Multistage stochastic capacitated discrete lot-sizing with lead times: problem definition, complexity analysis and tighter formulations

Carlos E. TESTURI

Programa de Posgrado del Doctorado en Informática
Programa de Desarrollo de las Ciencias Básicas – PEDECIBA
Instituto de Computación, Facultad de Ingeniería
Universidad de la República

Montevideo – Uruguay
March 2020

DOCTORAL THESIS

Multistage stochastic capacitated discrete lot-sizing with lead times: problem definition, complexity analysis and tighter formulations

Author:
Carlos E. TESTURI

Thesis supervisors:
Dr. Víctor M. ALBORNOZ
Dr. Héctor CANCELA
Academic supervisor:
Dr. Héctor CANCELA

Thesis presented in compliance with the requirements to obtain the title of ***Doctor en Informática*** awarded by
Programa de Desarrollo de las Ciencias Básicas – PEDECIBA,
Instituto de Computación, Facultad de Ingeniería,
Universidad de la República

Montevideo – Uruguay
March 2020

Testuri, Carlos E.,

Multistage stochastic capacitated discrete lot-sizing with lead times: problem definition, complexity analysis and tighter formulations. Carlos E. Testuri. - Montevideo: Universidad de la República, Instituto de Computación, Facultad de Ingeniería, 2020.

XV, 45 p.; 29,7 cm

Directores de tesis:

Dr. Víctor M. ALBORNOZ

Dr. Héctor CANCELA

Director académico:

Dr. Héctor CANCELA

Tesis de Doctorado – Universidad de la República, Programa del Doctorado en Informática – PEDECIBA, 2020.

Referencias bibliográficas: p. 35 – 41.

1. Determinación de lotes estocástica, 2. Programación entera-mixta estocástica múltiple-etapa, 3. Inecuaciones válidas, 4. Tiempo de espera

Members of the thesis evaluation committee

Prof. Dr. Andrés L. Medaglia

Prof. Dr. Virgílio José Martins Ferreira Filho

Prof. Dr. Horacio H. Yanasse

Prof. Dr. Javier Baliosian

Prof. Dr. Marcos Viera

Dedicated to the memory of my parents
and all those that promote education

Acknowledgements

I would like to especially thank my advisors, Víctor Albornoz and Héctor Cancela, for their guidance, support and encouragement during the development of this thesis. Particularly for their time, patience and wisdom dedicated in many face-to-face and remote affable meetings, and insightful work reviews.

Additionally, I am grateful to the reviewers of this thesis, Andrés L. Medaglia and Virgílio José Martins Ferreira Filho for their comments and suggestions, and the remaining members of the thesis committee, Horacio H. Yanasse, Javier Baliosian and Marcos Viera, for their availability to evaluate this dissertation.

I am also grateful to my colleagues on the shared teaching assignment by covering my absences and sabbatical leave, especially Pedro Piñeyro, German Ferrari, Héctor Cancela, Antonio Mauttone, Luis Stábile and Fernando Islas, and my colleagues of the recent teams of *Programación 3* course.

Also, I am grateful to my colleagues in the *Departamento de Investigación Operativa*, and my teaching and staff colleagues in the *Instituto de Computación* for their encourage, service and operation of the department and the institute.

In addition, I am grateful to all those who carry out the graduate program and the institutions that supports this work, *Programa de Desarrollo de las Ciencias Básicas*, and *Comsión Sectorial de Investigación Científica* and *Facultad de Ingeniería, Universidad de la República*, respectivelly.

Finally, I would like to thank my family: Nicolás, Natalia and Sonia, for their loving support.

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Abstract

Facultad de Ingeniería
Instituto de Computación

Doctor en Informática

Multistage stochastic capacitated discrete lot-sizing with lead times: problem definition, complexity analysis and tighter formulations

by Carlos E. TESTURI

A stochastic capacitated discrete procurement problem with lead times, cancellation and postponement is addressed. The problem determines the expected cost minimization of satisfying the uncertain demand of a product during a discrete time planning horizon. The supply of the product is made through the purchase of optional distinguishable orders of fixed size with lead time. Due to the uncertainty of demand, corrective actions, such as order cancellation and postponement, may be taken with associated costs and time limits. The problem is modeled as an extension of a capacitated discrete lot-sizing problem with uncertain demand and lead times through a multistage stochastic mixed-integer programming approach. To improve the resolution of the model by tightening its formulation, valid inequalities are generated based on conventional inequalities. Subsets of approximately non-dominated valid inequalities are determined heuristically. A procedure to tighten an upgraded formulation based on a known scheme of pairing of inequalities is proposed. Computational experiments are performed for several instances with different uncertainty information structure. The experimental results allow to conclude that the inclusion of subsets of the generated valid inequalities enable a more efficient resolution of the model.

Keywords: stochastic lot-sizing, multistage stochastic mixed-integer programming, valid inequality, lead time

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

Resumen

Facultad de Ingeniería
Instituto de Computación

Doctor en Informática

Multistage stochastic capacitated discrete lot-sizing with lead times: problem definition, complexity analysis and tighter formulations

por Carlos E. TESTURI

Se aborda un problema de adquisición discreta capacitado estocástico con plazos de entrega, cancelación y postergación. El problema determina la minimización de costos esperados de satisfacer la demanda incierta de un producto durante un horizonte de planificación de tiempo discreto. El suministro del producto se realiza mediante la compra de pedidos distinguibles opcionales de tamaño fijo con tiempo de entrega. Debido a la incertidumbre de la demanda, se pueden tomar medidas correctivas, como la cancelación y la postergación de pedidos, con los costos y límites de tiempo asociados. El problema se modela como una extensión de un problema de tamaño de lote discreto capacitado con demanda incierta y plazos de entrega a través de un enfoque de programación entera-mixta estocástica con múltiples etapas. Para mejorar la resolución del modelo mediante el ajuste de su formulación, se generan desigualdades válidas basadas en desigualdades convencionales. Se determinan heurísticamente subconjuntos de desigualdades válidas aproximadamente no dominadas. Además, se propone un procedimiento para ajustar una formulación mejorada basada en un esquema conocido de emparejamiento de desigualdades. Para validar la metodología se realizan experimentos computacionales para varios casos con diferente estructura de información de la incertidumbre. Los resultados experimentales permiten concluir que la inclusión de subconjuntos de las desigualdades válidas generadas permite una resolución más eficiente del modelo.

Palabras claves: determinación de lotes estocástica, programación entera-mixta estocástica múltiple-etapa, inecuaciones válidas, tiempo de espera

Contents

List of Figures

List of Tables

List of Abbreviations

DCS	Deterministic problem formulation
EVPI	**E**xpected **v**alue (of) **p**erfect **i**nformation
GMCKP	**G**eneralized **m**ultiple-**c**hoice **k**napsack **p**roblem
MIP	**M**ixed **i**nteger **p**rogramming
MIR	**M**ixed **i**nteger **r**ounding
RP	**R**ecourse **p**roblem
SCS	Stochastic problem formulation
WS	**W**ait (and) **s**ee

Chapter 1

Introduction

Procurement problems are central to production planning or supply planning problems. In general, these problems require timely decisions on the acquisition of products with the objective to efficiently meet their forecast demand. Due to their properties, usually these problems do not have a direct analytical resolution. Instead, frequently they are algebraically modeled, and their models solved numerically by algorithms. For some problem cases, their complexity is such that no efficient algorithmic resolution is known. This situation leads to the search for appropriate resolution methodologies, which is the area where this thesis seeks to make a contribution.

The problem under study tries to satisfy the uncertain demand of a product at minimum expected cost during a time planning horizon. The product procurement is accomplished by the optional acquisition of distinguishable orders of fixed size with lead time. Due to orders lead time and the revealing of demand uncertainty, corrective actions may be taken on the acquisition decisions with associated costs and within time limits. One objective of this thesis is to algebraically formulate the problem by multistage stochastic mixed-integer programming and establish that it belongs to the $\mathcal{NP}$-hard temporal complexity class. A second objective is to tighten the formulation through the development of associated valid inequalities. Another aim is to approximately determine non-dominated sets of valid inequalities. A final objective is to validate the effectiveness of the entire development through computational experimentation.

This thesis is structured around a compendium of articles. It is composed of a first part structured in chapters where the context and subject of it, its contributions and a summary of computational experiments that validate them are presented. The second part is composed by an appendix that includes three articles supporting it.

In Chapter 2, an introductory background of the problem is given. Together with a description of the problem, a summary of the relevant literature associated with the analysis of the selected formulation and the methodology used for the problem is presented. In addition, a brief description of the contextual literature that highlights the milestones of the methodology is provided.

Chapter 3 describes an algebraic formulation of the problem based on the stochastic mixed-integer programming approach. At the beginning, an introductory formulation of the problem is developed assuming deterministic data. After defining the information structure of the uncertainty from a discrete stochastic process, the stochastic variant of the formulation is presented.

Chapter 4 studies the time complexity of the problem based on its deterministic formulation. It includes the derivation of valid inequalities to strengthen the stochastic formulation of the problem. It also includes, a heuristic to determine non-dominated valid inequalities. Finally, an orderly tightening scheme of pairing of valid inequalities is proposed.

Computational experiments of the developed methodology on several data instances are summarized in Chapter 5. The value of information and solution of the stochastic formulation is obtained. The performance of formulations derived from valid inequalities is determined. Experiments are performed to evaluate a stochastic formulation obtained by the heuristic determination of non-dominated valid inequalities.

The epilogue of this thesis is provided in Chapter 6 with inferred conclusions and suggestions for further research.

Finally, additional details of results of this thesis were included in the following articles published or submitted for consideration for publication. The article "Stochastic discrete lot-sizing with lead times for fuel supply optimization", published in the *Pesquisa Operacional* journal, describes the context, the problem and its formulation. The development of valid inequalities to strengthen the stochastic formulation of the problem is included in article "Valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement" that was submitted to a journal for consideration for publication and is currently under the review process. The article "Non-dominated valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement", published in *Proceedings of the 8th International Conference on Operations Research and Enterprise Systems*, depicts the heuristic to determine non-dominated valid inequalities. Appendix A contains a copy these articles.

Chapter 2

A Procurement Problem

This chapter presents a variant of the problem of satisfying the uncertain demand of a product at minimum expected cost during a time planning horizon. The supply of the product is made through the purchase of optional distinguishable orders of fixed size with lead time. Due to the uncertainty of demand, corrective actions, such as order cancellation and postponement, may be taken with associated costs and time limits. The background of the problem is outlined in Section 2.1. The description of the procurement problem in a context of demand uncertainty and delivery times is presented in Section 2.2. A summary of relevant and contextual literature associated with the problem and the used methodology is presented in Sections 2.3 and 2.4, respectively. The description of the problem presented here was introduced in the paper Testuri, Cancela, and Albornoz (2019a), which is a part of the thesis work.

2.1 Problem Background

Distribution and storage of primary products are downstream oil supply chain activities. These involve complex logistic planning under uncertainty of product features and resources. In most non-oil producing countries, or where the production is not sufficient to cover the internal demand, it is necessary to import either crude oil, or even refined products, to cover demand. One of the most important and cheapest transportation modes is by ship. The use of ship transportation has an important impact in the supply chain, as it is necessary to negotiate not only volume and price, but also how the delivery will be carried out; shipments are of fixed sizes, ship routes are complex and travel times are usually long. This means that supply contracts must be fixed much in advance of the actual times where the fuel will be needed. Additionally, demand may vary significantly from the best forecasts; this stochastic component introduces an additional complexity and is a source of costs.

The problem background of the present work is a real problem arising in a state-owned Uruguayan oil company that deals with fuel acquisition under contractual and logistic conditions for the energy sector. The demand that the company faces is uncertain, given that thermal electricity generation, as a complement in an electrical system, is highly dependent on renewable sources (Testuri, Zimberg, and Ferrari, 2012; Zimberg, Testuri, and Ferrari, 2019). While the particular situation of the company, as discussed in the mentioned papers, focuses on fuel procurement for thermal generation, the present thesis discusses the formulation and solution of a more general variant of the problem core, which can represent situations in which a product procurement is carried out by selecting distinguishable discrete supply options in the context of uncertain demand.

2.2 Problem Description

In the problem under study, the decision maker aim is to meet the uncertain demand of a product over a finite discrete time planning horizon while minimizing the expectation of the costs incurred. To meet the product demand, there are optional distinguishable purchase orders with an indivisible amount of the product. Each order can be purchased at most once within the planning horizon, with an associated cost and a delivery time. After satisfying the demand of the product in a given period, the remaining quantity is stored, up to a certain capacity, to flexibly satisfy future demand in subsequent periods. The orders have significant delivery times within the planning horizon. Therefore a considerable amount of time elapses between the purchase decision and the moment when the product is received. As the time passes, the uncertainty of the demand is revealed. Then it can happen that, at a given time, a purchase order which has not yet been received is no longer necessary. In this case, it could be decided to cancel its acquisition or postpone its delivery; decisions that in turn, have minimum execution times in relation to the time of delivery and associated costs.

2.3 Relevant Literature

The uncapacitated lot-sizing problem formulated by Wagner and Whitin (1958) can be used as a baseline to formulate the problem of this work. This problem determines a minimum cost scheme of decisions of set-up, production or replenishment, and storage for a product in order to satisfy its demand over a discrete time planning horizon. In relation to this work we are interested in classifying the problem in variants according to i) the certainty of the data (determininistic or stochastic), ii) if production capacities are established as constant or variable, and iii) the sizing of the lot (continuous or discrete).

