Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

Scheduled service network design with quality targets and stochastic travel times

Giacomo Lanza[a,*], Teodor Gabriel Crainic[b], Walter Rei[b], Nicoletta Ricciardi[c]

[a] *Department of Statistical Sciences, Sapienza University of Rome, Italy*
[b] *CIRRELT and Department of Management and Technology, Université du Québec à Montréal, Canada*
[c] *CIRRELT and Department of Statistical Sciences Sapienza University of Rome, Italy*

ARTICLE INFO

Article history:
Received 7 January 2019
Accepted 16 May 2020
Available online 23 May 2020

Keywords:
Transportation
Service network design
Stochastic travel time
Two-stage formulation
Progressive hedging meta-heuristic

ABSTRACT

We address the stochastic scheduled service network design problem with quality targets and uncertainty on travel times. This important problem, raising in the tactical planning process of consolidation-based freight carriers, has been little studied up to now. We define the problem considering quality targets for on-time operation of services and delivery of demand loads to destinations. We introduce a two-stage mixed-integer stochastic model defined over a space-time network, with quality targets modeled through penalties. We also propose an effective progressive-hedging-based meta-heuristic, based on a partial-decomposition concept aiming to address the challenges raised by the presence of flow-distribution decisions in the first-stage problem and by the flow-related degeneracy particular to network design. The results of an extensive numerical experimentation emphasize the worthiness of the formulation, as well as the very good performance of the proposed meta-heuristic when compared to a well-known commercial solver.

1. Introduction

Transportation plays a vital role in the social, political, and economic development of our society. It supports production, trade, consumption, and leisure activities, and ensures the movement of people and freight in a timely and efficient manner. The transportation industry, freight carriers in particular, operates in a highly competitive, cost and quality-of-service-driven environment. We focus on consolidation-based, long-haul freight transportation carriers, e.g., railways, liner-shipping companies, less-than-truckload motor carriers, etc., which operate more or less tightly scheduled services to answer demand. Consolidation-based carriers take advantage of economies of scale to lower costs and, thus, service prices, by moving the loads of different customers in the same vehicle or convoy, for all or part of their itineraries from origins to destinations. Consolidation-based carriers operate a set of regular scheduled transportation services between particular terminals in their network. The schedule covers a certain length of time and it is repeatedly executed for a given planning horizon, e.g., a weekly schedule repeated for the six months of a so-called season. Deciding on the services and schedule to operate for the next season is a rather complex tactical-planning problem that is traditionally addressed through a *scheduled service network design* (*SSND*) methodology.

The scope of SSND is to produce the set of scheduled services, together with planned routes for forecast regular demand (services used and terminals passed through), to achieve the economic and quality targets of the carrier. The latter reflects carrier's internal decisions concerning the actual performance to achieve during the planning horizon with the selected service network, and normally relates to the degree of violation acceptable with respect to what is externally proposed to the customers (schedule and due dates).

While there is quite a body of literature on SSND models for consolidation-based transportation (see, e.g., the survey of Crainic & Kim, 2007), few address quality target issues, and even fewer account for the fluctuations in travel times and the resulting delays and breaches, with monetary and possible market-loss consequences. In fact, according to our best knowledge, none has jointly addressed the design of an efficient service network and the consideration of travel time uncertainty impacting its reliability in terms of on-time service arrival and freight delivery. We aim to fill this gap by defining the *Stochastic Scheduled Service Network Design Problem with Quality Targets* (*SSND-QT*) and proposing a formulation to address this tactical planning problem while explicitly considering travel time uncertainty. The formulation takes the

[*] Corresponding author.
E-mail addresses: giacomo.lanza@di.unipi.it, giacomo.lanza@uniroma1.it (G. Lanza), teodorgabriel.crainic@cirrelt.net (T.G. Crainic), rei.walter@uqam.ca (W. Rei), nicoletta.ricciardi@uniroma1.it (N. Ricciardi).

https://doi.org/10.1016/j.ejor.2020.05.031
0377-2217/© 2020 Elsevier B.V. All rights reserved.

form of a two-stage mixed-integer linear stochastic model over a space-time network. The first stage addresses the selection of services and the routing of freight flows. Quality targets are modeled through penalties. Then, when travel time realizations become known, penalties are assigned to the appropriate services and deliveries.

Network design problems are NP-Hard and display high computational difficulty, stochastic ones even more so (Powell & Topaloglu, 2003). Thus, only small- to medium-sized instances can be solved optimally, and heuristic methods are generally required. We therefore propose an effective progressive hedging (PH)-based meta-heuristic. We believe this to be the first attempt to apply this methodology to a SSND problem with uncertainty on travel times. We present a tailored approach (compared to, e.g., the more traditional one in Crainic, Fu, Gendreau, Rei, & Wallace, 2011) to overcome challenges related to the quadratic reformulation and flow-related degeneracy that arise when PH is applied straightforwardly to the SSND-QT formulation. The numerical results of the experiments we conducted show the meta-heuristic is efficient and yields good-quality solutions.

The contributions of this paper therefore are: (1) Propose a new SSND problem setting where uncertainty in travel times is explicitly considered and specific attention is given to quality targets; (2) Provide a two-stage stochastic linear mixed-integer programming formulation for the SSND-QT where quality targets are modeled through penalties; (3) Develop a PH-based meta-heuristic able to effectively find good-quality solutions to larger-scale problem instances when compared to a well-known commercial solver; (4) Present the results of an extensive experimentation underscoring the relevance of the formulation when quality targets are of interest, as well as the performance of the proposed meta-heuristic compared to an exact solution method.

The paper is organized as follows. We recall the problem context and review the relevant yet sparse literature on stochastic SSND in Section 2. Section 3 then presents the problem setting we address and the modeling approach we propose. Section 4 introduces the notation and the mathematical formulation. We describe the meta-heuristic in Section 5. We present the experimental plan in Section 6, and analyse the results in Section 7. Conclusions and future research issues are discussed Section 8.

2. Context and literature review

Freight consolidation-based carriers operate over a physical network of uni or intermodal terminals connected by infrastructure (rail, road) or conceptual (navigation) links. They set up and exploit transportation services, according to given schedules, to satisfy the forecast regular demand in a cost and resource-utilization efficiently way. *Demand* is represented in terms of commodities, i.e., collections of similar products. Each commodity requires the transportation of a certain *quantity* of freight from a particular *origin* terminal, available at a certain *availability date*, to be delivered to a *destination* terminal by a *due date*. Additional physical attributes (e.g., weight, volume) and shipment requirements (e.g., type of vehicle) may characterize each commodity. The *service network* moves the demand flows. Each service is characterized by its *origin* and *destination* terminals, its *schedule*, i.e., the departure time at origin, the departure and arrival times at intermediate stops (if any), and the arrival time at destination, as well as a number of characteristics such as its capacity, speed and priority, etc. To take advantage of economies of scale, the loads of different demands are *consolidated*, loaded together, into the same vehicles, which, in some cases (e.g., rail, barge or road trains) are further consolidated into convoys. Freight may thus be moved by a sequence of services between its origin and destination, undergoing consolidation (accompanied possibly by loading/unloading) and service-to-service transfer operations at intermediate terminals. Terminal activities follow a strict schedule as well - coordinated with services arrival times - in order to efficiently perform loading, unloading, transshipment, and consolidation activities. Services arriving early need generally to wait and are processed when planned (e.g., ships waiting at sea at the port entrance); late arrived services instead are processed immediately to avoid increasing the delay any further.

The methodology of choice to design such a transportation plan represents the problem over a space-time network appropriate for the schedule length, made up of the set of all services that might potentially be operated by the carrier to address the demand. Each potential service is described based on its own characteristics (physical route, stops, schedule) as a path in the space-time network. The aim is to select those services that allow for the satisfaction of demand and meet the economic and quality targets chosen by the carrier. The corresponding Scheduled Service Network Design model then takes the form of a fixed-cost, capacitated, time-dependent network design formulation.

Surveys on Service Network Design issues and formulations can be found in Assad (1980),Crainic (1988),Cordeau, Toth, and Vigo (1998) for rail transportation, Christiansen, Fagerholt, and Ronen (2004) and Christiansen, Fagerholt, Nygreen, and Ronen (2007) for maritime transportation, Crainic and Roy (1988), Crainic (2000), and Crainic (2003) for land-based long-haul transportation, and Crainic and Kim (2007) for intermodal transportation. As shown in these surveys, as well as in the more recent literature, most of the research related to SSND considers deterministic problem settings. Uncertain factors such as stochastic demand and variability in travel times have been generally overlooked and addressed in recent years only.

Most of the studies on stochastic SSND focus on demand uncertainty. Thus, e.g., Lium, Crainic, and Wallace (2007), Lium, Crainic, and Wallace (2009), Bai, Wallace, Li, and Chong (2014), Jiang, Bai, Aickelin, and Landa-Silva (2017), and Wang, Crainic, and Wallace (2020) discuss two-stage formulations and address the question of what may be lost by not integrating information about the stochastic nature of demand directly into the tactical planning methodology, highlighting consolidation as a powerful mean to hedge against fluctuation.

Travel time uncertainty is very rarely addressed in the SSND literature. We are aware of very few contributions addressing the integration of travel time and the quality of the service performed. Crainic, Ferland, and Rousseau (1984), for rail and generalized by Crainic and Rousseau (1986) for consolidation-based carrier planning, propose path-based SND formulations with generalized-cost objective functions combining fixed and variable operating costs together with time-related costs. Time representations take the form of non-linear congestion functions for terminals and particular lines (e.g., rail tracks), and are transformed into costs through service and commodity-specific unit time costs. Service-quality measurements are considered through penalties on the variances of the service and demand total travel times. A similar approach considering congestion functions is also used in Crainic, Florian, Guélat, and Spiess (1990a) and Crainic, Florian, and Léal (1990b) for national/regional planning of multi-commodity multimodal freight transportation systems. These contributions assume deterministic travel times.

Among the recent contributions considering travel times uncertainty, Demir et al. (2016) focus on a multimodal consolidation-based transportation system in which motor-carrier transportation services must catch rail and maritime transportation services operating according to fixed schedules. The goal is to select motor-carrier services and route freight by minimizing costs and exploiting the given rail and maritime services as much as possible to reduce gas emissions. The authors propose a multi-objective SND formulation and consider a chance constraint approach to ensure, with a given probability, safe connections between the selected motor-carrier services and the available "green" rail and maritime services. A similar problem is addressed by Van Hui, Gao, Leung, and Wallace (2014), where additional costs arise when connections are missed. In the proposed two-stage job-assignment model, the first stage selects the motor-carrier services minimizing the operation costs plus the expected costs of adjusting the plan in the second stage, when delays to upcoming shipments are observed. The recourse in those cases is based on what the authors call "common industrial practice" in their application, that is, "forget the consolidation pertaining to the late shipments, release the on-time shipments according to the plan, and send the late shipments using the fastest available means". A different problem is considered in Wang and Meng (2012a) and Wang and Meng (2012b), who address the optimization of the sailing speed of a container-ship fleet to achieve target arrival times at a sequence of ports, while accounting for energy consumption. The ships should respect a given schedule, but late departure times caused by longer-than-planned port operations are possible and these affect the sailing time toward the subsequent port. The authors focus on operation time uncertainty and propose mixed-integer non-linear convex stochastic minimization models. A similar problem is considered in Song, Li, and Drake (2015), where it is addressed through a stochastic multi-objective optimization formulation.

This brief overview highlights the novelty of the uncertain travel time problem in consolidation-based transportation, as well as the need for further research in the area. We propose a more general approach with respect to the existing literature to address this challenge. We account for variations in travel times, their consequences on delays for services and demand flows, and the propagation of these delays over the network. We define two types of quality targets, for services and demand, which need to be considered separately, and account for their violation. To the best of our knowledge, these features have never been jointly considered in previous stochastic SSND contributions.

3. Problem statement and proposed modeling approach

We address a general scheduled service network design problem setting for consolidation-based freight carriers while explicitly accounting for the usual variability in travel times. The latter belongs to the so-called business-as-usual uncertainty type (e.g., Klibi, Martel, & Guitouni, 2010), that is, uncertainty one observes over the duration of the tactical planning horizon and that one needs to address when the plan is applied day after day during actual operations. Such variations in travel times and, possibly, the associated delays in operating services and delivering flows, occur even in the most tightly operated systems due to congestion conditions, adverse weather, and so on, and may jeopardize what the carrier announces and "promises" to customers, namely, on-time operations with respect to the service schedule and the demand due dates at destinations, with negative impacts on reputation and revenues. Our goal therefore is a SSND formulation that explicitly accounts for such possible day-by-day variations, and their consequences for the carrier, at the planning phase, yielding a regular transportation plan requiring less adjustments and costs during day-by-day operations when the actual variations become known.

