POLITECNICO DI TORINO

Dip. di Automatica e Informatica (DAUIN)
Dept. of Control and Conputer Engineering

Optimization and Operations Research Group

Multi-period Transshipment Location-Allocation Problem with Stochastic Synchronized Operations

Riccardo Giusti
Daniele Manerba
Roberto Tadei

Nov 2019

DAUIN-ORO-2019-07

Multi-period Transshipment Location-Allocation Problem with Stochastic Synchronized Operations

Riccardo Giusti[1 1], Daniele Manerba[1,2], Roberto Tadei[1]

[1]Department of Control and Computer Engineering, Politecnico di Torino - 10129 Turin, Italy

[2]ICT for City Logistics and Enterprises Lab, Politecnico di Torino - 10129 Turin, Italy

[1]Corresponding author: riccardo.giusti@polito.it
Corso Duca degli Abruzzi, 24 - 10129 Torino (TO), Italy. Phone: +39 011 090 7083.
Other e-mail addresses: daniele.manerba@polito.it (D. Manerba), roberto.tadei@polito.it (R. Tadei)

Abstract

The Transshipment Location-Allocation Problem consists of locating transshipment facilities (e.g., inter-modal hubs) of a transportation network and allocating freight flows trough them, from several origins to several destinations, in order to satisfy demand and supply constraints. The objective is to maximize the total net transportation utility given by the total shipping utility minus the total cost to locate the facilities. Moreover, flow synchronization at the facilities must be also ensured. Unfortunately, the flow synchronization depends on a large set of unknown events, which could cause both unexpected reductions of the facility capacity and uncertain utility of handling operations. In this paper, we want first to evaluate how uncertainty on facility capacity and handling operations utility affects the Transshipment Location-Allocation Problem in terms of complexity, net gain, and optimal solutions. Moreover, we extend the problem from a single to a multi-period setting to have a wider view of future scenarios realizations and consequently synchronize the flows by using different facilities on different periods. We propose a two-stage Stochastic Programming formulation with recourse and analyze, over a ground set of instances, some well-known economic measures to derive managerial insights on the importance to address uncertainty for the problem. Finally, given the computational burden of solving the deterministic equivalent problem, we propose several heuristics based on Progressive Hedging and test their performance.
Keywords: Transshipment Location-Allocation Problem, Synchromodality, Stochastic Programming, Progressive Hedging.

0.1 Introduction

The Transshipment Problem consists of minimizing the total transportation cost over a network by using transshipment facilities as intermediate points for shipping freight from origins to destinations (Okiemute et al., 2017). In the literature, the utilization of transshipment facilities addresses several kind of issues, such as consolidation, packing and unpacking operations, inter-modal or inter-vehicle changes, and so on. Di Francesco et al. (2019) studied the problem of a forwarder that need to ship containers filled with pallets to intermediate depots where pallets are unpacked and sent to different destinations. Zhao et al. (2018) approached the location of consolidation centers in China as transshipment facilities to ship freight by rail routes from China to Europe.

While classical transshipment problems only consider allocation of flows to the existing facilities, in our study we consider a more general location-allocation problem, which consists in deciding the right number and location of the facilities to correctly satisfy the demand. A similar setting can be found in (Cooper, 1963). However, our Transshipment Location-Allocation Problem (TLAP) is a variant aiming at the maximization of the total net utility (i.e., the total shipping utility minus the total cost to locate the facilities) by finding transshipment facilities locations and allocating flows through them to satisfy customers demand. In the TLAP, both a network design (facility location) and a network flow planning (flows allocation) problems are embedded, that are respectively part of the strategic and tactical planning (Steadieseifi et al., 2014).

Recently, a new paradigm called *synchromodality* has emerged in the field of transportation and logistics. Synchromodality, among several other aspects, focuses on the importance of synchronizing operations to improve the efficiency and the sustainability of the supply chain. For its implementation in realistic settings, Pfoser et al. (2016) identified two main critical success factors, namely, the development of sophisticated planning methods and the use of efficient physical infrastructures (which in turn implies planning a good network design). On the other hand, the increase of freight transportation, and e-commerce in particular, is fostering the need to better synchronize operations and modes, as well as to address different sources of uncertainty. Jin et al. (2018) pointed out that unsynchronized shipping services at hub ports generate operation problems and high costs. To improve synchronization they designed feeder vessel services to pick up from and deliver containers to neighbouring local ports, working like transshipment facilities, and synchronize them with long-haul services improving the efficiency of container transshipment. Other benefits of the synchronization in transshipment facilities are studied in Neves-Moreira et al. (2016). The authors developed a method to provide long-haul services by means of short-haul jobs, over a logistics network of freight transportation in which trucks are allowed to exchange semi-trailers through several transshipment points. In this case, the synchronization helps to reduce the empty truck journeys drastically and finds applications in a decision support system of a Portuguese logistics operator. Finally, as pointed out by Giusti et al. (2019), dealing with uncertainty has a fundamental role to reduce the negative impact of disruptions in strategic/tactical planning. In fact, over a medium or long-term time horizon, a transshipment facility is subjected to events with unknown probability, such as lateness of the incoming/outgoing transportation modes, congestion, or unexpectedly slow execution of the handling operations. These events could cause significant stand-by time for vehicles in a facility and, in turn, loss of transshipment connections and reduction on the expected available capacity of the facilities. To show the benefits of synchromodality with respect to rigid planning, Qu et al. (2019) recently considered synchronization of transshipment flows at intermediate terminals, pointing out that in some cases overloaded terminals could cause delays and delay propagation. Hence, locating the right facilities in advance and synchronizing flows can have a great impact to limit this issue.

In this perspective, some recent papers have studied the problem at hand taking into account uncertainty. Tadei et al. (2012) proposed a model to maximize the total net utility, calculated by subtracting the total fixed cost of the located facilities from the total shipping utility. The shipping utility is given by a deterministic utility for shipping freight from origins to destinations via transshipment facilities plus a stochastic handling utility at the facilities. Eventually, a deterministic approximation of the problem was provided by using some results from the theory of extreme values (Tadei et al., 2018, Roohnavazfar et al., 2019). Instead, the model proposed in Baldi et al. (2012) aimed to minimize the total cost by finding an optimal location of the transshipment facilities for which the freight throughput costs are random variables with an unknown probability distribution. Here, the total cost is given by a deterministic fixed cost plus the expected cost of the total freight flow. Lastly, Wang et al. (2019) proposed different types of belief degree constrained programming models (optimistic value, pessimistic value, and Hurwicz criterion), in which costs and demands are studied as uncertain variables. They also addressed a sustainability problem by fixing the maximal CO_2 emissions that cannot be exceeded by the whole transportation system.

Hence, similarly to Tadei et al. (2012), our goal consists in finding a usage plan of a set of transshipment facilities that maximizes the total net transportation utility, but with a crucial focus on the consequences (in terms of transshipment capacity and loss) of possible unforeseen events in the transshipment synchronization. For this reason, we introduce a new variant of the transshipment problem where, for the first time in the literature, the uncertainty coming from the capacity of the facilities and the handling utilities are considered simultaneously. The uncertainty regarding the loss of capacity is explicitly considered to model possible leftovers at the facilities due to non-correctly synchronized operations. Finally, we will consider a multi-period problem instead of a single-period one since, in strategic/tactical planning, decisions must be taken on the long or medium-term horizon, which implies to have a wider view of the problem and not only on the next imminent set of operations. Considering only one period would possibly lead to design a network that can perform well in many cases, but then having catastrophic losses of revenue in other cases. Hence, having more periods allows to have a more precise view of the possible realization of the uncertainty and to avoid as much as possible the worst consequences. In conclusion, it becomes important to locate facilities to enable synchronization mechanism, so to provide a good trade-off between locating capacities that remain unused and the risk of leftovers.