For the deterministic uncapacitated and continuous lot-sizing case, Wagner and Whitin (1958) and Wagelmans, Hoesel, and Kolen (1992) showed that the problem has efficient resolution through dynamic programming applied to the original mixed-integer formulation. In addition, there are known tighter formulations which determine the convex hull of the decisions' feasible region: the extended facility location formulation of Krarup and Bilde (1977) and the valid inequalities formulation of Barany, Van Roy, and Wolsey (1984). In the extended facility location formulation the quantity produced in a period is distinguished according to the period of the demand it serves. Barany, Van Roy, and Wolsey (1984) derived valid inequalities, denoted $(\ell, \mathcal{S})$, of the original formulation that allow to obtain its convex hull.

For the problem variant with constant production capacities, Pochet and Wolsey (1993) derived valid inequalities that generate all facets of the convex hull of feasible solutions. This authors also showed that a subset of inequalities suffices to solve the problem when the Wagner-Whithin assumption is satisfied (Pochet and Wolsey, 1994).

A variant of the lot-sizing problem with capacity production was established by Florian and Klein (1971). Florian, Lenstra, and Rinnooy Kan (1980) showed that this variant belongs to the $\mathcal{NP}$-hard time complexity class for some combinations of convexity on objective functions and capacity limits. Bitran and Yanasse (1982) established that this variant is a generalization of the binary knapsack problem, and that it belongs to the $\mathcal{NP}$-hard class for a concave objective function with linear production and storage costs under several cost properties.

Fleischmann (1990) established a variant of the lot-sizing problem with discrete production; where a branch and bound procedure with Lagrangian relaxation solved by dynamic programming is used. Van Hoesel et al. (1994) reformulated the single-item discrete lot-sizing problem as a linear assignment problem under specific conditions that allow resolution by linear and dynamic programming. Van Eijl and Van Hoesel (1997) presented a partial linear description of the convex hull of the single-item lot-sizing problem with Wagner-Whitin costs. Tight mixed-integer programming formulations, with mixed-integer sets, of variants of the discrete lot-sizing problems are proposed by Miller and Wolsey (2003).

In the case that problem data is uncertain (stochastic variant), it can be formulated by means of stochastic programming (Birge and Louveaux, 2011; Kall and Wallace, 1994). Haugen, Løkketangen, and Woodruff (2001) proposed a progressive hedging meta-heuristic solution for the continuous problem. Ahmed, King, and Parija (2003) established a tightened extended formulation of the stochastic continuous uncapacitated problem and showed that the Wagner-Whitin conditions are not satisfied by the formulation. Guan et al. (2006) showed that the valid inequalities proposed by Barany, Van Roy, and Wolsey (1984) are also valid for the stochastic continuous variant, and they extend the inequalities to a general class that allow to define facets of the feasible set.

Other variants of the deterministic continuous uncapacitated lot-sizing problem model delivery time of the lots (e.g. due to production time). Lee, Çetinkaya, and Wagelmans (2001) presented a variant in which demands have a compliance interval, and they presented and efficient resolution by dynamic programming. Brahimi, Dauzère-Pérès, and Najid (2006) presented two variants according to whether the lots are or are not distinguishable with respect to delivery times. These authors proposed efficient algorithms based on dynamic programming for the distinguishable case and for the undistinguishable case when the order-delivery windows are not inclusive. For these variants, Wolsey (2006) set tight extended formulations. For the stochastic case, Huang and Küçükyavuz (2008) established that the problem with random lead times can be efficiently solved when delivery windows do not intersect in time. Jiang and Guan (2011) established a quadratic polinomial time algorithm. Liu and Küçükyavuz (2018) proposed valid inequalities for the static probabilistic lot-sizing problem. Other approaches to stochastic lot-sizing problems deals with scheduling of multiple products (Sox et al., 1999; Beraldi et al., 2006; Vargas and Metters, 2011; Cunha Neto, Ferreira Filho, and Arruda, 2015; Hu and Hu, 2018).

2.4 Contextual Literature

The methodology used to model the problem is stochastic mixed-integer (linear) programming. A mixed-integer programming problem is about the search of a solution over a set of combined continuous and discrete domain, frequently described as linear constraints, that optimizes a linear objective function (Schrijver, 1986; Nemhauser and Wolsey, 1988; Bertsimas and Weismantel, 2005). The leading attempt of Dantzig, Fulkerson, and Johnson (1959) (belatedly published) to solve the traveling salesman problem by describing the convex hull of the tours ushered the cutting plane method independently provided later by Gomory (1958). Land and Doig (1960) presented the successful branch and bound method to solve integer programming problems; which was later integrated with the cutting plane method. Edmonds (1968) showed for several cases of integer programming problems that their convex hull may be obtained by a combinatorial set of cutting planes. Following the characterization of decision problems on the $\mathcal{NP}$-complete time complexity class by Cook (1971), Karp (1972) proved that several integer programming problems belongs to the optimization related $\mathcal{NP}$-hard class. Balas (1979) showed that integer programming problems could be extended by considering disjunctions of polyhedra, and that valid inequalities for the disjunction of polyhedra can be deduced from inequalities of the original polyhedra. Lenstra, Lenstra, and Lovász (1982) began a primary approach to solve the problem with Lovász's basis reduction algorithm (Aardal, Weismantel, and Wolsey, 2002). Nemhauser and Wolsey (1990) developed a general recursive procedure to obtain disjunctive and mixed integer rounding (MIR) inequalities. Günlük and Pochet (2001) showed how known MIR inequalities can be combined to generate new strong valid inequalities. Several valid inequalities are derived from the disjunctive and MIR approaches (Cornuéjols, 2008).

Stochastic programming, a variant of the mathematical programming approach, seeks to determine the solution of optimization problems that involves uncertain data. A relevant feature of this framework is that the decision process modeled must be consistent with the temporal structure of the information. In this scheme, decisions and random events are interspersed in two or more time stages, where decisions depend on the probabilistic distribution of future events, but cannot anticipate them with certainty. Dantzig (1955) and Beale (1955) independently presented the earlier development of two-stage stochastic programming. Wets (1966), Walkup and Wets (1967) and Kall (1976) developed the fundamentals of two-stage stochastic programming. Due to the uncertainty modeling, problem instances become large. Therefore, decomposition mechanisms were developed to treat this effect

(Dantzig and Medansky, 1961; Benders, 1962). Based on Benders decomposition, Van Slyke and Wets (1969) developed the L-shape resolution method. Birge and Louveaux (1988) developed a multi-cut variant of the method. The assessing information metrics "expected value of perfect information" and "value of the stochastic solution" were developed based on decision theory. Madansky (1960), Dempster (1981), and Birge (1982) participated in their development. Several authors participated in the extension of the scheme from two to multiple stages (Beale, Forrest, and Taylor, 1980; Louveaux, 1980; Birge, 1985; Noël and Smeers, 1986). The integer L-shape method was proposed by Laporte and Louveaux (1993). Carøe and Tind (1998) proposed a decomposition of the method based on duality theory. Norkin, Ermoliev, and Ruszczyński (1998) developed a variant of the branch and cut method based on a statistical estimation of the recourse function. Sen and Higle (2005) developed a decomposition-based algorithm for the two-stage stochastic integer programming problem. Ahmed (2013) proposed a scenario decomposition algorithm for stochastic binary programming.

Chapter 3

Algebraic Modeling

This chapter is devoted to an algebraic formulation of the problem introduced in Chapter 2. The formulation is developed using an integer programming approach. The modelled problem deals with the minimization of the cost expectation incurred in decisions made to meet the uncertain demand of a product over a finite discrete time planning horizon. To satisfy the demand, there are optional distinguishable shipments, denoted as orders, with an indivisible quantity of the acquired product. The orders have relevant delivery times in the planning horizon; so that a significant amount of time elapses between the purchase decision and when the order is received. Due to the passage of time and after demand uncertainty has been revealed, it could happen that at any given time an order that was previously acquired and not yet received is no longer necessary. In this case it could be decided to cancel its acquisition or postpone its delivery; decisions, which, in turn, have minimum execution times in relation to the time of delivery and associated costs. After attending the demand in a given period, the amount of remaining product is stored up to a certain capacity, to satisfy future demands in a flexible manner.

The algebraic formulation of the problem is presented below. First, a deterministic model of the problem is developed, where definitions and general structure are established. After defining the information structure of the uncertainty from a discrete stochastic process, the stochastic variant of the model is presented. The models presented in this chapter are part of the publication Testuri, Cancela, and Albornoz (2019a), part of this thesis, and included in Appendix A.

3.1 Deterministic model

The following is a formulation of the deterministic problem, where the demand is assumed to be known with certainty. A discrete time sequential decision process is considered, in which decisions are made to acquire possible orders, or to cancel and postpone orders already acquired. Main entities of the problem are described as index sets in Table 3.1. The planning time is represented by the set T of discrete time periods, from initial period 1 up to horizon period $\hbar$. The set O of orders is partitioned in two subsets: the set A of already acquired orders –orders established in previous time execution of the model– that are pending reception, and the set F of possible (future) orders to be acquired from now on. Acquisition decisions are made on orders in set F, and cancellation or postponement decisions are taken on orders in set A.

TABLE 3.1: Index sets

T	periods, $\{1, ..., \hbar\}$ (ordered set)
A	already acquired orders
F	possible (future) orders to be acquired
O	orders, $A \cup F$

Due to the nature of the problem, these decisions have relevant compliance times in the planning horizon. Each order has a minimum delivery time, between the time the purchase decision is made and the order is received. Decisions to cancel and postpone an already purchased order must be made before a certain minimum time, prior to the receipt of the order. In addition, when deciding to postpone a previously purchased order, the time elapsed between the original reception period and the postponement period cannot be less than a given minimum postponement time. All these constraints cause some latency in the decision making process.

Parameters

Parameters are described in Table 3.2. The amount demanded d_t for the product in each period t is known. Due to storage constraints, the inventory of the product at the end of each period is restricted between a minimum amount, $\underline{s}$, and a maximum amount, $\overline{s}$, and there is an initial storage amount, s_0, at the beginning of the planning horizon.

The period τ^i in which an already acquired order i is received is fixed, and it is decided in previous acquisitions (i.e. previous model resolutions). Each order i has a given amount of product, q^i. Decisions on each order have latency times measured in periods. The delivery time of order i, γ^i, establishes the length of the wait time (measured in periods) between the acquisition decision and the actual arrival of the order. The minimum time for cancellation of order i, δ^i, establishes the minimum number of periods prior to the delivery period at which the order may be cancelled. The minimum postponement time of order i, ϵ^i, establishes the minimum number of periods after the delivery period in which the postponed order can be received. The achievement period of decisions on acquisition, cancellation and postponement must take place within the planning horizon.

For each order i there are unitary costs associated with the decisions to acquire, ca^i, cancel, cc^i, and postpone, cp^i, it. In addition, there is a unitary storage cost h_t at each period t.

The already acquired amount that is scheduled to be received in each period $t \in T$ is determined by the sum of the amount of the orders that are received in that period,

$$a_t := \sum_{\{i \in A | \tau^i = t\}} q^i; \tag{3.1}$$

this is an auxiliary aggregate parameter.

TABLE 3.2: Parameters

d_t	demand amount in period $t \in T$
s_0	initial inventory amount
$\underline{s}, \overline{s}$	minimum and maximum storage capacities by period
τ^i	period in which already acquired order $i \in A$ is received
q^i	amount of order $i \in O$
γ^i	delivery time of order $i \in F$, such that $0 \leq \gamma^i \leq \hbar - 1$
δ^i	cancellation minimum time of already acquired order $i \in A$, such that $0 \leq \delta^i \leq \tau^i - 1$
ϵ^i	postponement minimum time of already acquired order $i \in A$, such that $0 \leq \epsilon^i \leq \hbar - \tau^i$
ca^i	acquisition unit cost of order $i \in O$
cc^i	cancellation unit cost of order $i \in O$
cp^i	postponement unit cost of order $i \in O$
h_t	storage unit cost in period $t \in T$
a_t	already acquired amount that is received in period $t \in T$

Variables

Variables of the deterministic model are summarized in Table 3.3. The continuous variable s_t represents the storage amount of the product at the end of each period t. The acquisition decision of each order i to be taken in period t, subject to its delivery period γ^i, is modeled by the binary variable v_t^i. Decisions to cancel each order i in period t, prior to their minimum cancellation time, are established using binary variables x_t^i. When postponing an order reception, decisions must be made about when and until when it is done. Since both decisions are independent, the decision to postpone an order i is modeled with a prior cancellation decision and a decision whether to delay its receipt to another period t after its minimum delay time, represented by binary variables z_t^i. In addition, there are continuous aggregate variables that consolidate, per period t, the acquired amount incoming at period t, u_t, the cancelled amount outgoing from period t, w_t, and the postponed amount incoming at period t, y_t. These variables facilitate the legibility of inventory balance constraints.