As indicated in the previous section, the physical network supporting the operations of the carrier in tactical-planning modeling is represented by a network with nodes standing for the terminals and arcs modeling the connections between them the carrier may use. An estimation of the *usual travel time* (for each service type when appropriate) is associated to each link. The SSND formulation is then defined over a time-space network where the nodes are copies of the physical terminals at all periods of the schedule length, while the arcs are defined by the potential services the carrier might operate and among which the final service network is to be selected. Section 4 details the definition of the service arcs (or segments) as well as that of the holding arcs linking terminal nodes at consecutive time periods which are also included in the model. Each service is defined by its origin and destination terminals plus, possibly, a number of intermediate stops on its physical path among the two, as well as its *schedule*, called *usual* in the following. The latter is based on the usual travel times of the physical links and gives the arrival and departure times at each terminal on the route (capacity and other characteristics are also part of the definition as detailed in the next section).

We represent the business-as-usual variability in travel times through probability distributions associated to each physical link and service arcs. This information is then considered with respect to the quality targets of the carrier and integrated in the selection of the service network. We identify two major types of quality targets a carrier might want to consider in evaluating the quality of the proposed service network: 1) the *schedule target* related to the conformity to the scheduled arrival time at each stop for services, and 2) the *delivery target* defining the respect of the given delivery due dates for demand.

We address the basic version of the problem setting in this first modeling attempt, still retaining the main characteristics of stochastic SSND. All services are of the same type in terms of speed, priority, and capacity. Services may arrive early at a terminal but must wait, at no cost, until the scheduled time (e.g., ships at anchor at sea outside the port). Services may also arrive late, in which case terminal operations begin immediately (recall that service time at terminals is deterministic) and the departure time from that stop is modified to account for the late arrival and the service time. The on-time arrival time at the following stops may thus be compromised, unless the travel times on the subsequent links are lower than the usual ones. Delays for services are thus propagated through the service network, i.e., a delay on a service-link is propagated to the subsequent links and, thus, to the complete route (path) of the service. The problem setting considered does not include explicit penalty costs for commodities missing a transfer at a terminal due to service delays. We do, however, enforce "safe" connections through the penalties described below on late service arrival. Notice that late arrival of a service at some terminal is not important from the commodity point of view as long as it is delivered on time. The target penalties defined in the following address these issues.

Recall that tactical plans are built for a certain planning-horizon length, also called season of a few months to a year long, some time before the beginning of the season, to be used repeatedly each "day" of the season when adjustments to plans are performed within the limits of operational feasibility to reflect the new information that become available. Two-stage stochastic SSND formulations are then proposed to reflect this decision structure, the first stage defining the service network based on a forecast of uncertain parameters, the second being dedicated to adjusting the plan. We follow this approach in this paper, and need to address a number of interrelated questions: how we model targets, when the new information becomes available, and how we react when the realization of travel times is observed.

One way to model quality targets is through probabilistic conditions by exploiting travel time distributions. Thus, e.g., a schedule target may specify that each service must respect its planned arrival time at least with probability α, considering its occurrences during the planning horizon, while delays should not be greater than a pre-specified amount of time (e.g., a specific percentile of the travel time probability distribution or the expected value plus the standard deviation), with probability 1. Similarly, a delivery target may specify that each demand must arrive at its final destination respecting its due date at least with probability β. The issue when considering probabilistic constraints, however, is that they place hard restrictions on decisions, so that violations beyond the specific probabilistic threshold are not possible (Birge & Louveaux, 2011). The direct effect of this is that such formulations ignore some solutions which could be a good compromise between a violation, which comes at a cost, and an acceptable solution for the carrier. This flexibility is often recommended, if not required, when uncertainty must be faced at a tactical level and may be achieved by considering a soft reformulation of those conditions, as extensively discussed in King and Wallace (2012). This reformulation transforms hard into soft restrictions through a *penalty function*, which increases its value as the target is missed, becoming part of the objective, where the costs and benefits can be compared. We adopt this penalty-formulation approach in modeling quality targets and define two penalty terms. The first relates to the delays of each service at each of its stops, the second to the delay of each commodity at destination.

The *information revelation process* defines how and when the values of the stochastic parameters are observed. The arrival times of services and demand flows at their respective destinations, as well as at intermediate service stops, are the stochastic parameters in this study, their values being a direct result of the variations in the times required to travel along physical links. This implies that the delays incurred by services and demand flows, as well as the computation of the penalties corresponding to violated quality targets, may be observed only after services complete their movements, that is, at the second stage of decision making.

The aim of the proposed two-stage SSND-QT model is thus to select among the set of potential services that might be offered by the carrier, given their respective usual schedules, random travel times, and physical routes, those services, and consequent demand itineraries, that yield the lowest total cost, combining the cost of planned operations (first stage) and the additional expected penalty costs for violated quality targets (computed at the second stage).

The difference between the simple recourse of SSND-QT and the network recourse of most SSND models with stochastic demand may be worth noticing. In the latter case, the realization of demand takes place before transportation activities start and, thus, it makes sense to re-optimize, partially or completely, the demand distribution over the service network. This is not possible in our case, as information is revealed once transportation has been performed. Even when partial information is available (e.g., after the first service leg), re-routing is not possible or too costly in most cases when business-as-usual is addressed, as freight is loaded into vehicles which often are captive of the infrastructure (e.g., trains of rail tracks and ships of ports of call and navigation lines). Moreover, the lateness of a service at a stop does not necessarily imply that the transported demand will be late at its final destination as shorter travel times may occur in the rest of their itineraries. The simple recourse of paying the penalties is then the only possible recourse, which justifies addressing the issue in the tactical-planning SSND model to select services and flow itineraries that mitigate these costs. The model is introduced next.

4. Model formulation

The physical network on which the carrier operates is represented by $\mathcal{G}^P = (\mathcal{N}^P, \mathcal{A}^P)$, with node set $\mathcal{N}^P$ representing the physical terminals and arc set $\mathcal{A}^P$ representing the direct connections between terminals. Usual and random travel times are associated to each arc in $\mathcal{A}^P$.

We model the dynamics of the SSND-QT through a *space-time network*, $\mathcal{G} = (\mathcal{N}, \mathcal{A})$. We discretize the schedule length into T time periods of equal length through T+1 time instants. The set of physical nodes is replicated T+1 times, resulting in set $\mathcal{N}$. Demand is represented on $\mathcal{G}$ by the set of commodities $\mathcal{K}$, each commodity $k \in \mathcal{K}$ requiring the transport of a certain volume $w(k)$ from origin $o(k) \in \mathcal{N}$ to destination $d(k) \in \mathcal{N}$ according to its origin and destination terminals, as well as its entry, $a(k)$, and due, $b(k)$, dates. Notice that the latter is the *due date* specified in the understanding with the customer.

The set of arcs $\mathcal{A}$ is composed of the set of *holding* arcs $\mathcal{A}^H$ and the set of *moving* arcs $\mathcal{A}^M$. The former includes arcs between the representations of the same node in two consecutive periods and is used to model *idle time* at terminal for freight or *operation time* at terminal for services. The latter stands for the service legs of the potential services, that is, the movements of services between two different nodes in two different time periods.

Let $\mathcal{R}$ be the set of potential services that the carrier may use. Each service $r \in \mathcal{R}$ has a capacity u_r and a route in the physical network, specifying the set $\sigma(r) = \{z(r)_n \in \mathcal{N}_P,\ n = 1, \ldots, |\sigma(r)|\}$ of consecutive terminals visited between its origin $z(r)_1$ and destination $z(r)_{|\sigma(r)|}$ as well as the arcs in $\mathcal{A}$ used to perform the movement (Crainic, Errico, Rei, & Ricciardi, 2015). Timing information indicating the usual departure time $\zeta_{o(r)}$ at origin $o(r)\ (= z(r)_1)$, the usual arrival time $e_{d(r)}$ at destination $d(r)\ (= z(r)_{|\sigma(r)|})$, as well as the arrival/departure times at the other terminals visited in usual conditions without any delay are also defined. Combined with the timing information, the physical route yields the route the service performs in the space-time network, which we define as the sequence of *service legs* $\mathcal{L}(r) = \{l(r)\},\ \forall r \in \mathcal{R}$. The collection of all service legs of all potential services yields the set of moving arcs $\mathcal{A}^M = \bigcup_{r\in\mathcal{R}} \bigcup_{l\in\mathcal{L}(r)} l(r)$.

A usual travel time, $\hat{\tau}_{l(r)}$ and a random travel time, $\tau_{l(r)}$, are associated to each service leg $l(r) \in \mathcal{L}(r)$. A service leg has initial and terminal stops, z_l and z_{l+1} in $\sigma(r)$, and travels the corresponding path in the physical network. Consequently, its usual and random travel times are computed as the sum of the usual travel times and the convolution of the random travel times, respectively, of the physical arcs making up this path. The same time Δ is associated to terminal activities for all terminals and services.

A *fixed* selection (operation) cost f_r is associated to each potential service $r \in \mathcal{R}$. A unit commodity *transportation* cost $c^k_{l(r)}$ is also associated to each service leg $l(r) \in \mathcal{L}(r),\ r \in \mathcal{R}$ and commodity $k \in \mathcal{K}$. When appropriate, the arc-based notation, $c^k_{ij},\ (i,j) \in \mathcal{A}^{\mathcal{M}}$, is also used. The unit commodity handling cost at terminals is then given by $c^k_{ij},\ k \in \mathcal{K},\ (i,j) \in \mathcal{A}^{\mathcal{H}}$. Table 1 summarizes the parameters used to describe the model.

Let Ω define the set of possible outcomes of the arc travel time random variables and let ω be an element in that set. Two sets of first-stage decision variables are defined:

- $y_r \in \{0, 1\},\ r \in \mathcal{R}$, represent whether service r is selected ($y_r = 1$), or not ($y_r = 0$);
- $x^k_{ij} \geq 0,\ k \in \mathcal{K},\ (i,j) \in \mathcal{A}$, stand for the flow of commodities in the space-time network, moving, $(i,j) \in \mathcal{A}^M$, or waiting at a terminal, $(i,j) \in \mathcal{A}^H$.

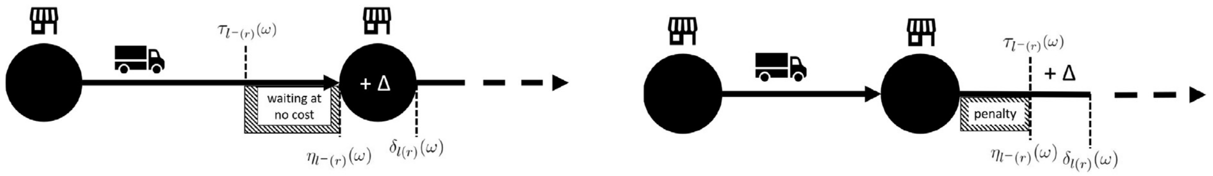
Three sets of second-stage variables are defined to model how to address the delays given a travel time realization $\omega \in \Omega$:

- $\delta_{l(r)}(\omega) \geq 0,\ l(r) \in \mathcal{L}(r),\ r \in \mathcal{R}$, represent the time instant at which service r begins to execute (move) its service leg $l(r)$ (obviously, it is the usual departure time $\zeta_{o(r)}$ for the first leg starting at $o(r)$);
- $\eta_{l(r)}(\omega) \geq 0,\ l(r) \in \mathcal{L}(r),\ r \in \mathcal{R}$, stand for the time instant at which service r ends its movement on service leg $l(r)$;
- $\varepsilon_k(\omega) \geq 0,\ k \in \mathcal{K}$, represent the time instant at which commodity k is delivered at its destination.

As already indicated, we express quality targets through a set of penalties on lateness and add those to the objective function.

Table 1
Parameters used to describe the model.

Sets	
Notation	Description
T	set of time periods in which the schedule length is discretized
$\mathcal{N}$	set of nodes of the space-time network defined as the set of physical terminals $\mathcal{N}^p$ replicated over T
$\mathcal{A}$	set of arcs of the space-time network defining movements or idle time
$\mathcal{K}$	set of commodities
$\mathcal{R}$	set of potential services
$\mathcal{L}(r)$	set of service legs of service $r \in \mathcal{R}$
Ω	set of possible travel time outcomes
$\mathcal{S}$	set of scenarios
Parameters	
Notation	Description
f_r	selection (operation) cost of service $r \in \mathcal{R}$
u_r	capacity of service $r \in \mathcal{R}$
$o(r)$	origin of service $r \in \mathcal{R}$
$\zeta_{o(r)}$	usual departure time at origin of service $r \in \mathcal{R}$
z_l	origin terminal of service leg $l(r) \in \mathcal{L}(r)$ of service $r \in \mathcal{R}$
$e_{l(r)}$	usual arrival time at destination terminal of service leg $l(r) \in \mathcal{L}(r)$ of service $r \in \mathcal{R}$
$\hat{\tau}_{l(r)}$	usual travel time associated to service leg $l(r) \in \mathcal{L}(r)$ of service $r \in \mathcal{R}$
$\tau_{l(r)s}$	travel time realization associated to service leg $l(r) \in \mathcal{L}(r)$ of service $r \in \mathcal{R}$ in scenario $s \in \mathcal{S}$
$\lambda^r_{l(r)}$	penalty cost for short delay for service leg $l(r) \in \mathcal{L}(r)$ of service $r \in \mathcal{R}$
$\Lambda^r_{l(r)}$	penalty cost for long delay for service leg $l(r) \in \mathcal{L}(r)$ of service $r \in \mathcal{R}$
B	maximum acceptable service delay
c^k_{ij}	transportation or handling cost of commodity $k \in \mathcal{K}$ associated to arc $(i, j) \in \mathcal{A}$
$w(k)$	volume of commodity $k \in \mathcal{K}$
$o(k)$	origin of commodity $k \in \mathcal{K}$
$d(k)$	destination of commodity $k \in \mathcal{K}$
$b(k)$	due date of commodity $k \in \mathcal{K}$
λ^k	penalty cost for short delay of commodity $k \in \mathcal{K}$ at destination
Λ^k	penalty cost for long delay of commodity $k \in \mathcal{K}$ at destination
B^k	maximum acceptable delivery delay of commodity $k \in \mathcal{K}$ at destination
Δ	service time for terminal activities



(a) An early arriving service waits until its usual arrival time at terminal to start terminal operations

(b) A late arriving service starts terminal operations immediately

Fig. 1. Examples of arrival and departure time to and from a terminal.

