EURO Journal on Transportation and Logistics 11 (2022) 100095

Contents lists available at ScienceDirect

EURO Journal on Transportation and Logistics

journal homepage: www.sciencedirect.com/journal/euro-journal-on-transportation-and-logistics

Scheduling cross-docking operations under uncertainty: A stochastic genetic algorithm based on scenarios tree

Andrea Gallo [a], Riccardo Accorsi [a,*], Renzo Akkerman [b], Riccardo Manzini [a]

[a] Department of Industrial Engineering, University of Bologna, Italy
[b] Operations Research and Logistics Group, Wageningen University, the Netherlands

ARTICLE INFO

Keywords:
Cross-docking
Operations scheduling
Uncertainty
Scenario tree
GA

ABSTRACT

A cross-docking terminal enables consolidating and sorting fast-moving products along supply chain networks and reduces warehousing costs and transportation efforts. The target efficiency of such logistic systems results from synchronizing the physical and information flows while scheduling receiving, shipping and handling operations. Within the tight time-windows imposed by fast-moving products (e.g., perishables), a deterministic schedule hardly adheres to real-world environments because of the uncertainty in trucks arrivals. In this paper, a stochastic MILP model formulates the minimization of penalty costs from exceeding the time-windows under uncertain truck arrivals. Penalty costs are affected by products' perishability or the expected customer' service level. A validating numerical example shows how to solve (1) dock-assignment, (2) while prioritizing the unloading tasks, and (3) loaded trucks departures with a small instance. A tailored stochastic genetic algorithm able to explore the uncertain scenarios tree and optimize cross-docking operations is then introduced to solve scaled up instaces. The proposed genetic algorithm is tested on a real-world problem provided by a national delivery service network managing the truck-to-door assignment, the loading, unloading, and door-to-door handling operations of a fleet of 271 trucks within two working shifts. The obtained solution improves the deterministic schedule reducing the penalty costs of 60%. Such results underline the impact of unpredicted trucks' delay and enable assessing the savings from increasing the number of doors at the cross-dock.

1. Introduction

The pressure for reducing the distribution time of products is increasing due to supply chain competition and services (Yu et al., 2016; Abad et al., 2018). Consumer demands for urgent deliveries together with the complexity of supply chain networks compel adopting cross-docking hubs to reduce holding costs and lead time (Chuang & Yin, 2016). Cross-docking avoids the most time-intensive tasks of warehousing, i.e., storage and retrieving, saving up to 70% of costs (Vahdani and Zandieh 2010) being convenient for fast-moving products (e.g. perishables). The cross-docking operations are as follows. Trucks depart from the suppliers and are assigned to a dock-door waiting until it becomes newly available. Workers unload the trucks, check documents, label pallets or cartons and sort within a temporary storage area (Yu and Egbelu, 2008). The operators consolidate products at the outbound dock door according to their destination, and, once loaded, the truck departs (Kuo, 2013). As only temporary storage is allowed, cross-docking enhances the customer service level in terms of on-time deliveries while decreasing holding costs (Joo and Kim 2013). The efficacy of cross-docking arises from the simultaneous arrival of trucks (Lee et al., 2006), the prioritization of unloading and loading operations at the inbound/outbound docks, and the consolidation of loads in agreement with the vehicle routing strategy. Synchronizing such processes is of vital importance and determines the ability of cross-docking to achieve its intended aims in practice (Buijs et al., 2014).

Optimization aids such synchronization. Although literature concentrates on deterministic formulation of the cross-docking operations management (Larbi et al., 2011), more recently significant contributions dealing with uncertain truck arrivals are proposed (Wide, 2020). The main sources of uncertainty affecting truck arrivals are weather conditions, i.e., rainy, snowing or foggy weather, roadways traffic and congestion, and lack of transport tracking or tracing infrastructures (Dulebenets, 2019). Unpredictable trucks arrivals result in disrupting cross-docking operations and decisions incurring in decreased service

* Corresponding author. Department of Industrial Engineering, Alma Mater Studiorum – University of Bologna, Viale Risorgimento 2, 40136, Bologna, Italy.
E-mail addresses: andrea.gallo7@unibo.it (A. Gallo), riccardo.accorsi2@unibo.it (R. Accorsi), renzo.akkerman@wur.nl (R. Akkerman), riccardo.manzini@unibo.it (R. Manzini).

https://doi.org/10.1016/j.ejtl.2022.100095
Received 19 July 2021; Received in revised form 16 April 2022; Accepted 30 September 2022
Available online 5 October 2022
2192-4376/

level and delay-induced costs. Hence, the consideration of arrival time as a stochastic variable is crucial in real-world cross-docking decision-making (Shen et al., 2016; Khani 2019).

We formulate a stochastic MILP model integrating the scheduling of truck-to-door, door-to-door handling, and door-to-truck loading tasks while minimizing the cross-docking makespan and delivery delays under uncertain trucks arrival time. The combined management of these aspects into a unique formulation represent the first novel contribution of this study. This model is validated with a numerical example showcasing the benefits achieved by a stochastic formulation. When the instance scales up, optimization fails to provide a solution in reasonable time, whereas heuristics or meta-heuristic algorithms provide quasi-optimal solutions (Joo and Kim, 2013). This paper then illustrates a novel stochastic genetic algorithm (GA) implementing scenario tree (SGA-ST) to minimize penalty costs resulting from late deliveries, representing the second novel contribution to the literature. The proposed metaheuristic schedules the door-to-door handling operations at the cross-dock, prioritizing unloading and loading tasks while minimizing the lateness of goods deliveries. The algorithm is applied to a real case provided by an Italian pallet delivery network provider. This company collects fully-loaded-pallet of perishable products from the suppliers, sorts and consolidates the shipments at the cross-dock, and loads departing trucks for delivery to the retailers. To further support cross-docking practitioners and logistic providers, we develop a ready-to-use decision-support tool (DST) incorporating the novel SGA-ST to aid timely daily decision-making. This DST, representing the third contribution of this paper, solves the door-assignment and scheduling operations for hundreds of trucks within a computational time of on average 1 hone hour, enabling timely daily decision-making and workflow organization. The solution resulting from the scenario tree exploration showcases the savings compared to deterministic decision-making.

The remainder of this paper is organized as follows. Section 2 presents a literature review support-decision models, heuristics and meta-heuristics methods for cross-docking operations management. Section 3 formulates stochastic MILP model validated with a numerical example. A SGA to solve large-instanceistance problem is illustrated and described in Section 4, whilst in Section 5 validates the SGA-ST compared to the SMILP optimizationoptimaztion. Section 6 applies SGA-ST to a real-world case study from a national perishables delivery network from suppliers to retailers. Section 7 discusses the managerial implications of this research while Section 6 concludes the paper and highlights possible future research directions.

2. Literature review

The literature on cross-docking is quite recent. It covers broad decisions on two dimensions (Buijs et al., 2014). The “time-dependent” decisions distinguishes among strategic, i.e. facility location and cross-dock shape and capacity setting (Kheirkhah and Rezaei 2016; Bartholdi and Gue 2000; Mousavi and Tavakkoli-Moghaddam 2013); and operational i.e. scheduling and vehicle routing (Larbi et al., 2011; Ahmadizar et al., 2015). The second dimension pertains the “scope” involving network-wide decisions (Sung and Yang 2008; Wisittipanich et al., 2019; Castellucci et al., 2021) and single-hub decisions (Liao et al., 2012).

Operational models gained increasing attentionattettion since synchronizing operations and dock door-assignment were identified as the main issues of cross-docking. Lee et al. (2006) integrates vehicle routing and dock-assignment by managing truck arrivals and consolidation simultaneously without temporary storage. Yu and Egbelu (2008) introduced the truck-docking scheduling problem to set the priority of dock assignments, using conveyors for door-to-door handling. Musa et al. (2010) developed an algorithm to manage load consolidation. Others formulated objective functions for the makespan to minimize the time spent at the cross-dock (Vahdani and Zandieh 2010; Arabani et al., 2011). Joo and Kim (2013) also considered compound trucks as service vehicles. Agustina et al. (2014) integrated vehicle scheduling and routing into a cross-docking model for perishable products. Moghadam et al. (2014) considered a fleet of trucks. Ahmadizar et al. (2015) illustrated a two-level vehicle routing problem to manage consolidation at the cross-dock and perform last-mile deliveries. Küçükoğlu and Öztürk (2015) formulated a packing problem for departing vehicle loading. Mohtashami et al. (2015) proposed a multi-objective model to minimize the cross-docking makespan and transportation costs simultaneously. Küçükoğlu and Öztürk (2017) proposed a two-stage model to manage supply chain shipments across a cross-dock hub. Others considered environmental impacts (Evangelista 2014; Yin and Chuang 2016; Abad et al., 2018), labour and workforce tasks requirements (Rezaei and Kheirkhah 2018; Tadumadze et al., 2019), or reverse logistics of end-of-life products in closed-loop networks (Rezaei and Kheirkhah 2017).

Table 1 classifies operational support-decision models for cross-docking. Some attempt to minimize the cross-docking makespan despite it does not necessarily convey service level increase. Delay-induced costs must be included in the objective function (Agustina et al., 2014) particularly to deal with perishables deliveries, where unmet due dates incurs in losses. In such circumstance, time-windows constraints could be relaxed instead and violation allowed with penalties (Theophilus et al., 2021).

Only a few studies have attempted to tackle the uncertainty in cross-docking scheduling problems (Theophilus et al., 2019). Mousavi et al. (2014) and Mousavi and Vahdani (2017) employed fuzzy programming and robust optimization, respectively. Larbi et al. (2011) provided three formulations for cross-docking operations scheduling problem with full, partial, and no information regarding the truck arrival times. Uncertainty is addressed in naval container shipping considering weather conditions or port congestion (Li et al. 2016; Wang and Meng 2012), in flights route scheduling (Kenan et al., 2018) or in the urban public transportation time-tables (Tong and Wong, 1999; Vodopivec and Miller-Hooks, 2017). To the best of our knowledge, no formulations deal with uncertainty in truck arrival at cross-dock hub using stochastic programming. We hereby formulates a stochastic MILP that schedule dock-assignment, door-to-door handling with temporary storage, and door-to-truck loading tasks under uncertain arrivals, while minimizes the costs from the violating the time-windows.

Because of the inherentlyinerenthly NP-hard complexity of cross-docking scheduling problems, alternative solving methods have been explored to provide good solutions in reasonable time. Cross-docking algorithms can be classified for network-oriented or single terminal algorithms (Buijs et al., 2014).

2.1. Cross-docking networks algorithms

Sung and Yang (2008) proposed an exact branch-and-price algorithm for cross-docking network design. Others integrate a terminals location problem with flows allocation decisionsdeicisions (Sung and Song, 2003), dock-assignment (Küçükoğlu and Öztürk, 2017), or vehicle routing (Mousavi and Tavakkoli-Moghaddam, 2013; Mokhtarinejad et al., 2015). Rezaei and Kheirkhah (2018) addressed sustainability aspects modeling a closed-loop network where the cross-dock manage even the reverse flow of end-of-life.