First, we provide a Stochastic Programming (SP) formulation of the problem in order to evaluate the possible potentialities to explicitly consider uncertainty. More precisely, according to realistic decision making processes in logistics, a two-stage SP model with recourse is proposed where the location of facilities is decided at the first stage while the allocation of the freight flows is decided at the second stage. In order to achieve a model that can be practically solved through mathematical programming techniques, a deterministic equivalent formulation of the stochastic model is derived by assuming certain probability distributions for the random variables and discretizing them through the use of a finite (although large) set of scenarios. The number of scenarios required to discretize the random variables with a good approximation is experimentally found by performing stability analysis. Then, several standard SP measures (VSS, EVPI, LUSS, and LUDS) are calculated and analyzed to show that the problem is worth to be studied, since an explicit consideration of the stochasticity can lead to conspicuous gains. Finally, to overcome the computational burden of solving the problem by state-of-the-art commercial solvers (like CPLEX or GUROBI), we introduce several matheuristic procedures based on the well-known Progressive Hedging (PH) approach Rockafellar and Wets (1991). The accuracy and efficiency of the proposed algorithms are tested over a large set of representative instances.

The rest of paper is organized as follows. In Section 0.2, we introduce the problem and we present its stochastic (a two-stage model with recourse) and deterministic equivalent formulations. In Section 0.3, we analyze the effects of introducing stochasticity in the problem through the calculation of insightful SP measures. The generation of the test instances and the stability analysis are also described there. In Section 0.4, we present the scenario decomposition required to implement the PH, the steps of the algorithm, the initialization and update of PH parameters. Section 0.5 provides the description of some PH variants based on heuristic ideas to improve the computational performance. In Section 0.6, we discuss the results of the computational experiments regarding the PH-based heuristic algorithms. Section 0.7 proposes conclusions of the work and future developments.

0.2 Problem definition and mathematical models

Let us consider the following sets

- I: set of origins
- J: set of destinations
- K: set of potential transshipment locations

- T: set of time periods representing the optimization horizon (say, a week).

Let P_i be the flow of freight supplied by origin $i \in I$ and Q_j be the demand required by destination $j \in J$. Given a unit of freight shipped from origin $i \in I$ to destination $j \in J$ via transshipment location $k \in K$, $v_{ijk}(\xi)$ represents its unknown marginal shipping utility, where ξ is a random variable defined over a certain probability space. On the other hand, using a facility $k \in K$ has a fixed cost f_k (contract cost) and for each time period $t \in T$ an utilization cost h_k (handling cost). Moreover, the freight flow transshipped at facility $k \in K$ cannot exceed its maximum deterministic capacity C_k. Being affected by uncertainty, we further define a stochastic capacity reduction $\widehat{C}_k^t(\xi) \geq 0$ for a certain facility $k \in K$ and in a specific time period $t \in T$. This capacity fluctuations implicitly model the freight leftover in a facility because of possible non-synchronization or handling delays in previous time periods.

The stochastic multi-period transshipment location-allocation problem aims at finding a facility location and flow allocation plan that maximizes the total net transportation utility given by the total shipping utility minus the total contract and handling costs of the located facilities, subject to capacity constraints.

In the following, without loss of generality, we will assume that the system is balanced, i.e. $\sum_{i \in I} P_i = \sum_{j \in J} Q_j$ (standard methods for balancing a network can be found in Ahuja et al., 1993).

0.2.1 A two-stage Stochastic Programming formulation

In the following we provide a two-stage SP formulation of the problem. Readers are referred to Birge and Louveaux (1997) and King and Wallace (2012) for an overview on this modeling paradigm. According to a realistic decision making process, the first stage is about deciding which facilities should be used (tactical planning), while the second-stage recourse actions are about how to manage the flows at the transshipment hubs (operational planning). Moreover, to always maintain the feasibility of the problem with respect to the supply/demand constraints, the transportation company is also allowed to pay for an external effort to ship the freight. For the first stage, we define a boolean variable y_k^t, $\forall t \in T, \forall k \in K$, taking value 1 if facility k is used for a transshipment in time period t, and 0 otherwise. Moreover, we define a boolean variable x_k, $\forall k \in K$, taking value 1 if facility k is used for a transshipment in any time period, and 0 otherwise. Lastly, we define $\mathbb{E}[Q(\mathbf{y}, \xi)]$ as the expected total shipping utility on the whole time horizon.

Then, our multi-period TLAP can be modeled as follows

$$\max_{x,y} \quad \mathbb{E}[Q(\mathbf{y}, \xi)] - \sum_{k \in K} f_k x_k - \sum_{k \in K} \sum_{t \in T} h_k y_k^t \tag{1}$$

subject to

$$y_k^t \leq x_k, \qquad k \in K, t \in T \tag{2}$$

$$x_k \in \{0,1\}, \qquad k \in K \tag{3}$$

$$y_k^t \in \{0,1\}, \qquad k \in K, t \in T. \tag{4}$$

Constraints (2) states that a facility is used only if a transshipment is done in at least one of the time period. Constraints (3) and (4) are binary condition on the variables. The objective function (1) expresses the maximization of the total net transportation utility given by the expected total shipping utility minus the total contract and handling costs of the located facilities. The function $Q(\mathbf{y}, \xi)$ corresponds to the objective function value of the following second-stage optimization problem

$$Q(\mathbf{y}, \xi) := \max_{z,w} \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} \sum_{t \in T} \left[v_{ijk}(\xi) \left(z_{ijk}^t - w_{ijk}^t \right) - c_{ijk} \, w_{ijk}^t \right] \tag{5}$$

subject to

$$\sum_{j \in J} \sum_{t \in T} \sum_{k \in K} z_{ijk}^t = P_i, \qquad i \in I \tag{6}$$

$$\sum_{i \in I} \sum_{t \in T} \sum_{k \in K} z_{ijk}^t = Q_j, \qquad j \in J \tag{7}$$

$$\sum_{i \in I} \sum_{j \in J} (z_{ijk}^t - w_{ijk}^t) \leq [C_k - \widehat{C}_k^t(\xi)] \, y_k^t, \qquad k \in K, t \in T \tag{8}$$

$$\sum_{i \in I} \sum_{j \in J} z_{ijk}^t \leq C_k \, y_k^t, \qquad k \in K, t \in T \tag{9}$$

$$w_{ijk}^t \leq z_{ijk}^t, \qquad i \in I, j \in J, k \in K, t \in T \tag{10}$$

$$z_{ijk}^t, w_{ijk}^t \geq 0, \qquad i \in I, j \in J, k \in K, t \in T \tag{11}$$

where variables z_{ijk}^t, for each $i \in I, j \in J, k \in K, t \in T$, represent the freight from origin i to destination j transshipped via facility k in time period t, while variables w_{ijk}^t, for each $i \in I, j \in J, k \in K, t \in T$, represent the freight transshipment from origin i to destination j that is done in facility k in period t by an external transportation company at a unit cost c_{ijk}. The objective function (5) aims at maximizing the total shipping utility given by the freight correctly transshipped at the facilities over the time horizon minus the cost to use external shipments.

Constraints (6) and (7) ensure that supplies in all origins are collected and that all demands in all destinations are satisfied. Constraints (8) ensure that, in each time period and for each located facility, shipments does not exceed the available capacity. Constraints (9) ensure that, in each time period flows pass only through facility that we decided to use. Constraints (10) ensure that the external shipments never exceed the total freight leftover. Finally, constraints (11) are non-negative conditions on z and w variables.

0.2.2 Deterministic Equivalent Problem

In the Deterministic Equivalent Problem (DEP), instead of working with stochastic variables, we explicitly deal with a set S of scenarios, each scenario $s \in S$ having probability π^s to occur (such that the standard axiom holds, i.e. $\sum_{s\in S} \pi^s = 1$). With this new formulation the second stage variables z_{ijk}^{ts} and w_{ijk}^{ts} become dependent on the scenarios, as well as the stochastic variables v_{ijk}^{s} and $\widehat{C}_k^{ts}$. Resorting to the solution of a DEP is, in the most cases, the only way to duly approximate the SP model (see Wallace and Ziemba, 2005).