Lateness of service is considered as soon as the observed arrival time at a stop exceeds the usual arrival time for that stop. Thus, for each service leg, we compute

$$\eta_{l(r)}(\omega) = \delta_{l(r)}(\omega) + \max\left(\hat{\tau}_{l(r)}, \tau_{l(r)}(\omega)\right) \qquad \forall\, l(r) \in \mathcal{L}(r) \qquad (1)$$

and

$$\delta_{l(r)}(\omega) = \begin{cases} \zeta_{o(r)} & \text{when } z_l = o(r) \\ \eta_{l^-(r)}(\omega) + \Delta & \text{when } z_l \neq o(r) \end{cases} \qquad \forall\, l(r) \in \mathcal{L}(r), \forall\, r \in \mathcal{R} \quad (2)$$

where $\tau_{l(r)}(\omega)$ is the travel time realization of service leg $l(r)$. Thus, when the service arrives earlier than planned, it must wait for the terminal operations and the usual travel time is considered (as depicted in Fig. 1a). When late, terminal operations start on arrival and the next leg starts once these are completed (as depicted in Fig. 1b). Expression (2) is also directly involved in the calculation of the delay propagation of the service r. A $\lambda^r_{l(r)}$ penalty proportional to the $\eta_{l(r)}(\omega) - e_{l(r)}$ difference is then applied. A similar approach is applied to the maximum acceptable delay B, the unit penalty cost $\Lambda^r_{l(r)}$ being applied to the difference $\eta_{l(r)}(\omega)$ and B. The same logic is followed to model late deliveries of demand flows, i.e., when $\varepsilon_k(\omega) > b(k),\ k \in \mathcal{K}$, with unit penalty cost λ^k, and the maximum acceptable delivery delay, B^k, with unit penalty cost Λ^k applied to the $\varepsilon_k(\omega) - B^k$ difference.

We approximate uncertainty through a set $\mathcal{S} = \{s\}$ of scenarios, each of which has dimension $|A^P|$. A probability p_s is assigned to each scenario, with $0 \leq p_s \leq 1,\ \forall s \in \mathcal{S}$ and $\sum_{s \in \mathcal{S}} p_s = 1$. Instantiating the previous decision variables and problem parameters for each scenario, the two-stage formulation of the SSND-QT may be written as:

$$\min \sum_{r \in \mathcal{R}} f_r y_r + \sum_{(i,j) \in \mathcal{A}} \sum_{k \in \mathcal{K}} c^k_{ij} x^k_{ij} + \sum_{s \in \mathcal{S}} p_s [Q(y, x; \tau_{l(r)}(s))] \qquad (3)$$

$$\sum_{j \in \mathcal{N}^+(i)} x^k_{ij} - \sum_{j \in \mathcal{N}^-(i)} x^k_{ji} = \begin{cases} w(k) & \text{if } i = o(k), \\ 0 & \text{if } i \neq o(k),\ i \neq d(k), \\ -w(k) & \text{if } i = d(k), \end{cases} \qquad \forall\, i \in \mathcal{N},\ \forall\, k \in \mathcal{K}, \qquad (4)$$

$$\sum_{k\in\mathcal{K}} x^k_{l(r)} \le u_r y_r \qquad \forall\, l(r) \in \mathcal{L}(r),\ \forall\, r \in \mathcal{R}, \tag{5}$$

$$x^k_{ij} \ge 0 \qquad \forall\, k \in \mathcal{K},\ \forall\, (i,j) \in \mathcal{A}, \tag{6}$$

$$y_r \in \{0,1\} \qquad \forall\, r \in \mathcal{R}, \tag{7}$$

where

$$\begin{aligned} Q(y,x;\tau_{l(r)}(s)) = & \sum_{r\in R}\ \sum_{l(r)\in L(r)} \lambda^r_{l(r)}(\eta_{l(r)s} - e_{l(r)})_+ \\ & + \sum_{r\in R}\ \sum_{l(r)\in L(r)} \Lambda^r_{l(r)}(\eta_{l(r)s} - B)_+ \\ & + \sum_{k\in K} \lambda^k(\varepsilon_{ks} - b(k))_+ + \sum_{k\in K} \Lambda^k(\varepsilon_{ks} - B^k)_+ \end{aligned} \tag{8}$$

and variables $\delta_{l(r)s}$, $l(r) \in \mathcal{L}(r), r \in \mathcal{R}$ and $\eta_{l(r)s}$, $l(r) \in \mathcal{L}(r), r \in \mathcal{R}$, are computed similarly to (1) and (2), respectively.

The objective (3) is to minimize the total cost of the system, including the fixed cost of selecting and operating services, the transportation costs for routing commodities, and the expected cost of recourse for applying the chosen plan when travel delays are observed. The second-stage function $Q(y,\ x;\ \tau_{l(r)}(s))$ depends on both design and routing decisions, as well as on the realizations of the random variables $\tau_{l(r)}$ expressed through the scenarios. Then, in the second-stage, (8) computes the total penalty costs of service and demand late arrivals, where the first two terms relate to schedule targets and the last two to delivery targets. The operator $(x-y)_+$ returns the difference between x and y if positive and 0 otherwise. Eq. (4) represent the commodity flow conservation constraints. Relations (5) represent the linking-capacity constraints, which state that the flow can use selected services only and that the total commodity flow on any service leg cannot exceed the capacity of the corresponding service. Relations (6) and (7) represent the non-negativity and binary constraints that define the domains of the decision variables.

The model includes the two quality targets considered. It can be easily modified to focus on one target type only. The solution method presented in the next section applies to all cases.

5. A Progressive Hedging-Based Meta-heuristic

The meta-heuristic we propose is based on the concepts of scenario-decomposition and *progressive-hedging* (Rockafellar & Wets, 1991). The idea is to call on augmented Lagrangian techniques to decompose the original multi-scenario problem into single-scenario-based sub-problems, iteratively solve those, build an aggregated solution out of the sub-problem ones, and, for as long as the sub-problem solutions do not agree (i.e., do not produce identical solutions), adjust the decomposition parameters to “guide” the method toward a “well hedged” solution to the original problem.

Convergence to the global optimum has been proven for continuous stochastic programs (Rockafellar, 1982). No such result exists in the integer case and, thus, only PH-based meta-heuristics are proposed for integer formulations. The method has been proven to be computationally efficient, however, for a wide range of problem settings, such as, operation planning (Gonçalves, Finardi, & da Silva, 2012), lot-sizing (Haugen, Løkketangen, & Woodruff, 2001), portfolio management (Mulvey & Vladimirou, 1991), unit commitment and server location (Gade et al., 2016; Guo, Hackebeil, Ryan, Watson, & Woodruff, 2015), scheduling (Carpentier, Gendreau, & Bastin, 2013), resource allocation (Watson & Woodruff, 2011), capacity planning and stochastic bin packing (Crainic, Gobbato, Perboli, & Rei, 2016), network design (Fan & Liu, 2010; Hvattum & Løkketangen, 2009; Mulvey & Vladimirou, 1991), and SND with demand uncertainty (Crainic et al., 2011; Crainic, Hewitt, & Rei, 2014a; Jiang et al., 2017).

A few important algorithmic issues must be carefully addressed when considering a PH-based meta-heuristic and are developed in the following sections:

- The *decomposition strategy*, which separates the formulation and defines sub-problems according to the given scenario set (Section 5.1);
- The *aggregation process* to synthesize an estimated global solution out of the local solutions provided by the sub-problems (Section 5.2);
- The *search of consensus* strategy, which takes advantage of the global and local information to modify the settings of the sub-problems and, thus, hopefully guide the meta-heuristic toward a solution to the original problem corresponding to a consensus among the scenario sub-problems (Section 5.3);
- The general algorithmic design of the meta-heuristic (Section 5.4).

5.1. Decomposition strategy

The formulation of the SSND-QT involves both design and routing variables as first-stage decisions, which is different from most previous applications to network design where demand was assumed stochastic and routing was decided in the second stage (e.g., Crainic et al., 2011). A standard application of the PH approach to the present case yields difficult non-linear sub-problem formulations, as well as raising issues regarding the consensus-seeking strategy. We thus introduce a new methodological framework for PH-based meta-heuristics aimed at network design formulations with design and flow variables in the first stage.

Following the standard approach (detailed in Annex 2), first-stage decision variables are defined in each scenario, and non-anticipativity constraints (10)–(13) are added to ensure all scenarios obtain the same design and routing solution. This yields

$$\begin{aligned} \min \sum_{s\in\mathcal{S}} p_s \Bigg(& \sum_{r\in\mathcal{R}} f_r y_{rs} + \sum_{(i,j)\in\mathcal{A}}\ \sum_{k\in\mathcal{K}} c^k_{ij} x^k_{ijs} \\ & + \sum_{r\in\mathcal{R}}\ \sum_{l(r)\in\mathcal{L}(r)} \lambda^r_{l(r)}(\eta_{l(r)s} - e_{l(r)})_+ + \sum_{r\in\mathcal{R}}\ \sum_{l(r)\in\mathcal{L}(r)} \Lambda^r_{l(r)}(\eta_{l(r)s} - B)_+ \\ & + \sum_{k\in\mathcal{K}} \lambda^k(\varepsilon_{ks} - b(k))_+ + \sum_{k\in\mathcal{K}} \Lambda^k(\varepsilon_{ks} - B^k)_+ \Bigg) \end{aligned} \tag{9}$$

subject to constraints (4)–(7) written for all scenarios $s \in \mathcal{S}$, and the non-anticipativity constraints

$$y_{rs} = \bar{y}_r \qquad \forall\, r \in \mathcal{R},\ \forall\, s \in \mathcal{S}, \tag{10}$$

$$x^k_{ijs} = \bar{x}^k_{ij} \qquad \forall\, k \in \mathcal{K},\ \forall\, (i,j) \in \mathcal{A},\ \forall\, s \in \mathcal{S}, \tag{11}$$

$$\bar{y}_r \in \{0,1\} \qquad \forall\, r \in \mathcal{R}, \tag{12}$$

$$\bar{x}^k_{ij} \ge 0 \qquad \forall\, k \in \mathcal{K},\ \forall\, (i,j) \in \mathcal{A}, \tag{13}$$

where $\bar{y}_r$ and $\bar{x}^k_{ij}$ are the overall design and routing solutions, respectively.

Let ϕ_{rs} and ϕ^k_{ijs} be the Lagrangian multipliers associated to constraints (10) and (11), respectively, and let ρ and ψ be the penalty factors associated to the respective quadratic terms of augmented Lagrangian relaxation. Following straightforward manipulations, and taking advantage of the binary nature of the service design variables, the objective function becomes

$$\begin{aligned}\min \sum_{s\in\mathcal{S}} p_s \Bigg(& \sum_{r\in\mathcal{R}} (f_r + \phi_{rs} + \frac{1}{2}\rho - \rho\bar{y}_r) y_{rs} \\ & + \sum_{(i,j)\in A} \sum_{k\in\mathcal{K}} (c^k_{ij} + \phi^k_{ijs} + \frac{1}{2}\psi x^k_{ijs} - \psi \bar{x}^k_{ij}) x^k_{ijs} \\ & + \sum_{r\in\mathcal{R}} \sum_{l(r)\in\mathcal{L}(r)} \lambda^r_{l(r)} (\eta_{l(r)s} - e_{l(r)})_+ + \sum_{r\in\mathcal{R}} \sum_{l(r)\in\mathcal{L}(r)} \Lambda^r_{l(r)} (\eta_{l(r)s} - B)_+ \\ & + \sum_{k\in\mathcal{K}} \lambda^k (\varepsilon_{ks} - b(k))_+ + \sum_{k\in\mathcal{K}} \Lambda^k (\varepsilon_{ks} - B^k)_+ \Bigg)\end{aligned} \tag{14}$$

The resulting formulation is scenario separable. Yet, it also raises significant questions. First, sub-problems are non-linear, which complicates addressing an already computationally difficult NP-Hard problem. Second, it requires consensus on multi-commodity flows. This is troubling, as the well-known nature of multi-commodity network design problems regarding the degeneracy of their flow distributions would make reaching such a consensus almost impossible. Even assuming consensus is reached for design variables, several equivalent flow path solutions could still exist and force the algorithm to continue.