Vehicle routing solving algorithms for cross-docking networks are investigated. Musa et al. (2010) proposed an ant colony optimization (ACO) algorithm to minimize transportation to customers with multiple pick-up terminals. Ahmadizar et al. (2015) proposed a GA, while Mousavi and Vahdani (2017) implemented a self-adaptive imperialist competitive algorithm (ICA).

2.2. Single-terminal algorithms

These focus on a single cross-docking terminal. Most of the literature

Table 1
Classification of cross-docking modeling.

	Dock-door assignments	Vehicle routing	Operations scheduling	Temporary storage	Model Formulation	Objective Function	Uncertainty
Abad et al., (2018)	X	✓	✓	✓	MILP	Costs & fuel	X
Agustina et al., (2014)	✓	✓	✓	✓	MILP	Costs	X
Ahmadizar et al., (2015)	X	✓	X	✓	MILP	Costs	X
Alpan et al., 2011	✓	✓	✓	✓	DP	Costs	X
Arabani et al., (2011)	X	X	✓	✓	MILP	Makespan	X
Joo and Kim (2013)	✓	X	✓	✓	MILP	Makespan	X
Kheirkhah and Rezaei (2016)	X	✓	X	X	MILP	Costs	X
Kucukoglu and Ozturk 2015	X	✓	X	X	MILP	Costs	X
Kucukoglu and Ozturk 2016	✓	✓	✓	X	MILP	Costs	X
Lee et al., (2006)	X	✓	X	X	IP	Costs	X
Moghadam et al., (2014)	X	✓	✓	X	MIP	Costs	X
Mohtashami et al., (2015)	X	✓	✓	X	MILP	Makespan, costs & trips	X
Mokhtarinejad et al., (2015)	X	✓	✓	X	MILP	Costs & waiting time	X
Mousavi and Tavakkoli-Moghaddam (2013)	X	✓	X	X	MIP	Costs	X
Mousavi and Vahdani (2017)	X	✓	✓	✓	MILP	Costs	✓
Mousavi et al., (2014)	X	✓	✓	✓	MILP	Costs	✓
Musa et al., (2010)	X	✓	X	X	IP	Costs	X
Rezaei and Kheirkhah (2017)	X	✓	X	X	MILP	Costs	X
Rezaei and Kheirkhah (2018)	X	✓	X	X	MILP	Sustainability	X
Santos et al., (2011)	X	✓	X	X	IP	Costs	X
Santos et al., (2013)	X	✓	X	X	IP	Costs	X
Sung and Song (2003)	X	✓	X	X	IP	Costs	X
Sung and Yang (2008)	X	✓	X	X	IP	Costs	X
Vahdani and Zandieh (2010)	X	X	✓	✓	MILP	Makespan	X
Wen et al., (2009)	X	✓	✓	X	MILP	Travel time	X
Wisittipanich and Hengmeechai (2017)	✓	X	✓	✓	MIP	Makespan	X
Yin and Chuang (2016)	X	✓	X	X	IP	Costs	X
Yu and Egbelu (2008)	X	X	✓	✓	MILP	Makespan	X
Yu et al., (2016)	X	✓	X	X	MILP	Costs	X
This paper	✓	**X**	✓	✓	**SMILP**	**Costs**	✓

formulates meta-heuristics to solve vehicle routing problems starting at the cross-dock. Lee et al. (2006) and Wen et al. (2009) use tabu search (TS). Yu et al. (2016) adopt simulated annealing (SA) considering the simultaneous arrival of all trucks. Santos et al. (2011) proposed an heuristic based on branch-and-price algorithm to minimize vehicle routing costs. Grangier et al. (2017) proposed a large neighborhood search algorithm. Küçükoğlu and Öztürk (2015) combined TS-SA to solve a vehicle routing and packing problem. Yin and Chuang (2016) introduced environmental impact constraints within adaptive artificial bee colony (ABC) algorithm.

Others focused on vehicle-induced operations scheduling at the terminal. The objective function is commonly minimizing the makespan (Mohtashami, 2015; Wisittipanich and Hengmeechai, 2017; Arabani et al., 2012) or the transportation costs (Mohtashami et al., 2015). Some compares the computational performance of alternative meta-heuristic algorithms (Vahdani and Zandieh, 2010; Arabani et al., 2011; Joo and Kim 2013). Moghadam et al. (2014) merge ACO and SA to solve a vehicle routing and scheduling problem that minimizes the total cost. Others solve vehicle pick-up and delivery problems (Abad et al., 2018; Santos et al., 2013) and truck arrivals sequencing problems (Liao et al., 2012; Yu and Egbelu, 2008).

2.3. Classification of cross-docking algorithms

Table 2 classifies the most significant contributions upon the scope and goal of the problem and the type of solving method adopted. For further details we remand to Theophilus et al. (2019).

The pioneering work by Fathollahi-Fard et al. (2019) illustrates the adoption of the novel Social Engineering Optimizer (SEO) to solve large-instance truck-induced operations scheduling and minimizing the makespan. Shahmardan and Sajadieh (2020) solve a truck scheduling problem for a single-terminal where inbound trucks are also used for deliveries through a reinforced learning of a tailored neighborhood search. Dulebenets (2021) sheds a light on new efficient hybrid solving method for cross-docking operations scheduling by proposing a Adaptive Polyploid Memetic Algorithm (APMA) which outperform the state of the art of meta-heuristics algorithms.

Considering the reported literature, solely Larbi et al. (2011), Rahbari et al. (2019) and Xi et al. (2020) addressed uncertain in truck arrivals. Larbi et al. propose two heuristics for partial and no information on truck arrivals with the attempt to prioritize the departures by decreasing probability to be completely loaded, while minimizing costs for handling, transportation and penalties. Rahbari et al. provide a bi-objective formulation for an integrated VRP and scheduling for perishables suggesting the adoption of meta-heuristics or genetics to solve the problem with large real-world instances. Xi et al. brought out an exact method based on column and constraint generation to minimize cost under uncertain truck arrivals.

In this paper, we illustrate a two-stage SGA implementing a scenario tree (SGA-ST) to solve each realization of the stochastic parameters, and scheduling dock-assignmentassingment, door-to-door handling, and door-to-truck loading task while minimizing penalties from undelivered products (i.e. losses) time-windows violation. Developing a tailor-made SGA-ST to implement the scenario tree of a stochastic cross-docking scheduling problem in single-terminal operations represents the second novel contribution of this research.

3. Stochastic problem formulation

3.1. Problem boundaries

A delivery service network for perishables is organized by a 3 PL

Table 2
Classification of solving methods for cross-docking problems.

Authors	Year	Scope	Objective Function	Meta-heuristics	Heuristics	Uncertainty
Sung and Song	2003	Network design, flows allocation	Min. costs	✔	X	X
Lee et al.	2006	Vehicle routing	Min. costs	✔	X	X
Sung and Yang	2008	Network design	Min. costs	X	✔	X
Yu and Egbelu	2008	Truck sequencing problem	Min. makespan	X	✔	X
Wen et al.	2009	Vehicle routing	Min. travel time	✔	X	X
Musa et al.	2010	Vehicle routing	Min. costs	✔	X	X
Vahdani and Zandieh	2010	Trucks operations scheduling	Min. makespan	✔	X	X
Arabani et al.	2011	Trucks operations scheduling	Min. makespan	✔	X	X
Larbi et al.	2011	Trucks operations scheduling	Min. costs	X	✔	✔
Santos et al.	2011	Vehicle routing	Min. costs	X	✔	X
Arabani et al.	2012	Trucks operations scheduling	Min. makespan & lateness	✔	X	X
Liao et al.	2012	Truck sequencing problem	Min. makespan	✔	X	X
Joo and Kim	2013	Trucks operations scheduling	Min. makespan	✔	X	X
Kuo	2013	Truck sequencing and dock assignment	Min. makespan	✔	X	X
Mousavi and Moghaddam	2013	Network design, vehicle routing	Min. costs	✔	X	X
Santos et al.	2013	Pick-up and delivery problem	Min. costs	X	✔	X
Moghadam et al.	2014	Trucks operations scheduling	Min. costs	✔	X	X
Ahmadizar et al.	2015	Vehicle routing	Min. costs	✔	X	X
Kucukoglu and Ozturk	2015	Vehicle routing	Min. costs	✔	X	X
Mohtashami et al.	2015	Trucks operations scheduling	Min. makespan, costs & trips	✔	X	X
Mohtashami	2015	Trucks operations scheduling	Min. makespan	✔	X	X
Mokhtarinejad et al.	2015	Network design, vehicle routing	Min. costs & truck waiting time	✔	X	X
Kucukoglu and Ozturk	2016	Network design, dock assignment	Min. costs	✔	X	X
Yin and Chuang	2016	Vehicle routing	Min. costs	✔	X	X
Yu et al.	2016	Vehicle routing	Min. costs	✔	X	X
Grangier et al.	2017	Vehicle routing	Min. costs	✔	X	X
Mousavi and Vahdani	2017	Vehicle routing	Min. costs	✔	X	X
Wisittipanich and Hengmeechai	2017	Trucks operations scheduling	Min. makespan	✔	X	X
Rezaei and Kheirkhah	2018	Network design	Min. costs, environmental impact, max. social benefit	✔	X	X
Fathollahi-Fard et al.	2019	Truck operations scheduling	Min makespan	✔	✔	X
Xi et al.	2020	Truck operations scheduling	Min. costs	X	X	✔
Shahmardan and Sajadieh	2020	Truck operations scheduling	Min makespan	✔	✔	X
Dulebenets	2021	Truck operations scheduling	Min. costs	✔	✔	X
This paper		Truck operations scheduling	Min. costs (direct and delays)	✔	X	✔

company through cross-docking terminal which consolidates palletized products from growers and suppliers and deliver, after consolidation, to the retailers, C (Fig. 1). For each retailer $c \in C$, the accepted receiving time-window opens at time otw_c and closes at ctw_c. This time window is affected by a service level agreement, corresponding to the level of target freshness guaranteed by the retailer to the costumers. When a delivery exceeds the time window, a reward is paid to the retailer for decayed products shelflife. Retailers prefer on-time partial deliveries rather than delayed complete order. This yields a penalty cost upc_c paid for pallet unit delivered to c and time-unit after ctw_c.