The Deterministic Equivalent model is as follows

$$\max \sum_{s\in S} \pi^s \sum_{i\in I}\sum_{j\in J}\sum_{k\in K}\sum_{t\in T} \left[v_{ijk}^{s}\left(z_{ijk}^{ts} - w_{ijk}^{ts}\right) - c_{ijk}\, w_{ijk}^{ts}\right] - \sum_{k\in K} f_k x_k - \sum_{k\in K}\sum_{t\in T} h_k y_k^t \qquad (12)$$

subject to

$$y_k^t \le x_k, \qquad k \in K, t \in T \qquad (13)$$

$$\sum_{j\in J}\sum_{t\in T}\sum_{k\in K} z_{ijk}^{ts} = P_i, \qquad i \in I, s \in S \qquad (14)$$

$$\sum_{i\in I}\sum_{t\in T}\sum_{k\in K} z_{ijk}^{ts} = Q_j, \qquad j \in J, s \in S \qquad (15)$$

$$\sum_{i\in I}\sum_{j\in J}(z_{ijk}^{ts} - w_{ijk}^{ts}) \le (C_k - \widehat{C}_k^{ts})\, y_k^t, \qquad k \in K, t \in T, s \in S \qquad (16)$$

$$\sum_{i\in I}\sum_{j\in J} z_{ijk}^{ts} \le C_k\, y_k^t, \qquad k \in K, t \in T, s \in S \qquad (17)$$

$$w_{ijk}^{ts} \le z_{ijk}^{ts}, \qquad i \in I, j \in J, k \in K, t \in T, s \in S \qquad (18)$$

$$x_k \in \{0,1\}, \qquad k \in K \qquad (19)$$

$$y_k^t \in \{0,1\}, \qquad k \in K, t \in T \qquad (20)$$

$$z_{ijk}^{ts}, w_{ijk}^{ts} \ge 0, \qquad i \in I, j \in J, k \in K, t \in T, s \in S. \qquad (21)$$

The constraints (13)–(21) correspond to the constraints of the stochastic model (2)–(4) and (6)–(11) exploded by scenarios when depending by stochastic parameters. The objective function (12) maximizes the total net transportation utility given by the total shipping utility over all possible realizations of the scenarios minus the total contract and handling costs of the located facilities.

The DEP becomes a mixed-integer linear problem, which is in general difficult to solve for real-life instances. Moreover, the complexity of the problem also grows with the cardinality of S, i.e., the number of scenarios used to discretize the probability distributions of the random variables.

0.3 Analysis of the stochasticity

In this section, we want to analyze the economical impact of explicitly consider stochasticity in our multi-period TLAP instead of using classical deterministic estimators such as the expected values for approximating the stochastic parameters. Since our problem incorporates two main sources of uncertainty (transshipment capacity and handling utility), we also assess the impact of each source separately. Moreover, we consider some properties of the solutions obtained by the deterministic estimation, in order to derive useful algorithmic insights for the efficient solution of the stochastic problem. In Section 0.3.1, we describe how deterministic instances are generated and stochastic parameters are modeled. Then, in Section 0.3.2, we presents the results of the stability analysis, which helped us to define the number of scenarios needed to calculate the SP measures presented in Section 0.3.3.

0.3.1 Instances generation

In our experimental tests we consider instances with different size, which is defined by the combination of the number of origins $|I|$, destinations $|J|$, transshipment facilities $|K|$, and time periods $|T|$. To be consistent with the literature, the generation procedure is adapted from Tadei et al. (2012) (in which a similar problem is studied). However, we propose some modifications in order to take into account the multi-periodic structure of the problem.

For each instance, the representative parameters are generated as follows (we use the notation $\mathcal{U}[a, b]$ to represent a *Uniform* distribution in the range $[a, b]$)

- the flow supply P_i, $i \in I$, is drawn from $\mathcal{U}[900|T|, 1000|T|]$;
- the flow demand Q_j, $j \in J$,, is drawn from $\mathcal{U}[0.5\bar{Q},\ \bar{Q}]$, where $\bar{Q} = \dfrac{\sum_{i \in I} P_i}{|J|}$. The demand in the last destination is simply adjusted to ensure that the system is balanced;
- the expected deterministic capacity C_k is drawn from different distribution ranges to create three different types of facilities (small-, medium-, and large-sized). For each size, about $\frac{1}{3}|K|$ facilities are generated. The distributions used for small-, medium-, and large-sized facilities are $\mathcal{U}[4\bar{C},\ 8\bar{C}]$, $\mathcal{U}[8\bar{C},\ 12\bar{C}]$, and $\mathcal{U}[12\bar{C},\ 16\bar{C}]$, respectively, where $\bar{C} = \dfrac{\sum_{j \in J} Q_j}{|K||T|}$;
- the expected deterministic utility u_{ijk}, $i \in I$, $j \in J$, $k \in K$, is drawn from $\mathcal{U}[u_{min}, 10u_{min}]$ with $u_{min} = 1$;
- the contract f_k and handling h_k costs, $k \in K$, are equal to $C_k \dfrac{0.3\, u_{tot}}{|I||J||K|}$, where $u_{tot} = \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} u_{ijk}$;

- unitary cost c_{ijk} of using an external transportation company, $i \in I$, $j \in J$, $k \in K$, is equal to $1.3\ (\bar{u}_k + u_{ijk} + \dfrac{f_k + h_k}{C_k})$, where $\bar{u}_k$ is the average utility for each facility, i.e. $\bar{u}_k = \dfrac{\sum_{i\in I}\sum_{j\in J} u_{ijk}}{|I||J|}$. This is done to avoid pathological situations in which it would be more convenient to use an external carrier instead of the predefined transportation network.

Given any deterministic instance generated as above, we derive the data dependent on each scenario $s \in S$ as follows. The realized utility is calculated for each $i \in I, j \in J, k \in K$ as $v^s_{ijk} = u_{ijk} + v^s$, where v^s is drawn from a *Gumbel* distribution having a location parameter $\mu = \bar{u}_k$ and a scale parameter $\beta = 0.32\ \bar{u}_k$ in the range $[u_{min}, 20\ u_{min}]$. Instead, the realized loss of capacity $\widehat{C}^{ts}_k$, for each $k \in K, t \in T$, is drawn from a Normal distribution having a mean value $\mu = 0.3 * C_k$ and a standard deviation $\sigma = 2C_k$ in the range $[0, C_k]$. Basically, we are simulating that, on average, the whole capacity of the facility will not be available and the probability of having almost all the capacity is higher than having no space at all.

0.3.2 Stability analysis

We performed an in-sample stability on 10 different instances with different combinations of $|I|$, $|J|$, $|K|$, and $|T|$. For each instance, we calculated the value of the recourse problem (RP) by solving the Deterministic Equivalent Problem (see Section 0.2.2), and then we computed the percentage ratios between the standard deviation and the mean of the objective function of 10 runs in which the stochastic variables are drawn using different seeds. We repeated this calculation for all the quantity of scenarios $|S|$ that we want to test (5, 10, 25, 50, 75, 100). The DEP solutions are obtained by applying the CPLEX solver on the plain DEP formulation, with a MIP gap of 0.5% and a Benders strategy. The results of this analysis are presented in the graph shown in Figure 1 and can help us to determine the quantity of scenarios needed to have a stable enough RP solution. To take this decision we must consider that we used a MIP gap of 0.5% that can cause small errors to calculate the ratios. Even if 50 and 75 scenarios seem to have already good ratios, we preferred to use 100 scenarios to ensure stability that guarantees a ratio under 0.5%.