Objective function

The aim is to determine a minimum cost scheme of inventory, acquisition, cancellation and postponement of orders that satisfy demand during the planning horizon. The costs of acquisitions, cancellations and postponements accrue at the time of decision making. The objective function includes budget costs of acquisition, cancellation and postponement, and inventory

TABLE 3.3: Variables of the deterministic model

s_t	inventory amount at the end of period $t \in T$
u_t	acquired amount at period $t \in T$
v_t^i	if order $i \in F$ is acquired in period $t \in \{1, ..., \hbar - \gamma^i\}$ (binary)
w_t	cancelled amount outgoing from period $t \in T$
x_t^i	if already acquired order $i \in A$ is cancelled in period $t \in \{1, ..., \tau^i - \delta^i\}$ (binary)
y_t	postponed amount incoming in period $t \in T$
z_t^i	if already acquired order $i \in A$ is postponed to period $t \in \{\tau^i + \epsilon^i, ..., \hbar\}$ (binary)

costs,

$$\min \sum_{t \in T} \Big[\sum_{\{i \in F | t \leq \hbar - \gamma^i\}} ca^i q^i v_t^i \tag{3.2}$$

$$+ \sum_{\{i \in A | t \leq \tau^i - \delta^i\}} (cc^i - ca^i) q^i x_t^i \tag{3.3}$$

$$+ \sum_{\{i \in A | \tau^i + \epsilon^i \leq t\}} (cp^i + ca^i - cc^i) q^i z_t^i \tag{3.4}$$

$$+ h_t s_t \Big]. \tag{3.5}$$

The expression (3.2) represents the costs of acquiring possible orders, the expression (3.3) represents the costs of cancellation minus the budgeted acquisition costs of already acquired orders that are cancelled, the expression (3.4) represents the acquisition costs minus the costs of cancellation plus the costs of postponement of the already acquired orders that are postponed, and the expression (3.5) represents inventory costs. The postponement cost includes the subtraction of cancellation costs, since a postponement is represented by a prior cancellation, but it does not incur cancellation costs.

Constraints

The main requirement is to satisfy demand while maintaining the inventory balance with contributions of the product previously stored, what was already acquired, what is acquired, and what is cancelled, postponed and remains available in storage. This is determined, for all $t \in T$, by the equation

$$s_{t-1} + a_t + u_t + y_t = d_t + w_t + s_t, \tag{3.6}$$

where s_0 is the given initial inventory.

The amount stored in each period $t \in T$ is constrained between lower and upper bounds by

$$\underline{s} \leq s_t \leq \bar{s}. \tag{3.7}$$

The amount of acquired product that is received in each period is determined by the sum of orders acquired in the range of the delivery periods of the orders. This is accomplished for each period $t \in T$ by

$$u_t = \sum_{\{i \in F | \gamma^i + 1 \leq t\}} q^i v^i_{t-\gamma^i}. \tag{3.8}$$

If an order is acquired, the acquisition is decided in a single period before or equal to its possible receiving period less its delivery time. This is established for each order $i \in F$ by

$$\sum_{t=1}^{\hbar-\gamma^i} v^i_t \leq 1. \tag{3.9}$$

The already acquired amount that is cancelled in each period is determined by the cancellations of the orders in the possible range of the corresponding cancellation periods. This is ensured for each period $t \in T$ by

$$w_t = \sum_{\{i \in A | \tau^i = t\}} \left(q^i \sum_{t'=1}^{\tau^i - \delta^i} x^i_{t'} \right). \tag{3.10}$$

If an order $i \in A$ is cancelled, the cancellation is decided in a single period before or equal to its receiving period τ^i minus its cancellation time δ^i, as established by

$$\sum_{t=1}^{\tau^i-\delta^i} x^i_t \leq 1. \tag{3.11}$$

The postponement of an order is modeled by the use of cancellation, that is to say, it is only possible to postpone orders that are cancelled. The already acquired amount that is postponed to a certain period is determined by the postponements of the orders in the possible range of the corresponding postponement periods. For a period $t \in T$ this is accomplished by

$$y_t = \sum_{\{i \in A | \tau^i + \epsilon^i \leq t\}} q^i z^i_t. \tag{3.12}$$

If an order $i \in A$ is postponed, it is to be received in a single period subsequent to or equal to its original receiving period τ^i plus its delay time ϵ^i, as established by

$$\sum_{t=\tau^i+\epsilon^i}^{\hbar} z^i_t \leq 1. \tag{3.13}$$

An orden can be postponed, if its original arrival decision has been cancelled. This is accomplished for each order $i \in A$ by

$$\sum_{t=\tau^i+\epsilon^i}^{\hbar} z^i_t \leq \sum_{t=1}^{\tau^i-\delta^i} x^i_t. \tag{3.14}$$

Finally, there are domain constraints of the variables as stated by

$$\begin{aligned} & s_t, u_t, w_t, y_t \geq 0, \quad \text{for all } t \in T, \\ & v_t^i \in \{0,1\}, \quad \text{for all } i \in F \text{ and } t \in \{1, ..., \hbar - \gamma^i\}, \\ & x_t^i \in \{0,1\}, \quad \text{for all } i \in A \text{ and } t \in \{1, ..., \tau^i - \delta^i\}, \\ & z_t^i \in \{0,1\}, \quad \text{for all } i \in A \text{ and } t \in \{\tau^i + \epsilon^i, ..., \hbar\}. \end{aligned} \tag{3.15}$$

The deterministic problem formulation, (3.2)-(3.15), is denoted as (DCS)[1]. The (DCS) formulation is a generalization of the discrete lot-sizing problem, and it belongs to the $\mathcal{NP}$-hard time complexity class, since the discrete lot-sizing problem belongs to this class (Bitran and Yanasse, 1982; Wolsey, 2002).

The feasibility of this formulation is conditioned to the timely availability of the product acquisition to satisfy its demand, as stated for each period $t \in T$ by the condition

$$s_0 + \sum_{\{i \in A | \tau^i \leq t\}} q^i + \sum_{\{i \in F | \gamma^i + 1 \leq t\}} q^i \geq \sum_{t'=1}^{t} d_{t'}. \tag{3.16}$$

3.2 Stochastic model with uncertain demand

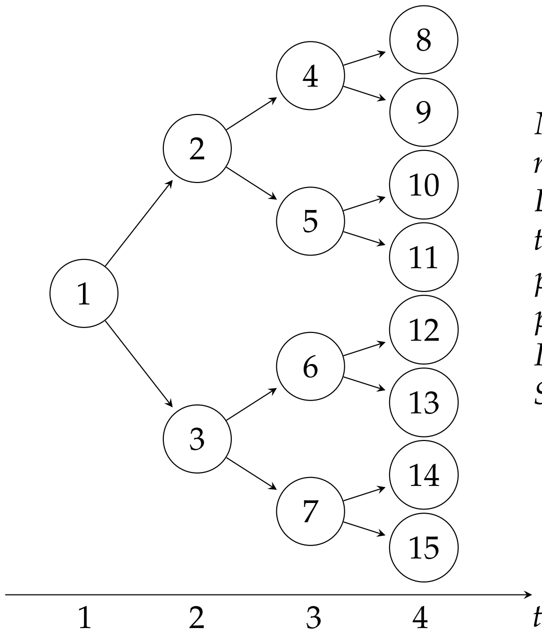
The following model is a stochastic extension of the deterministic model in which demand is a random parameter. Previously, the uncertain information structure was established through stochastic optimization (Birge and Louveaux, 2011). Subsequently, the entities and formulation of the stochastic model are reported.

Uncertain information structure

The uncertain demand is represented by a discrete-time stochastic process indexed in the planning periods; in such a way that each stage of the stochastic process is associated to a period. The process is defined in a finite probability space. It is assumed that the demand of the first period is deterministic, and that the demands of the remaining periods are random with known distribution function. The decisions of a period only depend on the event outcomes of the random parameters of previous periods. This process is non-anticipatory of the future decisions or the realizations of the random events. This information structure can be represented by a arborescence of events with $\hbar$ levels or stages called *tree of scenarios* (Römisch and Schultz, 2001). The arborescence is a perfect directed tree, with the root node event in period $t = 1$ and with leaf node events in period $t = \hbar$. Each path of events from the root node to a leaf node is denoted as a *scenario*.

Each node of the scenario tree describes the state of the process and is identified by a period and a scenario. An alternative abbreviated notation is to identify the nodes by a single index n in an numerable set of nodes, N. For the first period, $t = 1$, there is a unique node, called r, that represents the root

[1]The same denomination used in the articles is maintained for compatibility reasons.



(A) Indexed scenario tree

$N = \{1, 2, ..., 15\}$
$r = 1$
$L = \{8, 9, ..., 15\}$
$t(4) = 3$
$p(4) = 2$
$p(4, 2) = 1$
$P(4) = (1, 2, 4)$
$S(2) = \{4, 5, 8, 9, 10, 11\}$

(B) Set-value functions on nodes

FIGURE 3.1: Example of a scenario tree with four periods and set-value functions on nodes

of the tree. Each node $n \in N$ has an immediate time predecessor $p(n)$ node; the auxiliary node 0 is defined as the predecessor of the root node, $0 := p(r)$, such that $0 \notin N$. The period corresponding to each node n is defined as $t(n)$. The probability of the state of each node n is defined as π_n, such that $\pi_n \geq 0$ and $\sum_{n \in N | t(n)=t} \pi_n = 1$, for all $t = 1, ..., \hbar$. The k-th time predecessor of node n is defined as $p(n, k) := p(p(n, k-1))$ for $k = 2, ..., t(n) - 1$, such that $p(n, 1) := p(n)$. The ordered set of nodes on the path from the root node to n is defined as sequence $\big(r \equiv p(n, t(n) - 1), p(n, t(n) - 2), ..., p(n, 1), n\big)$. The set of nodes successors of node $n \in N$ is defined as $S(n) := \{n' \in N, k = 1, ..., \hbar - t(n) | n = p(n', k)\}$. The set of leaf nodes is defined as $L := \{n \in N | t(n) = \hbar\}$. An example of a perfect binary scenario tree with four periods in conjunction with set-value functions is depicted in Figure 3.1.

The stochastic parameters and set-value functions on the nodes of the stochastic model are summarized in Table 3.4. The stochastic demand parameter, d_n, which in the deterministic model depends on the periods, in the stochastic model depends on the nodes of the tree.

Derived subsets of the sets of nodes and periods that are indexed in the parameters are established in Table 3.5 in order to facilitate the formulation. There are subsets to abbreviate the denomination of nodes where it is possible to acquire each order $i \in F$, N^i_γ, and where it is possible to cancel and postpone each order $i \in A$, N^i_δ. In addition, subsets of periods to where it is possible to postpone each order $i \in A$ are established, T^i_ϵ. The subscripts of these subsets are part of their denomination.

In the stochastic model, decisions depend on the nodes of the tree instead of the periods as in the deterministic formulation. The deterministic formulation variables are redefined according to Table 3.6. In contrast to the

TABLE 3.4: Parameters and set-value functions on nodes

d_n	demand at node $n \in N$
$t(n)$	period of node $n \in N$
$p(n)$	immediate time predecessor node of node $n \in N$ in the tree
$p(n,k)$	k-th time predecesor node of node $n \in N$ in the tree
π_n	probability of node $n \in N$
$P(n)$	set of ordered nodes in the path from root node to node $n \in N$
$S(n)$	set of successor nodes of node $n \in N$ in the tree
L	set of leaf nodes of the tree

TABLE 3.5: Derived index sets

N^i_γ	nodes where it is possible to acquire order $i \in F$, $\{n \in N \| t(n) \leq \hbar - \gamma^i\}$
N^i_δ	nodes where it is possible to cancel and postpone order $i \in A$, $\{n \in N \| t(n) \leq \tau^i - \delta^i\}$
T^i_ϵ	periods to where it is possible to postpone order $i \in A$, $\{t \in T \| t \geq \tau^i + \epsilon^i\}$

deterministic model, the decision to postpone a order, into a given period, could be taken in different nodes; this is modeled by including a node index into variable z.

TABLE 3.6: Variables of the stochastic model

s_n	inventory amount at the end of period in node $n \in N$
u_n	acquired amount incoming at node $n \in N$
v^i_n	if order $i \in F$ is acquired in node $n \in N^i_\gamma$ (binary)
w_n	cancelled amount outgoing of node $n \in N$
x^i_n	if an already acquired order $i \in A$ is cancelled in node $n \in N^i_\delta$ (binary)
y_n	postponed amount incoming at node $n \in N$
z^i_{nt}	if already acquired order $i \in A$ is postponed in node $n \in N^i_\delta$ to period $t \in T^i_\epsilon$ (binary)

The indexes of periods in expressions of deterministic parameters or variables may refer to the temporal realization of a node n by $t(n)$. This is the case for parameters corresponding to the already acquired amount and storage unit cost, which could be indexed as $a_{t(n)}$ and $h_{t(n)}$, respectively.

Based on the previous definitions of index sets, parameters and variables a multistage stochastic mixed-integer programming formulation of the stochastic capacitated sizing problem is

$$(\text{SCS}): \min \sum_{n \in N} \pi_n \Big[\sum_{\{i \in F | n \in N^i_\gamma\}} ca^i q^i v^i_n \tag{3.17}$$

$$+ \sum_{\{i \in A | n \in N^i_\delta\}} (cc^i - ca^i) q^i x^i_n \tag{3.18}$$

$$+ \sum_{\{i \in A, t \in T^i_\epsilon | n \in N^i_\delta\}} (cp^i + ca^i - cc^i) q^i z^i_{nt} \tag{3.19}$$

$$+ h_{t(n)} s_n \Big], \tag{3.20}$$

s.t.