Let IT be the set of inbound trucks arriving at the cross-dock. Each of the trucks $i \in IT$ delivers products $p \in P$ in quantity $iq_{i,p}$. Incoming truck must be assigned to an inbound dock door $\in ID$. Pallets are unloaded from trucks with time ut and temporary stored in front of the door

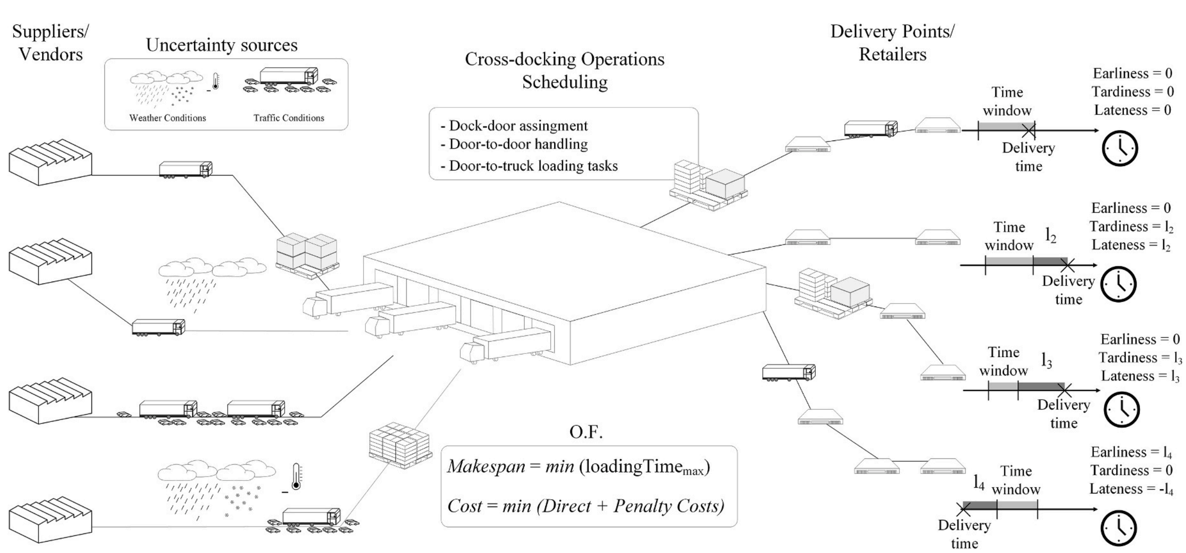

Fig. 1. Stochastic cross-docking scheduling problem boundaries.

waiting for door-to-door handling. The mean transfer time for door-to-door handling is *trt*. Each pallet is assigned to available departing trucks *OT* under a capacity *cap* [pallets]. The cross-dock operations are pulled by a demand $dem_{p,c}$. The joint availability of trucks and workers trigger the loading task in time *lt*. When loading is completed, trucks head to retailers with routing time $tt_c, c \in C$. Deliveries can be split to reduce penalties. Secondary deliveries are enabled by an urgent call for transport which incurs a further cost $ucsd_c$.

The proposed formulation undertakes the following assumptions, considered in the literature when no otherwise expressed (Theophilus et al., 2019):

- The temporary storage capacity inside the hub is unlimited;
- Trucks and dock doors are dedicated to inbound or outbound operations only;
- External cross-dock area for trucks waiting is unlimited;
- Unlimited outbound fleet for secondary deliveries;
- No preemption of loading and unloading tasks is allowed.

3.2. Stochastic truck arrivals

The arrival time of an inbound truck $i \in IT$ at the cross-docking hub is denoted by $\widetilde{\tau}_i$ as in Eq. 1:

$$\widetilde{\tau}_i = \tau_i^{dep} + \widetilde{\tau}_i^{tt}, \forall i \in IT, \tag{1}$$

where τ_i^{dep} is the departure time of truck *i* from the supplier, and $\widetilde{\tau}_i^{tt}$ is the corresponding travel time. This time is affected by weather conditions or roadways state (i.e., traffic, congestion, unpaved roads) as well as other drivers like seasonality. These result in uncertain arrival which lead to poor accuracy of the scheduling.

$\widetilde{\tau}_i^{tt}$ is defined as a stochastic variable with probability distribution function $f(\tau_i^{tt})$ defined within the interval $[\tau_i^{min}, \tau_i^{max}]$, where τ_i^{tt} is a realization of the random variable $\widetilde{\tau}_i^{tt}$, and τ_i^{min} and τ_i^{max} are two positive numbers representing the minimum and maximum travelling times respectively.

As a consequence, $\widetilde{\tau}_i$ is also a random variable with a probability distribution function $f(\tau_i)$ defined over the interval $[\tau_i^{dep} + \tau_i^{min}, \tau_i^{dep} + \tau_i^{max}]$. The distribution of random variable $\widetilde{\tau}_i$ can be discretized by performing a frequency analysis on the historical travel times per each supplier-terminal route. The values assumed by the stochastic variable i.e., variable realization, correspond to a scenarios $s \in S$. The basic scenario with predictable on-time arrivals (i.e., arrivals at the expected time) is defined by s_0. By introducing the set of scenarios *S*, the stochastic arrival times of the inbound trucks is denoted by τ_{is}. Each scenario is characterized by the realization's probability p_s of a discrete random variable, following the property:

$$\sum_{s \in S} p_s = 1 \tag{2}$$

The probability of scenario *s* is estimated as the relative frequency of τ_{is}. The variability of the arrival times affect the robustness of the scheduling. To not exceed the drivers' working shift, the arrivals and the associated unloading operations must be scheduled precisely along with the occupation of the dock-doors. By incorporating the stochastic nature of the arrivals, decision-making conveys reducing lead time at the cross-dock and optimizing the delivery service level.

3.3. Estimation of delivery routing time

To decrease the problem's complexity, the retailers served by a single truck are considered destinations of a generic delivery route whose demand is the sum of the single orders:

$$dem_{p,c} = \sum_{i \in I_c} dem_{p,i}, \text{ where } I_c \subseteq C \tag{3}$$

This prompts implementing the vehicle routing problem separately and estimating travel time or evaluating the time-windows violation. Good approximations can be obtained using quasi-optimal methods, as showcased by Figliozzi (2008). The approximation of the vehicle routing time leads to several simplifications of the model:

- The scheduling problem does not involve vehicle routing for departing trucks.
- The number of retailers $c \in C$ is a subset holding groups of delivery points served by a single truck;
- Each outbound truck serves a cluster of retailers whose route is denoted by *c*;
- Each vehicle must deliver within a single time-window.

3.4. Two-stage stochastic model

Sets, variables, parameters are defined in this section according to the scheme of Figs. 1 and 2. A set of linear constraints ensure that the precedence graph is respected and explain the link between variables.

3.4.1. Sets

$f \in F$ Suppliers
$c \in C$ Retailers
$i \in I$ Inbound trucks
$o \in O$ Outbound trucks
$p \in P$ Products
$id \in ID$ Inbound dock doors
$od \in OD$ Outbound dock doors
$s \in S$ index of scenarios

3.4.2. Parameters

ctwc closing time of the time window of client *c*
upcc unit penalty cost for each pallet delivered one time-unit late
iqi, pallets of product *p* carried by inbound truck *i*
ut unit unloading time
trt door-to-door average transfer time.
lt unit loading time
cap capacity of outbound trucks
demp, demand of product *p* by client *c*
ttc travel time from the cross-dock to client *c*
ucsdc unit penalty cost for each late pallet delivered with a secondary shipment
$\tau$is arrival time of inbound truck *i* in scenario *s*
ps probability of scenario *s*.

3.4.3. Decision variables

lat_c absolute lateness value of the load delivered to customer *c*
$dt_{o,c}$ delivery time of outbound truck *o* to client *c*; $dt_{o,c} = 0$ if outbound truck *o* does not serve client *c*
$del_{o,c} = 1$ if outbound truck *o* serves client *c* (0, otherwise).
su_{is} starting time of the unloading process of inbound truck *i* in scenario *s*
$pl_{i_1,i_2,o} = 1$ if the products from inbound truck i_1 are loaded into outbound truck *o* before products from inbound truck i_2 (0, otherwise).
$sl_{i,o}$ starting time of the loading process for products coming from inbound truck *i* into outbound truck *o*
$trp_{p,i,o,s}$ number of pallets of product *p* transferred from inbound truck *i* to outbound truck *o* in scenario *s*
$ltp_{c,p,i,o,s}$ number of pallets of product *p* from inbound truck *i* that should have been carried by outbound truck *o* to customer *c* but arrived late and must be delivered with a second truck
$exch_{i,o} = 1$ if inbound truck *i* transfers products to outbound truck *o* (0, otherwise).
$uda_{id_i s} = 1$ if inbound truck *i* is assigned to inbound dock door d_i in scenario *s* (0, otherwise).

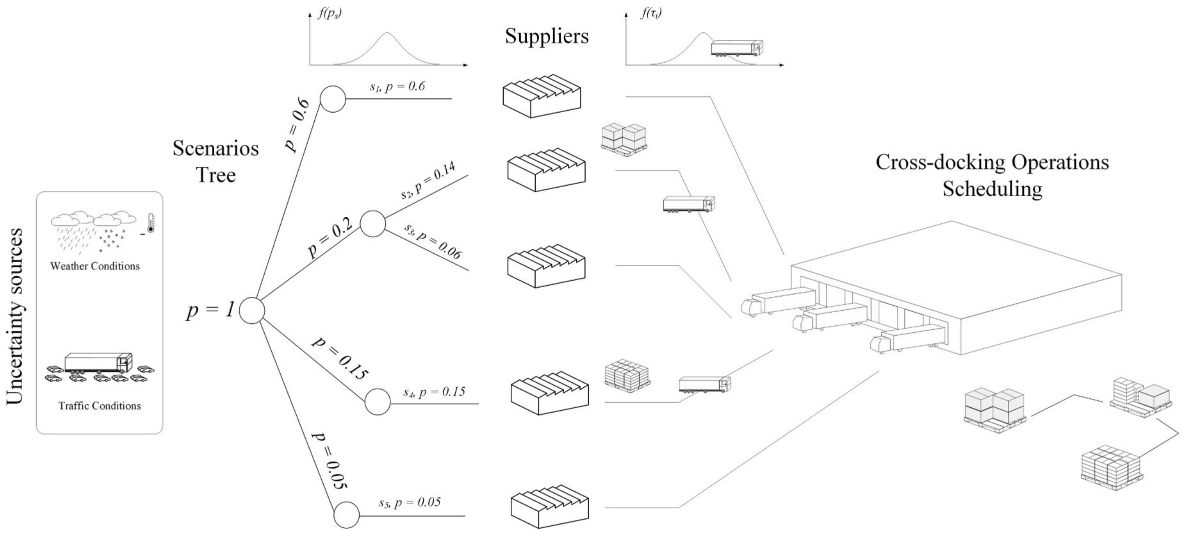

Fig. 2. Formulating stochastic arrivals with scenario tree.

$lda_{od_o} = 1$ if outbound truck o is assigned to outbound dock door d_o (0, otherwise).

$pdii_{d_i,i_1,i_2,s} = 1$ if inbound truck i_1 immediately precedes inbound truck $i_2$at dock door d_i in scenario s (0, otherwise).

$pdia_{d_i,i_1,i_2,s} = 1$ if inbound truck i_1 precedes inbound truck $i_2$at dock door d_i in scenario s (0, otherwise).

$pdoi_{d_o,o_1,o_2} = 1$ if outbound truck o_1 immediately precedes outbound truck $o_2$at dock door d_o (0, otherwise).

$pdoa_{d_o,o_1,o_2} = 1$ if outbound truck o_1 precedes outbound truck $o_2$at dock door d_o (0, otherwise).