0.3.3 Stochastic Programming measures

In this section we present the SP measures of three different type of tests: one considering both stochastic utilities and losses of capacity (Test A), the second one considering only stochastic utilities (Test B), and the last one considering only stochastic losses of capacity (Test C). Before presenting the tests we briefly describe the measures computed by comparing the value of the RP solution to other solutions.

The value of the stochastic solution VSS, introduced by Birge (1982), is a SP measure largely used in the analysis of stochastic models, to understand the value of studying the stochastic model

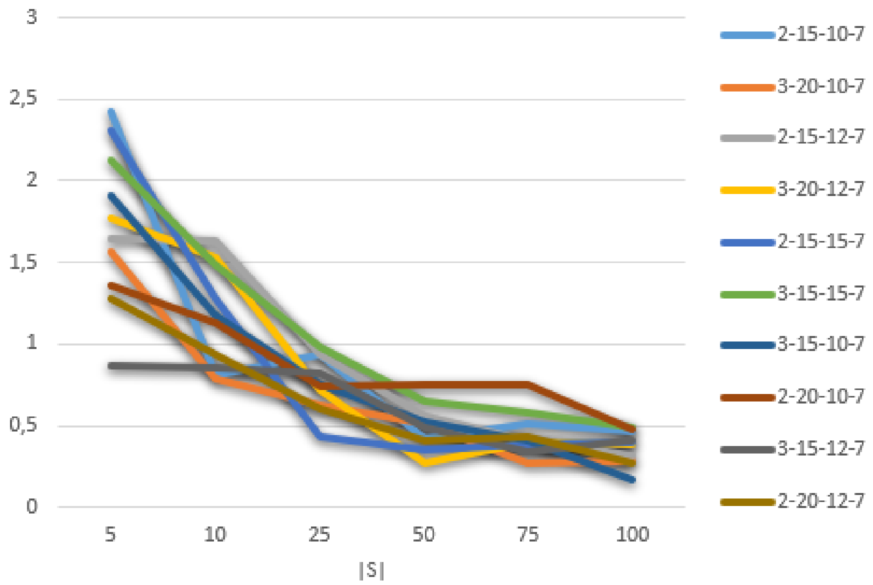

Figure 1: Percentage ratio between the standard deviation and the mean of the objective function of 10 instances ($|I|-|J|-|K|-|T|$) over 10 runs.

with respect to the deterministic one. First, the expected value problem (EV) is computed by solving the deterministic problem of the average scenario. Then, the expected value solution (EEV) is computed by fixing the first-stage variables x and y to the value of the EV solution and solving the deterministic equivalent problem. Lastly, VSS can be calculated by comparing the EEV solution with the RP solution ($VSS\% = 100\ (RP - EEV)\ /\ RP$). Another largely studied SP measure is the expected value of the perfect information (EVPI), presented in the work of Dempster (1981), that show the value of possible investments to forecast uncertainties. The EVPI is calculated by comparing the RP solution with the wait-and-see (WS) solution, calculated by computing the mean of the deterministic model solved separately for each scenario ($EVPI\% = 100\ (WS - RP)\ /\ RP$). We also present the loss using the skeleton solution (LUSS) and the loss of upgrading the deterministic solution (LUDS), by referring to the work of Maggioni and Wallace (2013). These measures are not used frequently in literature, but they can provide more insights about the deterministic solution and the characteristics of the problem. LUSS is calculated by comparing the RP solution with the expected skeleton solution value (ESSV), that is computed by fixing x and y when they assume the value of the lower bound in the EV solution ($LUSS\% = 100\ (RP - ESSV)\ /\ RP$). Instead, LUDS is calculated by comparing the RP with the expected input value (EIV), computed by fixing the lower bounds of x and y to the value of the EV solution ($LUDS\% = 100\ (RP - EIV)\ /\ RP$). Since all first-stage variables are binary, ESSV was calculated by excluding all the facilities that were not selected in the EV and EIV solutions by preventively selecting all the opened facilities of the EV solution.

Table 1: SP measures of 10 instances for test A (stochastic utilities $v_{ijk}(\xi)$ and losses of capacity $\widehat{C}_k^t(\xi)$)

Instance				Test A			
$\|I\|$	$\|J\|$	$\|K\|$	$\|T\|$	**VSS%**	**EVPI%**	**LUSS%**	**LUDS%**
2	15	10	7	9.21	22.75	9.21	1.04
3	20	10	7	8.32	21.98	8.32	0.35
2	15	12	7	9.81	26.76	9.81	1.05
3	20	12	7	8.50	24.27	8.50	1.45
2	15	15	7	7.79	21.02	7.79	0.41
3	15	15	7	7.16	24.21	7.16	0.59
3	15	10	7	7.94	21.81	7.94	0.47
2	20	10	7	9.34	24.84	9.34	1.10
3	15	12	7	9.45	27.24	9.45	1.76
2	20	12	7	9.11	21.85	9.11	0.58
			Avg	8.66	23.67	8.66	0.88
			Min	7.16	21.02	7.16	0.35
			Max	9.81	27.24	9.81	1.76

First, we analyse the results of Test A in which the two stochastic variables are studied together. The high VSS percentages show that the deterministic solution does not perform well and considering the RP solution can lead to an increase on revenues. In fact, the EV solution tends to be more conservative and use the least number of facilities to ship all the demand, causing a great amount of leftovers when stochastic losses of capacity are considered to compute the EEV solution. Then, we can notice that LUSS and VSS are always the same, that implies that $EEV = ESSV$. In practice, the EV solution tends to use the least possible number of facilities. To calculate the ESSV we can only set to 0 first-stage variables that were set to 1 in the EV solution. So, to avoid recourse costs the ESSV solution will select the same facilities of the EV solution. This is true also in tests B and C. Even if the EV solution does not perform well, LUDS percentages tend to 0 indicating that the deterministic solution can be used as a good lower bound for the stochastic one. Hence, the computation of the EIV solution starts with a subset of facilities of the RP solution already opened and then new ones are selected to find a good balance between facility cost and the risk of having leftovers. Lastly, the EVPI shows that at most we should spend that percentage of the RP solution to predict which scenario we will face. In this case, it is clear that by knowing the exact realization of the scenarios we will increase our profit.

In test B, we can see that the percentage of the VSS tends to 0 and this seems reasonable because leftovers has the greater negative impact on revenues and the recourse function adds costs if we are exceeding the available capacity. Hence, by considering losses of capacity as deterministic parameters is sufficient to use the EEV results to avoid leftovers. This implies that considering only the stochastic utilities does not bring great economical benefits in comparison to the average utilities. Moreover, we can see that also LUDS and VSS are almost the same for all instances and that indicates that the deterministic solution cannot be upgraded. Since the EV solution already

Table 2: SP measures of 10 instances for test B (stochastic utilities $v_{ijk}(\xi)$) and C (stochastic losses of capacity $\widehat{C}_k^t(\xi)$)

Instance	Test B				Test C			
$\|I\|$ $\|J\|$ $\|K\|$ $\|T\|$	VSS%	EVPI%	LUSS%	LUDS%	VSS%	EVPI%	LUSS%	LUDS%
2 15 10 7	0.49	9.68	0.49	0.47	6.58	11.72	6.58	0.34
3 20 10 7	0.39	9.56	0.39	0.39	6.47	10.78	6.47	0.50
2 15 12 7	0.92	12.48	0.92	0.85	5.74	9.76	5.74	0.55
3 20 12 7	0.71	12.65	0.71	0.71	5.58	10.06	5.58	0.39
2 15 15 7	0.32	10.80	0.32	0.32	5.21	9.17	5.21	0.23
3 15 15 7	0.56	12.65	0.56	0.56	5.18	9.78	5.18	0.48
3 15 10 7	0.27	9.87	0.27	0.27	6.70	11.36	6.70	0.41
2 20 10 7	0.31	11.0	0.31	0.31	6.73	11.85	6.73	0.34
3 15 12 7	0.46	14.03	0.46	0.46	5.78	10.33	5.78	0.37
2 20 12 7	0.19	9.63	0.19	0.15	5.48	9.59	5.48	0.25
Avg	0.46	11.23	0.46	0.45	5.95	10.44	5.95	0.39
Min	0.19	9.56	0.19	0.15	5.18	9.17	5.18	0.23
Max	0.92	14.03	0.92	0.85	6.73	11.85	6.73	0.55

selected enough capacity, it does not seem convenient to have unnecessary capacity and open more facilities, even if they have better utilities in a stochastic environment. Hence, adding more costs related to facilities is not compensated by better utilities. Instead, the results of test C show that losses of capacity has the greater impact on VSS. Moreover, we can also see that the characteristic of having an EV solution that is close to the perfect upgreadability ($EIV = RP$) is due to losses of capacity. The values of the EVPI for tests B and C are more of less half of the ones in Test A, showing that both stochastic variables have an impact on that measure.