We therefore introduce a partial scenario decomposition, based on a "hierarchy of importance" among first-stage variables. According to this decomposition strategy, the non-anticipativity constraints on the design variables are kept and relaxed as before through the augmented Lagrangian mechanism, while, on the other hand, the flow non-anticipativity constraints are discarded. Consequently, consensus is explicitly sought by the meta-heuristic on the design variables, while it is only indirectly looked for in the flows. The partial scenario decomposition applied to the SSND-QT formulation then yields the following sub-problem formulations:

$$\begin{aligned}\min \sum_{r\in\mathcal{R}} & \Big(f_r + \phi_{rs} + \frac{1}{2}\rho - \rho\bar{y}_r\Big) y_{rs} + \sum_{(i,j)\in\mathcal{A}} \sum_{k\in\mathcal{K}} c^k_{ij} x^k_{ijs} \\ & + \sum_{r\in\mathcal{R}} \sum_{l(r)\in\mathcal{L}(r)} \lambda^r_{l(r)} (\eta_{l(r)s} - e_{l(r)})_+ + \sum_{r\in\mathcal{R}} \sum_{l(r)\in\mathcal{L}(r)} \Lambda^r_{l(r)} (\eta_{l(r)s} - B)_+ \\ & + \sum_{k\in\mathcal{K}} \lambda^k (\varepsilon_{ks} - b(k))_+ + \sum_{k\in\mathcal{K}} \Lambda^k (\varepsilon_{ks} - B^k)_+\end{aligned} \tag{15}$$

Subject to

$$\sum_{j\in\mathcal{N}^+(i)} x^k_{ijs} - \sum_{j\in\mathcal{N}^-(i)} x^k_{jis} = \begin{cases} w(k) & \text{if } i = o(k) \\ 0 & \text{if } i \neq o(k),\ i \neq d(k) \\ -w(k) & \text{if } i = d(k) \end{cases} \qquad \forall i \in \mathcal{N},\ \forall k \in \mathcal{K},\ \forall s \in \mathcal{S}, \tag{16}$$

$$\sum_{k\in\mathcal{K}} x^k_{l(r)s} \leq u_r y_{rs} \qquad \forall l(r) \in \mathcal{L}(r),\ \forall r \in \mathcal{R},\ \forall s \in \mathcal{S}, \tag{17}$$

$$x^k_{ijs} \geq 0 \qquad \forall k \in \mathcal{K},\ \forall (i,j) \in \mathcal{A},\ \forall s \in \mathcal{S}, \tag{18}$$

$$y_{rs} \in \{0, 1\} \qquad \forall r \in \mathcal{R},\ \forall s \in \mathcal{S}. \tag{19}$$

In this paper, sub-problems are solved separately, yielding two vectors containing respectively the design, i.e., the selected services, and the corresponding flow distribution, i.e., the commodity flows on the service legs. This is used to build an aggregated design for the complete problem and to adjust multipliers and penalty factors to guide the meta-heuristic, as detailed in the following sections.

5.2. Solution aggregation

The overall design summarizes in a single design vector $\bar{y}^\nu$ the information obtained by solving the sub-problems at a given iteration ν. It represents both an estimation of global trends and a reference point to "guide" the sub-problem solutions, iteration after iteration, toward consensus. A weighted average function is used as aggregation operator (as in, e.g., Crainic et al., 2011; Rockafellar & Wets, 1991) yielding

$$\bar{y}^\nu_r = \sum_{s\in S} p_s y^\nu_{rs} \qquad \forall r \in \mathcal{R} \tag{20}$$

The values of $\bar{y}^\nu_r$ are between 0 and 1. When all scenarios agree on the selection status of a service r, consensus is observed, and $\bar{y}^\nu_r$ equals 1 when all scenarios agree on activating service r, or 0 when all scenarios agree on non-selecting it. Most of the time, however, one observes that $0 < \bar{y}^\nu_r < 1$. We assume in this case that a high (close to 1) or a low (close to 0) value for $\bar{y}^\nu_r$ indicates a trend toward selecting or not selecting the service, respectively, and use this information in the adjustment of parameters to guide the meta-heuristic.

5.3. Search for consensus

The local information provided by solving the sub-problems together with the aggregated reference solution are used to modify the values of the Lagrangian parameters and the fixed service costs within each sub-problem, to guide the solutions of the sub-problems toward reaching a consensus. The "distance" of local solutions to the reference one is an important criterion in the adjustment strategies, which aim to induce selection or non-selection of a service when its status is different from that in the current overall design.

We propose two adjustment strategies. The first follows the literature and makes use of the design information only. The second adds the flow information produced by the sub-problems to the criteria used to locally adjust the service fixed costs.

Both strategies start with the update of the augmented Lagrangian multipliers and parameter ρ, considering the current reference design $\bar{y}^\nu$ and the designs y^ν_s provided by the sub-problems $s \in \mathcal{S}$ (Løkketangen & Woodruff, 1996; Rockafellar & Wets, 1991).

Three cases may occur when considering any design variable y^ν_{rs}, which is either 1 or 0 in scenario sub-problem s at iteration ν:

- $y^\nu_{rs} < \bar{y}^\nu_r$. Service r is not selected in scenario s, but the trend among the other scenarios, as reflected by the reference point $\bar{y}^\nu_r$, is opposite. The goal is, then, to promote the activation of service r by reducing its cost in the sub-problem s;
- $y^\nu_{rs} > \bar{y}^\nu_r$. The opposite situation, when service r is selected in scenario s, but most other scenarios do not agree on this decision. The cost is then adjusted so as to dissuade selecting service r in the sub-problem s; Notice, that, the higher the number of other sub-problems not agreeing to select the service, the stronger the disincentive;
- $y^\nu_{rs} = \bar{y}^\nu_r$. Total consensus is observed among all sub-problems and the fixed costs remain unchanged.

The parameters are then updated as follows:

$$\phi^\nu_{rs} \leftarrow \phi^{\nu-1}_{rs} + \rho^{\nu-1}(y^\nu_{rs} - \bar{y}^{\nu-1}_r) \qquad \forall r \in R \tag{21}$$

$$\rho^\nu \leftarrow \gamma \rho^{\nu-1} \tag{22}$$

Update rule (21) represents the steepest ascent step in the space of the dual problem (Mulvey & Vladimirou, 1991) and depends on parameter ρ^ν. Initially, ρ^0 is set to an arbitrarily positive small value and is dynamically adjusted at each iteration through a given parameter $\gamma > 1$. As seen in the literature, the adjustment should be smooth. Although dynamic adjustments of the penalty parameter are not covered by the convergence theory for the PH algorithm, Mulvey and Vladimirou (1991) found that this strategy can improve the overall convergence behavior.

Fixed costs are adjusted at each iteration. According to the usual approach, this adjustment is global and targets the design arcs with a status different from what a majority of arcs seems to agree upon at the current iteration (Crainic et al., 2011). As above, the trend is given by the value of the current reference solution $\bar{y}_r^\nu,\ r \in \mathcal{R}$, a high/low value of $\bar{y}_r^\nu$ indicating a trend to select/not select service r. As discussed in Crainic et al. (2011), we assume that when $\bar{y}_r^\nu$ is less than a given threshold $thres_{low}$, increasing its cost f_r will result in it becoming less interesting in the sub-problems, resulting eventually in its non-selection. Symmetrically, when $\bar{y}_r^\nu$ is higher than a given threshold $thres_{high}$, lowering the cost f_r of service r will make it more attractive and lead to its selection.

The first strategy then adjusts the fixed costs according to Crainic et al. (2011):

$$f_r^\nu = \begin{cases} \beta f_r^{\nu-1} & \text{if } \bar{y}_r^{\nu-1} < thres_{low} \\ \dfrac{1}{\beta} f_r^{\nu-1} & \text{if } \bar{y}_r^{\nu-1} > thres_{high} \\ f_r^{\nu-1} & \text{otherwise} \end{cases} \tag{23}$$

with f_r^ν representing the modified selection cost of service r at iteration ν, and parameters $\beta > 1,\ 0 < thres_{low} < 0.5$, and $0.5 < thres_{high} < 1$.

The second adjustment strategy we propose extends the previous one by adding a local adjustment to the global one. The local adjustment is particular to each sub-problem and is based on the respective flow distribution at the current iteration. Let $X^\nu(r) = \sum_{l(r) \in \mathcal{L}(r)} \sum_{k \in \mathcal{K}} x_{k(r)s}^{k\nu}$ be the total flow moved by service r in the solution of sub-problem s at iteration ν. Let's define two threshold parameters $0 < local_{low} < u_r/2$ and $u_r/2 < local_{high} < u_r$ to qualify the local flow distribution. Four occurrences may be observed at iteration ν when contrasting the global and local trend information. The local adjustment rule stated by relations (24) - (27) addresses the first three.

$$\begin{cases} f_{rs}^\nu = \dfrac{1}{\beta^2} f_r^\nu & if \bar{y}_r^\nu > thres_{high} and X^\nu(r) > local_{high} \qquad (24) \\ \dfrac{1}{\beta} f_r^\nu & if \bar{y}_r^\nu > thres_{high} and X^\nu(r) < local_{low} \qquad (25) \\ \beta f_r^\nu & if \bar{y}_r^\nu < thres_{low} and X^\nu(r) < local_{low} \qquad (26) \\ f_r^\nu & if \bar{y}_r^{\nu-1} = 0.5 \qquad (27) \end{cases}$$

Cases (24) and (26) define the two situations when the global and the local information agree to select or not, respectively, service r. The local adjustments we propose thus reinforce the global information favoring/discouraging the selection of the service by lowering/increasing its fixed cost. In the case (25), the local trend is to make a rather limited use of a service for which a strong selection trend exists at the global level. One can interpret the local decision as a "safety" selection to hedge against uncertainty and use for excess flow one cannot move on the most sought-after services. We therefore reinforce the global trend but less strongly compared to case (24). Finally, the fixed cost is not changed when there is a fifty-fifty global split on a particular service.

The global and local trends clash in the forth case

$$\bar{y}_r^\nu < thres_{low} \quad \text{and} \quad X^\nu(r) > local_{high}, \tag{28}$$

a majority of sub-problems avoiding certain services, which are heavily favored locally. The question here is "how important these services may be with respect to the final design?", that is, to what extent a rather small subset of highly used services in some sub-problems should influence the final plan? The adjustment strategy based on global information only simply ignores these services. We propose to account for this information but to try to limit its impact, by limiting the number of services that belong to condition (28).

Let $C_s^\nu = \{r \in \mathcal{R} : \bar{y}_r^\nu < tres_{low}$ and $\sum_{k \in K} x_{l(r)s}^{k\nu} > local_{high}^k\}$ for each sub-problem s at iteration ν. Let $h_s^{\nu-1} = |C_s^{\nu-1}|$ be the number of services satisfying condition (28) at iteration $\nu - 1$ for sub-problem s. Also, let the parameter $a_{rs}^\nu,\ \forall r \in \mathcal{R},\ \forall s \in \mathcal{S}$, to assume value 1 when $r \in C_s^\nu$ and 0 otherwise. Then, we add constraint (29) to sub-problem s at iteration ν and solve the new formulation (after all other usual Lagrangian parameters and fixed costs modifications). If a solution is found, it indicates that the parameter and cost updates allowed to switch flow to other services. When a solution is not found, we assume the services in $C_s^{\nu-1}$ are necessary to find a feasible solution for sub-problem s and are therefore kept in subsequent iterations.

$$\sum_{r \in \mathcal{R}} a_{rs}^\nu y_{rs} \le h_s^{\nu-1} - 1 \tag{29}$$

5.4. The PH-based meta-heuristic for the SSND-QT

We address a SND and face the challenges of "converging" or, more precisely, of achieving consensus for all design variables within an acceptable computing time (Crainic et al., 2011). We thus build a two-phase meta-heuristic, schematically illustrated in Algorithm 1. The first phase executes the PH-based search, with the algorithmic components defined in the previous sections, for a certain number of iterations. As illustrated in Algorithm 1, the procedure solves the sub-problems with the current fixed costs, builds the reference solution out of the sub-problem solutions, updates the augmented Lagrangian multipliers and penalty term ρ, adjusts

Algorithm 1 Hierarchic progressive hedging-based meta-heuristic.

```
Initialization $\nu = 0$
$\phi_{rs}^\nu \leftarrow 0, \forall r \in \mathcal{R}, \forall s \in \mathcal{S}$;
$\rho^\nu \leftarrow \rho^0$;
$f_{rs}^\nu \leftarrow f_r, \forall r \in \mathcal{R}, \forall s \in \mathcal{S}$;
Solve the corresponding $|S|$ SSND-QT sub-problems;
First Phase:
while stopping criterion is not met do
    $\bar{y}_r^\nu \leftarrow \sum_{s \in S} p_s y_{rs}^\nu, \forall r \in \mathcal{R}$;
    Adjust globally $f_r^\nu, \forall r \in \mathcal{R}$, using Eq. (23);
    for $s = 1, \ldots, |S|$ do
        if PHRL = TRUE then
            Adjust locally $f_{rs}^\nu, \forall r \in \mathcal{R}$ using the rule (24) - (27);
            $h_s^\nu \leftarrow |C_s^\nu|$;
            Add constraint (29) with $h_s^{\nu-1}$;
        end if
        Solve the modified SSND-QT sub-problem $s$;
        if PHRL = TRUE then
            while Solve == TRUE do
                $h_s^{\nu-1}$ ++;
                Solve the modified SSND-QT sub-problem $s$;
            end while
        end if
    end for
    $\nu \leftarrow \nu + 1$;
    Update:
$\phi_{rs}^\nu \leftarrow \phi_{rs}^{\nu-1} + \rho^{\nu-1}(y_{rs}^\nu - \bar{y}_r^{\nu-1}), \forall r \in \mathcal{R}, \forall s \in \mathcal{S}$;
$\rho^\nu \leftarrow \gamma \rho^{\nu-1}$;
end while
Second Phase:
Fix the design variables for which consensus is obtained;
Solve the restricted multi-scenario formulation.
```

the fixed costs, and iterates. We identify the first adjustment strategy, based on the reference solution only, as *PHR*, and *PHRL* the strategy performing local adjustments as well. At the initialization step, $\nu = 0$ and the service fixed costs are the original ones. In the current implementation, sub-problems are solved optimally by using a commercial solver.