$tmc_{o,c}$ departure time of outbound truck o from the cross-dock traveling to client c; $tmc_{o,c} = 0$ if outbound truck o does not serve client c

3.4.4. Objective function

The objective function minimizes the penalty costs incurred for violating the time-windows. These costs are split into penalties for late prime delivery or urgent secondary delivery. The freshness of delivered products, affected by delays, is agreed with the retailers resulting into a target service level. Even deliveries before the time-window are not acceptable by the retailer generating a cost. The objective function is expressed as follows:

$$\min \sum_{c\in C}\sum_{p\in P} lat_c upc_c dem_{p,c} + \sum_{s\in S}\sum_{c\in C}\sum_{i\in IT}\sum_{p\in P} p_s ltp_{c,p,i,o,s} ucsd_c. \tag{4}$$

The first term of Eq. (4) corresponds to the total costs for earliness and tardiness of the deliveries. Such cost is avoided when a truck respects the retailer time-window. The costs of secondary deliveries i.e., urgent transport services, are not time-dependent.

3.4.5. Constraints

$$lat_c \geq otw_c - \sum_{o\in OT} dt_{oc} \quad \forall c \in C \tag{5}$$

$$lat_c \geq \sum_{o\in OT} dt_{oc} - ctw_c \quad \forall c \in C \tag{6}$$

$$\sum_{o\in OT} del_{oc} \geq 1 \quad \forall c \in C \tag{7}$$

$$\sum_{c\in C} del_{o,c} = 1 \quad \forall o \in OT \tag{8}$$

$$dt_{o,c} \leq M \bullet del_{o,c} \quad \forall o \in OT, c \in C \tag{9}$$

$$tmc_{o,c} \leq M \bullet del_{o,c} \quad \forall o \in OT, c \in C \tag{10}$$

$$\left(tmc_{o,c} + del_{o,c} \bullet tt_c\right) = dt_{o,c} \quad \forall o \in OT, c \in C \tag{11}$$

$$\sum_{s\in S}\sum_{i\in IT}\left(tr_{p,i,o,s} + ltp_{c,p,i,o,s}\right) \geq \sum_{c\in C} dem_{p,c} \bullet del_{o,c} \quad \forall p \in P, o \in OT, s \in S \tag{12}$$

$$\begin{cases} tr_{p,i,o,s} \leq iq_{i,p} \\ if\ sl_{i,o} \geq su_{i,s} + \sum_{p\in P} iq_{i,p}(ut + trt) - M\left(1 - exch_{i,o}\right) \\ tr_{p,i,o,s} = 0 \\ otherwise \end{cases} \quad \forall p \in P, i \in I, s \in S \tag{13}$$

$$\sum_{p\in P} tr_{p,i,o,s} \leq M \bullet exch_{i,o} \quad \forall i \in IT, o \in OT, s \in S \tag{14}$$

$$\sum_{o\in OT}\left(tr_{p,i,o,s} + \sum_{c\in C} ltp_{c,p,i,o,s}\right) \leq iq_{i,p} \quad \forall p \in P, i \in I, s \in S \tag{15}$$

$$tmc_{o,c} \geq sl_{i,o} + \sum_{p\in P} lt_{i,o} \bullet tr_{p,i,o,1} - M\left(1 - exch_{i,o}\right) \quad \forall o \in OT, c \in C, i \in IT \tag{16}$$

$$sl_{i,o} \geq su_{i,1} + \sum_{p\in P} iq_{i,p} \bullet (ut + trt) - M\left(1 - exch_{i,o}\right) \quad \forall i \in IT, o \in OT \tag{17}$$

$$su_{i,s} \geq \tau_{is} \quad \forall i \in IT, s \in S \tag{18}$$

$$\sum_{i_1\in IT, i_1} pl_{i_1,i_2,o} = exch_{i_2,o} \quad \forall i_2 \in IT, o \in OT \tag{19}$$

$$\sum_{i_1\in IT} pl_{i_2,i_1,o} = exch_{i_2,o} \quad \forall i_2 \in IT, o \in OT \tag{20}$$

$$\sum_{i\in IT} pl_{0,i,o} = 1 \quad \forall o \in OT \tag{21}$$

$$\sum_{i\in IT} pl_{i,H,o} = 1 \quad \forall o \in OT \tag{22}$$

$$sl_{i,o} \geq sl_{i_2,o_2} + \sum_{p \in P} lt \bullet tr_{p,i_2,o_2,s_0} - M\left(1 - \sum_{d \in OD} pdoi_{d,o_2,o}\right) \quad \forall i, i_2 \in IT, o, o_2 \in OT \tag{23}$$

$$sl_{i,o} \geq sl_{i1,o} + \sum_{p \in P} lt \bullet tr_{p,i_1,o,s_0} - M\left(1 - pl_{i_1,i,o}\right) \quad \forall i, i_1 \in IT, o \in OT \tag{24}$$

$$su_{i,s} \geq su_{i_1,s} + \sum_{p \in P} ut \bullet iq_{i_1,p} - M\left(1 - \sum_{d \in ID} pdii_{d,i_1,i,s}\right) \quad \forall i, i_1 \in IT, s \in S \tag{25}$$

$$\sum_{d \in ID} uda_{d,i,s} = 1 \quad \forall i \in IT, s \in S \tag{26}$$

$$\sum_{d \in OD} lda_{d,o} = 1 \quad \forall o \in OT \tag{27}$$

$$uda_{0,d,s} = 1 \quad \forall d \in ID, s \in S \tag{28}$$

$$uda_{H,d,s} = 1 \quad \forall d \in ID, s \in S \tag{29}$$

$$lda_{0,d} = 1 \quad \forall d \in OD \tag{30}$$

$$lda_{H,d} = 1 \quad \forall d \in OD \tag{31}$$

$$\sum_{i_1 \in IT:\ i_1 < H} pdii_{d,i_1,i,s} = uda_{i,d,s} \geq pdia_{d,i,i_2,s} \quad \forall i, i_2 \in IT, d \in ID, s \in S \tag{32}$$

$$\sum_{o_1 \in OT:\ o_1 < H} pdoi_{d,o_1,o} = lda_{o,d} \geq pdoa_{d,o,o_2} \quad \forall o, o_2 \in OT, d \in OD \tag{33}$$

$$\sum_{i_1 \in IT:\ i_1 > 0} pdii_{d,i,i_1,s} = uda_{i,d,s} \geq pdia_{d,i_2,i,s} \quad \forall i, i_2 \in IT, d \in ID, s \in S \tag{34}$$

$$\sum_{o_1 \in OT:\ o_1 > 0} pdoi_{d,o,o_1} = lda_{o,d} \geq pdoa_{d,o_2,o} \quad \forall o, o_2 \in OT, d \in OD \tag{35}$$

$$pdii_{d,i_1,i_2,s} \leq pdia_{d,i_1,i_2,s} \leq 1 - pdia_{d,i_2,i_1,s} \quad \forall i_1, i_2 \in IT, d \in ID, s \in S \tag{36}$$

$$doi_{d,o_1,o_2} \leq pdoa_{d,o_1,o_2} \leq 1 - pdoa_{d,o_2,o_1} \quad \forall o_1, o_2 \in OT, d \in OD \tag{37}$$

$$pdia_{d,0,i,s} \leq uda_{i,d,s} \geq pdia_{d,i,H,s} \quad \forall i \in IT, d \in ID, s \in S \tag{38}$$

$$pdoa_{d,0,o} \leq lda_{o,d} \geq pdoa_{d,o,H} \quad \forall o \in OT, d \in OD \tag{39}$$

$$pdia_{d,i,0,s} = pdia_{d,H,i,s} = 0 \quad \forall i \in IT, d \in ID, s \in S \tag{40}$$

$$pdoa_{d,o,0} = pdoa_{d,H,o} = 0 \quad \forall o \in OT, d \in OD \tag{41}$$

$$pdia_{d,i_2 i_3,s} \geq pdii_{d,i_1 i_2,s} + pdia_{d,i_1 i_3,s} - 1 \quad \forall d \in \mathrm{ID}, i_1, i_2, i_3 \in \mathrm{IT}, s \in S \tag{42}$$

$$pdoa_{d,o_2 o_3} \geq pdoi_{d,o_1 o_2} + pdoa_{d,o_1 o_3} - 1 \quad \forall d \in \mathrm{OD}, o_1, o_2, o_3 \in \mathrm{OT} \tag{43}$$

$$del_{o,c}, pl_{i_1,i_2,o}, exch_{i,o}, uda_{i,d_i,s}, lda_{o,d_o}, pdii_{d_i,i_1,i_2,s}, pdia_{d_i,i_1,i_2,s}, pdoi_{d_o,o_1,o_2}, pdoa_{d_o,o_1,o_2} \in \{0,1\}, \forall c \in C, o, o_1, o_2 \in OT, \forall i, i_1, i_2 \in IT, d_i \in ID, d_o \in OD, s \in S$$

$$lat_c, dt_{o,c}, su_{i,s}, sl_{i,o}, tmc_{o,c} \in \mathbb{R}^+, \forall c \in C, o \in OT, \forall i \in IT, s \in S$$

$$tr_{p,i,o,s}, ltp_{c,p,i,o,s} \in \mathbb{Z}^+, \forall c \in C, p \in P, \forall i \in IT, o \in OT, s \in S$$

Constraints (5) and (6) define the absolute lateness value. Constraints (7–9) ensure that each client is served by one outbound trucks. The delivery time of each truck is calculated as the departing time from the cross-dock plus the routing time (10,11). The reatiler's demand must be fulfilled, either with the scheduled delivery or through a secondary delivery (12, 15). Constraints (13) and (14) impose deliveries only for received pallets.

Inequalities (16–18) define the timing of the cross-dock operations. An outbound truck departs from the terminal only after the loading activities have been completed (16); Loading starts after door-to-door handling (17); Unloading is triggered by inbound trucks arrival (18).

Constraints (19,20, 24) prioritizes the loading operations, imposing the FIFO sequence. Each truck is preceded by the dummy truck *0* (21) and succeeded by the dummy truck *H* (22). Inequalities (23,24) define the minimum time to start loading of an outbound truck. Loading of a new truck starts when loading of previous have been completed (23). The minimum loading times are estimated on the expected arrival times of inbound trucks given scenario s_0 (i.e., no late arrivals).

For inbound trucks, the unloading starts once the previous truck assigned to a door is completely unloaded (25). Unloading tasks are driven arrivals, so that this time is determined for each possible representation of the stochastic variable τ_{is} (i.e., arrival time). Conversely, for departing trucks, the cross-docker establishes when loading begins.

All trucks are assigned to a dock-door (26,27), except for dummy trucks *0* and *H* assigned to all the doors simultaneously (28–31). The remaining (32–43) define the precedence order of inbound and outbound trucks at the corresponding dock doors, prioritizing the operations at the terminal.

All the inbound operations are dependent on scenario *s*. Outbound operations need to be scheduled in advance to guarantee the trucks' availability. This requires such operations to be scheduled in advance without knowing the actual arrival time.

The proposed formulation is a two-stage stochastic model with complete recourse so that the feasibility of the solution at the second stage is always guaranteed. Such assumption is not unrealistic owing to the broad availability of extra-paid delivery trucks in the transport market. Whether all carriers are unavailable at a given time, they will reach the cross-docking hub as soon as they complete their previous missions.