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n, \quad n \in N, \tag{3.21}$$

$$\underline{s} \leq s_n \leq \bar{s}, \quad n \in N, \tag{3.22}$$

$$u_n = \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} q^i v^i_{p(n,\gamma^i)}, \quad n \in N, \tag{3.23}$$

$$\sum_{n' \in P(n)} v^i_{n'} \leq 1, \quad i \in F, n \in N, t(n) = \hbar - \gamma^i, \tag{3.24}$$

$$w_n = \sum_{\{i \in A | t(n) = \tau^i\}} \left(q^i \sum_{\{n' \in P(n) | t(n') \leq \tau^i - \delta^i\}} x^i_{n'} \right), \quad n \in N, \tag{3.25}$$

$$\sum_{n' \in P(n)} x^i_{n'} \leq 1, \quad i \in A, n \in N, t(n) = \tau^i - \delta^i, \tag{3.26}$$

$$x^i_n \geq z^i_{nt}, \quad i \in A, n \in N^i_\delta, t \in T^i_\epsilon, \tag{3.27}$$

$$y_n = \sum_{\{i \in A | t(n) \geq \tau^i + \epsilon^i\}} \left(q^i \sum_{\{n' \in P(n) \cap N^i_\delta\}} z^i_{n',t(n)} \right), \quad n \in N, \tag{3.28}$$

$$\sum_{\{n' \in P(n), t \in T^i_\epsilon\}} z^i_{n't} \leq 1, \quad i \in A, n \in N^i_\delta, \tag{3.29}$$

$$s_n, u_n, w_n, y_n \geq 0, \quad n \in N, \tag{3.30}$$

$$v^i_n \in \{0, 1\}, \quad i \in F, n \in N^i_\gamma, \tag{3.31}$$

$$x^i_n, z^i_{nt} \in \{0, 1\}, \quad i \in A, n \in N^i_\delta, t \in T^i_\epsilon. \tag{3.32}$$

This formulation takes into account the information structure of the scenario tree. The objective function minimizes expected costs. The expected costs can be factored by stage, where for each stage there is a distribution of costs with associated probability. The objective include expectation costs of acquisition (3.17), cancellation less acquisition in case of cancellation (3.18), postponement plus acquisition minus cancellation (3.19) –a postponement is modeled in conjunction with a cancellation–, and storage costs (3.20).

Constraints (3.21) set the product material flow conservation over time for each node, where the left and right expressions represent the incoming and outgoing flow, respectively. The lower and upper storage bounds at each node are determined by constraints (3.22). The amount of acquired product that is received at each node is determined by acquisitions of orders in the possible range of the corresponding acquisition periods according to (3.23). Constraints (3.24) state that each order is acquired at a single node at most in each path from the root node to a node whose period coincides with the receiving period minus the delivery time of the order. The product previously acquired that is cancelled at each node is determined by the cancellations of the nodes in the path from the root node to the node, whose cancellation periods are less than the delivery period less the cancellation time, according to (3.25). Constraints (3.26) state that each order to be cancelled is at a single node in each path from the root node to a node whose period coincides with the receiving period minus the cancellation time of the order. The postponement of the orders is modeled in conjunction with the cancellation, i.e. only cancelled orders can be postponed, (3.27). The already acquired amount that is postponed in a node is determined by the postponements of the orders in the nodes in the path from the root to the node for all periods superior to the period of reception plus the delay time of the node, according to (3.28). Constraints (3.29) state that each order to be postponed is at a single node in each path from the root node to a node in some period greater than the receiving period plus the time of postponement of the node. Constraints (3.30)–(3.32) state the domain of the variables. The set of feasible solutions of (SCS) is denoted by X_{SCS}.

This formulation is a generalization of the deterministic one. The deterministic formulation is a special case of the stochastic for the case of a unique scenario in which the tree of scenarios is reduced to a path. Therefore, the feasibility of the stochastic formulation is conditioned to the timely availability of the product acquisition to satisfy its demand along the path of each node of the tree, as stated for each $n \in N$ by the condition

$$s_0 + \sum_{\{i \in A | \tau^i \leq t(n)\}} q^i + \sum_{\{i \in F | \gamma^i + 1 \leq t(n)\}} q^i \geq \sum_{n' \in P(n)} d_{n'}.$$

Furthermore, the formulation does not guarantee that all solutions that satisfy constraints at initial stages are also feasible in the remaining stages; that is, it does not have “complete recourse” (Walkup and Wets, 1967).

Chapter 4

Valid Inequalities

In this chapter, a study of the time complexity of the problem based on its deterministic formulation is presented. The problem is shown to belong to the time complexity class $\mathcal{NP}$-hard. Therefore, there is no known polyhedral description of the convex hull of X_{SCS}. It is nevertheless interesting to derive valid inequalities which can be used to strengthen the formulation. In some cases adding these inequalities can directly improve the capability of solvers to find solutions for larger instances in shorter times. Even when this is not the case, they may be used within a more sophisticated solving strategy, such as branch and cut methods relying on constraint separation. It includes a development of valid inequalities to strengthen the stochastic formulation of the problem. It also includes a heuristic to determine non-dominated valid inequalities. The chapter concludes with a proposal of an orderly tightening scheme of a known pairing of valid inequalities.

4.1 Problem Complexity

Without loss of generality with respect to the stochastic formulation, the complexity of the problem is analyzed from the (DCS) deterministic formulation. (DCS) is a generalization of the discrete lot-sizing problem described by Wolsey (2002). The discrete lot-sizing problem is a special case of (DCS) in which the acquisition variables are selected among a special ordered set of type one (v. constraint (3.9)).

Proposition 1. (DCS) belongs to the time complexity class $\mathcal{NP}$-hard

Proof. The proof shows that (DCS) belongs to $\mathcal{NP}$ and that (DCS) is at least as hard as the generalized multiple-choice knapsack problem (GMCKP) (Pisinger, 2001). Since (GMCKP) belongs to $\mathcal{NP}$-hard, then (DCS) also belongs to this class.

Assuming that the feasibility condition (3.16) is met, there is an optimal solution on (v, x, z, s) in which s is the solution of a network flow problem in which in turn (v, x, z) are assumed independent. This polinomial size solution can be used to obtain a bounded cost objective (Yes answer) by satisfying the constraints in polinomial time.

It is proved that (GMCKP) is polynomially reducible to (DCS). As a previous step, it is necessary to establish a reduced equivalent instance of (DCS) that still encodes (GMCKP). It is considered an instance of (DCS) in which the delivery time of every order to be acquired is zero, $\gamma^i = 0$ for all $i \in F$, there are no already acquired orders on which to decide cancellations or postponements, $A = \emptyset$, there are no inventory bounds, $\underline{s}_t = 0$ and $\bar{s}_t = \infty$ for all $t \in T$, the initial inventory is zero, $s_0 = 0$, and the inventory costs are zero, $h_t = 0$ for all $t \in T$. Let this instance be

$$
\begin{aligned}
\text{(DCSI)}: \min \sum_{t \in T} \sum_{i \in F} & ca^i q^i v_t^i \\
\text{s.t.} & \\
& s_{t-1} + u_t = d_t + s_t, \quad t \in T, && (4.1)\\
& u_t = \sum_{i \in F} q^i v_t^i, \quad t \in T, && (4.2)\\
& \sum_{t \in T} v_t^i \leq 1, \quad i \in F, \\
& s_0 = 0, s_t, u_t \geq 0, \quad t \in T, \\
& v_t^i \in \{0,1\}, \quad i \in F, t \in T.
\end{aligned}
$$

(DCSI) can be equivalently formulated in the space of the v variables. Since $s_0 = 0$, equality constraints (4.1) may be substituted by inequality constraints of the form

$$
\sum_{t' \in \{1,\dots,t\}} u_{t'} \geq \sum_{t' \in \{1,\dots,t\}} d_{t'}, \quad t \in T, \tag{4.3}
$$

and then substituting u variables in constraints (4.3) by equations (4.2). This transformation leads to the formulation

$$
\begin{aligned}
\text{(DCSIR)}: \min \sum_{t \in T} \sum_{i \in F} & ca^i q^i v_t^i \\
\text{s.t.} & \\
& \sum_{t' \in \{1,\dots,t\}} \sum_{i \in F} q^i v_{t'}^i \geq \sum_{t' \in \{1,\dots,t\}} d_{t'}, \quad t \in T, && (4.4)\\
& \sum_{t \in T} v_t^i \leq 1, \quad i \in F, \\
& v_t^i \in \{0,1\}, \quad i \in F, t \in T.
\end{aligned}
$$

Furthermore, (DCSIR) is still constrained by establishing $d_t = 0$ for $t = 1, \dots, \hbar - 1$. Then constraints (4.4) are separated according to their bound, whether it is zero or not. Constraints with zero bound are dropped since they are redundant. This reduction leads to the formulation

$$\begin{aligned}
\text{(DCSIR0)} : \min & \sum_{t \in T} \sum_{i \in F} ca^i q^i v_t^i \\
\text{s.t.} & \sum_{t \in T} \sum_{i \in F} q^i v_t^i \geq \sum_{t \in T} d_t, \\
& \sum_{t \in T} v_t^i \leq 1, \quad i \in F, \\
& v_t^i \in \{0, 1\}, \quad i \in F, t \in T.
\end{aligned}$$

Given an instance of the (complementary) generalized multiple-choice knapsack problem for a set $i = 1, ..., m$ of items and a set $j = 1, ..., n$ of variants of the items. Let C be the capacity, and p_{ij}, ω_{ij} and χ_{ij} be the price, weight and binary variable, respectively, for each $i = 1, ..., m$ and $j = 1, ..., n$. Then the problem formulation is

$$\begin{aligned}
\text{(GMCKP)} : \min & \sum_{i=1}^{m} \sum_{j=1}^{n} p_{ij} \chi_{ij} \\
\text{s.t.} & \sum_{i=1}^{m} \sum_{j=1}^{n} \omega_{ij} \chi_{ij} \geq C, \\
& \sum_{i=1}^{m} \chi_{ij} \leq 1, \quad j = 1, ..., n, \\
& \chi_{ij} \in \{0, 1\}, \quad i = 1, ..., m, j = 1, ..., n.
\end{aligned}$$

Given a (DCSIR0) instance with $|T| = m$, $|F| = n$, $\sum_{t \in T} d_t = C$, and $p_{ij} = ca^j q^j$ and $\omega_{ij} = q^j$ for $i = 1, ..., m$ and $j = 1, ..., n$, then for any feasible solution of the (DCSIR0) instance there is a corresponding feasible solution of the (GMCKP) instance, and viceversa.

□

4.2 Valid Inequalities

Valid inequalities for X_{SCS} are derived from the $(\ell, \mathcal{S})$ valid inequalities formulation for the deterministic uncapacitated lot-sizing problem of Barany, Van Roy, and Wolsey (1984) while considering the extension for the stochastic case of Guan et al. (2006). The derived inequalities establish bounds on decision variables for the nodes of possible paths in the scenario tree.

Definition 1. An inequality

$$\sum_{j=1}^{m} \alpha_j \chi_j \geq \alpha_0,$$

is a *valid inequality* for a set $X \subseteq \mathbb{R}^m$ if $\sum_{j=1}^{m} \alpha_j \chi_j \geq \alpha_0$ for all $\chi \in X$.

Two sets of valid inequalities are derived:

(i) from equations (3.25) and inequalities (3.26) it holds that

$$w_n \leq \sum_{\{i \in A | t(n) = \tau^i\}} q^i, \quad \text{for all } n \in N.$$

From these inequalities and definition of a_t (v. (3.1)) it holds that

$$w_n \leq a_{t(n)}, \quad \text{for all } n \in N. \tag{4.5}$$

Consider the material balance equation (3.21) of model (SCS), for all $n \in N$

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n,$$

given that $s_{p(n)}, a_{t(n)}, y_n \geq 0$ and inequalities (4.5) it follows that

$$u_n \leq d_n + s_n. \tag{4.6}$$

From inequalities (4.6) the following valid inequalities can be established

$$u_n \leq d_n \beta(n) + s_n, \quad \text{for all } n \in N, \tag{4.7}$$

where $\beta(n) := \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} v^i_{p(n,\gamma^i)}$, since for all $n \in N$, if $\beta(n) = 0$, then from (3.23) $u_n = \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} q^i v^i_{p(n,\gamma^i)} = 0$. Otherwise, if $\beta(n) \geq 1$, then (4.7) holds.

In general, from the sum of material balance equation (3.21) between $n \in N$ and $\ell \in S(n)$, the following condition holds

$$u_n \leq d_{n\ell} + s_\ell, \tag{4.8}$$

where $d_{n\ell} := \sum_{n' \in P(\ell) \setminus P(p(n))} d_{n'}$ is the accumulated demand in the nodes in the path from n to ℓ.

From (4.8) the following valid inequalities can be established

$$u_n \leq d_{n\ell} \beta(n) + s_\ell, \quad \text{for all } n \in N, \ell \in S(n), \tag{4.9}$$

similar to the provision for (4.7).

The valid inequalities (4.12) are obtained by adding the inequalities (4.9) for each subset of the set of nodes of the path from the root node to the ℓ node.

(ii) Given the material balance equation (3.21) of model (SCS), for all $n \in N$

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n,$$

given that $s_{p(n)}, a_{t(n)}, y_n \geq 0$ it follows that

$$u_n \leq d_n + w_n + s_n. \tag{4.10}$$

From inequalities (4.10) and following similar provisions than case (i) the following valid inequalities can be established

$$u_n \leq d_{n\ell}\beta(n) + w_{n\ell} + s_\ell, \quad \text{for all } n \in N, \ell \in S(n). \tag{4.11}$$

The valid inequalities (4.13) are obtained by adding the inequalities (4.11) for each subset of the set of nodes of the path from the root node to the ℓ node.

