287/ijoc.1090.0335>

- 1 Grove, P.G., O'Kelly, M.E., 1986. Hub Networks and Simulated Schedule Delay. *Pap. Reg. Sci.* 59, 103–119.  
2 <https://doi.org/10.1111/j.1435-5597.1986.tb00985.x>
- 3 Hu, L., Zhu, J. X., Wang, Y., & Lee, L. H. (2018). Joint design of fleet size, hub locations, and hub capacities for third-party  
4 logistics networks with road congestion constraints. *Transportation Research Part E: Logistics and Transportation*  
5 *Review*, 118, 568-588.
- 6 Ishfaq, R., & Sox, C. R., 2012. Design of intermodal logistics networks with hub delays. *European Journal of Operational*  
7 *Research*, 220(3), 629-641.
- 8 Jourdan, L., Dhaenens, C., Talbi, E.-G., 2006. Using Datamining Techniques to Help Metaheuristics: A Short Survey, in:  
9 Almeida, F., Blesa Aguilera, M.J., Blum, C., Moreno Vega, J.M., Pérez Pérez, M., Roli, A., Sampels, M. (Eds.), *Hybrid*  
10 *Metaheuristics, Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 57–69.
- 11 Karimi-Mamaghan, M., Mohammadi, M., Jula, P., Pirayesh, A., & Ahmadi, H. (2020). A Hybrid Metaheuristic for a Multi-  
12 objective Agile Inspection Planning Model under Uncertainty. *European Journal of Operational Research*. In press. DOI:  
13 10.1016/j.ejor.2020.01.061.
- 14 Kian, R., & Kargar, K. (2016). Comparison of the formulations for a hub-and-spoke network design problem under  
15 congestion. *Computers & Industrial Engineering*, 101, 504-512.
- 16 Kleinrock, L., 2007. *Communication Nets: Stochastic Message Flow and Delay*. Courier Corporation.
- 17 Lee, A.H.I., Huang, T.-H., Kang, H.-Y., 2008. A priority mix planning model for semiconductor fabrication under an  
18 uncertain information environment. *J. Inf. Optim. Sci.* 29, 377–400. <https://doi.org/10.1080/02522667.2008.10699811>
- 19 Likas, A., Vlassis, N., J. Verbeek, J., 2003. The global k-means clustering algorithm. *Pattern Recognit., Biometrics* 36, 451–  
20 461. [https://doi.org/10.1016/S0031-3203\(02\)00060-2](https://doi.org/10.1016/S0031-3203(02)00060-2)
- 21 Lo, H.K., Tung, Y.-K., 2003. Network with degradable links: capacity analysis and design. *Transp. Res. Part B Methodol.* 37,  
22 345–363. [https://doi.org/10.1016/S0191-2615\(02\)00017-6](https://doi.org/10.1016/S0191-2615(02)00017-6)
- 23 Marianov, V., Serra, D., 2003. Location models for airline hubs behaving as M/D/c queues. *Comput. Oper. Res.* 30, 983–  
24 1003. [https://doi.org/10.1016/S0305-0548\(02\)00052-7](https://doi.org/10.1016/S0305-0548(02)00052-7)
- 25 Mohammadi, M., Dauzère-Pérès, S., Yugma, C., 2018. Performance evaluation of single and multi-class production systems  
26 using an approximating queuing network. *Int. J. Prod. Res.* 0, 1–27. <https://doi.org/10.1080/00207543.2018.1492163>
- 27 Mohammadi, M., Jolai, F., Rostami, H., 2011. An M/M/c queue model for hub covering location problem. *Math. Comput.*  
28 *Model.* 54, 2623–2638. <https://doi.org/10.1016/j.mcm.2011.06.038>
- 29 Mohammadi, M., Jolai, F., Tavakkoli-Moghaddam, R., 2013. Solving a new stochastic multi-mode p-hub covering location  
30 problem considering risk by a novel multi-objective algorithm. *Appl. Math. Model.* 37, 10053–10073.  
31 <https://doi.org/10.1016/j.apm.2013.05.063>
- 32 Mohammadi, M., Jula, P., Tavakkoli-Moghaddam, R., 2019a. Reliable single-allocation hub location problem with  
33 disruptions. *Transp. Res. Part E Logist. Transp. Rev.* 123, 90–120. <https://doi.org/10.1016/j.tre.2019.01.008>
- 34 Mohammadi, M., Dauzère-pérès, S., Yugma, C., Karimi-Mamaghan, M., 2019b. A queue-based aggregation approach for  
35 performance evaluation of a production system with an AMHS. *Computers & Operations Research*, 2019, doi:  
36 10.1016/j.cor.2019.104838.
- 37 Mohammadi, M., Jula, P., Tavakkoli-Moghaddam, R., 2017. Design of a reliable multi-modal multi-commodity model for  
38 hazardous materials transportation under uncertainty. *Eur. J. Oper. Res.* 257, 792–809.  
39 <https://doi.org/10.1016/j.ejor.2016.07.054>
- 40 Mohammadi, M., Siadat, A., Dantan, J.-Y., Tavakkoli-Moghaddam, R., 2015. Mathematical modelling of a robust inspection  
41 process plan: Taguchi and Monte Carlo methods. *Int. J. Prod. Res.* 53, 2202–2224.  