4. Stochastic GA implementing scenario tree

Cross-docking scheduling problems are inherently NP-hard (Abad et al., 2018) and are not solvable by a commercial solver in a reasonable time for real-world sized problems (Joo and Kim, 2013). GA proved to be effective in solving NP-hard optimization problems, particularly for VRP and cross-dock scheduling (Ahmadizar et al., 2015; Kusolpuchong et al., 2019). GAs can provide a quasi-optimal solution for complex instances with dozens of trucks and outperform most state-of-art meta-heuristics (Vahdani and Zandieh, 2010). In this paper, a tailor-made SGA implementing the scenario tree (SGA-ST) is proposed to solve the problem formulation in 3.4 when the instance scales up.

4.1. Chromosome definition

Each chromosome is made of an array of genes that represent the entities/variables of the problem to solve. The proposed GA encodes a random sequence of trucks into each chromosome. Each gene labels a specific truck. Since no compound trucks exist and each truck can be either devoted to inbound or outbound, this algorithm introduces two types of chromosomes, namely *inbound* and *outbound* chromosomes.

Inbound chromosomes count a number of genes equal to the incoming trucks. The sequence encoded into such chromosomes represents the priority for the unloading operations. SGA-ST converts such priority into a solution for the inbound scheduling problem through the application of the algorithm described in Section 4.3. Likewise, the priority order given by the outbound chromosome generates a solution for departures scheduling as illustrated in Section 4.3. At the first stage, the proposed SGA-ST sets $|S| + 1$ chromosomes: $|S|$ inbound chromosomes whose number is equal to the number of scenarios of different arrivals, and one outbound chromosome. The latter encoded the final schedule of the outbound trucks loading and departing operations aimed to minimize the penalty costs from late deliveries. For each inbound chromosome (i.e. scenario), SGA-ST extracts a random sequence between 1 and $|I|$. Similarly, a random sequence between 1 and $|O|$ is also generated for the outbound chromosome.

Given the two-stage nature of the problem, the assignment of inbound trucks to the corresponding dock-door represents a second-stage decision. Therefore, by providing a schedule for each scenario, the cross-docker implements the optimal decision based on the actual leaf of the scenarios tree.

Outbound chromosomes count a number of genes equalizing the departing vehicles, and consequently, the sequence of the chromosome represents the priority for loading. Such list indicates the candidate trucks to be assign to the next available door. Scheduling outbound trucks' operations consists of the first-stage decision as it needs a-priori management of the vehicles fleet for deliveries.

We then solve the stochastic scenario tree built upon the realization of the uncertain arrival time. Each feasible solution of the scheduling problem is represented by the inbound chromosomes corresponding to scenario *s*. Since a scenario corresponds to sequence of arrivals, SGA-ST provides a schedule for each sequence. Settings two sets of chromosomes for inbound and outbound respectively, enables generating only feasible solutions and avoids time-consuming cuts. Fig. 3 draws the pattern used to set chromosomes, and the main steps of the proposed genetic algorithm.

4.2. Fitness function

A fitness function determines the most promising chromosomes that will evolve to the next generation. As chromosomes represent feasible solutions for the cross-docking scheduling problem, the objective function determines (4) provides a formulation of the fitness function.

4.3. Decision rules

The algorithm randomly generates a chromosome's array. A set of tailor-made rules are applied to obtain the decision variables and quantify the fitness function.

4.3.1. Door assignment procedure

The truck with the highest priority (i.e., the first gene in the chromosome) is assigned to the first available dock door. When no doors are available, the algorithm calculates the release time $rt{:=}su_{i_0,s} + \sum_{p\in P} ut \bullet iq_{i_0,p}\ \forall id \in ID$, and assigns the incoming truck to the door with the earlier release time.

Because of the uncertain arrival time, the door-assignment algorithm pseudo-code is repeated for each scenario *s* thus providing $|S|$ different door-assignments and truck-induced operations schedules. The pseudo code 2., based on the comparison between release *rt* and available time, $availTime{:}= su_{i_0 s} + \sum_{p\in P} ut \bullet iq_{i_0,p} + trt \bullet tr_{p,i,o,s}$, aids determining the assignment of outbound dock-doors *OD*.

Door assignment algorithm for inbound trucks *i*

```
rt := 0
door := 1
for each inbound dock door d
    if there are no trucks assigned to d
        door := d
        rt := 0
    else
        Pick the last truck i_0 assigned to door d
        time := su_{i_0,s} + ∑_{p∈P} ut · iq_{i_0,p}
        if time < rt
            door := d
            rt := time
        end
    end
end
Add truck i to the list of trucks assigned to door
uda_{i,door,s} := 1
su_{i,s} := rt
```

Pseudo code 1. Inbound dock-door assignment.

Door assignment algorithm for outbound trucks o

```
rt := 0
door := 1
for each outbound dock door d
    if there are no trucks assigned to d
        door := d
        rt := 0
    else
        Pick the last truck o_0 assigned to door d
        time := ∑_{i∈I} sl_{i,o,s} + ∑_{p∈P} ∑_{i∈I} lt · tr_{p,i,o_0,s}
        if time < rt
            rt := time
            door := d
        end
    end
end
Pick the client c assigned to truck o
for each product p
    if dem_{p,c} > 0
        for each inbound truck i
            if tr_{p,i,o,s} > 0
                availTime := su_{i_0 s} + ∑_{p∈P} ut · iq_{i_0,p} + trt · tr_{p,i,o,s}
                sl_{i,o,s} := availTime
                if availTime > rt
                    rt := availTime
                end
            end
        end
    end
end
lda_{i,door} := 1
```

Pseudo code 2. Outbound dock-door assignment.

4.3.2. Final outbound schedule

The dock-door assignment procedure determines a schedule for the inbound operations per each realization of the stochastic parameter τ_{is}. It is then possible to obtain an outbound schedule by assessing the expected costs for each scenario $s \in S$ and the probability ps of its realization.

The final outbound schedule procedure proposed in Pseudo code 3. determines the optimal schedule of the outbound truck loading operations based on the fitness function value described in Equation (4). It evaluates $\forall$ scenario s_1 the cost function $upc_c \bullet \left(\sum_{p\in P} dem_{p,c} - latePal\right) \bullet lat_c + latePal \bullet ucsd_c \bullet p_{s_2}$ resulting from the outbound schedule based on s_1. Assuming this schedule, the procedure computes the associated error cost due to the realization of any other scenario s_2. This cost is weighted to the probability of s_2 given by p_s. The final outbound schedule is that which minimizes the error cost from the inaccurate forecast of trucks' arrival.

Final outbound schedule for outbound truck o serving client c

$optScen := 0$
$optCost := \infty$
for each scenario s_1
  $latePal := 0$
  $cost := 0$
  for each scenario s_2
    for each inbound truck i
      if $\sum_{p \in P} tr_{p,i,o,s_2} > 0$
        if $su_{i,s_2} + \sum_{p \in P} ut \cdot iq_{i_0,p} + trt \cdot tr_{p,i,o,s_2} > sl_{i,o,s_1}$
          $latePal := \sum_{p \in P} tr_{p,i,o,s_2}$
          $ltp_{c,p,i,o,s_1} := ltp_{c,p,i,o,s_1} + tr_{p,i,o,s_2}$
          $delTime := \sum_{i \in I} sl_{i,o,s_2} + \sum_{p \in P} \sum_{i \in I} lt \cdot tr_{p,i,o,s_2} + tt_c$
          $lat_c := 0$
          if $delTime > ctw_c$
            $lat_c := ctw_c - delTime$
          end
          $cost := cost + upc_c \cdot (\sum_{p \in P} dem_{p,c} - latePal) \cdot lat_c + latePal \cdot ucsd_c \cdot p_{s_2}$
          if $cost < optCost$
            $optCost := cost$
            $optScen := s_2$
          end
        end
      end
    end
  end
end

Pseudo code 3. Outbound truck scheduling.

4.4. Parents selection, crossover and swap

The parents selection procedure links two consecutive generations of GA. The most promising chromosomes (i.e., with high fitness value) transmit their pattern to the next generations. Once the parents are selected, they undergo recombination of their genes to shape the offspring. Parents' selection mechanism uses a random selection based on roulette wheel pattern (Ahmadizar et al., 2015). The new generation recombines with a new population randomly generated to prevent the risk of local-minimum considering a mutation rate (MR) of 0.1.

Crossover combines the parents' genes to shape two new solutions (Arabani et al., 2011). In agreement with the stochastic two-stage formulation, the cross-over rate (CR) change at each generation in the interval [0,1]. The crossover position r is indeed chosen randomly using a roulette wheel. The genes in the batch $[1, r]$ are exchanged as in Fig. 3. The generation of feasible solutions is guaranteed by removing duplicated genes, replaced by new random sequences.

The swap operator performs a second random rearrangement of the genes at each chromosome (Joo and Kim, 2013). Here, the roulette wheel extracts two random positions r_1 and r_2 to swap in the interval [1, |I|] or [1, |O|].

5. SGA-ST validation

A gap estimation is carried out to assess the quality of the SGA-ST solution. It consists of two steps intended for lower and upper bound respectively. By definition, the lower bound could be unfeasible because of the constraints' relaxation, while the upper bound always coincides with a feasible solution.

5.1. Lower-Upper bound estimation

To get the lower bound, we relax the constraints of the dock-doors' availability (Constraints 23–25). The workers unload the trucks as soon as they reach the cross-docking terminal. The same relaxation applies to outbound dock doors. Trucks are loaded as soon as the ordered pallets are available in the temporary storage, with no delays due to limited capacity. These steps are applied to each chromosome, calculating a fitness value, and providing a lower bound to each solution.

The upper bound estimation procedure generates a feasible solution by determining a ranked sequence for inbound and outbound trucks. Inbound trucks are scheduled according to the FIFO policy. When the full order is ready, the gene corresponding to the outbound truck is added to the chromosome. The procedure illustrated in the previous section evaluates the chromosomes and tallies the fitness value of the upper bound.