The results of test A are sort of a combination of the results of tests B and C. In fact, we can see that all the percentages of all the measures tend to increase. Even if utilities lead to small values of VSS when considered individually, when we include both variables as stochastic the VSS imputed to losses of capacity tends to increase. In conclusion, this analysis shows that the stochastic problem with both stochastic utilities and losses of capacity is interesting to be studied and can bring to great economical benefits.

Unlucky, CPLEX and other commercial solvers do not perform well in terms of computational time to solve the RP model, especially when we use larger instances. Hence, we decided to develop a PH approach to overcome this limit. Moreover, the analysis of LUDS showed that it is possible, in some cases, to derive first-stage solutions very close to the optimal ones. The EIV solution can be computed quite fast and it could be a good heuristic on its own. However, we cannot ensure that, by making simple modifications to the instances and scenarios generation, and to the distribution of the stochastic variables, the LUDS will not get worse. In fact, the results of test C show that by considering the two combined sources of uncertainty the average LUDS increases and gets closer to 1-2%. Based on these insights, we decided to improve the proposed PH algorithm by adding a heuristic that uses the EIV solution (see Section 0.5.2) in order to provide better lower bounds.

0.4 Progressive Hedging-based heuristic algorithm

The basic PH algorithm has been introduced by Rockafellar and Wets (1991). It is based on the decomposition of the problem by scenarios, after having relaxed the non-anticipativity constraints, and on a subgradient method to converge to the first-stage decisions consensus. Variant of this algorithm have been used in recent studies with good results in terms of approximation of the exact solution and computational time on various kinds of problems. For instance PH has been used to solve problems regarding social engagement paradigm (Fadda et al., 2019), scheduling and planning in industry (Peng et al., 2019), selection of suppliers (Manerba and Perboli, 2019), expansion plans for electric vehicle charging station (Kabli et al., 2019), and fixed-charge capacitated multicommodity network design (Crainic et al., 2011). Considering the performance of the PH algorithm we decided to implement the generic version and some heuristics to solve large instances in which commercial solvers required too much time to find an optimal solution.

0.4.1 Scenario decomposition

To implement a progressive hedging approach we need to make our model decomposable by scenarios by making the first-stage variables scenario dependent. Hence, the model can be rewritten as follows

$$\max \sum_{s\in S} \pi^s \sum_{i\in I}\sum_{j\in J}\sum_{k\in K}\sum_{t\in T} \left[v^s_{ijk}\left(z^{ts}_{ijk} - w^{ts}_{ijk}\right) - c_{ijk}\, w^{ts}_{ijk}\right] - \sum_{k\in K} f_k x^s_k - \sum_{k\in K}\sum_{t\in T} h_k y^{ts}_k \tag{22}$$

subject to

$$y_k^{ts} \leq x_k^s, \qquad k \in K, t \in T, s \in S \tag{23}$$

$$\sum_{j\in J}\sum_{t\in T}\sum_{k\in K} z_{ijk}^{ts} = P_i, \qquad i \in I, s \in S \tag{24}$$

$$\sum_{i\in I}\sum_{t\in T}\sum_{k\in K} z_{ijk}^{ts} = Q_j, \qquad j \in J, s \in S \tag{25}$$

$$\sum_{i\in I}\sum_{j\in J}(z_{ijk}^{ts} - w_{ijk}^{ts}) \leq (C_k - \widehat{C}_k^{ts})\, y_k^{ts}, \qquad k \in K, t \in T, s \in S \tag{26}$$

$$\sum_{i\in I}\sum_{j\in J} z_{ijk}^{ts} \leq C_k\, y_k^{ts}, \qquad k \in K, t \in T, s \in S \tag{27}$$

$$w_{ijk}^{ts} \leq z_{ijk}^{ts}, \qquad i \in I, j \in J, k \in K, t \in T, s \in S \tag{28}$$

$$x_k^s = \bar{x}_k, \qquad k \in K, s \in S \tag{29}$$

$$y_k^{ts} = \bar{y}_k^t, \qquad k \in K, t \in T, s \in S \tag{30}$$

$$x_k^s \in \{0,1\}, \qquad k \in K, s \in S \tag{31}$$

$$y_k^{ts} \in \{0,1\}, \qquad k \in K, t \in T, s \in S \tag{32}$$

$$z_{ijk}^{ts}, w_{ijk}^{ts} \geq 0, \qquad i \in I, j \in J, k \in K, t \in T, s \in S. \tag{33}$$

To ensure that x and y keep the same values in all scenarios we introduced constraints (29) and (30), where $\bar{x}_k \in \{0,1\}$ are first-stage decisions on the selection of each facility $k \in K$, whereas $\bar{y}_k^t$ are first-stage decisions on the usage of each facility $k \in K$ in any time period $t \in T$. Then, we use an Augmented Lagrangian technique to relax constraints (29) and (30), by introducing the free Lagrangian multipliers λ_k^s and μ_k^{ts} and the penalty factors $\rho_1 \geq 0$ and $\rho_2 \geq 0$. Finally, the scenario-separable model for each scenario $s \in S$ can be rewritten as follows

$$\begin{aligned}
\max \pi^s \sum_{i\in I}\sum_{j\in J}\sum_{k\in K}\sum_{t\in T} \left[v_{ijk}^s\ (z_{ijk}^{ts} - w_{ijk}^{ts}) - c_{ijk}\ w_{ijk}^{ts}\right] + \\
- \sum_{k\in K} x_k^s (f_k + \lambda_k^s + \frac{\rho_1}{2} - \rho_1 \bar{x}_k) + \bar{x}_k (\frac{\rho_1}{2} - \lambda_k^s) + \\
- \sum_{k\in K}\sum_{t\in T} y_k^{ts} (h_k^t + \mu_k^{ts} + \frac{\rho_2}{2} - \rho_2 \bar{y}_k^t) + \bar{y}_k^t (\frac{\rho_2}{2} - \mu_k^{ts})
\end{aligned} \tag{34}$$

subject to

$$
\begin{aligned}
y_k^{ts} &\leq x_k^s, && k \in K, t \in T && (35)\\
\sum_{j\in J}\sum_{t\in T}\sum_{k\in K} z_{ijk}^{ts} &= P_i, && i \in I && (36)\\
\sum_{i\in I}\sum_{t\in T}\sum_{k\in K} z_{ijk}^{ts} &= Q_j, && j \in J && (37)\\
\sum_{i\in I}\sum_{j\in J}(z_{ijk}^{ts} - w_{ijk}^{ts}) &\leq [C_k - \widehat{C}_k^{ts}]\, y_k^{ts}, && k \in K, t \in T && (38)\\
w_{ijk}^{ts} &\leq z_{ijk}^{ts}, && i \in I, j \in J, k \in K, t \in T && (39)\\
x_k^s &\in \{0,1\}, && k \in K && (40)\\
y_k^{ts} &\in \{0,1\}, && k \in K, t \in T && (41)\\
z_{ijk}^{ts}, w_{ijk}^{ts} &\geq 0, && i \in I, j \in J, k \in K, t \in T && (42)
\end{aligned}
$$