Theorem 1. Let $\ell \in N$ and $\mathcal{S} \subseteq P(\ell)$ then the SCS-$(\ell, \mathcal{S})$ inequalities

$$\text{(i)} \quad \sum_{n\in\mathcal{S}} u_n \leq \sum_{n\in\mathcal{S}} d_{n\ell}\beta(n) + s_\ell, \tag{4.12}$$

$$\text{(ii)} \quad \sum_{n\in\mathcal{S}} u_n \leq \sum_{n\in\mathcal{S}} d_{n\ell}\beta(n) + \sum_{n\in\mathcal{S}} w_{n\ell} + s_\ell, \tag{4.13}$$

are valid for X_{SCS}.

Proof. The proof of (4.12) is based on the deterministic case presented by Barany, Van Roy, and Wolsey (1984). Given a point $(s, v, x, z) \in X_{\text{SCS}}$ there are two cases.

1) If $\beta(n) = \sum_{\{i\in F|t(n)\geq\gamma^i+1\}} v^i_{p(n,\gamma^i)} = 0$ for all $n \in \mathcal{S}$, then $u_n = \sum_{\{i\in F|t(n)\geq\gamma^i+1\}} q^i v^i_{p(n,\gamma^i)} = 0$ for all $n \in \mathcal{S}$ and $s_\ell \geq 0$, therefore the inequality holds.

2) Otherwise, there exists $n \in \mathcal{S}$ such that $\beta(n) = 1$. Let $n' = \text{argmin}\{t(n)|n \in N, \beta(n) = 1\}$. Then $\beta(n) = 0$ and $u_n = 0$ for all $n \in \mathcal{S} \cap P(p(n'))$. Thus $\sum_{n\in\mathcal{S}} u_n \leq \sum_{n\in P(\ell)\setminus P(p(n'))} u_n \leq d_{n'\ell} + s_\ell \leq \sum_{n\in\mathcal{S}} d_{n\ell}\beta(n) + s_\ell$.

The proof of (4.13) is similar considering that $\sum_{n\in\mathcal{S}} w_{n\ell} \geq 0$. □

Lemma 1. The SCS-$(\ell, \mathcal{S})$ inequalities can be written alternatively as

$$\text{(i)} \quad \sum_{n\in P(\ell)\setminus\mathcal{S}} u_n + \sum_{n\in\mathcal{S}} d_{n\ell}\beta(n) + \sum_{n\in P(\ell)} (y_n - w_n) \geq d_{1\ell} - \sum_{n\in P(\ell)} a_{t(n)} - s_0, \quad \text{for all } \ell \in N, \mathcal{S} \subseteq P(\ell). \tag{4.14}$$

$$\text{(ii)} \quad \sum_{n\in P(\ell)\setminus\mathcal{S}} u_n + \sum_{n\in\mathcal{S}} d_{n\ell}\beta(n) + \sum_{n\in P(\ell)} (y_n - w_n) + \sum_{n\in\mathcal{S}} w_{n\ell} \geq d_{1\ell} - \sum_{n\in P(\ell)} a_{t(n)} - s_0, \quad \text{for all } \ell \in N, \mathcal{S} \subseteq P(\ell). \tag{4.15}$$

Proof. The sum of equations (3.21), for all $n \in P(\ell)$ of a given $\ell \in N$, results in

$$s_0 + \sum_{n\in P(\ell)} a_{t(n)} + \sum_{n\in P(\ell)} y_n + \sum_{n\in P(\ell)} u_n = d_{1\ell} + w_{1\ell} + s_\ell,$$

from where it is possible to solve for s_ℓ and substitute it in (4.12) and (4.13), obtaining an alternative representation of the valid inequalities without inventory variables, denoted as SCS-$(\ell, \mathcal{S})$-i and SCS-$(\ell, \mathcal{S})$-ii, respectively. □

The formulation variants in which the inequalities (4.14) and (4.15) are added to (SCS) are called (SCS-$(\ell, \mathcal{S})$-i) and (SCS-$(\ell, \mathcal{S})$-ii), respectively.

4.3 Non-dominated Valid Inequalities

Given that the derived valid inequalities are highly dominated for most experimental instances, a heuristic scheme is established to determine non-dominated ones.

Definition 2. Given a pair of valid inequalities, defined on vector variable $\chi \in \mathbb{R}^m_+$,

$$\sum_{j=1}^{m} \alpha_j \chi_j \geq \alpha_0, \tag{4.16}$$

$$\sum_{j=1}^{m} \beta_j \chi_j \geq \beta_0, \tag{4.17}$$

for a set $X \subseteq \mathbb{R}^m_+$. It is established that inequality (4.16) *dominates* inequality (4.17) if there is $\mu > 0$ such that $\alpha_j \leq \mu \beta_j$ for all $j = 1, ..., m$, $\alpha_0 \geq \mu \beta_0$, and there exist $j = 1, ..., m$ such that $\alpha_j \neq \mu \beta_j$ or $\alpha_0 \neq \mu \beta_0$.

If (4.16) dominates (4.17) then $\{\chi \in \mathbb{R}^m_+ : (4.16)\} \subseteq \{\chi \in \mathbb{R}^m_+ : (4.17)\}$.

Depending on the instance values of parameters q and d, some of the SCS-$(\ell, \mathcal{S})$ valid inequalities of a given subset $\mathcal{S} \subseteq P(\ell)$, $\ell \in N$, may be dominated by other inequalities of a different subset. Therefore, a heuristic procedure was established to determine non-dominated inequalities on the power set of $P(\ell)$.

Let $\chi := (v, x, z) \in \{0, 1\}^m$ be the compound variable of variables v, x and z such that $m = \sum_{i \in F} |N^i_\gamma| + \sum_{i \in A} |N^i_\delta| + \sum_{i \in F, n \in N^i_\delta} |T^i_\epsilon|$. For each $\ell \in N$, the inequalities (4.14) and (4.15) can be rewritten by a projection of the $n \in P(\ell)$ terms of the compound variable as $\alpha(\ell)^T \chi(\ell) \geq \alpha_0(\ell)$, where its left-hand side coefficients and variables are represented by $\alpha(\ell)$ and $\chi(\ell)$, respectively, and the right-hand side by $\alpha_0(\ell)$.

Given the power set of $P(\ell)$, $S^p_\ell := \{S_1, ..., S_{K_\ell}\}$, let $\alpha_k(\ell)$ be the coefficient vector of variable $\chi(\ell)$ for subset $S_k, k \in \{1, ..., K_\ell\}$. Therefore, inequalities (4.14) and (4.15) can be denoted as

$$\alpha_k(\ell)^T \chi(\ell) \geq \alpha_0(\ell), \quad k \in \{1, ..., K_\ell\}, \ell \in N. \tag{4.18}$$

For a given $\ell \in N$, let $k_1, k_2 \in \{1, ..., K_\ell\}$ and that χ is nonnegative, $\alpha_{k_1}(\ell)^T \chi(\ell) \geq \alpha_0(\ell)$ dominates $\alpha_{k_2}(\ell)^T \chi(\ell) \geq \alpha_0(\ell)$, if the component-wise comparison of vectors $\alpha_{k_1}(\ell)$ and $\alpha_{k_2}(\ell)$ is such that $\alpha_{k_1 j}(\ell) \leq \alpha_{k_2 j}(\ell)$ for each component $j = 1, ..., m(\ell)$ and at least for one component the inequality is strict (v. Definition 2 for $\mu = 1$). Let S^d_ℓ be the subset of *dominant* inequalities on S^p_ℓ.

The procedure to obtain S^d_ℓ by pairwise comparison of inequalities has and upper bound of $O(K^2_\ell m)$ operations. If S^d_ℓ has few elements, an efficient heuristic to obtain a non-dominated inequality candidate within the set,

$$k^* := \text{argmin}_{k \in K_\ell} \sum_{j=1}^{m(\ell)} \alpha_{kj}, \tag{4.19}$$

takes $\Theta(K_\ell m(\ell))$ operations. It is denoted $S_\ell^{d*} := \{k^*\}$.

A variant of the (SCS) formulation is generated by including to it the inequalities SCS-$(\ell, \mathcal{S})$-ii for the set S_ℓ^{d*}, for each $\ell \in N$, establishing a formulations denoted as (SCS-$(\ell, \mathcal{S})$-ii) subject to S^{d*}.

4.4 Tightening the Pairing of Valid Inequalities

The objective is to combine valid inequalities, for an integer set, to obtain new tight inequalities. For this purpose, the pairing procedure developed by Guan, Ahmed, and Nemhauser (2007) is adapted to knapsack-set valid inequalities and used in conjunction with the tightening of inequality coefficients.

Definition 3 (Guan, Ahmed, and Nemhauser (2007))**.** Given a pair of valid inequalities

$$\sum_{j=1}^{m} \alpha_j \chi_j \geq \alpha_0 \quad \text{and} \quad \sum_{j=1}^{m} \beta_j \chi_j \geq \beta_0 \quad \text{for a set } X \subset \{0,1\}^m$$

represented by vectors $\alpha = (\alpha_0, \alpha_1, ..., \alpha_m) \in \mathbb{R}^{m+1}$ and $\beta = (\beta_0, \beta_1, ..., \beta_m) \in \mathbb{R}^{m+1}$. Their *pairing*, denoted as $\alpha \circ \beta$, for the case that $\beta_0 \geq \alpha_0$, is defined component-wise as

$$\begin{aligned}
(\alpha \circ \beta)_0 :=& \beta_0, \quad \text{and} \\
(\alpha \circ \beta)_j :=& \begin{cases} \alpha_j & \text{if } \alpha_j \geq \beta_j \\ \beta_j & \text{if } \alpha_j \leq \beta_j, \beta_j \leq \alpha_j + \beta_0 - \alpha_0 \\ \alpha_j + \beta_0 - \alpha_0 & \text{if } \alpha_j \leq \beta_j, \beta_j \geq \alpha_j + \beta_0 - \alpha_0, \end{cases} \\
& \text{for all } j = 1, ..., m.
\end{aligned}$$

Theorem 2 (Guan, Ahmed, and Nemhauser (2007))**.** If vectors α and β define two valid inequalities for X such that $\beta_0 \geq \alpha_0$, then $\alpha \circ \beta$ defines a valid inequality for X.

Definition 4. A valid inequality defined by vector α for a set $X \subset \{0,1\}^m$ may be *tightened* to a new inequality, denoted as $\underline{\alpha}$, by reduction of its components α_j to $\underline{\alpha}_j := \min(\alpha_j, \alpha_0)$ for $j = 1, ..., m$.

Then by Definition 2 inequality $\sum_{j=1}^{m} \underline{\alpha}_j \chi_j \geq \alpha_0$ may dominate inequality $\sum_{j=1}^{m} \alpha_j \chi_j \geq \alpha_0$, thus $\{\chi \in [0,1]^m : \sum_{j=1}^{m} \underline{\alpha}_j \chi_j \geq \alpha_0\} \subseteq \{\chi \in [0,1]^m : \sum_{j=1}^{m} \alpha_j \chi_j \geq \alpha_0\}$.

For the case of knapsack-set valid inequalities the result of applying the pairing and tightening procedures is dependent on their sequence order.

Theorem 3. If vectors α and β define two valid inequalities for a set $X \subset \{0,1\}^m$ such that $\beta_0 \geq \alpha_0$, then $\underline{\alpha} \circ \underline{\beta}$ may dominate $\underline{\alpha \circ \beta}$.

Proof. A case based proof is presented depending on the relationship between the values of α_j and β_j, for $j = 1, ..., m$, and α_0 and β_0. Table 4.1 depicts the 12 cases of relationships with the corresponding component-wise evaluations of $(\underline{\alpha} \circ \underline{\beta})_j$ and $(\underline{\alpha \circ \beta})_j$.

TABLE 4.1: Component-wise evaluations of procedures sequence by relationship case

#	Relationship	$(\underline{\alpha} \circ \underline{\beta})_j$	?	$(\underline{\alpha \circ \beta})_j$
1	$\alpha_j \leq \alpha_0 \leq \beta_j \leq \beta_0$	$\min\{\beta_j, \alpha_j + \beta_0 - \alpha_0\}$	$=$	$\min\{\beta_j, \alpha_j + \beta_0 - \alpha_0\}$
2	$\alpha_j \leq \beta_j \leq \alpha_0 \leq \beta_0$	$\min\{\beta_j, \alpha_j + \beta_0 - \alpha_0\}$	$=$	$\min\{\beta_j, \alpha_j + \beta_0 - \alpha_0\}$
3	$\alpha_0 \leq \alpha_j \leq \beta_j \leq \beta_0$	β_j	$=$	β_j
4	$\alpha_0 \leq \beta_j \leq \alpha_j \leq \beta_0$	β_j	$\leq$	α_j
5	$\beta_j \leq \alpha_j \leq \alpha_0 \leq \beta_0$	α_j	$=$	α_j
6	$\beta_j \leq \alpha_0 \leq \alpha_j \leq \beta_0$	α_0	$\leq$	α_j
7	$\alpha_j \leq \alpha_0 \leq \beta_0 \leq \beta_j$	$\alpha_j + \beta_0 - \alpha_0$	$=$	$\alpha_j + \beta_0 - \alpha_0$
8	$\alpha_0 \leq \alpha_j \leq \beta_0 \leq \beta_j$	β_0	$=$	β_0
9	$\beta_j \leq \alpha_0 \leq \beta_0 \leq \alpha_j$	α_0	$\leq$	β_0
10	$\alpha_0 \leq \beta_j \leq \beta_0 \leq \alpha_j$	β_j	$\leq$	β_0
11	$\alpha_0 \leq \beta_0 \leq \alpha_j \leq \beta_j$	β_0	$=$	β_0
12	$\alpha_0 \leq \beta_0 \leq \beta_j \leq \alpha_j$	β_0	$=$	β_0

For most cases $(\underline{\alpha} \circ \underline{\beta})_j$ is equal to $(\underline{\alpha \circ \beta})_j$, except for cases 4, 6, 9, and 10, where $(\underline{\alpha} \circ \underline{\beta})_j$ is less than or equal $(\underline{\alpha \circ \beta})_j$. The claim follows from Definition 2.