The goal of the first phase is to identify a subset of arcs for which “consensus” appears possible. Phase one stops on one of a number of criteria, e.g., time limit, maximum number of iterations, a minimum percentage of arcs for which consensus is reached. The second phase of the meta-heuristic is then called upon to compute the complete solution to the original problem, by fixing the design variables for which consensus has been reached and solving the reduced formulation. When the first phase yields a consensual design, i.e., all sub-problems agree, this corresponds to solving the capacitated multi-commodity network flow problem. In most cases, however, consensus is not reached, and the corresponding reduced network design formulation is solved.

6. Experimental plan

We performed two sets of experiments. The first aimed to quantify the benefits of explicitly considering stochastic travel time in the SSND model. This experiment therefore involved solving exactly the stochastic and deterministic versions of the SSND-QT formulation for small- and medium-sized problem instances only, but with diverse characteristics in terms of level of variability, number of commodities, size of delivery time windows, and penalty costs. The results were then evaluated through a Monte Carlo simulation and compared. In the same set of experiments, we also investigated the impact of the penalty parameter on the stochastic solutions, in particular, on the reliability of the resulting service networks. Problem instances were thus solved for increasing values of penalties, keeping all other parameters fixed. The second set of experiments focused on qualifying the performance of the proposed meta-heuristic. A subset of instances used in the previous experiment were addressed with both PHR and PHRL approaches, and the solutions were compared to quantify the gain of using local flow information. Meta-heuristic results were then compared to the exact solutions obtained by solving the multi-scenario formulations with CPLEX.

Deterministic and stochastic mixed-integer linear programming models, as well as the meta-heuristic, were implemented using the OPL language. All experiments were conducted on an Intel Xeon X5675 computer with 3.07 GHz and 48 Gigabyte of RAM. The CPLEX 12.6 solver (IBM ILOG, 2016) was used to implement the solution methods applied.

Before detailing the results of the sets of experiments, we present the instance generation and the stability analysis we performed.

6.1. Instances and scenario generation

Performance was assessed by comparing the meta-heuristic results to the optimal results provided by CPLEX. Consequently, for most of our experiments, the instances had to be of dimensions that could be tackled by the software. Somewhat larger instances were also generated. The physical service network we consider in all our experiments is inspired by that used in Crainic, Hewitt, Toulouse, and Vu (2014b), which consists of 5 physical nodes and 10 physical arcs and is shown in Fig. 2a. The service network is defined for a schedule length of 15 periods and displays a cyclic nature (Crainic et al., 2014b), as illustrated in Fig. 2b.

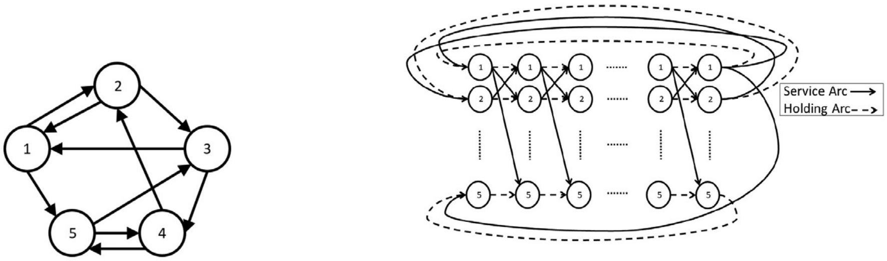
We consider 8 *demand classes* defined by the number of commodities and the time available to deliver them. We used four *levels of demand*, from 1, corresponding to a low (15) number of commodities, to 2 and 3 for medium sizes (20 and 25, respectively), and to 4 for a relatively high number of commodities (50). Two values were considered for the *delivery-time windows*, loose (*l*), with due dates between 11 and 14 periods after the availability date (given a schedule length of 15 periods), and tight (*t*) with due dates between 9 and 12 periods after the availability dates. In the tables of this and the following sections, the demand classes are represented by *DClass*(· ·) with the respective values for these two attributes. The potential service network is the same for all instances and includes 150 direct services and 7 services with one intermediate stop. We generated services with usual duration of 3, 4 or 5 periods (the usual duration of a service leg was a smaller but integer number of periods). The fixed cost of a direct service is proportional to its usual duration. The fixed cost of a service with intermediary stop is 35% less than the cost of the two direct services one would need for the same path.

We aimed for a set of test instances that provides the means to assess the impact of travel time variability on the performance of the model and algorithm, as well as on the resulting service networks. We thus selected a class of probability distributions for the random event that allowed us to control the main elements defining the travel times, usual value, variability, and ranges, defined as the difference between the maximum and minimum travel times possible on the arcs of the network.

In our case, arc travel times do have a minimum value, corresponding to the free running time of a single service under perfect conditions, and a maximum value since we do not address neither major disturbances nor highly hazardous and catastrophic events. Given that, in practice, delays occur much more frequently than early arrivals, we aimed for a probability distribution that increases rapidly to the value of the usual travel time, followed by a gradual decrease until the maximum travel time value (i.e., the tail should be skewed to the right). We thus selected the class of *Truncated Gamma* (*TG*) probability distributions (Chapman, 1956; Coffey & Muller, 2000), fitting these requirements (see Fig. 3a).

The scenario generation process was performed by sampling random values from TG distributions defined by particular values for mode, variance, and range of travel times. We assumed independent sampling and, thus, travel time random variables. Note that this hypothesis only affects the scenario generation procedure, and not the performance of our scenario-based model. The mode of a service (service leg) is its usual duration. Nine *scenario classes*, *SClass*, were generated by considering three variability levels and three ranges. The former was measured in terms of standard deviation, low for level 1, medium for level 2, and high for level 3. We set the same lower bound for all cases but varied the range by using three different upper bounds. Recall that, first, our problem setting is interurban travel where in most cases time periods are lengthy (e.g., of the order of half a day or a day) and, second, we address business-as-usual delays. Consequently, the ranges should be reasonably “close”. We thus defined a tight range, *t*, computed as the mode −30% of a time-period duration, medium, *m*, computed as mode plus one time period, and loose, *l*, computed as medium plus 30% of a time period. The looser the range, the wider is the concept of “usual” travel time. Scenario classes, *SClass*(· ·) are thus identified by the pair *level of variability* (1, 2, or 3) and *range* (t, m, or l). Fig. 3b displays distributions for the same range but different standard deviations, while Fig. 3c and 3d display different ranges and their tails, respectively, for the same level of variability.

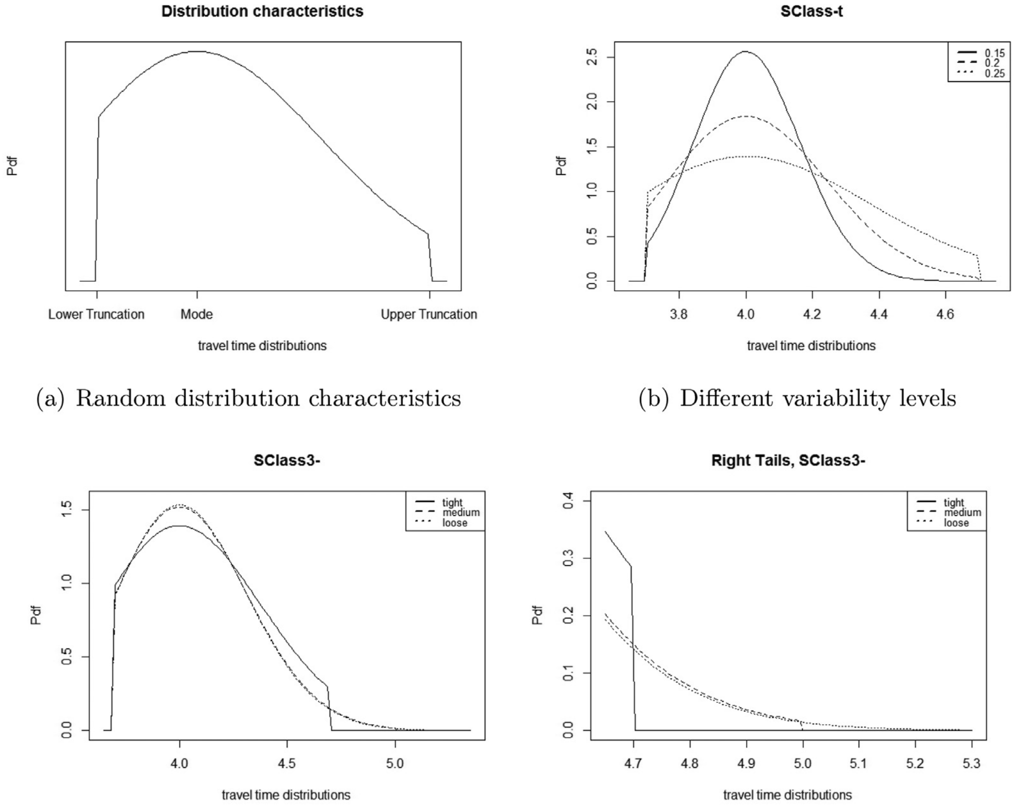
The mode was also used as the travel time point estimation in the deterministic formulation of the original problem. Experiments were performed under three levels of increasing penalty costs for each of the two schedule targets, on-time arrival and maximum acceptable delay. For services, the first level of on-time-arrival penalty was set to 175% of the cost of the most expensive



(a) Physical service network

(b) Space-time service network

Fig. 2. Physical and space-time service networks.



(a) Random distribution characteristics

(b) Different variability levels

(c) Different ranges

(d) Tails for different ranges

Fig. 3. Travel time truncated gamma distributions.

service, while the first level of the maximum-delay penalty was set to 215% of the same value. The second and third levels are obtained by doubling and tripling these values, respectively. A similar process was performed for the delivery targets, where the on-time-arrival penalty was set to the cost of the most expensive service, while the first level of the maximum-delay penalty was set to 175% of the same value. To address the single-target formulations, we set the penalties to 0 for the target not considered in that experiment.

6.2. In-sample and out-of-sample stability

We performed *in-sample* and *out-of-sample* stability verification to assess the correctness of the scenario generation procedure and the representativity of the generated scenarios, and, thus, avoid introducing bias in the results of the optimization model. The two types of stability are different and it is recommended to test them both (Kaut, Vladimirou, Wallace, & Zenios, 2007). We performed stability tests for the highest variability level (level 3), with varying ranges and levels of demand.

For in-sample stability test, we considered ten instances for each problem class. Each instance was solved with different scenario-sized sets and for each scenario-size the difference between the highest and lowest optimal values across instances was collected. In-sample stability achieving a difference of less than 1% between the highest and lowest optimal values for each problem class was obtained by using sets of 30 scenarios. Table 2a displays results for the third demand class and the third penalty level.

We then tested out-of-sample stability by using 30-scenario-sized sets to find solutions and 100-scenario-sized sets as the “true” stochastic phenomenon (generated from the same TG distributions used to construct the scenario sets for the optimization process). A procedure similar to Monte-Carlo simulation was used to evaluate the solutions. The evaluation was performed by fixing the first-stage variables obtained solving the stochastic programs

Table 2
Stability results for the scenario-generation procedure.

In-sample stability test		
	DClass(3t)	DClass(3l)
SClass(3t)	0.72	0.65
SClass(3m)	0.77	0.75
SClass(3l)	0.56	0.51
Out-of-sample stability test		
	DClass(3t)	DClass(3l)
SClass(3t)	2.24	0.61
SClass(3m)	1.08	1.44
SClass(3l)	2.33	0.79

for the 30-scenario-sized sets and optimizing the demand flows by solving the second-stage for the 100-scenario-sized sets. We computed the difference between the highest and lowest optimal-solution values for the 100 scenarios. In the worst-case situation, we obtained an error of 2.33%. Table 2b displays the results for the same problem classes, scenario classes, and penalty as for the in-sample experiments. We also tested out-of-sample stability for larger scenario sizes. Nevertheless, we were able to only reduce this error at a marginal level (decimal order) but at the expenses of a tremendous increase of running time.