0.4.2 PH Algorithm

Algorithm 1 PH algorithm.

```
τ = 0
Solve the EV problem to find the initial x̄_k and ȳ_k^t
Initialize all the Lagrangian multipliers and penalties (Section 0.4.3)
τ = 1
while τ ≤ τ_max do
    for each scenario s ∈ S do
        Solve the corresponding sub-problem
    end for
    Update x̄_k and ȳ_k^t
    if full consensus is met then
        break
    else
        Update the Lagrangean multipliers and penalties (Section 0.4.3)
    end if
    τ = τ + 1
end while
Fix the first-stage variables for which the consensus is met and solve the model
```

The PH algorithm (Algorithm 1) can be subdivided into three phases: initialization, consensus, and solving. In the description of the algorithm, we use the symbol τ to indicate the iteration number which is initialized to 0. In the initialization phase, we need to solve the model by using the average scenario by computing the mean of all scenarios (Step 2). The solution found is used to build the temporary global solution (TGS). Then, we need to initialize the Lagrangian multipliers and the penalties (Step 3). After the initialization phase we pass to the consensus phase (Steps 4-16), that can last for a maximum of τ_{max} iterations (Step 5). In each iteration we solve all the scenario sub-problems (Steps 6-8) and we use the results to update the TGS (Step 9). Then we check if the consensus is met (Step 10) and we terminate the iterations in that case (Step 11), otherwise we update the Lagrangian multipliers and penalties (Step 13). After the consensus phase we move to the solving phase, in which we fix the first-stage variables that reached the consensus and we solve the simplified MILP problem (Step 17).

0.4.3 Initialize and update PH parameters

PH parameters ($\bar{x}_k^{(\tau)}$, $\bar{y}_k^{t(\tau)}$, $\lambda_k^{s(\tau)}$, $\mu_k^{ts(\tau)}$, $\rho_1^{(\tau)}$, $\rho_2^{(\tau)}$) must be initially set and then updated at every iteration τ. Initially ($\tau = 0$), we set the TGS by using the result values of the EV solution ($\bar{x}_k^{(0)} = x_k^{(EV)}, k \in K$ and $\bar{y}_k^{t(0)} = y_k^{t(EV)}, k \in K, t \in T$), hence $\bar{x}_k^{(0)}$ and $\bar{y}_k^{t(0)}$ can be either 0 or 1. The Lagrangian multipliers start with a value equal to 0 ($\lambda_k^{s(0)} = 0, k \in K, s \in S$ and $\mu_k^{ts(0)} = 0, k \in K, t \in T, s \in S$). Lastly, the starting values for penalties ρ_1 and ρ_2 must be small and positive ($\rho_1^{(0)} = 0.005$ and $\rho_2^{(0)} = 0.01$).

At the end of each iteration we update the parameters. First, we need to calculate the TGS (43)-(44) to check the consensus and because we will need these values to update the multipliers (45)-(46). Then the penalties are updated by using constant factors α_1 and α_2 that should be strictly greater than 1, (47)-(48). We set $\alpha_1 = 1.5$ and $\alpha_2 = 3$.

$$\bar{x}_k^{(\tau)} = \sum_{s \in S} \pi^s x_k^{s(\tau)}, \qquad \tau > 0, k \in K \tag{43}$$

$$\bar{y}_k^{t(\tau)} = \sum_{s \in S} \pi^s y_k^{ts(\tau)}, \qquad \tau > 0, k \in K, t \in T \tag{44}$$

$$\lambda_k^{s(\tau)} = \lambda_k^{s(\tau-1)} + \rho_1^{(\tau-1)}(x_k^{s(\tau)} - \bar{x}_k^{(\tau)}), \qquad \tau > 0, k \in K, s \in S \tag{45}$$

$$\mu_k^{ts(\tau)} = \mu_k^{ts(\tau-1)} + \rho_2^{(\tau-1)}(y_k^{ts(\tau)} - \bar{y}_k^{t(\tau)}), \qquad \tau > 0, k \in K, t \in T, s \in S \tag{46}$$

$$\rho_1^{(\tau)} = \alpha_1 \rho_1^{(\tau-1)}, \qquad \tau > 0, k \in K, t \in T \tag{47}$$

$$\rho_2^{(\tau)} = \alpha_2 \rho_2^{(\tau-1)}, \qquad \tau > 0, k \in K, t \in T \tag{48}$$

The initial set of penalties ρ_1 and ρ_2 and the choice of the updating factors α_1 and α_2 can be used to tune the algorithm. We decided to have smaller penalties for the parameters related to x variables, since excluding a facility makes impossible its utilization for any time period.

0.5 PH variants and acceleration strategies

0.5.1 PH-Bounds

To improve the performance of the PH algorithm we implemented a variant named `PH-Bounds` (Algorithm 2). Compared to the classic PH presented in Section 0.4.2, we added few steps to force the consensus by fixing variables with a TGS outside an upper and a lower bounds, called respectively *exclusion bound* (EB) and *consensus bound* (CB). The rationale of this variant of the algorithm is to try to simplify at each iteration the underlying model to be solved by the PH main operations.

Algorithm 2 PH-Bounds algorithm.

```
1:  τ = 0
2:  Solve the EV problem to find the initial x̄_k and ȳ_k^t
3:  Initialize all the Lagrangian multipliers and penalties (Section 0.4.3)
4:  Initialize EB and CB
5:  τ = 1
6:  while τ ≤ τ_max do
7:     for each scenario s ∈ S do
8:        Solve the corresponding sub-problem
9:     end for
10:    Update x̄_k and ȳ_k^t
11:    if full consensus is met then
12:       break
13:    else
14:       Update the Lagrangean multipliers and penalties (Section 0.4.3)
15:       Fix to 1 all x and y with x̄ ≥ CB and ȳ ≥ CB
16:       Fix to 0 all x and y with x̄ ≤ EB and ȳ ≤ EB
17:       Update EB and CB
18:    end if
19:    τ = τ + 1
20: end while
21: Fix the first-stage variables for which the consensus is met and solve the model
```

First, both bounds must be initialized to a value representing a percentage of consensus (Step 4). During the iterations, every time the consensus is evaluated and is not reached we fix to 1 the variables whose value is not less than CB (Step 15) and to 0 the ones that do not exceed EB (Step

16). To speed up the variables fixing process, EB is increased and CB is decreased at each iteration (Step 17). For our tests, we set a starting percentage of CB to 100% ($CB = 1.0$) and EB to 0% ($EB = 0$). Then, we decrease CB of 2% ($CB^{(\tau)} = CB^{(\tau-1)} - 0.02$) at each iteration, whereas EB is incremented of 2% ($EB^{(\tau)} = EB^{(\tau-1)} + 0.02$).

0.5.2 PH-LUDS

Since we recognize that the values of LUDS tend to the perfect upgreadability, we want to implement a variant of the PH based on a LUDS based heuristic (`PH-LUDS`). In practice, we initialize the TGS with the EIV solution. Then, we execute the PH algorithm and we compare its solution with the EIV one keeping the best one. Using the EIV solution ensures to have already a solution with a good approximation, especially in the case that the termination criterion of the PH leads to solutions far from the optimal.

0.5.3 Primal LP-based heuristic

To have intermediate solutions during the PH we implemented a LP-based heuristic (Algorithm 3) that computes a solution at the end of each iteration if the consensus is not met. This algorithm has a low cost in terms of computational time, because we fix all the binary variables and solve LP problems (that are simple to be solved). This heuristic is used in all the PH variants presented.