□

Example 1. Given the set

$$X = \{\chi \in \{0,1\}^3 : \chi_1 + 4\chi_2 + 8\chi_3 \geq 2,\ 4\chi_1 + \chi_2 + 6\chi_3 \geq 3\}$$

The inequalities of X are denoted by $\alpha = (2,1,4,8)$ and $\beta = (3,4,1,6)$. Then the inequality corresponding to $\alpha \circ \beta$ is $2\chi_1 + 4\chi_2 + 8\chi_3 \geq 3$, and the one corresponding to $\underline{\alpha \circ \beta}$ is $2\chi_1 + 3\chi_2 + 3\chi_3 \geq 3$. Moreover, the inequality corresponding to $\underline{\alpha} \circ \underline{\beta}$ is $2\chi_1 + 2\chi_3 + 3\chi_3 \geq 3$. While inequality $\underline{\alpha \circ \beta}$ allows to deduce inequality $\chi_2 + \chi_3 \geq 1$, inequality $\underline{\alpha} \circ \underline{\beta}$ allows to deduce inequalities $\chi_1 + \chi_3 \geq 1$ and $\chi_2 + \chi_3 \geq 1$, and cut out point $(0,1,0)$. ■

This procedure may be sequentially applied to cases of inequalities (4.14) and (4.15) for different $\ell \in N$ to obtain new tight inequalities. Most advanced algebraic languages and MIP solvers tighten constraint coefficients in their presolving phases; therefore, it could be advantageous to coordinate presolving phases with the pairing procedure.

Chapter 5

Computational experiments

A series of computational experiments and results for the solution of the (SCS) stochastic formulation and its variants are presented. The chapter summarizes experiments to determine the value of information and solution of the stochastic formulation. It outlines the experimentation performed to determine the performance of formulations derived from valid inequalities. Also, it recapitulates experiments performed to evaluate a stochastic formulation obtained by the heuristic determination of non-dominated of valid inequalities. In the thesis work various experiments were done, and the detail of the experiments was published in works Testuri, Cancela, and Albornoz (2019a) and Testuri, Cancela, and Albornoz (2019b), and under review process in work Testuri, Cancela, and Albornoz (2019c), which are annexed in Appendix A. The current chapter presents a summary, to allow for a more compact and high-level vision of the results, and to gather all of them in one place.

5.1 Value of information and solution of (SCS)

Computational experiments were carried out to assess the goodness of the formulation by calculating stochastic programming metrics that allow the comparison of stochastic and deterministic formulations. Details of the experiments are published in the work Testuri, Cancela, and Albornoz (2019a) (v. Section A.1). It is important to note that, in the aforementioned publication, orders are denoted as cargos, and that they are indexed by character c instead of i.

The experiments were performed on the resolution of various instances of the (SCS) formulation for four information structures of uncertainty (scenario tree). The sizes of each scenario tree (number of scenarios and nodes) and formulation instance (numbers of constraints, variables, and binary variables) for a given distribution of scenario tree arity, periods and already acquired orders (A) and possible orders to be acquired (F) are shown in Table 5.1. Three data instances were randomly generated for each tree structure and cargo distribution, totaling 12 instances.

Computational implementation was performed using AMPL (Fourer, Gay, and Kernighan, 2002) for the algebraic coding of the stochastic model, and GUROBI 6.5 (Gurobi Optimization, LLC, 2018) for the resolution of the instances through its branch and cut solver. The calculations were carried out

TABLE 5.1: Instance size by scenario tree and order distribution

Arity	Periods	Orders [$|O|$ ($|A|+|F|$)]	Scenarios	Nodes	Cons.	Var.	(binary)
2	5	10 (2+8)	16	31	225	249	(124)
2	6	12 (3+9)	32	63	480	549	(296)
3	5	10 (3+7)	81	121	827	809	(324)
3	6	12 (4+8)	243	364	2542	2485	(1028)

on an Intel Core i7 5960X 3.5GHz computer with 20MB cache and 64GB RAM, operating with CentOS-7 Linux system, with a time limit of 900 s.

Optimal value of the instances for the original, "recourse problem" (RP), and the corresponding "wait and see" (WS) and "expected value of perfect information" (EVPI) metrics are shown in Table 5.2 together with average RP resolution time. RP represents the value of the optimum for the original problem. WS represent the lower bound of the optimum in the idealized case of knowing the future with certainty; and EVPI is defined as the difference between RP and WS. It can be seen that EVPI shows a good performance, since large values validate the use of the stochastic programming approach. Additional experiments were performed to evaluate the "value of the stochastic solution" (VSS) metric for each period of a pair of instances, following the proposal of Escudero et al. (2007) (v. Testuri, Cancela, and Albornoz (2019a)).

TABLE 5.2: Optimal values of recourse problem (RP) and "wait and see" approach (WS), and expected value of perfect information measure (EVPI) by instance

Instance	Arity	Periods	Orders	RP	Time(s)	WS	EVPI
1				10,254	0.07	8,993	1,261
2	2	5	10	11,699	0.32	10,428	1,271
3				12,755	0.66	10,601	2,154
4				8,091	6.65	6,961	1,130
5	2	6	12	11,760	14.76	10,415	1,345
6				17,014	31.59	15,561	1,453
7				8,186	1.14	6,177	2,009
8	3	5	10	13,284	20.76	10,945	2,339
9				16,719	17.05	14,294	1,632
10				12,371	(2.34%)†	10,008	2,363
11	3	6	12	6,346	202.68	4,468	1,878
12				11,580	(1.60%)†	9,003	2,577

(†) MIP gap for instances that reached the time limit of 900 s.

The experimental optimal values and stochastic rating metrics obtained show the validity and interest of the stochastic formulation, as well as the benefits that can be obtained with respect to a deterministic variant of the model that considers average demand.

5.2 Valid inequality formulations

Computational experiments were carried out to evaluate if the derived valid inequality formulation (SCS-$(\ell,\mathcal{S})$-i) improve the quality and resolution time (Testuri, Cancela, and Albornoz, 2019c) (v. Section A.2).

The experiments were carried out to compare the resolution of various instances of the (SCS) formulation and the derived valid inequalities formulation (SCS-$(\ell,\mathcal{S})$-i) with set S fixed with the root node. Six scenario tree structures were considered based on rooted perfect directed trees by number of immediate time successors of each node (Arity) and number of periods of the planning horizon as depicted in Table 5.3.

TABLE 5.3: Size of scenario tree structures

Arity(g)	Horizon($\hbar$)	Scenarios($\|L\|$)	Nodes($\|N\|$)
2	5	16	31
2	6	32	63
2	7	64	127
3	5	81	121
3	6	243	364
3	7	729	1,093

A distribution of orders by quantity is associated to each tree structure. Each distribution of orders by quantity ($|O|$) is identified by the sum of numbers of already acquired orders ($|A|$) and possible orders to be acquired ($|F|$), as showed in column labelled "Orders" in Table 5.4. The 3-uple ⟨*Arity, Horizon, Orders*⟩ identifies table rows, denominated as *categories* of data instances. The table depicts, for each category, the numbers of constraints, $(\ell,\mathcal{S})$ inequalities, total variables and binary variables.

TABLE 5.4: Size of instance categories defined by scenario tree structure and order distribution

Arity(g)	Horizon($\hbar$)	Orders($\|O\|$)	(SCS)-const.	$(\ell,\mathcal{S})$-ineq.	Variables	(binary)
2	5	10 (2+ 8)	225	31	249	(124)
2	6	12 (3+ 9)	480	63	549	(296)
2	7	14 (3+11)	1,012	127	1,223	(714)
3	5	10 (3+ 7)	827	121	809	(324)
3	6	12 (4+ 8)	2,542	364	2,485	(1,028)
3	7	14 (4+10)	7,987	1,093	8,091	(3,718)

Thirty data instances were randomly generated for each of the six instance categories, totaling 180 instances. The same software and hardware described in Section 5.1 was used for the computational implementation. For each instance, the formulations were solved either within a time limit of 900 s or without a gap between the objective and its lower bound (MIP-gap = 0). The instances average results of the (SCS) and (SCS-$(\ell,\mathcal{S})$-i) formulations by instance category are presented in Table 5.5 and Table 5.6, respectively (v. Testuri, Cancela, and Albornoz (2019c) for detailed results)

TABLE 5.5: Average results of 30 instances of formulation (SCS) by instance category

g	$\hbar$	$\|O\|$	Time(s)	MIP-gap(%)	Nodes	Cuts	LP-gap(%)
2	5	10	1.60	0	15,261	159	10.31
2	6	12	27.61	0	111,228	473	11.25
2	7	14[a]	868.58	0.73	2,752,608	855	9.89
3	5	10	21.99	0	42,727	444	12.40
3	6	12[b]	821.46	2.86	868,850	1,797	19.69
3	7	14[c]	900.20	5.81	46,457	2,031	24.43
3	*7*	*14*[cd]	*900.14*	*4.47*	*33,177*	*1,520*	*21.32*

[a]28 of 30 instances reach the 900 s time limit. [b]27 of 30 instances reach the 900 s time limit. [c]All instances reach the 900 s time limit. [d]Median value results.

TABLE 5.6: Average results of 30 instances of formulation (SCS-$(\ell, \mathcal{S})$-i) by instance category

g	$\hbar$	$\|O\|$	Time(s)	MIP-gap(%)	Nodes	Cuts	LP-gap(%)
2	5	10	0.67	0	3,835	108	7.48
2	6	12	17.65	0	61,852	431	9.51
2	7	14[a]	542.96	0.48	1,311,339	1,218	7.01
3	5	10	7.94	0	18,532	318	9.74
3	6	12[b]	728.13	2.46	495,200	2,211	17.26
3	7	14[c]	900.32	7.15	55,149	2,048	26.08
3	*7*	*14*[cd]	*900.16*	*4.75*	*31,722*	*1,716*	*17.08*

[a]15 of 30 instances reach the 900 s time limit. [b]22 of 30 instances reach the 900 s time limit. [c]All instances reach the 900 s time limit. [d]Median value results.

The results depicted are solver mean elapsed time at columns "Time", solver mean relative mixed-integer programming gap for instances that reach the time limit of 900 s at column "MIP-gap(%)", solver mean number of nodes of solver branch and cut method at column "Nodes", mean number of cuts added by solver's branch and cut method at column "Cuts", and mean relative linear programming relaxation gap at column "LP-gap(%)". The types of cuts most frequently added by the solver are mixed integer rounding, generalized upper bound cover, infeasibility proof, Gomory, mod-k, cover and network.

The results of formulation (SCS-$(\ell, \mathcal{S})$-i) are better than those of formulation (SCS). Formulation (SCS-$(\ell, \mathcal{S})$-i) has shorter execution times and smaller number of nodes of the branch and cut tree than formulation (SCS) for all instance categories except for category (3,7,14) that are similar. For this category median results are presented on an additional italic font row, due to the presence of outliers. For all instance categories where some of its instances reach the 900 s time limit, the formulation (SCS-$(\ell, \mathcal{S})$-i) obtains a lower or equal number of these instances and a lower MIP-gap average than formulation (SCS). Specifically in the case of instance category (2,7,14), while formulation (SCS) is solved to optimality for 12 instances in the allotted time, formulation (SCS-$(\ell, \mathcal{S})$-i) is solved for 18 instances. This is also the case for category (3,6,12), whereas formulation (SCS) is optimally solved for 3 instances, formulation (SCS-$(\ell, \mathcal{S})$-i) is solved for 8 instances.

5.3 Non-dominated valid inequality formulation

Computational experiments were carried out to evaluate the stochastic formulation (SCS-$(\ell, \mathcal{S})$-ii) instantiated by non-dominated sets of valid inequalities $\mathcal{S}^{d*}$ (Testuri, Cancela, and Albornoz, 2019b) (v. Section A.3).

The non-dominated sets were obtained heuristically. To implement the heuristic, it was necessary to instantiate the constraints, in symbolic form, with data of the instances. Because the indexing of some variables in constraints is not a simple function of the data (e.g. (3.23)), it is difficult to factor the constraints' coefficients of each variable in symbolic form (as required by the compounded notation (4.18)), although the functions are total computable (Enderton, 1977). When instantiating a model with its data, AMPL generates (scalar) identifiers of constraints and variables, but it does not generate identifiers of coefficients of variables in constraints. Therefore, in order to determine the input of the heuristic, it was necessary to parse the instantiated constraints to uniquely determine the coefficients of each variable by constraint. The parser and the heuristic were implemented as a callback procedures with C++ programming language.

The experiments were performed on the same data categories and instances of Section 5.2 (v. Table 5.4). The same computational tools from the previous sections were used, except the GUROBI solver that was upgraded to the 7.5 version.

For each instance, the formulations (SCS) and (SCS-$(\ell, \mathcal{S})$-ii) were solved, and a summary of the results is presented in Table 5.7 and Table 5.8.