Approximating a probability distribution through scenarios will inexorably bring some noise into results. The error of such an approximation will be reduced as the number of scenarios increases. On the other hand, as we increase the number of scenarios, the difficulty of obtaining optimal solutions increases as well. There is thus a trade-off between the stability of the stochastic solutions and the problem growing too large. We consider the results of stability tests performed with 30 scenarios satisfactory and consistent with our scopes of finding optimal solutions in acceptable time. Note that the reported results relate to the highest variability level, which defines the most volatile situations. In less variable cases, such results are even lower. In the following thus, results are obtained by considering 30 as the scenario-size.

7. Results and analyses

Results of the experimentation are presented and discussed first with respect to the evaluation of the SSND-QT formulation and, then, relative to the performance of the meta-heuristic. Several other results on the formulation are presented in Lanza, Crainic, Rei, and Ricciardi (2018).

7.1. Formulation evaluation

The evaluation analysis was performed considering 6 demand classes derived by the combined use of the first 3 levels of demand and the 2 delivery-time windows. We generated 10 instances for each demand class, for a total of 60 deterministic instances. For each deterministic instance, 27 stochastic instances were constructed, combining the 3 levels of variability, 3 ranges, and 3 penalty rates.

A solution, whether for the deterministic variant (*SSND*) or for the stochastic one, consists of a set of selected services and the paths used to transport commodity flows to their destinations. Solutions were found to the three stochastic formulations and are identified in the following as *SSND-QT* for the complete formulation, *SSND-QT-S* and *SSND-QT-D* when only the schedule- or the delivery-targets were considered, respectively. Comparisons among methods and solutions are discussed relative to the set-up cost of the network (service selection plus routing costs), as well as to their behavior into a stochastic environment through a Monte Carlo-like procedure with a set of 100 scenarios. The section finishes with an analysis of the impact of penalties.

7.1.1. Cost comparisons

The deterministic solution SSND displays, in general, characteristics typical of consolidation-based transportation networks, where different commodities share the capacity of single services for most of their journeys, passing through several intermediary stops, where they often wait, before arriving at destination. One also observes the just-in-time arrival, with respect to due dates, of freight at destination. Furthermore, one-stop services are usually favored when possible, rather than no-stop services in order to lower the fixed costs.

The stochastic SSND-QT-S set-up costs are generally similar to those of the corresponding deterministic SSND, but the structure of the service network is markedly different. In almost all cases, less services operate in SSND-QT-S than in SSND, even though the two solutions share part of them. This may be explained as follows by the increased risk of delay of multi-stop services. Thus, if a service experiences a delay in its first leg, it will most likely arrive at destination (its second stop) later than scheduled, unless, in the second leg, the observed travel time is much lower than usual and absorbs the delay. Given the assumed distributions, complete absorption is not very likely and one-stop services have a higher risk of paying for delays. Consequently, the model would move the solution away from less-expensive multi-stop services to more expensive direct connections, lowering the risk of extra costs when the services operate. The observed trend of SSND-QT-S solutions is to select only the strictly necessary direct services to fulfill demand. This often results in higher flow distribution costs and more complex distribution patterns. As fewer services are available, commodity paths will be more tangled and involve more services and transfers, the latter implying additional idle time at intermediary terminals. The SSND-QT-S set-up cost is thus the result of two opposite effects: on the one hand, fixed costs, as well as the number of selected services, are reduced while, on the other hand, freight distribution costs are increased. This characteristic seems to contrast SSND-QT-S solutions to the documented effect of stochastic demand, which generally increases the number of services (Wang, Crainic, & Wallace, 2018). Table 3 illustrates this analysis displaying the average cost characteristics - Set-up, Fixed (service selection), and Routing (freight distribution) - together with the associated number of selected services for instances belonging to the third demand class that were solved with increasing level of variability and the highest penalty level.

The opposite behavior is observed for the SSND-QT-D for which, generally, set-up costs are generally more expensive compared to the corresponding deterministic version. Moreover, unlike the SSND-QT-S case, but similar to the case of SSND with stochastic demand, more services are selected and the increase in set-up costs follows directly. SSND-QT-D networks appear to be built to bring commodity flows as early as possible to destination, at least one period before due date. When avoiding just-in-time arrival is not possible for the total quantity of a commodity, the flow is sometimes split, and a major part is shipped in advance. Such behavior requires more services to be selected. Table 4 displays the average results for the same demand classes, scenario classes, and penalty level considered above.

Let the *full cost* of a network be the set-up cost plus the penalties incurred for delays when simulating the behavior of the network with the Monte Carlo procedure. SSND-QT-S and SSND-QT-D display full costs that are always lower than that of the corresponding SSND. This shows that explicitly considering the stochastic nature of the travel times in the tactical planning model may hedge against, or at least reduce, the effects and consequences of uncertainty, despite an initial higher set-up cost. To illustrate, we display in Tables 5 and 6, the average results of the Monte-Carlo simulation procedure for two stochastic versions and the same demand classes, scenario classes and penalty level considered above.

Table 3
Cost characteristics of SSND-QT-S solutions.

	DClass(3t)				DClass(3l)			
	Set-up	Fixed	Num.Serv.	Routing	Set-up	Fixed	Num.Serv.	Routing
SClass(1m)	6326.5	136.2	31.3	6190.3	6684.8	167.2	37.4	6517.6
SClass(2m)	6330.7	135.9	31.1	6194.8	6705.2	163.1	37	6542.1
SClass(3m)	6333.6	131	30.5	6202.6	6713.7	158.4	36.6	6555.3
SSND	6340.8	139.8	31.4	6201	6694.8	177.1	39.5	6517.7

Table 4
Cost characteristics of SSND-QT-D solutions.

	DClass(3t)				DClass(3l)			
	Set-up	Fixed	Num.Serv.	Routing	Set-up	Fixed	Num.Serv.	Routing
SClass(1m)	6344.8	144.2	32.4	6200.6	6697	178.4	39.8	6518.6
SClass(2m)	6350.6	144.4	32.9	6206.2	6715.2	176.9	40	6538.3
SClass(3m)	6351.9	144.1	32.9	6207.8	6717.1	175.9	39.7	6541.2
SSND	6340.8	139.8	31.4	6201	6694.8	177.1	39.5	6517.7

Table 5
Monte Carlo simulation cost analysis of SSND-QT-S.

	DClass(3t)				DClass(3l)			
	Full Cost	Tot Penalty	Short	Long	Full Cost	Tot Penalty	Short	Long
SClass(1m)	17871.2	11544.7	11118.2	426.4	20636.1	13951.2	13474.9	476.3
SSND	18221.3	11880.5	11437.4	443.0	21550.5	14855.7	14336.4	519.3
SClass(2m)	36835.2	30504.48	13162.2	17342.2	42864	36158.8	16443.6	19715.2
SSND	39439.9	33099.1	13214.0	19885.1	51719.8	45025	16959.2	28065.8
SClass(3m)	46531.2	40197.5	15873.4	24324.1	54481	47767.7	19589	28178.7
SSND	50362.1	44021.3	16532.2	27489.0	65478.2	58783.4	20940.6	37842.7

Table 6
Monte Carlo simulation cost analysis of SSND-QT-D.

	DClass(3t)				DClass(3l)			
	Full Cost	Tot Penalty	Short	Long	Full Cost	Tot Penalty	Short	Long
SClass(1m)	8647	2302.2	2222.9	79.3	10186.5	3489.5	3314.7	174.7
SSND	11068.5	4727.7	3069.3	1658.4	14803.6	8108.8	3826.6	4282.2
SClass(2m)	13388.8	7038.1	2485.5	4552.6	17215.1	10499.9	3773.5	6726.3
SSND	21332	14991.2	3421.5	11569.6	28410.2	21715.4	4275.3	17440
SClass(3m)	15615.6	9263.7	2877.3	6386.4	20735.9	14018.8	4537.2	9481.5
SSND	25199.9	18859.1	4327.9	14531.1	33396.2	26701.4	5335.5	21365.8

The tables display the average full costs, total penalty, as well as the split of the latter according to weather the penalty corresponds to the short (i.e., the actual target) or a long (relative to the maximum time allowed B or B^k) delay.

The general SSND-QT yields service designs that appear as a compromise between SSND-QT-D and SSND-QT-S, displaying both the direct-service and early-freight-arrival trends. It is the schedule targets, however, that influence the SSND-QT the most. The set-up cost display exactly the same behavior as for SSND-QT-S when penalties or variability levels increase (a decrease in the number of selected services and fixed costs and a consequent increase in routing costs). Nevertheless, the delivery targets try to limit just-in-time deliveries, favoring delivery one period before the due date, if not earlier. Routing is complicated as, in the majority of the cases, demand is not only delivered in advance to lower the expenses related to the late arrivals of freight, but it is also moved through the network by less services than in the SSND. Freight paths thus seem even more tangled than in SSND-QT-S and include longer idle times at some intermediary stops. Average set-up costs and performance results are shown in Tables 7 and 8.

7.1.2. Impact of penalties

The amplitude of the penalties for late arrival directly influences the design. The higher the level of penalties the more the model aims to build a service network that will perform as planned when travel times vary. Table 9 displays the results when the level of penalties increases. The average increase in set-up costs and average decrease in total delay are shown, in percentages, for solutions obtained with penalty levels 2 and 3, compared relative to solutions obtained with penalty level 1. The same demand and scenario classes considered in the previous experiments were also used here.

Focusing on the SSND-QT-S, we observe that, as anticipated, increasing the penalty enforces the reliability by decreasing the number of multi-stop services. The main delay decrease concerns the most expensive delay, i.e., the long delay over the threshold B. The higher the penalty, the lower are such delays. Increasing penalties threefold yields an increase in the fixed cost of the network by some 0.03%, with a decrease in the amount of total delay of some 3% for the short delays and 10% for the long ones.

Similar results were observed for SSND-QT-D. Here, an increase in the penalty level makes the optimization increase the number of selected services to deliver the highest number of commodities possible at least one period before due date. The percentage of early freight arrivals is increased by 8% when the solutions based on the highest and lowest penalty level are compared (total number of commodities is 25). The total amount of delay decreases by 12%, the vast majority of delays being of the large variety (over

Table 7
Cost characteristics of SSND-QT solutions.

	DClass(3t)				DClass(3l)			
	Set-up	Fixed	Num.Serv.	Routing	Set-up	Fixed	Num.Serv.	Routing
SClass(1m)	6347.2	136.9	31.6	6210.3	6703.5	170.2	38.6	6533.3
SClass(2m)	6333.6	136.6	31.4	6219.4	6742.2	161.9	37.3	6580.3
SClass(3m)	6365.9	135.1	31.2	6230.8	6765.5	156.4	36.3	6609.1
SSND	6340.8	139.8	31.4	6201	6694.8	177.1	39.5	6517.7

Table 8
Monte Carlo simulation cost analysis of SSND-QT.

	DClass(3t)					
	Full Cost	Tot Penalty	Short S	Long S	Short D	Long D
SClass(1m)	20245.6	13898.4	11227.4	419.8	2190.4	60.7
SSND	22949.04	16608.2	11437.4	443	3069.3	1658.3
SClass(2m)	44871.2	38515.3	13173.7	18233.4	2541.6	4566.5
SSND	54431.2	48090.4	13214	19885.1	3421.5	11569.6
SClass(3m)	57512	51146.1	16247.7	25483.8	2986.8	6427.7
SSND	69221.2	62880.4	16532.2	27489	4327.9	14531.1
	DClass(3l)					
	Full Cost	Tot Penalty	Short S	Long S	Short D	Long D
SClass(1m)	24389.8	17686.3	13792.7	469.6	3270.4	153.5
SSND	29659.4	22964.6	14336.4	519.3	3826.6	4282.2
SClass(2m)	54048.8	47306.6	16332.5	19924.8	4032.1	7017.1
SSND	73435.3	66740.5	16959.2	28065.8	4275.3	17440.1
SClass(3m)	67839.4	47306.6	16332.5	19924.8	4032.1	7017.1
SSND	92179.6	85484.8	20940.6	37842.8	5335.5	21365.8

Table 9
Effects of penalty increase on SSND-QT-D and SSND-QT-S behavior.

	Delivery target			Schedule target		
	Fixed cost (%)	Short delay (%)	Long delay (%)	Fixed cost (%)	Short delay (%)	Long delay (%)
Penalty 1	-	-	-	-	-	-
Penalty 2	+0.01	-3.26	-9.99	+0.01	-1.06	-5.66
Penalty 3	+0.05	-7.48	-18.36	+0.03	-3.38	-9.95

the threshold B^k), at the expense of additional set-up cost of 0.05% only.