Algorithm 3 Primal LP-based heuristic at each iteration

1: Sort x and y according to the consensus of the TGS (higher consensus first)
2: **while** $n_{x=1} \leq n_x^{(EV)}$ **or** next $\bar{x} \geq 0.5$ **do**
3:     Fix the next x to 1
4: **end while**
5: **while** $n_{y=1} \leq 1.2\, n_y^{(EV)}$ **or** next $\bar{y} \geq 0.5$ **do**
6:     **if** the next y as a corresponding $x = 1$ **then**
7:         Fix the next y to 1
8:     **end if**
9: **end while**
10: Fix all the others x and y to 0
11: Solve the LP problem

The first solution is calculated before the iterations start by using the results of EV (or EIV for the `PH-LUDS`). Then, at the end of each iteration (after Step 14 of Algorithm 1), we sort in non-increasing order $x's$ and $y's$ according to the TGS $\bar{x}'s$ and $\bar{y}'s$ (Step 1 of Algorithm 3). Then, we iterate through the list of x, by fixing x to 1 until we do not have at least the same number of facilities opened ($n_{x=1}$) equal to the one in the EV solution ($n_x^{(EV)}$) or still other facilities in the list with $\bar{x}$ rounded to the closer integer equal to 1 (Steps 2-4). We do the same for y but we ensure that we only select y if the corresponding x is selected (Steps 5-9). Moreover, taking into account that the EV solution tends to open facilities in less time periods we ensure that at least $1.2\ n_y^{(EV)}$ are used. We have an exception to this rule for the `PH-LUDS`, in which we want at least the same number of opened facilities of the EIV solution ($n_y^{(EIV)}$). Lastly, we fix to 0 the remaining x and y to ensure that all the binary variables are fixed (Step 10) and we have simply to solve a LP problem (Step 11). Every time we find a solution that is better than the previous, we save that solution and we keep track of the time passed since the PH has started as well as the number of iterations. This will help us to understand the impact of this heuristic on the solutions.

0.6 Computational tests

We want to test the computation performance of the different implementations of the PH by performing tests on two different set of instances. The first set is composed by the instances used for the stability analysis and for the SP measures (small instances).We test them primarily to find out if our methods have a good approximation of the optimum. Instead, the second set is composed by larger instances. We use them to understand the computational performance of the PHs. For both sets, we compare the results of the classic PH and its variants (`PH-Bounds` and `PH-LUDS`) with the CPLEX solver, by setting a time limit of three hours. CPLEX is also used in the PH with a Benders' strategy and a MIP gap of 1%. The MIP gap is set to 5% during the consensus phase to speed up the solving time of the single scenarios problem. Moreover, in all PHs we set the maximum number of iterations to 15 ($\tau_{max} = 15$). In the results, we present for CPLEX the total computation time (t), the time to reach the best solution (ttb), and the gap%. For the three PH methods we present t, ttb, itb, i.e. the number of iterations needed to get the best solution, and the percentage gap of the PH solution with respect to the CPLEX solution ($\Delta\%$). A positive $\Delta\%$ indicates that CPLEX performs better, whereas a negative value indicates that the PH gets a better solution.

In Table 3 we present the results of the CPLEX solver for the set of small instances, whereas in Table 4 the results for the set of large instances. We can notice that on small instances the MIP gap is not too large, indicating that CPLEX can find solutions close to the optimum that can be use as a benchmark for the PHs. Instead for large instances, MIP gaps tend to increase a lot assuming almost always values greater than 10%, in some cases even above 100%, producing solutions very far from the optimal ones. Moreover, three hours are never enough to reach the optimum and complete the simulation.

In Table 5 we present the results of the three variants of the PH algorithm for the set of small instances. Regarding computation time, `PH-Bounds` is the fastest as expected, but its average

Table 3: Results of the CPLEX solver for the set of small instances

Instance				CPLEX		
$\|I\|$	$\|J\|$	$\|K\|$	$\|T\|$	t	ttb	gap%
2	15	10	7	10800	10775	2,98
3	20	10	7	10800	10762	1,63
2	15	12	7	10800	8991	3,89
3	20	12	7	10800	7471	1,96
2	15	15	7	10800	10394	0,29
3	15	15	7	10800	6573	3,03
3	15	10	7	10800	10560	0,78
2	20	10	7	10800	10419	1,9
3	15	12	7	10800	10158	3,77
2	20	12	7	10800	10165	0,74
			Avg	10800	9627	2,10

Table 4: Results of the CPLEX solver for the set of large instances

Instance				CPLEX		
$\|I\|$	$\|J\|$	$\|K\|$	$\|T\|$	t	ttb	gap%
5	30	15	7	10800	9155	11,87
3	25	15	10	10800	9878	91,55
2	20	10	31	10800	9837	31,01
3	30	20	7	10800	6814	117,98
5	30	20	7	10800	10800	564366
3	40	20	7	10800	10531	5,41
3	25	15	31	10800	10800	224642
5	30	20	10	10800	10800	567737
5	40	20	7	10800	10800	413959
3	30	20	31	10800	10800	470676
			Avg	10800	10022	224164

Table 5: Results of the the three PH variants for the set of small instances

Instance				PH				PH-Bounds				PH-LUDS			
$\|I\|$	$\|J\|$	$\|K\|$	$\|T\|$	t	ttb	itb	Δ%	t	ttb	itb	Δ%	t	ttb	itb	Δ%
2	15	10	7	281	55	3	-0,18	213	163	11	1,46	312	46	0	-0,29
3	20	10	7	574	38	1	1,56	351	351	15	-0,12	690	334	7	0,49
2	15	12	7	297	297	15	0,75	225	225	15	1,53	374	374	15	0,75
3	20	12	7	729	50	1	3,15	464	438	14	0,60	908	188	0	0,31
2	15	15	7	331	331	15	1,74	238	238	15	0,60	375	50	0	0,33
3	15	15	7	643	387	11	1,13	433	279	9	0,04	801	164	0	0,35
3	15	10	7	411	411	15	0,85	286	286	15	0,29	475	475	15	0,85
2	20	10	7	364	364	15	0,64	261	180	10	1,56	419	273	10	0,35
3	15	12	7	513	90	3	0,69	345	273	11	0,08	640	182	2	0,69
2	20	12	7	379	324	14	0,85	260	260	15	0,65	439	68	0	0,40
			Avg	452	235	9	1,12	308	269	13	0,67	543	215	5	0,42

ttb is the highest among the algorithms. In fact, `PH-Bounds` tends to find the best solution in an iteration very close to the last one. `PH-LUDS` has the lowest itb because the EIV solution is often the best one (itb = 0). In general, for small instances, all three methods have a good approximation

Table 6: Results of the the three PH variants for the set of large instances

Instance				PH				PH-Bounds				PH-LUDS			
$\|I\|$	$\|J\|$	$\|K\|$	$\|T\|$	t	ttb	itb	Δ%	t	ttb	itb	Δ%	t	ttb	itb	Δ%
5	30	15	7	3002	1631	11	-7,53	1851	1331	10	-8,25	3511	535	0	-8,00
3	25	15	10	1480	1480	15	-84,9	1083	1083	15	-85,6	1724	296	0	-86,7
2	20	10	31	1741	1741	15	-26,1	1181	1181	15	-27,1	2135	408	0	-28,0
3	30	20	7	1865	1865	15	-110,4	1254	1254	15	-110,3	2165	327	0	-111,9
5	30	20	7	4346	470	2	-1023	2651	448	2	-1023	5010	811	0	-1034
3	40	20	7	2859	2120	13	-1,70	1947	1947	15	-2,11	3120	2060	11	-2,20
3	25	15	31	5217	5217	15	-257,4	3791	2580	9	-258,2	7125	1617	0	-258,5
5	30	20	10	5288	5288	15	-722,7	3417	2446	9	-719,6	6672	6672	15	-722,7
5	40	20	7	7097	7097	15	-702,0	4100	4100	15	-698,3	8857	1757	0	-708,1
3	30	20	31	10894	6624	10	-480,7	7224	5145	9	-479,7	14206	2963	0	-485,4
			Avg	4379	3353	13	-341,6	2850	2152	11	-341,2	5453	1745	3	-344,6

to the CPLEX solution and in very few cases they get better results.