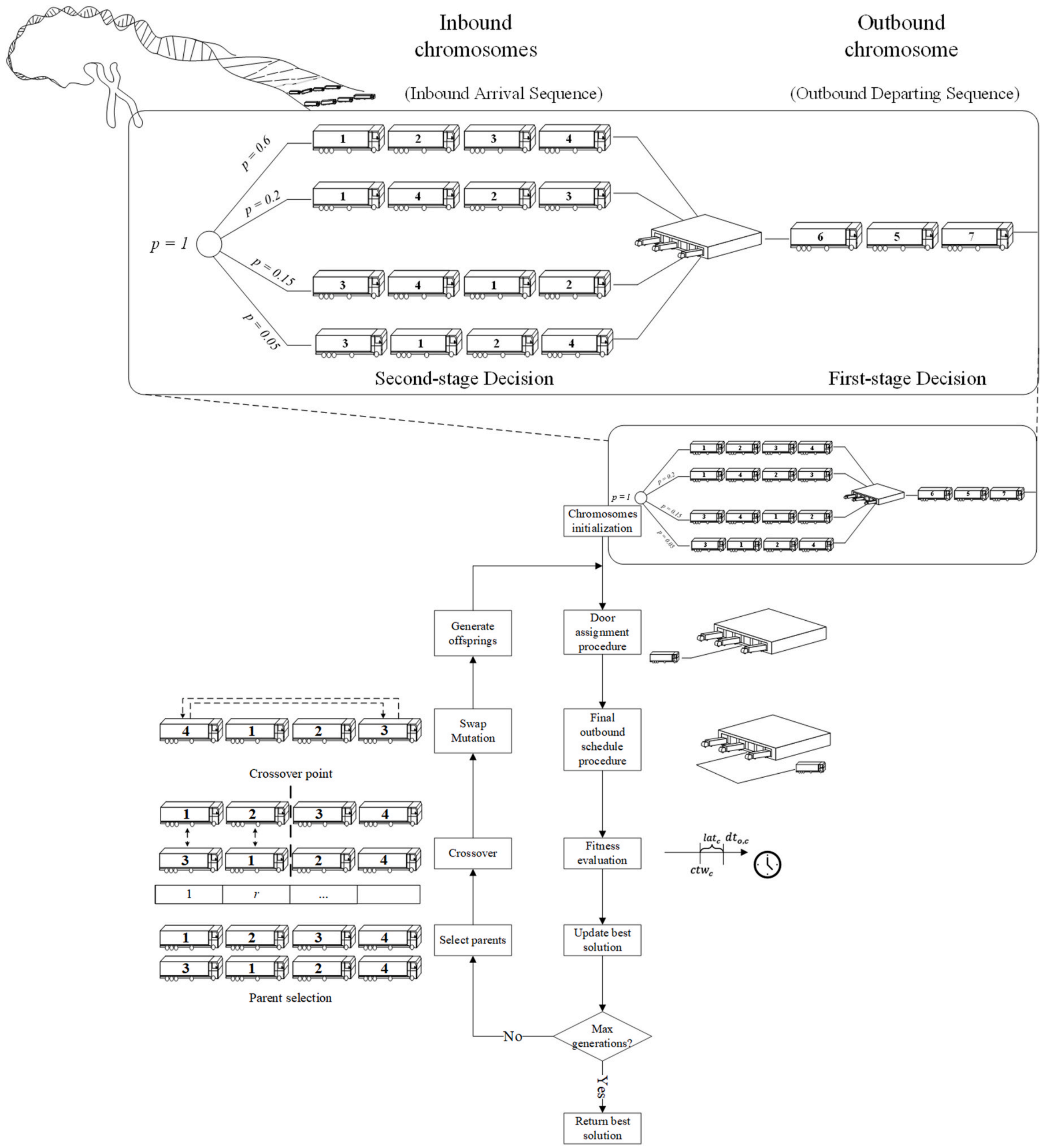

Fig. 3. Chromosomes definition over the scenario tree and SGA-ST steps.

5.2. SGA-ST vs. SMILP comparison

The validation of the proposed SGA-ST is carried out by comparing the solution obtained with a Stochastic MILP (SMILP) formulation defined in Section 3. The SMILP has been implemented in AMPL and solved using the Gurobi solver running on a 2.59 GHz Dual Core PC with 12 GB of RAM for a small instance. As the prosed stochastic scheduling problem is intrinsically NP-hard, validation is only possible with a small instance. While reporting estimates upper and lowe bounds, Table 3 underlines that SGA-ST achieves the optimum quickly when the instance scales up. When the total number of trucks exceeds 20, the optimization model lacks to provide a solution on time while the SGA-ST finds its best within 12 seconds.

6. Real-world application

The proposed SGA-ST algorithm is applied to a real-world scheduling problem of a national delivery service company. This company is specialized in delivering fast-moving and perishable products to retailers. Two type of services are ruled in agreement with the retailersreatilers. These correspond to two delivery time-windows, service levels, and penalties. The *Gold* service completes deliveries within 24 h hours and is intended for highly perishable products like some fresh spring varieties (e.g. cherries, strawberries). The *Silver* service is cheaper but ensures delivery within 48 h hours. Both time-windows begin when the truck departs from the supplier.

The cross-docking terminal layout is made of 14 dock-doors, split equally for inboundindound and outbound tasks. Table 4 summarizes the input data of the large-scale instance. Records of unloading, loading, and door-to-door handling tasks have been gathered and calculated

Table 3
Results of the comparison between the SGA-ST and a stochastic MILP model.

	Inbound data		Outbound data		Scenarios	SMILP solving time [s]	SGA-ST solving time [s]	SMILP optimal solution	SGA-ST best solution	Upper bound	Lower bound
	Trucks	Doors	Trucks	Doors							
1	4	1	2	1	2	5	8	27	27	46	3
2	4	2	5	2	1	399	0.5	104.364	104.364	118.93	90.32
3	4	2	5	2	2	47	0.5	118.785	118.785	127.76	112.58
4	6	2	8	2	1	140	1	174.32	174.32	196.52	170.12
5	10	2	10	2	1	NA	12	NA	208.88	309.87	195.48

Table 4
Input data of the case study

	Value
Inbound/Outbound trucks $\lvert I\rvert$; $\lvert O\rvert$	130; 141
Gold service deliveries	70 out of 141
Silver service deliveries	71 out of 141
Inbound/outboundOurbound dock-doors $\lvert ID\rvert$; $\lvert OD\rvert$	7; 7
Average load of inbound truck $\frac{\sum_i\sum_p iq_{ip}}{\lvert I\rvert}$ [pallets]	30
ut [s/pallet]	55
trt [s/pallet · trip]	46
lt [s/pallet]	55
upc_c	1
$ucsd_c$	10

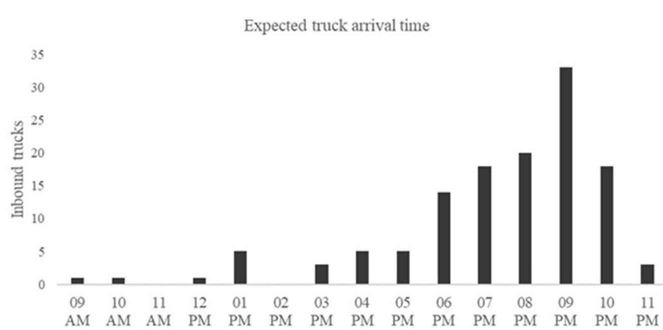

Fig. 4. Frequency distribution of expected truck arrival times.

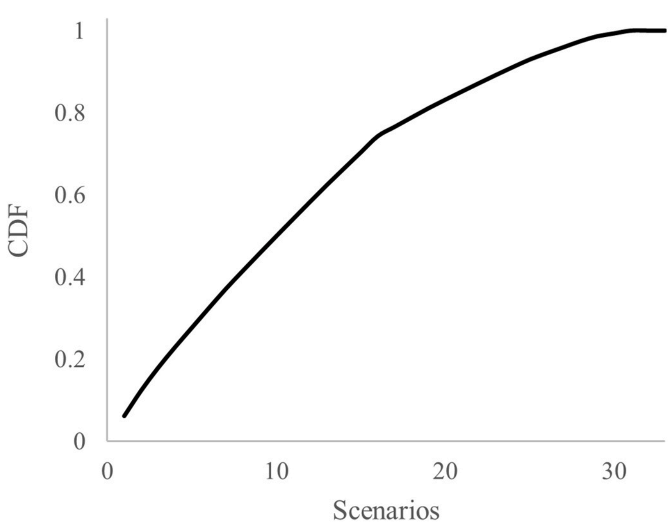

Fig. 5. Scenarios' Cumulative Distribution Function (CDF).

through discrete frequency analyses conducted on field. Fig. 4 shows the distribution of the arrival time τ_i for the inbound trucks departing from the 130 Italian provinces. The best-fitting PDF $f(\tau_i)$ can vary with the supplier, the type of route, and with seasonality on weather and traffic conditions. The collection of historical arrivals recommends setting a ST made of 33 scenarios $s \in S$. Each scenario corresponds to different arrival times $\tau_{is}$and probability p_s of realization in agreement with the CDF of Fig. 5.

The proposed SGA-ST has been implemented into a expert system developed in .NET C# programming language and run on a 2.59 GHz Dual Core PC with 12 GB of RAM. A graphic user interface (GUI) allows setting the number of mutations of the GA algorithm and the chromosomes population's size. At the first generation, the upper and lower bounds of the fitness function (4) are quantified. For the others, the decision-support tool provides the current best solution, the number of late deliveries, the best mutation, and the corresponding gap with the lower bound as in Fig. 6.

After instance-induced tuning, ten chromosomes are initialized at each generation. The SGA-ST requires one random chromosome for all the departing trucks and one chromosome per scenario *s* for the inbound trucks. A problem solution is encoded into 34 chromosomes. Each inbound chromosome schedules 130 trucks, whilst a outbound chromosome considers 141 trucks. The testbed of the algorithm demonstrates it converges after 10^3 generations, resulting into $34{\cdot}10^4$ chromosomes to solve such a instance.

Fig. 7 draws the dock-door utilization over the scheduling horizon of daily operations. The chart splits the inbound and outbound docks and shows up- and down-times of the unloading and loading operations carried out at each door $id \in ID$ and $od \in OD$ respectively. It provides a-priori practical insights to the cross-docker about which will the mostly utilized door will be, and how to organize workers' teams for unloading, handling and shipping preparation accordingly. Fig. 7 also highlights layout bottle necks and assessesasseses the contribution of the dock-doors to improve the overall cross-docking performance, to reduce penalties and increase retailers service level.

The dock-doors utilization is further explored via a sensitivity analysis. We tested other two cross-dock layout configuration with 12 and 16 doors respectively, still equally distributed between inbound and outbound. The comparison is illustrated in Fig. 8.

We confirm that dock-doors utilization rates are affected by their number, but such evidence can change with the seasonal traffic and delivery service intensity. More considerations arise from Table 5. SGA-ST schedules 271 trucks with an expected penalty costs (i.e. fitness function) of 45.78 [€/day] with $\lvert ID\rvert = 7$, $\lvert OD\rvert = 7$, and only three deliveries result to be late. The total lateness is 6.19 [h/day], mainly due the traveling time (i.e. 23 h/route) required to achieve the farthest retailer. The upper bound is quantified in 303.91 [€/day] resulting from 28 late deliveries, whilst the lower bound is 1.97 [€/day], with only one late delivery.

Because arrivals are concentrated in the time-window 06:00–10:00 p.m.PM, utilization is maximum in this batch requiring for accurate arrangement of the workforce during the day (Tadumadze et al., 2019). Whether we tried new layout configurations by increasing or decreasing the number of doors, the DST could estimates the penalties paid by the cross-docker during a broad horizon (i.e. one year) and assess whether or when layout re-design might pay off.