7.2. Meta-heuristic performance analyses

The meta-heuristic performance was evaluated through experiments conducted considering instances belonging to all demand classes, the loosest delivery time window $(11-14)$, the highest variability level $(\sigma = 0.25)$ and the loosest time distribution range (l). We still use 30 scenarios for each instance (this means that at each iteration the meta-heuristic solves 30 single-scenario sub-problems). The parameters were set to $\gamma = 1.1$ and $\beta = 1.1$ for the fixed-cost adjustments, while thresholds were set to $thres_{high} = 0.8$, $thres_{low} = 0.2$ for global adjustments and $local_{high} = 1 + u_r/2$ and $local_{low} = (u_r/2) - 1$ for local adjustments. Parameter values were chosen based on those used in Crainic et al. (2011), after not having observed any significant changes in the meta-heuristic behavior for small changes in those values during preliminary experiments. We considered, instead, different values of parameter ρ^0, ranging from 0.1 to 50, and different thresholds of consensus required to stop Phase 1, ranging from 80% to 95% of the services.

Table 10 displays the performance results of CPLEX and the meta-heuristic with the PHR and PHRL cost-adjustment strategies, with $\rho^0 = 5$. The first phase is stopped either when consensus is obtained for 90% of services, on reaching a total of 20 iterations, or after 4 hours of running time. The table also displays the total computation time (in seconds), the number of iterations, and the relative gaps with respect to the solution obtained with CPLEX.

As expected, CPLEX solves almost all instances in a very short time, requiring only a few seconds for the easiest instances and around 2 minutes for the most complicated. Meta-heuristic computing times are longer. This is not surprising. On the one hand, instances were selected in such a way to allow CPLEX to find the optimal solution to the stochastic formulation. On the other hand, in order to have a clear comparison in terms of solution quality, the optimal solution to each sub-problem at each iteration was determined using the same solver (future extensions could, of course, involve the use of meta-heuristics for the sub-problems). While this slows down the meta-heuristic, it provides the means to show its very good performance in terms of solution quality, both versions finding good quality solutions; 45% of the instances are solved optimally, while the optimality gaps for the remaining instances are less than 1%.

What is the effect of flow-information adjustment? The PHRL approach seems to perform slightly better compared to the PHR approach, yielding an average optimality gap of 0.022% compared to 0.054% for PHR. The improvement is reached, however, at the expense of a higher number of iterations (on average 4.63 instead of 3.58) and a much longer running time (PHRL needs on average 58% more time compared to PHR). This additional effort is due to the higher number of operations characterizing PHRL compared to PHR. In all cases, the PHR version stops Phase 1 because the chosen consensus threshold is reached, while in 25% of the cases, the PHRL stops because the time limit is reached.

We also tested the meta-heuristic over a set of larger instances built on the same physical network with 35 scenarios, 220 services and 85 commodities. CPLEX could not address instances of such

Table 10
Performances of CPLEX and meta-heuristic (PHR and PHRL strategies; $\rho^0 = 5$).

$\|K\|$	CPLEX		PHR				PHRL			
	Time (sec.)	Obj	Time (sec.)	Obj	Iter.	Gap (%)	Time (sec.)	Obj	Iter.	Gap (%)
15	13.11	3964.38	61.25	3964.38	2	0	82.66	3964.38	2	0
15	13.86	4538.18	114.24	4538.18	2	0	151.33	4538.18	2	0
15	12.95	3701.94	70.42	3712.86	2	0.29	104.88	3701.94	2	0
15	13.87	3834.58	101.16	3841.76	3	0.19	185.51	3841.76	3	0.19
15	14.21	4562.23	128.22	4565.61	2	0.07	395.12	4565.61	5	0.07
15	13.74	4679.08	120.91	4679.08	2	0	324.04	4679.08	4	0
20	32.53	6007.60	189.53	6007.60	2	0	230.42	6007.60	2	0
20	18.23	5061.42	172.41	5061.42	3	0	312.03	5061.42	4	0
20	23.14	6178.43	249.02	6181.83	3	0.06	544.75	6180.20	4	0.03
20	30.71	6532.49	350.09	6565.06	2	0.50	715.83	6532.49	4	0
20	16.15	4830.19	87.80	4830.19	3	0	151.25	4830.19	3	0
20	17.83	5216.48	116.97	5216.48	3	0	338.74	5216.48	5	0
25	24.90	7151.51	207.49	7153.12	2	0.02	313.53	7153.12	2	0.02
25	21.57	6245.17	104.80	6245.17	2	0	222.61	6250.66	3	0.09
25	27.10	6729.70	244.82	6729.70	4	0	828.99	6729.70	7	0
25	23.45	6968.70	176.93	6968.70	2	0	257.61	6968.70	2	0
25	29.54	7295.06	374.55	7300.43	4	0.07	887.82	7302.14	5	0.10
25	22.34	6638.79	89.24	6638.79	2	0	147.18	6639.35	3	0.01
50	48.26	14494.40	1682.03	14494.40	6	0	9313.16	14494.40	8	0
50	51.19	15562.70	1806.54	15575.10	6	0.08	9574.86	15564.30	8	0.01
50	90.65	15068.40	2514.37	15068.40	8	0	11129.99	15068.40	7	0
50	49.37	15317.40	1871.77	15317.40	7	0	8927.28	15317.40	8	0
50	123.89	14776.00	2024.23	14779.40	7	0.02	8850.26	14779.40	8	0.02
50	50.38	15035.60	2177.47	15035.60	7	0	9184.05	15035.60	10	0
Average	32.62		626.51		3.58	0.05	2632.25		4.63	0.02

Table 11
Average performance of PHR and PHRL with different values of ρ^0.

ρ^0	PHR			PHRL		
	Iterations	% of Optima	Gap (%)	Iterations	% of Optima	Gap (%)
0.1	3.17	70.83	0.0498	2.54	79.17	0.02215
0.5	2.5	66.67	0.0504	2.33	75	0.02239
1	2.17	66.67	0.0522	2.29	75	0.02471
5	1.88	62.5	0.0542	2.25	75	0.02617
10	1.83	62.5	0.0558	2.17	70.83	0.02863
20	1.71	62.5	0.0598	2.13	70.83	0.02938
50	1.63	58.33	0.0675	1.63	62.5	0.04406

sizes, while the meta-heuristic with the global fixed-cost adjustment strategy PHR (with $\rho^0 = 5$ and a consensus threshold of 90%) provided solutions in 4 hours, on average.

7.2.1. Impact of parameter ρ^0

The performance of the method is generally sensitive to the choice of the penalty parameter ρ^0, which scales the penalty term (Bertsekas, 2014). Theory suggests that high values of the penalty parameter should induce faster but often premature convergence leading to ill-conditioned solutions. Conversely, small values of ρ^0 yield a weaker enforcement of the non-anticipativity constraints, resulting in a more gradual convergence to typically better solutions after many iterations (Mulvey & Vladimirou, 1991; Wallace & Helgason, 1991). This is supported by empirical evidence in our case as well. We solved the chosen instance set considering different values of ρ^0, ranging from 0.1 to 50. The first phase is stopped when consensus is obtained for 80% of services.

Table 11 displays average performance results for the PHR and PHRL versions of the meta-heuristic, namely, the number of iterations, the percentage of instances solved to optimality, and the optimality gaps (with respect to CPLEX). In line with the results from the literature, a small ρ^0 value yields longer first-phase computations, but the mechanism has sufficient time to “absorb” the information from the scenarios and obtains good results. For large ρ^0 values, however, the meta-heuristic stops quite rapidly at a local optimum. The number of instances solved to optimality decreases (from 70% to 58% for PHR and from 79% to 62% for PHRL), as well as the accuracy of solutions, even though the optimality gaps are very low and less than 1%. This behavior characterizes both meta-heuristic variants but, as observed earlier, the PHRL always outperforms the PHR in relative gaps and number of instances optimally solved. It is noteworthy that the PHRL variant is able to solve at least 75% of instances optimally, maintaining this standard until $\rho^0 = 5$, while not only is the PHR unable to reach this threshold, but this percentage decreases as soon as ρ^0 increases. Regarding iterations, it seems that if ρ^0 is very low, the iterations needed by the PHRL are less than those needed by the PHR, as if the additional flow information not only influences the accuracy of the solutions but also allows the approach to converge sooner. This characteristic is lost when $\rho^0 = 1$.

7.2.2. Impact of the consensus threshold parameter

We solved the instance set considering different values of consensus threshold to stop Phase 1, ranging from 80% to 95% of the services. The value of ρ^0 was set to 5. Intuitively, the value of the threshold directly influences the number of iterations the algorithm needs to perform before stopping the first phase: the higher the value, the more iterations are needed in Phase 1 to reach the desired consensus threshold and go to Phase 2. At the same time, however, this allows Phase 2 to start with a higher number of fixed services, decreasing its computational effort. The results presented in Table 12 are consistent with this behavior. The table displays the average number of iterations performed by the PHR and PHRL

Table 12
Performance of PHR and PHRL with different consensus threshold values.

Threshold	PHR			PHRL		
	Iterations	% of Optima	Gap (%)	Iterations	% of Optima	Gap (%)
80%	1.88	62.5	0.0542	2.25	66.67	0.0263
85%	2.63	66.67	0.0338	3.21	75	0.0255
90%	3.58	66.67	0.0337	4.63	75	0.0173
95%	8.73	66.67	0.0331	11.21	75	0.0136

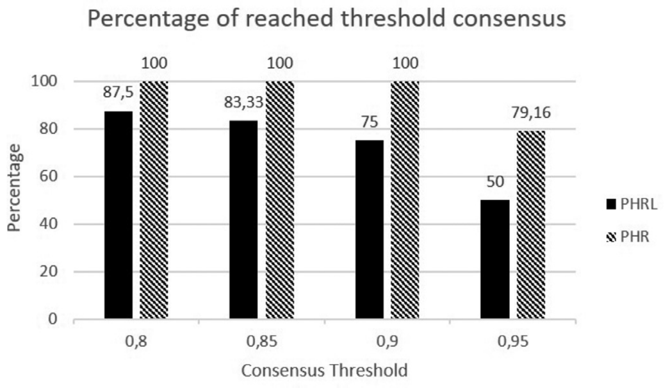


Fig. 4. Percentage of instances reaching consensus thresholds.

versions, the percentage of instances solved to optimality, and the average optimality gaps.

Both variants display the same behavior, with similar increases in the number of iterations and accuracy as the consensus threshold increases. PHRL constantly outperforms PHR in terms of accuracy but, as already observed, at the expense of a higher number of iterations (27% more on average). Thresholds higher than 85% seem to improve the accuracy level for both approaches, while the percentage of instances solved optimally does not change and the decrease in optimality gaps is marginal.

It should be noted, however, that the highest threshold (95%) seems too strict, and the first phases of both variants stopped, for most of the hardest cases, not because the consensus threshold was attained, but because the time limit (4 hours) was reached. Figure 4 plots the percentage of times the Phase 1 of PHR and PHRL stopped by reaching the chosen consensus threshold.

We also solved a number of instances without a time-limit stopping rule. Solutions and gaps do not change significantly (only after the third decimal place). What consistently changes is the number of iterations needed to reach this threshold and stop Phase 1, on average 20. This emphasizes that consensus thresholds and time limits should be appropriately chosen. The results of this section are a step in defining these criteria.

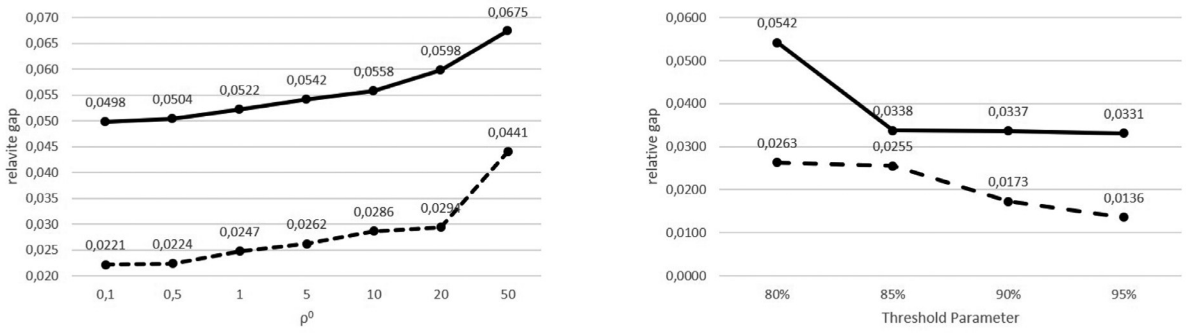
7.2.3. Final remarks about PHR and PHRL

To sum up the observations regarding the two versions of the fixed-cost adjustment strategy, both approaches result in good quality solutions, but display slightly different behaviors.PHRL seems to greatly benefit from small ρ^0 parameters (even when $\rho^0 = 0.1$ and $\rho^0 = 0.5$, PHRL outperforms PHR in terms of number of iterations) and high consensus thresholds. The PHR approach seems to have a more stable behavior with respect to optimality gaps but is characterized by a decreasing number of iterations when the parameters are changed. Optimality gaps are displayed for increasing values of ρ^0 and consensus threshold for the two variants in Fig. 5a and 5b.