The results presented in Table 6 show that for the set of large instances the three PHs outperform CPLEX finding largely better solutions with an average gap above 100%. In terms of total running time, `PH-Bounds` performs better than the other methods, whereas `PH-LUDS` tends to find the best solution in less time and almost always corresponds to the EIV solution. The values of $\Delta\%$ show that `PH-LUDS` finds on average the best solutions, but the other methods find solutions that are very close to the `PH-LUDS` ones.

In general, we can say that the three PH variants have very good performance in terms of both approximation and computation time. We should also consider that the PHs were using CPLEX to find the consensus and that was not so efficient to solve the single deterministic scenarios. In fact, we were using a MIP gap of 5% to speed up the process but that can decrease the quality of the solution. Hence, the methods can be further improved by developing a more efficient algorithm to solve the deterministic problem.

0.7 Conclusions and future developments

In this paper we presented a Transshipmnet Location-Allocation Problem by considering a synchromodal network in which flows are synchronized under the uncertainty of capacities and utilities. The SP measures studied in the paper pointed out that optimizing on a stochastic version of the problem provides better revenues by avoiding leftovers as mush as possible. In fact, the deterministic solution was always underestimating the number of facilities needed to reduce the impact of the worst scenario realizations, i.e. those with the highest losses of capacity. Moreover, studying LUDS was important to find out that by solving the deterministic problem a good subset of facilities is selected making possible to upgrade the solution almost to the stochastic one. This insight was used to develop a PH variant (`PH-LUDS`) and also to implement the primal LP-based heuristic. What will be interesting for the future is to study how other sources of uncertainty (e.g., demand and costs), distributions, and constraints impact on the measures and if the almost perfect upgreadability of the deterministic solution is related to the problem or only to this setting. Regarding computational tests, the PH methods proposed are able to perform far better than CPLEX for large instances in much lower computation time, but also to provide solutions very close to the CPLEX one for smaller instances. The limit of these methods mainly concerns the solving time of the single instances during the consensus phase, that leaded sometimes to high total computation time. To implement one fast exact method approach or even heuristic to solve the single scenarios will be of primarily importance to further improve the PH performance.

Bibliography

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Baldi, M. M., Ghirardi, M., Perboli, G., and Tadei, R. (2012). The capacitated transshipment location problem under uncertainty: A computational study. *Procedia - Social and Behavioral Sciences*, 39:424 – 436.

Birge, J. R. (1982). The value of the stochastic solution in stochastic linear programs with fixed recourse. *Mathematical Programming*, 24(1):314–325.

Birge, J. R. and Louveaux, F. V. (1997). *Introduction to Stochastic Programming*. Springer Verlag, New York.

Cooper, L. (1963). Location-allocation problems. *Operations Research*, 11(3):331–343.

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

Dempster, M. A. H. (1981). The expected value of perfect information in the optimal evolution of stochastic systems. In Arató, M., Vermes, D., and Balakrishnan, A. V., editors, *Stochastic Differential Systems*, pages 25–40, Berlin, Heidelberg. Springer Berlin Heidelberg.

Di Francesco, M., Gaudioso, M., Gorgone, E., and Zuddas, P. (2019). The forwarder planning problem in a two-echelon network. *Networks (in press)*.

Fadda, E., Perboli, G., and Tadei, R. (2019). A progressive hedging method for the optimization of social engagement and opportunistic iot problems. *European Journal of Operational Research*, 277(2):643–652.

Giusti, R., Manerba, D., Bruno, G., and Tadei, R. (2019). Synchromodal logistics: An overview of critical success factors, enabling technologies, and open research issues. *Transportation Research Part E: Logistics and Transportation Review*, 129:92–110.

Jin, J. G., Meng, Q., and Wang, H. (2018). Column generation approach for feeder vessel routing and synchronization at a congested transshipment port. In *Prooceedings of Odysseus2018 - 7th International Workshop on Freight Transportation and Logistics, June 3-8, 2018, Cagliari (Italy)*.

Kabli, M., Quddus, M., Nurre, S., Marufuzzaman, M., and Usher, J. (2019). A stochastic programming approach for electric vehicle charging station expansion plans. *International Journal of Production Economics (in press)*.

King, A. J. and Wallace, S. (2012). *Modeling with Stochastic Programming*. Springer Series in Operations Research and Financial Engineering. Springer-Verlag New York.

Maggioni, F. and Wallace, S. (2013). Analyzing the quality of the expected value solution in stochastic programming. *Annals of Operations Research*, 200:37–54.

Manerba, D. and Perboli, G. (2019). New solution approaches for the capacitated supplier selection problem with total quantity discount and activation costs under demand uncertainty. *Computers & Operations Research*, 101:29–42.

Neves-Moreira, F., Amorim, P., Guimarães, L., and Almada-Lobo, B. (2016). A long-haul freight transportation problem: Synchronizing resources to deliver requests passing through multiple transshipment locations. *European Journal of Operational Research*, 248(2):487–506.

Okiemute, G., Nsikan, A., and Akpan, N. (2017). Transshipment problem and its variants: A review. *Mathematical Theory and Modeling*, 7, No 2:19–32.

Peng, Z., Zhang, Y., Feng, Y., Rong, G., and Su, H. (2019). A progressive hedging-based solution approach for integrated planning and scheduling problems under demand uncertainty. *Industrial and Engineering Chemistry Research*, 58(32):14880–14896.

Pfoser, S., Treiblmaier, H., and Schauer, O. (2016). Critical success factors of synchromodality: Results from a case study and literature review. *Transportation Research Procedia*, 14:1463–1471.

Qu, W., Rezaei, J., Maknoon, Y., and Tavasszy, L. (2019). Hinterland freight transportation replanning model under the framework of synchromodality. *Transportation Research Part E: Logistics and Transportation Review*, 131:308–328.

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

Roohnavazfar, M., Manerba, D., De Martin, J. C., and Tadei, R. (2019). Optimal paths in multi-stage stochastic decision networks. *Operations Research Perspectives*, 6:100124.

Steadieseifi, M., Dellaert, N. P., Nuijten, W., Van Woensel, T., and Raoufi, R. (2014). Multimodal freight transportation planning: A literature review. *European Journal of Operational Research*, 233(1):1–15.

Tadei, R., Perboli, G., and Manerba, D. (2018). A recent approach to derive the multinomial logit model for choice probability. In Daniele, P. and Scrimali, L., editors, *New Trends in Emerging Complex Real Life Problems, AIRO Springer Series: ODS, Taormina, Italy, September 10–13, 2018*, volume 1, pages 473–481. Springer International Publishing.

Tadei, R., Perboli, G., Ricciardi, N., and Baldi, M. M. (2012). The capacitated transshipment location problem with stochastic handling utilities at the facilities. *International Transactions in Operational Research*, 19(6):789–807.

Wallace, S. and Ziemba, W. (2005). *Applications of Stochastic Programming*. MPS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics.

Wang, C., Hu, Z., Xie, M., and Bian, Y. (2019). Sustainable facility location-allocation problem under uncertainty. *Concurrency and Computation: Practice and Experience*, 31(9):e4521.

Zhao, L., Zhao, Y., Hu, Q., Li, H., and Stoeter, J. (2018). Evaluation of consolidation center cargo capacity and loctions for China railway express. *Transportation Research Part E: Logistics and Transportation Review*, 117:58–81.