42 <https://doi.org/10.1080/00207543.2014.980460>
- 43 Mohammadi, M., Tavakkoli-Moghaddam, R., 2016. Design of a fuzzy bi-objective reliable p-hub center problem. *J. Intell.*  
44 *Fuzzy Syst.* 30, 2563–2580. <https://doi.org/10.3233/IFS-151846>
- 45 Mohammadi, M., Tavakkoli-Moghaddam, R., Siadat, A., Rahimi, Y., 2016. A game-based meta-heuristic for a fuzzy bi-  
46 objective reliable hub location problem. *Eng. Appl. Artif. Intell.* 50, 1–19. <https://doi.org/10.1016/j.engappai.2015.12.009>
- 47 Mohammadi, M., Torabi, S.A., Tavakkoli-Moghaddam, R., 2014. Sustainable hub location under mixed uncertainty. *Transp.*  
48 *Res. Part E Logist. Transp. Rev.* 62, 89–115. <https://doi.org/10.1016/j.tre.2013.12.005>
- 49 Morrison, J. R., & Martin, D.P., 2007. Practical extensions to cycle time approximations for the g/g/m queue with  
50 applications. *IEEE Transactions on Automation Science and Engineering*, 4(4), 523-532.
- 51 Oliveto, P. S., & Witt, C. (2015). Improved time complexity analysis of the simple genetic algorithm. *Theoretical Computer*  
52 *Science*, 605, 21-41.
- 53 Özgün-Kibiroğlu, Ç., Serarslan, M. N., & Topcu, Y. İ. (2019). Particle swarm optimization for uncapacitated multiple  
54 allocation hub location problem under congestion. *Expert Systems with Applications*, 119, 1-19.
- 55 Rahimi, Y., Tavakkoli-Moghaddam, R., Mohammadi, M., Sadeghi, M., 2016. Multi-objective hub network design under  
56 uncertainty considering congestion: An M/M/c/K queue system. *Appl. Math. Model.* 40, 4179–4198.  
57 <https://doi.org/10.1016/j.apm.2015.11.019>

- 1 Rodríguez, V., Alvarez, M.J., Barcos, L., 2007. Hub location under capacity constraints. *Transp. Res. Part E Logist. Transp.*  
2 *Rev.* 43, 495–505. <https://doi.org/10.1016/j.tre.2006.01.005>
- 3 Satyam, K., Krishnamurthy, A., 2008. Performance evaluation of a multi-product system under CONWIP control. *IIE Trans.*  
4 40, 252–264. <https://doi.org/10.1080/07408170701488086>
- 5 Sedehzadeh, S., Tavakkoli-Moghaddam, R., Mohammadi, M., & Jolai, F., 2014. Solving a new priority m/m/c queue model  
6 for a multimode hub covering location problem by multi-objective parallel simulated annealing. *Economic Computation*  
7 *& Economic Cybernetics Studies & Research*, 48(4).
- 8 Sun, J., Zhang, H., Zhou, A., Zhang, Q., Zhang, K., 2019. A new learning-based adaptive multi-objective evolutionary  
9 algorithm. *Swarm Evol. Comput.* 44, 304–319. <https://doi.org/10.1016/j.swevo.2018.04.009>
- 10 Taleizadeh, Ata Allah, Maryam Karimi Mamaghan, and Seyed Ali Torabi, 2018. A possibilistic closed-loop supply chain:  
11 pricing, advertising and remanufacturing optimization. *Neural Computing and Applications*: 1-21.
- 12 Vahdani, B., Mohammadi, M., 2015. A bi-objective interval-stochastic robust optimization model for designing closed loop  
13 supply chain network with multi-priority queuing system. *Int. J. Prod. Econ.* 170, 67–87.  
14 <https://doi.org/10.1016/j.ijpe.2015.08.020>
- 15 Yang, T. H., & Chiu, T. Y. (2016). Airline hub-and-spoke system design under stochastic demand and hub congestion.  
16 *Journal of Industrial and Production Engineering*, 33(2), 69-76.
- 17 Wang, X., Tang, L., 2017. A machine-learning based memetic algorithm for the multi-objective permutation flowshop  
18 scheduling problem. *Comput. Oper. Res.* 79, 60–77. <https://doi.org/10.1016/j.cor.2016.10.003>
- 19 Zahiri, B., Tavakkoli-Moghaddam, R., Mohammadi, M., Jula, P., 2014. Multi-objective design of an organ transplant  
20 network under uncertainty. *Transp. Res. Part E Logist. Transp. Rev.* 72, 101–124.  
21 <https://doi.org/10.1016/j.tre.2014.09.007>
- 22 Zhalechian, M., Torabi, S.A., Mohammadi, M., 2018. Hub-and-spoke network design under operational and disruption risks.  
23 *Transp. Res. Part E Logist. Transp. Rev.* 109, 20–43. <https://doi.org/10.1016/j.tre.2017.11.001>
- 24 Zhang, J., Zhan, Z., Lin, Y., Chen, N., Gong, Y., Zhong, J., Chung, H.S.H., Li, Y., Shi, Y., 2011. Evolutionary Computation  
25 Meets Machine Learning: A Survey. *IEEE Comput. Intell. Mag.* 6, 68–75. <https://doi.org/10.1109/MCI.2011.942584>
- 26 Zhang, H., Zhou, A., Song, S., Zhang, Q., Gao, X.-Z., Zhang, J., 2016. A Self-Organizing Multiobjective Evolutionary Algorithm. *IEEE*  
27 *Trans. Evol. Comput.* 20, 792–806. <https://doi.org/10.1109/TEVC.2016.2521868>
- 28 Zheng, J., Zhang, W., Qi, J., & Wang, S. (2019). Canal effects on a liner hub location problem. *Transportation Research Part*  
29 *E: Logistics and Transportation Review*, 130, 230-247.