TABLE 5.7: Average results of 30 instances of formulation (SCS) by instance category

g	$\hbar$	$\lvert O\rvert$	Mean(s)	Median(s)	MIP(%)	Nodes	Cuts	LP(%)
2	5	10	0.68	0.48	-	6,449	125	10.31
2	6	12	13.07	8.09	-	30,706	189	11.25
2	7	14[a]	493.45	346.92	0.26	1,093,185	713	9.84
3	5	10	13.39	4.35	-	31,260	300	12.40
3	6	12[b]	758.80	900.32	1.90	733,244	1,319	19.64
3	7	14[c]	900.25	900.17	5.02	27,291	1,264	23.42

[a] 12 of 30 instances reach the time limit of 900 s. [b] 24 of 30 instances reach the time limit of 900 s. [c] All instances reach the time limit of 900 s.

TABLE 5.8: Average results of 30 instances of formulation (SCS-$(\ell, \mathcal{S})$-ii) subject to S^{d*} by instance category

g	$\hbar$	$\lvert O\rvert$	Mean(s)	Median(s)	Heur.(s)	MIP(%)	Nodes	Cuts	LP(%)
2	5	10	0.48	0.36	0.05	-	2,693	98	7.46
2	6	12	13.22	3.54	0.17	-	37,603	249	11.25
2	7	14[a]	435.25	186.76	0.67	0.28	1,005,992	1,024	6.97
3	5	10	5.87	2.73	0.15	-	12,281	239	9.72
3	6	12[b]	720.13	900.20	0.76	1.71	513,898	1,609	17.24
3	7	14[c]	900.19	900.13	5.34	4.66	41,495	1,367	20.53

[a] 12 of 30 instances reach the time limit of 900 s. [b] 12 of 30 instances reach the time limit of 900 s. [c] All instances reach the time limit of 900 s.

The average results depicted are solver mean and median elapsed times at columns "Mean" and "Median", respectively, parser and heuristic processing mean time at column "Heuristic", solver mean relative mixed-integer programming gap for instances that reach the time limit of 900 s at column "MIP", solver mean number of nodes of solver branch and cut method at column "Nodes", mean number of cuts added by solver's branch and cut method at column "Cuts", and mean relative linear programming relaxation gap at column "LP". Formulation (SCS-$(\ell, \mathcal{S})$-ii) subject to S^{d*} obtains better overall results than formulation (SCS) for all categories. Except for category (2,6,12), where their mean times are similar; but, their median times are 3.54 and 8.09 s, respectively.

Chapter 6

Conclusions

In this thesis, a stochastic multistage capacitated discrete procurement problem derived from a real fuel supply problem is formulated. Original and recourse decisions over distinguishable orders in the problem are modeled with time delays. The structure of the uncertain information is modeled by a discrete time stochastic process with finite probability, summarized in a scenario tree. Stochastic programming with entities indexed by functions of data was used to formulate a multistage stochastic discrete mixed-integer problem based on the lot-sizing scheme with lead times.

Computational experiments on the formulation were performed for several randomly generated instances within a variety of scenario trees. Most of the experiments were solved to optimality for medium-size instances. The experimental results showed the validity of the model with respect to deterministic formulations over expected value data.

To tighten the problem formulation, two variants of valid inequalities were generated based on the $(\ell, \mathcal{S})$ inequalities approach. Further, a tightened ordering scheme of a known pairing of valid inequalities was proposed. Computational experiments were carried out on two formulation variants associated with the derived valid inequality for randomly generated instances of six tree structures and order distributions categories. Most of the computational experiments were solved to optimality in the small and medium sized categories for the allocated time. One of the formulation variants shows improved performance over the original formulation in terms of execution time, MIP-gap, number of branch and bound nodes, and number of solver cuts.

Since the inequalities are highly dominated for most experimental instances, a heuristic scheme to determine non-dominated ones was established. Computational experiments were performed on one formulation variant of the valid inequalities over a subset of non-dominated ones. Most computational experiments could be solved to optimality for the small and medium-size categories on the allocated time. The non-dominated set variant formulations obtain a slightly better results than the original one.

One aspect of modeling that could be considered in future research may be the inclusion of cancellation and postponement decisions in the set of orders that will be acquired. This aspect was evaluated incipiently; but, it was ruled out given the complexity that the formulation would take. The number of constraints could increase considerably. In addition, nonlinear restrictions may be required for postponement decisions. Another task would be to consider alternative constraints for the generation of valid inequalities. Further direction of future research could be to experiment with the tightening ordering scheme of pairing of valid inequalities while considering the dominance of the inequalities involved.

For this type of formulations, it could be advantageous if the algebraic modeling language allowed the access and operation on "medium level entities" of a model obtained after the higher level symbolic entities are instantiated with data. Some of this medium level entities are already available in advanced languages, but it would be necessary to generate identifiers of coefficients of variables in constraints after data instantiation. This could be done after the current languages pre-solving stage or maybe, even better for control and not for efficiency, providing a novel previous stage of data instantiation. Then, the medium level entities should be available for operation with the current command sub-language (scripting), such that the entities (e.g. constraints) can be modified or generated, by using the high-level knowledge of the problem, and the model can be updated before the solving stage.

Bibliography

Aardal, Karen, Robert Weismantel, and Laurence A. Wolsey (2002). "Non-standard approaches to integer programming". In: *Discrete Applied Mathematics* 123.1, pp. 5 –74. ISSN: 0166-218X. DOI: 10.1016/S0166-218X(01)00337-7. URL: http://www.sciencedirect.com/science/article/pii/S0166218X01003377.

Ahmed, S. (2013). "A scenario decomposition algorithm for 0-1 stochastic programs". In: *Operations Research Letters* 41.6. cited By 25, pp. 565–569. DOI: 10.1016/j.orl.2013.07.009. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84883175358&doi=10.1016%2fj.orl.2013.07.009&partnerID=40&md5=385a669d01f1581065bfa08a97ca09b5.

Ahmed, Shabbir, Alan J. King, and Gyana Parija (2003). "A Multi-Stage Stochastic Integer Programming Approach for Capacity Expansion under Uncertainty". In: *Journal of Global Optimization* 26.1, pp. 3–24. ISSN: 1573-2916. DOI: 10.1023/A:1023062915106.

Balas, Egon (1979). "Disjunctive Programming". In: *Discrete Applied Mathematics 5*, pp. 3–51.

Barany, Imre, Tony J. Van Roy, and Laurence A. Wolsey (1984). "Mathematical Programming at Oberwolfach II". In: ed. by Bernhard Korte and Klaus Ritter. Berlin, Heidelberg: Springer Berlin Heidelberg. Chap. Uncapacitated lot-sizing: The convex hull of solutions, pp. 32–43. ISBN: 978-3-642-00915-0. DOI: 10.1007/BFb0121006. URL: http://dx.doi.org/10.1007/BFb0121006.

Beale, E. M. L. (1955). "On Minimizing A Convex Function Subject to Linear Inequalities". In: *Journal of the Royal Statistical Society. Series B (Methodological)* 17.2, pp. 173–184.

Beale, E.M.L., J.J.H. Forrest, and C.J. Taylor (1980). "Multi-time-period stochastic programming". In: *Stochastic programming (Proc. Internat. Conf., Univ. Oxford, Oxford, 1974)*. London: Academic Press, pp. 387–402.

Benders, J. F. (1962). "Partitioning procedures for solving mixed-variables programming problems". In: *Numerische Mathematik* 4.1, pp. 238–252. ISSN: 0945-3245. DOI: 10.1007/BF01386316. URL: https://doi.org/10.1007/BF01386316.

Beraldi, P. et al. (2006). "Scenario-based planning for lot-sizing and scheduling with uncertain processing times". In: *International Journal of Production Economics* 101.1 SPEC. ISS. cited By 16, pp. 140–149. DOI: 10.1016/j.ijpe.2005.05.018. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-29344461227&doi=10.1016%2fj.ijpe.2005.05.018&partnerID=40&md5=6a4368fd3cccb3338fdc9848148e12c9.

Bertsimas, D. and R. Weismantel (2005). *Optimization over integers*. Belmont, Ma.: Dynamic Ideas.

Birge, John R. (1982). "The value of the stochastic solution in stochastic linear programs with fixed recourse". In: *Math. Programming* 24.3, pp. 314–325. ISSN: 0025-5610.

— (1985). "Decomposition and partitioning methods for multistage stochastic linear programs". In: *Oper. Res.* 33.5, pp. 989–1007. ISSN: 0030-364X.

Birge, John R. and François Louveaux (2011). *Introduction to Stochastic Programming (2nd. ed.)* Springer Series in Operations Research and Financial Engineering. New York: Springer. DOI: 10.1007/b97617.

Birge, John R. and François V. Louveaux (1988). "A multicut algorithm for two-stage stochastic linear programs". In: *European J. Oper. Res.* 34.3, pp. 384–392. ISSN: 0377-2217.

Bitran, Gabriel R. and Horacio H. Yanasse (1982). "Computational Complexity of the Capacitated Lot Size Problem". In: *Management Science* 28.10, pp. 1174–1186. ISSN: 00251909, 15265501. URL: http://www.jstor.org/stable/2630946.

Brahimi, Nadjib, Stéphane Dauzère-Pérès, and Najib M. Najid (2006). "Capacitated Multi-Item Lot-Sizing Problems with Time Windows". In: *Operations Research* 54.5, pp. 951–967. ISSN: 0030364X, 15265463. URL: http://www.jstor.org/stable/25147026.

Carøe, Claus C. and Jørgen Tind (1998). "L-shaped decomposition of two-stage stochastic programs with integer recourse". In: *Math. Programming* 83.3, Ser. A, pp. 451–464. ISSN: 0025-5610.

Cook, Stephen A. (1971). "The Complexity of Theorem-proving Procedures". In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing*. STOC '71. Shaker Heights, Ohio, USA: ACM, pp. 151–158. DOI: 10.1145/800157.805047.

Cornuéjols, Gérard (2008). "Valid inequalities for mixed integer linear programs". In: *Mathematical Programming* 112.1, pp. 3–44. ISSN: 1436-4646. DOI: 10.1007/s10107-006-0086-0. URL: https://doi.org/10.1007/s10107-006-0086-0.

Cunha Neto, E., V.J.M. Ferreira Filho, and E.F. Arruda (2015). "Stochastic economic lot sizing and scheduling problem with pitch interval, reorder points and flexible sequence". In: *International Journal of Production Research* 53.19. cited By 1, pp. 5948–5961. DOI: 10.1080/00207543.2015.1033031. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-84938968546&doi=10.1080%2f00207543.2015.1033031&partnerID=40&md5=1dbb765c5b1bec31c8585d5a39f04b8f.

Dantzig, G. B., D. R. Fulkerson, and S. M. Johnson (1959). "On a Linear-Programming, Combinatorial Approach to the Traveling-Salesman Problem". In: *Operations Research* 7.1, pp. 58–66. ISSN: 0030364X, 15265463. URL: http://www.jstor.org/stable/167593.

Dantzig, George B. (1955). "Linear Programming under Uncertainty". In: *Management Science* 1.3-4, pp. 197–206.

Dantzig, George B. and Albert Medansky (1961). "On the Solution of Two-stage Linear Programs under Uncertainty". In: *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. Berkeley, Calif.: University of California

Press, pp. 165–176. URL: https://projecteuclid.org/euclid.bsmsp/1200512165.

Dempster, M. A. H. (1981). "The expected value of perfect information in the optimal evolution of stochastic systems". In: *Stochastic Differential Systems*. Ed. by M. Arató, D. Vermes, and A. V. Balakrishnan. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 25–40. ISBN: 978-3-540-38564-6.

Edmonds, J. (1968). "Matroid Partition". In: *Mathematics of the Decision Sciences: Part 1*. Ed. by G.B. Dantzig and A.F. Veinott. American Mathematical Society, pp. 335–345.

Enderton, Herbert B. (1977). "Elements of Recursion Theory". In: *Handbook Of Mathematical Logic*. Ed. by Jon Barwise. Vol. 90. Studies in Logic and the Foundations of Mathematics. Elsevier, pp. 527 –566. DOI: 10.1016/S0049-237X(08)71114-5. URL: http://www.sciencedirect.com/science/article/pii/S0049237X08711145.

Escudero, Laureano F. et al. (2007). "The value of the stochastic solution in multistage problems". In: *TOP* 15.1, pp. 48–64. ISSN: 1863-8279. DOI: 10.1007/s11750-007-0005-4. URL: https://doi.org/10.1007/s11750-007-0005-4.

Fleischmann, Bernhard (1990). "The discrete lot-sizing and scheduling problem". In: *European Journal of Operational Research* 44.3, pp. 337 –348. ISSN: 0377-2217. DOI: 10.1016/0377-2217(90)90245-7. URL: http://www.sciencedirect.com/science/article/pii/0377221790902457.

Florian, M., J.K. Lenstra, and A.H.G. Rinnooy Kan (1980). "Deterministic Production Planning: Algorithms and Complexity". In: *Management Science* 26.7, pp. 669–679.

Florian, Michael and Morton Klein (1971). "Deterministic Production Planning with Concave Costs and Capacity Constraints". In: *Management Science* 18.1, pp. 12–20.

Fourer, Robert, David M. Gay, and Brian W. Kernighan (2002). *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press.

Gomory, Ralph E. (1958). "Outline of an Algorithm for Integer Solutions to Linear Programs". In: *Bulletin of the American Mathematical Society* 64.5, pp. 275–278. DOI: 10.1090/S0002-9904-1958-10224-4.