The fitness's lower bound, obtained by relaxing the dock-doors' availability (23–25), seems not influenced by the number of dock-doors. The lower bound efficacy in the proposed SGA-ST increases with properly designed cross-docking layout, whose dook-doors are sufficient to

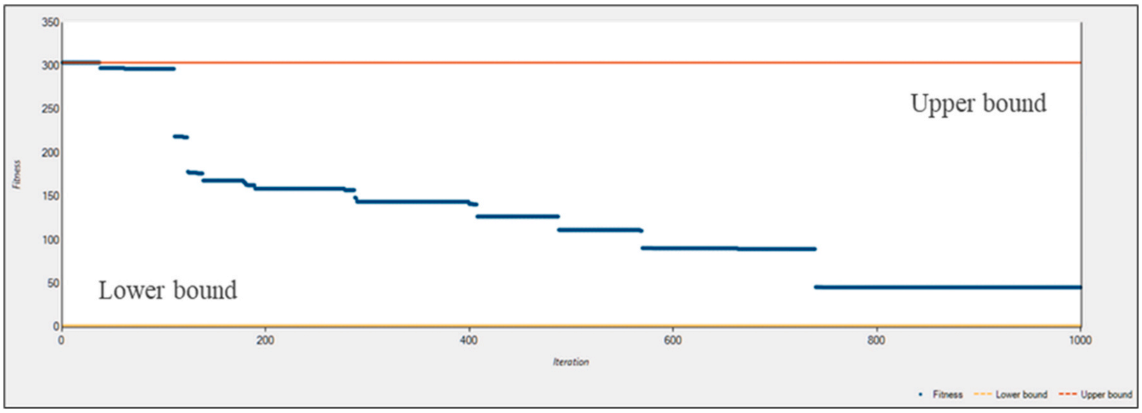

Fig. 6. Decision-support tool GUI: fitness functions throughout generations.

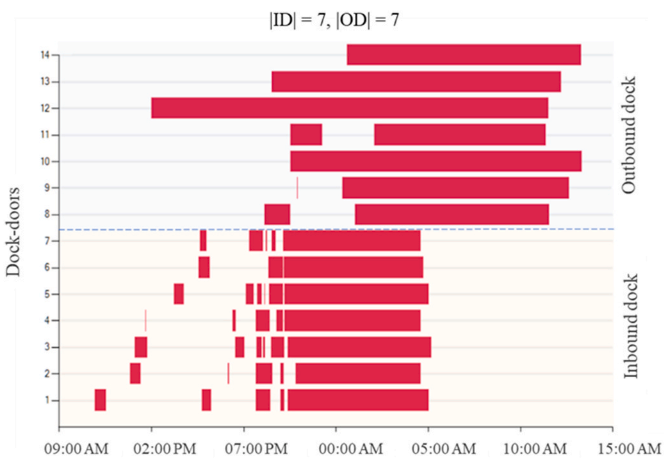

Fig. 7. Decision-support tool GUI: Dock-door utilization over the day.

avoid any trucks queues ($|ID|\rightarrow|I| \wedge |OD|\rightarrow|O|$) or when arrivals are uniformly distributed. Conversely, it is less effective when dock-doors are bottlenecks.

The solutions' benchmark is the Business-As-Usual (BAU) scenario which accounts penalties for 111.76 [€/day] resulting from 6 late deliveries with a total lateness of 7.5 [h]. The proposed SGA-ST provides a cost saving of 59.04%. compared to the BAU scenario.

7. Discussion of managerial implications

As uncertainty significantly affects cross-docking operations scheduling, the adoption of still fascinating deterministic decision-making, broadly explored in the literature (e.g. Tadumadze et al., 2019; Theophilus et al., 2021), could not convey cost savings in real-world applications characterized by uncertain truck arrivals. In this paper, we formulate a stochastic two-stage of a single-terminal cross-docking operations scheduling problem and develop a tailor-made SGA-ST solving method for large-scale instance under uncertainty. The savings provided by the SGA-ST are assessed and compared to the deterministic problem using two well-known estimators: the Benefit Stochastic Solution (BSS) and the Expected Value of Perfect Information (EVPI) (see Birge and Louveaux, 2011). The comparison entails the expected value (EV) of the solution which approximates the stochastic parameters with their EVs. The resulting optimal function (4) is equal to 362.52 [€/day], even worse than the upper bound with SGA-ST. The BSS quantifies the cost burden (i.e., corresponding to +87.38%) reached when approximation to EVs of the stochastic parameters are used instead of stochastic modeling.

To further benchmark the proposed SGA-ST, the EVPI, representing the gain from full information availability, is quantified. The full information availability (PI) consists of the actual arrival time of each incoming truck acknowledged before the scheduling. The SGA-ST estimates EVPI by considering the certain scenario s* i.e., PI, which generate a cost function of 36.68 [€/day] with a burden of 9.1 [€/day] (i.e. compared to 45.78 [€/day]). We reasonably conclude that the optimal schedule provided by SGA for each scenario (i.e. inbound chromosome) of the ST, supports decision-making under uncertainty achieving a cost reduction close to EVPI (i.e. full information availability). The managerial implications resulting from adopting the DST incorporating the novel SGA-ST consist on the chance for scheduling daily cross-docking

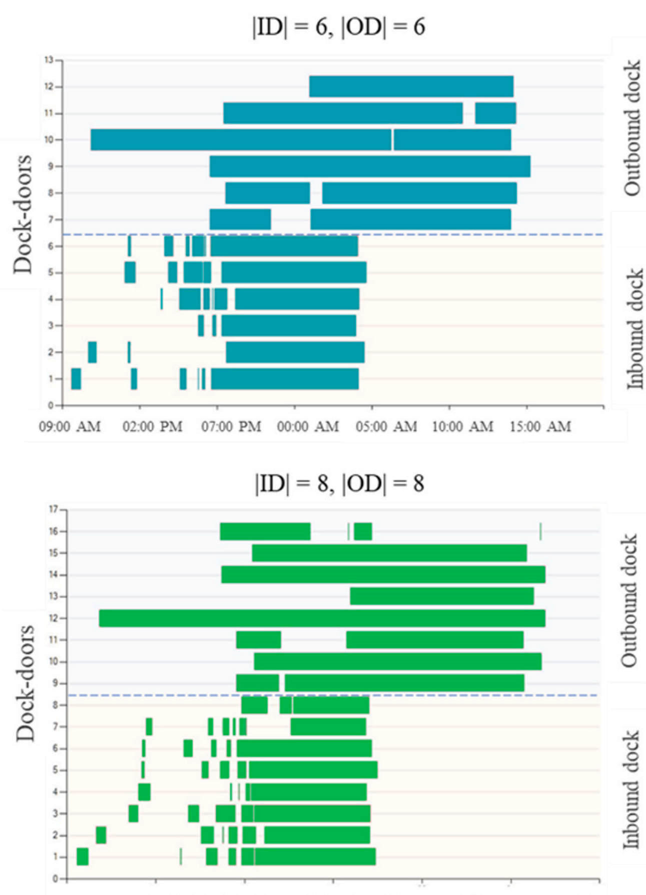

Fig. 8. Comparisons of two layout configurations based on available dock-doors.

Table 5
Results of the case study and sensitivity analysis on dock doors.

Solved Layout Scenario	Generations	Solutions per generation	Best fitness value (4) [€/day]	BAU Savings [%]	N. late deliveries	Upper bound	Lower bound
$\|ID\| = 7$, $\|OD\| = 7$	1000	10	45.78	-59.04	3	303.91	1.97
$\|ID\| = 6$, $\|OD\| = 6$	1000	10	135.69	+21.41	5	1043.7	1.97
$\|ID\| = 8$, $\|OD\| = 8$	1000	10	9.96	−91.08	2	37.6	1.97
BAU: $\|ID\| = 7$, $\|OD\| = 7$	1000	10	111.76	–	6	–	–

operations with no a-priori knowledge of the arrival times, a better organization of the workforce during the shift, and the allocation of the most-skilled workers to the expected bottlenecks or late deliveries with the highest penalty costs (i.e. highly perishable products' deliveries).

The conducted sensitivity analysis on the layout configuration in terms of dock-door availability (Table 5) provides other practical insights. It aids the terminal-logistic manager while assessing the daily costs resulting from uncertain arrivals, the unmet service level for the retailers, and the penalties paid due to the lack of dock-doors. According to the quantified savings or costs, two dimensions of improvement can be tried and simulated using the DST: (1) reconfiguring the terminal layout by increasing the number of dock-doors (if possible), or (2) reducing the unloading, door-to-door, and loading tasks' time through more expert labour or more efficient storage/handling equipment. SGA-ST enables predicting time and cost savings generated daily when the number of dock-doors increases. While solving the real-world instance, we reveal how reducing daily penalty costs of 80% is possible by establishing a new dock-door. Similarly, SGA-ST could assess the costs saving associated with a reduction of the cross-docking tasks time.

This research is triggered by an industrial and managerial decision-making issue: scheduling cross-docking operations under uncertain truck arrivals. With respect to excellent contributions to solve complex cross-docking scheduling (Fathollahi-Fard et al., 2019; Shahmardan et al., 2020; Theophilus et al., 2021; Dulebenets, 2021), we found a gap in the research where uncertainty is not broadly investigated. A first tentative to address this gap, is proposed in this paper with a tailor-made SGA-ST heuristics solving a two-stage SMILP when the instance scale up. We first respond to managerial needs with a user friendly DST incorporating the SGA and enabling timely operational decision-making and service level-induced costs optimization. Future research is intended for developing new functionalities in the DST and benchmarking the proposed SGA-ST with other solving algorithm (e.g., SA, VNS, TS, ACO, ICA, ABC, SEO, APMA) adopted to address other problems' dimensions to seek greater cross-docker's benefits.

8. Conclusions and future research

Cross-docking terminal enables consolidating and sorting fast-moving and perishables products throughout the supply chain networks and contributes to reducing warehousing costs and transportation efforts. Synchronization between inbound and outbound operations leads to cost reduction and increases on-time deliveries, but it is sensitive to uncertain truck arrivals.

Scheduling unloading, door-to-door and loading operations in a cross-docking terminal is a true challenge for practitioners, in real-world logistic environments characterized by uncertainty. Decision-support systems aid logistic managers to tackle such issue, but the inherentlyinerently complexity of stochastic optimizationoptimzation models discourages their adoption in practice. We found very few stochastic formulations and solving methods for cross-docking operations scheduling are illustrated in the literature.

This paper aims to fill this gap by proposing a tailor-made SGA-ST to schedule truck services and handling operations at the cross-docking terminal. The proposed SGA-ST algorithm optimizes the penalty costs paid to the retailers for late deliveries of perishable products. After validation and comparison with SMILP, we applied SGA-ST to a real-world instance of a national delivery service company of perishable products serving retailers operating. SGA-ST, fueled by the historical profile of the trucks arrival times, provides significant reduction of such costs and increases retailers service compared to a deterministic decision. Furthermore, GA overcomes the computational complexity of SMILP and enables solving large instances in reasonable time. To further support managerial decisions, the proposed SGA-ST has been incorporatedincoroporated into a user-friendly DST and GUIs to assess and quantify daily savings.