The PHRL variant appears to enable the algorithm to perform better compared to PHR when solution accuracy is of concern, and the higher computational effort is of lesser importance. On the other hand, it seems that, at least for the instances studied, the information on global trends is sufficient to find good solutions, making the PHR variant a suitable methodology to address problem instances quickly and efficiently.

8. Conclusion

We addressed the SSND-QT, the stochastic SSND problem with quality targets and travel time uncertainty. It is an important problem for the tactical planning process of consolidation-based freight carriers, as well as for the network design problem class. Very few papers in the literature address issues related to this problem, our contribution being, according to our best knowledge, the first to state and address the full problem. We defined the problem considering quality targets for on-time, according to a schedule, operation of services and on-time, according to carrier-customer contracts, delivery of commodity flows to destinations. We proposed



(a) Gaps for different ρ^0 values with consensus threshold at 90%

(b) Gaps for different consensus thresholds with $\rho^0 = 5$

Fig. 5. Optimality gap evolution for PHR (full line) and PHRL (dashed line).

a two-stage mixed-integer linear stochastic model defined over a space-time network, with quality targets modeled through penalties. The first stage addresses the selection of services and the routing of the freight flows, while the second stage verifies targets and computes penalties. We also proposed an effective PH-based meta-heuristic, based on a partial-decomposition concept aimed to address the challenges raised by the presence in the first-stage problem of flow-distribution decisions and by the flow-related degeneracy peculiar to network design. The results of an extensive numerical experimentation underscored the worthiness of the formulation when quality targets are of interest, highlighting specific features to hedge against time fluctuations. They also showed that the proposed meta-heuristic performs very well, being comparable to that of a well-known MIP solver, and outperforming the solver when instance dimensions increase. Several interesting research avenues are open. One relates to the introduction of uncertainty on terminal operations and, clearly in the longer-term, integrating in a unique formulation uncertainty related to both demand and time. A second avenue concerns the representation of more complex decisions/actions when delays are observed, addressing, for example, the case of missed connections. Extensions to explicitly account for the management of resources required to operate services is another interesting research avenue, as is the study and relaxation of the independence hypothesis regarding travel time distributions. Algorithmically speaking, continuing the study of partial decomposition and associated adjustment and consensus-seeking strategies is certainly of interest, as is the definition of multi-scenario subproblems. The latter issue belongs to the broader challenge of developing efficient solution methods for large instances of stochastic network design problems. We look forward to presenting results on some of these issues in the near future.

Acknowledgments

While working on this project, the second author was Adjunct Professor with the Department of Computer Science and Operations Research, Université de Montréal. Partial funding for this project has been provided by the Ministero dell'Istruzione, dell'Universitá e della Ricerca (MIUR) of Italy, through its Research Projects of Relevant National Interest (PRIN) program, the Sapienza Universitá di Roma, Italy, through its Progetto di Ateneo La Sapienza, and the Natural Sciences and Engineering Council of Canada (NSERC), through its Discovery Grant program. We gratefully acknowledge the support of the Fonds de recherche du Québec, the Canada Foundation for Innovation, and Ministére de l'Économie, de la Science et de l'Innovation Québec through their infrastructure grants. The project was conducted at CIRRELT - The Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation - and most final computations were carried on the supercomputers of Calcul Québec and Compute Canada. We gratefully acknowledge their support.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ejor.2020.05.031.

References

Assad, u. (1980). Models for Rail Transportation. *Transportation Research Part A: Policy and Practice, 14*, 205–220.

Bai, R., Wallace, S. W., Li, J., & Chong, A. Y.-L. (2014). Stochastic service network design with rerouting. *Transportation Research Part B: Methodological, 60*, 50–65.

Bertsekas, D. P. (2014). *Constrained optimization and lagrange multiplier methods*. Academic press.

Birge, J. R., & Louveaux, F. (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

Carpentier, P.-L., Gendreau, M., & Bastin, F. (2013). Long-term management of a hydroelectric multireservoir system under uncertainty using the progressive hedging algorithm. *Water Resources Research, 49*(5), 2812–2827.

Chapman, D. G. (1956). Estimating the parameters of a truncated gamma distribution. *The Annals of Mathematical Statistics*, 498–506.

Christiansen, M., Fagerholt, K., Nygreen, B., & Ronen, D. (2007). Maritime Transportation, *14*, 189–284.

Christiansen, M., Fagerholt, K., & Ronen, D. (2004). Ship Routing and Scheduling: Status and Perspectives. *Transportation Science, 38*(1), 1–18.

Coffey, C. S., & Muller, K. E. (2000). Properties of doubly-truncated gamma variables. *Communications in Statistics-Theory and Methods, 29*(4), 851–857.

Cordeau, u., Toth, P., & Vigo, D. (1998). A Survey of Optimization Models for Train Routing and Scheduling. *Transportation Science, 32*(4), 380–404.

Crainic, T. G. (1988). Rail tactical planning: Issues, models and tools. In L. Bianco, & A. La Bella (Eds.), *Freight transport planning and logistics* (pp. 463–509). Springer-Verlag, Berlin.

Crainic, T. G., Errico, F., Rei, W., & Ricciardi, N. (2015). Modeling demand uncertainty in two-tier city logistics tactical planning. *Transportation Science, 50*(2), 559–578.

Crainic, T. G., Fu, X., Gendreau, M., Rei, W., & Wallace, S. W. (2011). Progressive hedging-based metaheuristics for stochastic network design. *Networks, 58*(2), 114–124.

Crainic, T. G., Hewitt, M., & Rei, W. (2014a). Scenario grouping in a progressive hedging-based meta-heuristic for stochastic network design. *Computers & Operations Research, 43*(2), 90–99.

Crainic, T. G., Hewitt, M., Toulouse, M., & Vu, D. M. (2014b). Service network design with resource constraints. *Transportation Science, 50*(4), 1380–1393.

Crainic, u. (2000). Network Design in Freight Transportation. *European Journal of Operational Research, 122*(2), 272–288.

Crainic, u. (2003). Long-Haul Freight Transportation. In R. W. Hall (Ed.), *Handbook of Transportation Science* (pp. 451–516). Kluwer Academic Publishers, Norwell, MA.

Crainic, u., Ferland, J.-A., & Rousseau, J. M. (1984). A Tactical Planning Model for Rail Freight Transportation. *Transportation Science, 18*(2), 165–184.

Crainic, u., Florian, M., Guélat, J., & Spiess, H. (1990a). Strategic Planning of Freight Transportation: STAN, An Interactive-Graphic System. *Transportation Research Record, 1283*, 97–124.

Crainic, u., Florian, M., & Léal, J. E. (1990b). A Model for the Strategic Planning of National Freight Transportation by Rail. *Transportation Science, 24*(1), 1–24.

Crainic, u., Gobbato, L., Perboli, G., & Rei, W. (2016). Logistics Capacity Planning: A Stochastic Bin Packing Formulation and a Progressive Hedging Meta-heuristic. *European Journal of Operational Research, 253*(12), 404–417.

Crainic, u., & Kim, u. (2007). Intermodal Transportation. In C. Barnhart, & G. Laporte (Eds.), *Transportation*. In *Handbooks in Operations Research and Management Science, chapter 8: vol. 14* (pp. 467–537). North-Holland, Amsterdam.

Crainic, u., & Rousseau, J. M. (1986). Multicommodity, Multimode Freight Transportation: A General Modeling and Algorithmic Framework for the Service Network Design Problem. *Transportation Research Part B: Methodological, 20*, 225–242.

Crainic, u., & Roy, J. (1988). O.R. Tools for Tactical Freight Transportation Planning. *European Journal of Operational Research, 33*(3), 290–297.

Demir, E., Burgholzer, W., Hrušovskỳ, M., Arikan, E., Jammernegg, W., & Van Woensel, T. (2016). A green intermodal service network design problem with travel time uncertainty. *Transportation Research Part B: Methodological, 93*, 789–807.

Fan, Y., & Liu, C. (2010). Solving stochastic transportation network protection problems using the progressive hedging-based method. *Networks and Spatial Economics, 10*(2), 193–208.

Gade, D., Hackebeil, G., Ryan, S. M., Watson, J.-P., Wets, R. J.-B., & Woodruff, D. L. (2016). Obtaining lower bounds from the progressive hedging algorithm for stochastic mixed-integer programs. *Mathematical Programming, 157*(1), 47–67.

Gonçalves, R. E. C., Finardi, E. C., & da Silva, E. L. (2012). Applying different decomposition schemes using the progressive hedging algorithm to the operation planning problem of a hydrothermal system. *Electric Power Systems Research, 83*(1), 19–27.

Guo, G., Hackebeil, G., Ryan, S. M., Watson, J.-P., & Woodruff, D. L. (2015). Integration of progressive hedging and dual decomposition in stochastic integer programs. *Operations Research Letters, 43*(3), 311–316.

Haugen, K. K., Løkketangen, A., & Woodruff, D. L. (2001). Progressive hedging as a meta-heuristic applied to stochastic lot-sizing. *European Journal of Operational Research, 132*(1), 116–122.

Hvattum, L. M., & Løkketangen, A. (2009). Using scenario trees and progressive hedging for stochastic inventory routing problems. *Journal of Heuristics, 15*(6), 527–557.

Jiang, X., Bai, R., Aickelin, U., & Landa-Silva, D. (2017). Fuzzy c-means-based scenario bundling for stochastic service network design. In *Proceedings of the IEEE symposium series on computational intelligence (SSCI)* (pp. 1–8). IEEE.

Kaut, M., Vladimirou, H., Wallace, S. W., & Zenios, S. A. (2007). Stability analysis of portfolio management with conditional value-at-risk. *Quantitative Finance, 7*(4), 397–409.

King, A. J., & Wallace, S. W. (2012). *Modeling with Stochastic Programming*. Springer Science & Business Media.

Klibi, W., Martel, A., & Guitouni, A. (2010). The design of robust value-creating supply chain networks: a critical review. *European Journal of Operational Research, 203*(2), 283–293.

Lanza, G., Crainic, T. G., Rei, W., & Ricciardi, N. (2018). A study on travel time stochasticity in service network design with quality targets. In *Proceedings of the international conference on computational logistics* (pp. 401–416). Springer.

Lium, A.-G., Crainic, T. G., & Wallace, S. W. (2007). Correlations in stochastic programming: A case from stochastic service network design. *Asia-Pacific Journal of Operational Research, 24*(2), 161–179.

Lium, A.-G., Crainic, T. G., & Wallace, S. W. (2009). A study of demand stochasticity in service network design. *Transportation Science, 43*(2), 144–157.

Løkketangen, A., & Woodruff, D. L. (1996). Progressive hedging and tabu search applied to mixed integer (0, 1) multistage stochastic programming. *Journal of Heuristics, 2*(2), 111–128.

Mulvey, J. M., & Vladimirou, H. (1991). Applying the progressive hedging algorithm to stochastic generalized networks. *Annals of Operations Research, 31*(1), 399–424.

Powell, W. B., & Topaloglu, H. (2003). Stochastic programming in transportation and logistics. *Handbooks in operations research and management science, 10*, 555–635.

Rockafellar, R. T. (1982). Lagrange multipliers and subderivatives of optimal value functions in nonlinear programming. *Nondifferential and Variational Techniques in Optimization*, 28–66.

Rockafellar, R. T., & Wets, R. J.-B. (1991). Scenarios and policy aggregation in optimization under uncertainty. *Mathematics of Operations Research, 16*(1), 119–147.

Song, D.-P., Li, D., & Drake, P. (2015). Multi-objective optimization for planning liner shipping service with uncertain port times. *Transportation Research Part E: Logistics and Transportation Review, 84*, 1–22.

Van Hui, Y., Gao, J., Leung, L., & Wallace, S. (2014). Airfreight forwarderâs shipment planning under uncertainty: A two-stage stochastic programming approach. *Transportation Research Part E: Logistics and Transportation Review, 66*, 83–102.

Wallace, S. W., & Helgason, T. (1991). Structural properties of the progressive hedging algorithm. *Annals of Operations Research, 31*(1), 445–455.

Wang, S., & Meng, Q. (2012a). Liner ship route schedule design with sea contingency time and port time uncertainty. *Transportation Research Part B: Methodological, 46*(5), 615–633.

Wang, S., & Meng, Q. (2012b). Robust schedule design for liner shipping services. *Transportation Research Part E: Logistics and Transportation Review, 48*(6), 1093–1106.

Wang, X., Crainic, u., & Wallace, u. (2018). Stochastic Scheduled Service Network Design: The Value of Deterministic Solutions. *INFORMS Journal on Computing*. Forthcoming

Wang, X., Crainic, u., & Wallace, u. (2020). Stochastic Network Design for Planning Scheduled Transportation Services: The Value of Deterministic Solutions. *INFORMS Journal on Computing, 31*(1), 153–170.

Watson, J.-P., & Woodruff, D. L. (2011). Progressive hedging innovations for a class of stochastic mixed-integer resource allocation problems. *Computational Management Science, 8*(4), 355–370.