Future research is intended for handling strategic decisions like the optimal number of dock-doors to avoid queues and increase service level. Additionally, the minimization of forklift traveling time in door-to-door handling tasks can be considered into a bi-objective problem formulation. Lastly, further studies will benchmark the proposed SGA-ST with other metaheuristic algorithms and incorporate the best-performing techniques within the DST.

Declaration of competing interest

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

References

Abad, H.K.E., Vahdani, B., Sharifi, M., Etebari, F., 2018. A bi-objective model for pick-up and delivery pollution-routing problem with integration and consolidation shipments in cross-docking system. J. Clean. Prod. 193, 784–801.
Agustina, D., Lee, C., Piplani, R., 2014. Vehicle scheduling and routing at a cross docking center for food supply chains. Int. J. Prod. Econ. 152, 29–41. https://doi.org/10.1016/j.ijpe.2014.01.002.
Ahmadizar, F., Zeynivand, M., Arkat, J., 2015. Two-level vehicle routing with cross-docking in a three-echelon supply chain: a genetic algorithm approach. Appl. Math. Model. 39 (22), 7065–7081.
Alpan, G., Larbi, R., Penz, B., 2011. A bounded dynamic programming approach to schedule operations in a cross docking platform. Comput. Ind. Eng. 60, 385–396. https://doi.org/10.1016/j.cie.2010.08.012.
Arabani, A.B., Ghomi, S.F., Zandieh, M., 2011. Meta-heuristics implementation for scheduling of trucks in a cross-docking system with temporary storage. Expert Syst. Appl. 38 (3), 1964–1979.
Arabani, A.B., Zandieh, M., Ghomi, S.M.T.F., 2012. A cross-docking scheduling problem with sub-population multi-objective algorithms. Int. J. Adv. Manuf. Technol. 58 (5-8), 741–761.
Birge, J.R., Louveaux, F., 2011. Introduction and examples. Introduction Stochastic Programm. Springer Ser. Operations Res. Financ. Eng. 3–54. https://doi.org/10.1007/978-1-4614-0237-4_1.
Buijs, P., Vis, I.F., Carlo, H.J., 2014. Synchronization in cross-docking networks: a research classification and framework. Eur. J. Oper. Res. 239 (3), 593–608.
Castellucci, P.B., Costa, A.M., Toledo, F., 2021. Network scheduling problem with cross-docking and loading constraints. Comput. Oper. Res. 132, 105271.
Dulebenets, M.A., 2019. A delayed start parallel evolutionary algorithm for just-in-time truck scheduling at a cross-docking facility. Int. J. Prod. Econ. 212, 236–258.
Dulebenets, M.A., 2021. An Adaptive Polyploid Memetic Algorithm for scheduling trucks at a cross-docking terminal. Inf. Sci. 565, 390–421.
Fathollahi-Fard, A.M., Ranjbar-Bourani, M., Cheikhrouhou, N., Hajiaghaei-Keshteli, M., 2019. Novel modifications of social engineering optimizer to solve a truck scheduling problem in a cross-docking system. Comput. Ind. Eng. 137, 106103.
Grangier, P., Gendreau, M., Lehuédé, F., Rousseau, L.-M., 2017. A matheuristic based on large neighborhood search for the vehicle routing problem with cross-docking. Comput. Oper. Res. 84, 116–126.
Joo, C.M., Kim, B.S., 2013. Scheduling compound trucks in multi-door cross-docking terminals. Int. J. Adv. Manuf. Technol. 64 (5-8), 977–988.
Khani, A., 2019. An online shortest path algorithm for reliable routing in schedule-based transit networks considering transfer failure probability. Transp. Res. Part B Methodol. 126, 549–564.

Kheirkhah, A., Rezaei, S., 2016. Using cross-docking operations in a reverse logistics network design: a new approach. J. Inst. Eng. Prod. 10, 175–184. https://doi.org/10.1007/s11740-015-0646-3.

Küçükoğlu, I., Öztürk, N., 2015. A hybrid meta-heuristic algorithm for vehicle routing and packing problem with cross-docking. J. Intell. Manuf. 30 (8), 2927–2943.

Küçükoğlu, I., Öztürk, N., 2017. Two-stage optimisation method for material flow and allocation management in cross-docking networks. Int. J. Prod. Res. 55 (2), 410–429.

Kuo, Y., 2013. Optimizing truck sequencing and truck dock assignment in a cross docking system. Expert Syst. Appl. 40 (14), 5532–5541.

Kusolpuchong, S., Chusap, K., Alhawari, O., Suer, G., 2019. A genetic algorithm approach for multi objective cross dock scheduling in supply chains. Procedia Manuf. 39, 1139–1148.

Larbi, R., Alpan, G., Baptiste, P., Penz, B., 2011. Scheduling cross docking operations under full, partial and no information on inbound arrivals. Comput. Oper. Res. 38 (6), 889–900.

Lee, Y.H., Jung, J.W., Lee, K.M., 2006. Vehicle routing scheduling for cross-docking in the supply chain. Comput. Ind. Eng. 51 (2), 247–256.

Liao, T., Egbelu, P., Chang, P., 2012. Two hybrid differential evolution algorithms for optimal inbound and outbound truck sequencing in cross docking operations. Appl. Soft Comput. 12 (11), 3683–3697.

Moghadam, S.S., Ghomi, S.F., Karimi, B., 2014. Vehicle routing scheduling problem with cross docking and split deliveries. Comput. Chem. Eng. 69, 98–107.

Mohtashami, A., 2015. Scheduling trucks in cross docking systems with temporary storage and repetitive pattern for shipping trucks. Appl. Soft Comput. 36, 468–486.

Mohtashami, A., Tavana, M., Santos-Arteaga, F.J., Fallahian-Najafabadi, A., 2015. A novel multi-objective meta-heuristic model for solving cross-docking scheduling problems. Appl. Soft Comput. 31, 30–47. https://doi.org/10.1016/j.asoc.2015.02.030.

Mokhtarinejad, M., Ahmadi, A., Karimi, B., Rahmati, S.H.A., 2015. A novel learning based approach for a new integrated location-routing and scheduling problem within cross-docking considering direct shipment. Appl. Soft Comput. 34, 274–285.

Mousavi, S.M., Tavakkoli-Moghaddam, R., 2013. A hybrid simulated annealing algorithm for location and routing scheduling problems with cross-docking in the supply chain. J. Manuf. Syst. 32 (2), 335–347.

Mousavi, S.M., Vahdani, B., 2017. A robust approach to multiple vehicle location-routing problems with time windows for optimization of cross-docking under uncertainty. J. Intell. Fuzzy Syst. 32 (1), 49–62.

Mousavi, S.M., Vahdani, B., Tavakkoli-Moghaddam, R., Hashemi, H., 2014. Location of cross-docking centers and vehicle routing scheduling under uncertainty: a fuzzy possibilistic–stochastic programming model. Appl. Math. Model. 38, 2249–2264. https://doi.org/10.1016/j.apm.2013.10.029.

Musa, R., Arnaout, J.-P., Jung, H., 2010. Ant colony optimization algorithm to solve for the transportation problem of cross-docking network. Comput. Ind. Eng. 59 (1), 85–92.

Rahbari, A., Nasiri, M.M., Werner, F., Musavi, M.M., Jolai, F., 2019. The vehicle routing and scheduling problem with cross-docking for perishable products under uncertainty: two robust bi-objective models. Appl. Math. Model. 70, 605–625.

Rezaei, S., Kheirkhah, A., 2018. A comprehensive approach in designing a sustainable closed-loop supply chain network using cross-docking operations. Comput. Math. Organ. Theor. 24 (1), 51–98.

Rezaei, S., Kheirkhah, A., 2017. Applying forward and reverse cross-docking in a multi-product integrated supply chain network. J. Inst. Eng. Prod. 11, 495–509. https://doi.org/10.1007/s11740-017-0743-6.

Santos, F.A., Mateus, G.R., Cunha, A.S.D., 2011. A novel column generation algorithm for the vehicle routing problem with cross-docking. Lecture Notes in Computer Science Network Optimization 412–425.

Santos, F.A., Mateus, G.R., Cunha, A.S.D., 2013. The pick-up and delivery problem with cross-docking. Comput. Oper. Res. 40 (4), 1085–1093.

Shahmardan, A., Sajadieh, M.S., 2020. Truck scheduling in a multi-door cross-docking center with partial unloading-Reinforcement learning-based simulated annealing approaches. Comput. Ind. Eng. 139, 106134.

Shen, Y., Xu, J., Li, J., 2016. A probabilistic model for vehicle scheduling based on stochastic trip times. Transp. Res. Part B Methodol. 85, 19–31.

Sung, C.S., Song, S.H., 2003. Integrated service network design for a cross-docking supply chain network. J. Oper. Res. Soc. 54 (12), 1283–1295.

Sung, C.S., Yang, W., 2008. An exact algorithm for a cross-docking supply chain network design problem. J. Oper. Res. Soc. 59 (1), 119–136.

Tadumadze, G., Boysen, N., Emde, S., Weidinger, F., 2019. Integrated truck and workforce scheduling to accelerate the unloading of trucks. Eur. J. Oper. Res. 278 (1), 343–362.

Theophilus, O., Dulebenets, M.A., Pasha, J., Abioye, O.F., Kavoosi, M., 2019. Truck scheduling at cross-docking terminals: a follow-up state-of-the-art review. Sustainability 11 (19), 5245.

Theophilus, O., Dulebenets, M.A., Pasha, J., Lau, Y.Y., Fathollahi-Fard, A.M., Mazaheri, A., 2021. Truck scheduling optimization at a cold-chain cross-docking terminal with product perishability considerations. Comput. Ind. Eng. 156, 107240.

Vahdani, B., Zandieh, M., 2010. Scheduling trucks in cross-docking systems: robust meta-heuristics. Comput. Ind. Eng. 58 (1), 12–24.

Wen, M., Larsen, J., Clausen, J., Cordeau, J.-F., Laporte, G., 2009. Vehicle routing with cross-docking. J. Oper. Res. Soc. 60 (12), 1708–1718.

Wisittipanich, W., Hengmeechai, P., 2017. Truck scheduling in multi-door cross docking terminal by modified particle swarm optimization. Comput. Ind. Eng. 113, 793–802.

Wisittipanich, W., Irohara, T., Hengmeechai, P., 2019. Truck scheduling problems in the cross docking network. Int. J. Logist. Syst. Manag. 33 (3), 420–439.

Xi, X., Changchun, L., Yuan, W., Hay, L.L., 2020. Two-stage conflict robust optimization models for cross-dock truck scheduling problem under uncertainty. Transport. Res. E Logist. Transport. Rev. 144 (2020), 102123.

Yin, P.-Y., Chuang, Y.-L., 2016. Adaptive memory artificial bee colony algorithm for green vehicle routing with cross-docking. Appl. Math. Model. 40 (21-22), 9302–9315.

Yu, V.F., Jewpanya, P., Redi, A.A.N.P., 2016. Open vehicle routing problem with cross-docking. Comput. Ind. Eng. 94, 6–17.

Yu, W., Egbelu, P.J., 2008. Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. Eur. J. Oper. Res. 184 (1), 377–396.