Guan, Yongpei, Shabbir Ahmed, and George L. Nemhauser (2007). "Sequential pairing of mixed integer inequalities". In: *Discrete Optimization* 4.1. Mixed Integer Programming, pp. 21 –39. ISSN: 1572-5286. DOI: 10.1016/j.disopt.2006.10.003. URL: http://www.sciencedirect.com/science/article/pii/S1572528606000831.

Guan, Yongpei et al. (2006). "A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem". English. In: *Mathematical Programming* 105.1, pp. 55–84. ISSN: 0025-5610. DOI: 10.1007/s10107-005-0572-9. URL: http://dx.doi.org/10.1007/s10107-005-0572-9.

Günlük, Oktay and Yves Pochet (2001). "Mixing mixed-integer inequalities". In: *Mathematical Programming* 90.3, pp. 429–457. ISSN: 1436-4646. DOI: 10.1007/PL00011430. URL: https://doi.org/10.1007/PL00011430.

Gurobi Optimization, LLC (2018). *Gurobi Optimizer Reference Manual*. URL: https://www.gurobi.com/wp-content/plugins/hd_documentations/content/pdf/reference_manual.pdf.

Haugen, Kjetil K., Arne Løkketangen, and David L. Woodruff (2001). "Progressive hedging as a meta-heuristic applied to stochastic lot-sizing". In: *European Journal of Operational Research* 132.1, pp. 116–122. ISSN: 0377-2217. DOI: 10.1016/S0377-2217(00)00116-8. URL: http://www.sciencedirect.com/science/article/pii/S0377221700001168.

Hu, Z. and G. Hu (2018). "A multi-stage stochastic programming for lot-sizing and scheduling under demand uncertainty". In: *Computers and Industrial Engineering* 119. cited By 5, pp. 157–166. DOI: 10.1016/j.cie.2018.03.033. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85044543194&doi=10.1016%2fj.cie.2018.03.033&partnerID=40&md5=751564555031b717235394f89b394a1b.

Huang, Kai and Simge Küçükyavuz (2008). "On stochastic lot-sizing problems with random lead times". In: *Operations Research Letters* 36.3, pp. 303–308. ISSN: 0167-6377. DOI: 10.1016/j.orl.2007.10.009. URL: http://www.sciencedirect.com/science/article/pii/S0167637707001460.

Jiang, Ruiwei and Yongpei Guan (2011). "An $O(n^2)$-time algorithm for the stochastic uncapacitated lot-sizing problem with random lead times". In: *Operations Research Letters* 39.1, pp. 74 –77. ISSN: 0167-6377. DOI: 10.1016/j.orl.2010.10.004. URL: http://www.sciencedirect.com/science/article/pii/S0167637710001422.

Kall, Peter (1976). *Stochastic Linear Programming*. Berlin: Springer-Verlag.

Kall, Peter and Stein W. Wallace (1994). *Stochastic Programming*. Chichester: Wiley.

Karp, Richard M. (1972). "Reducibility Among Combinatorial Problems". In: *Complexity of Computer Computations*. Ed. by R.E. Miller and J.W. Thatcher. Plenum Press, pp. 85–103.

Krarup, Jakob and Ole Bilde (1977). "Plant location, Set Covering and Economic Lot Size: An $O(mn)$-Algorithm for Structured Problems". In: *Numerische Methoden bei Optimierungsaufgaben Band 3: Optimierung bei graphentheoretischen und ganzzahligen Problemen*. Ed. by L. Collatz, G. Meinardus, and W. Wetterling. Basel: Birkhäuser Basel, pp. 155–180. ISBN: 978-3-0348-5936-3. DOI: 10.1007/978-3-0348-5936-3_10. URL: http://dx.doi.org/10.1007/978-3-0348-5936-3_10.

Land, Ailsa H. and Alison G. Doig (1960). "An automatic method for solving discrete programming problems". In: *Econometrica* 28.3, pp. 497–520.

Laporte, Gilbert and François V. Louveaux (1993). "The integer L-shaped method for stochastic integer programs with complete recourse". In: *Oper. Res. Lett.* 13.3, pp. 133–142. ISSN: 0167-6377.

Lee, Chung-Yee, Sila Çetinkaya, and Albert P. M. Wagelmans (2001). "A Dynamic Lot-Sizing Model with Demand Time Windows". In: *Management Science* 47.10, pp. 1384–1395. ISSN: 00251909, 15265501. URL: http://www.jstor.org/stable/822493.

Lenstra, A. K., H. W. Lenstra, and L. Lovász (1982). "Factoring polynomials with rational coefficients". In: *Mathematische Annalen* 261.4, pp. 515–534.

ISSN: 1432-1807. DOI: 10.1007/BF01457454. URL: https://doi.org/10.1007/BF01457454.

Liu, Xiao and Simge Küçükyavuz (2018). "A polyhedral study of the static probabilistic lot-sizing problem". In: *Annals of Operations Research* 261.1, pp. 233–254. ISSN: 1572-9338. DOI: 10.1007/s10479-017-2641-x. URL: https://doi.org/10.1007/s10479-017-2641-x.

Louveaux, François V. (1980). "A solution method for multistage stochastic programs with recourse with application to an energy investment problem". In: *Oper. Res.* 28.4, pp. 889–902. ISSN: 0030-364X.

Madansky, Albert (1960). "Inequalities for Stochastic Linear Programming Problems". In: *Management Science* 6.2, pp. 197–204. ISSN: 00251909, 15265501. URL: http://www.jstor.org/stable/2627007.

Miller, Andrew J. and Laurence A. Wolsey (2003). "Tight MIP Formulations for Multi-Item Discrete Lot-Sizing Problems". In: *Operations Research* 51.4, pp. 557–565. ISSN: 0030364X, 15265463. URL: http://www.jstor.org/stable/4132419.

Nemhauser, George and Laurence Wolsey (1988). *Integer and Combinatorial Optimization*. New York: John Wiley & Sons, Inc. ISBN: 9781118627372. DOI: 10.1002/9781118627372.fmatter.

Nemhauser, George L. and Laurence A. Wolsey (1990). "A recursive procedure to generate all cuts for 0–1 mixed integer programs". In: *Mathematical Programming* 46.1, pp. 379–390. ISSN: 1436-4646. DOI: 10.1007/BF01585752.

Noël, Marie-Cécile and Yves Smeers (1986). "On the use of nested decomposition for solving nonlinear multistage stochastic programs". In: *Stochastic programming (Gargnano, 1983)*. Vol. 76. Lecture Notes in Control and Inform. Sci. Berlin: Springer, pp. 235–246.

Norkin, V.I., Yu.M. Ermoliev, and A. Ruszczyński (1998). "On Optimal Allocation of Indivisibles Under Uncertainty". In: *Operations Research* 46.3, pp. 381–395.

Pisinger, David (2001). "Budgeting with bounded multiple-choice constraints". In: *European Journal of Operational Research* 129.3, pp. 471 –480. ISSN: 0377-2217. DOI: 10.1016/S0377-2217(99)00451-8. URL: http://www.sciencedirect.com/science/article/pii/S0377221799004518.

Pochet, Yves and Laurence A. Wolsey (1993). "Lot-Sizing with Constant Batches: Formulation and Valid Inequalities". In: *Mathematics of Operations Research* 18.4, pp. 767–785. ISSN: 0364765X, 15265471. URL: http://www.jstor.org/stable/3690122.

— (1994). "Polyhedra for lot-sizing with Wagner—Whitin costs". In: *Mathematical Programming* 67.1, pp. 297–323. ISSN: 1436-4646. DOI: 10.1007/BF01582225. URL: https://doi.org/10.1007/BF01582225.

Römisch, Werner and Rüdiger Schultz (2001). "Multistage Stochastic Integer Programs: An Introduction". English. In: *Online Optimization of Large Scale Systems*. Ed. by Martin Grötschel, Sven O. Krumke, and Jörg Rambau. Springer Berlin Heidelberg, pp. 581–600. ISBN: 978-3-642-07633-6. DOI: 10.1007/978-3-662-04331-8_29. URL: http://dx.doi.org/10.1007/978-3-662-04331-8_29.

Schrijver, Alexander (1986). *Theory of Linear and Integer Programming*. West Sussex, England: John Wiley & Sons, Ltd.

Sen, Suvrajeet and Julia L. Higle (2005). "The C^3 Theorem and a D^2 Algorithm for Large Scale Stochastic Mixed-Integer Programming: Set Convexification". In: *Mathematical Programming* 104.1, pp. 1–20. ISSN: 1436-4646. DOI: 10.1007/s10107-004-0566-z. URL: https://doi.org/10.1007/s10107-004-0566-z.

Sox, C.R. et al. (1999). "Review of the stochastic lot scheduling problem". In: *International Journal of Production Economics* 62.3. cited By 87, pp. 181–200. DOI: 10.1016/S0925-5273(98)00247-3. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-0032685588&doi=10.1016%2fS0925-5273%2898%2900247-3&partnerID=40&md5=b3525f15e4d3546208f01c8e1c7a4f21.

Testuri, Carlos E., Héctor Cancela, and Víctor M. Albornoz (2019a). "Stochastic discrete lot-sizing with lead times for fuel supply optimization". In: *Pesquisa Operacional* 39, pp. 37 –55. ISSN: 0101-7438. DOI: 10.1590/0101-7438.2019.039.01.0037. URL: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0101-74382019000100002&nrm=iso.

— (2019b). "Undominated valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement". In: *Proceedings of the 8th International Conference on Operations Research and Enterprise Systems*. SCITEPRESS Digital Library, pp. 390–397. DOI: 10.5220/0007395203900397.

— (2019c). "Valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement". In: *Tech. report. Instituto de Computación, Facultad de Ingeniería, Universidad de la República* Under review process.

Testuri, Carlos E., Bernardo Zimberg, and Germán Ferrari (2012). *Modelado estocástico múltiple etapa de adquisición de combustible para la generación de electricidad bajo demanda incierta*. Tech. rep. INCO RT 12-07. Instituto de Computación, Facultad de Ingeniería, Universidad de la República. URL: https://www.fing.edu.uy/inco/pedeciba/bibliote/reptec/TR1207.pdf.

Van Eijl, C.A. and C.P.M. Van Hoesel (1997). "On the discrete lot-sizing and scheduling problem with Wagner-Whitin costs". In: *Operations Research Letters* 20.1, pp. 7 –13. ISSN: 0167-6377. DOI: 10.1016/S0167-6377(96)00043-0. URL: http://www.sciencedirect.com/science/article/pii/S0167637796000430.

Van Hoesel, Stan et al. (1994). "The single-item discrete lotsizing and scheduling problem: optimization by linear and dynamic programming". In: *Discrete Applied Mathematics* 48.3, pp. 289 –303. ISSN: 0166-218X. DOI: 10.1016/0166-218X(92)00182-L. URL: http://www.sciencedirect.com/science/article/pii/0166218X9200182L.

Van Slyke, R. and R.J-B. Wets (1969). "L-shaped linear programs with applications to control and stochastic programming". In: *SIAM Journal on Applied Mathematics* 17, pp. 638–663.

Vargas, V. and R. Metters (2011). "A master production scheduling procedure for stochastic demand and rolling planning horizons". In: *International Journal of Production Economics* 132.2. cited By 18, pp. 296–302. DOI: 10.1016/j.ijpe.2011.04.025. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-79957682405&doi=10.1016%2fj.ijpe.2011.04.025&partnerID=40&md5=db28551df0f61a3cbc32114923f41edb.

Wagelmans, Albert, Stan Van Hoesel, and Antoon Kolen (1992). "Economic Lot Sizing: An $O(n \log n)$ Algorithm That Runs in Linear Time in the Wagner-Whitin Case". In: *Operations Research* 40, S145–S156. ISSN: 0030364X, 15265463. URL: http://www.jstor.org/stable/3840844.

Wagner, Harvey M. and Thomson M. Whitin (1958). "Dynamic Version of the Economic Lot Size Model". In: *Management Science* 5.1, pp. 89–96. DOI: 10.1287/mnsc.5.1.89. URL: http://mansci.journal.informs.org/content/5/1/89.abstract.

Walkup, David W. and Roger J.B. Wets (1967). "Stochastic Programs with Recourse". In: *SIAM Journal on Applied Mathematics* 15.5, pp. 1299–1314. ISSN: 00361399. URL: http://www.jstor.org/stable/2099170.

Wets, Roger J.B. (1966). "Programming Under Uncertainty: The Equivalent Convex Program". In: *SIAM Journal on Applied Mathematics* 14.1, pp. 89–105. ISSN: 00361399. URL: http://www.jstor.org/stable/2946179.

Wolsey, Laurence A. (2002). "Solving Multi-Item Lot-Sizing Problems with an MIP Solver Using Classification and Reformulation". In: *Management Science* 48.12, pp. 1587–1602.

— (2006). "Lot-sizing with production and delivery time windows". In: *Mathematical Programming* 107.3, pp. 471–489. ISSN: 1436-4646. DOI: 10.1007/s10107-005-0675-3. URL: http://dx.doi.org/10.1007/s10107-005-0675-3.

Zimberg, Bernardo, Carlos E. Testuri, and Germán Ferrari (2019). "Stochastic modeling of fuel procurement for electricity generation with contractual terms and logistics constraints". In: *Computers & Chemical Engineering* 123, pp. 49 –63. ISSN: 0098-1354. DOI: 10.1016/j.compchemeng.2018.12.021. URL: http://www.sciencedirect.com/science/article/pii/S0098135418304939.

Appendix A

Publications

A.1 Stochastic discrete lot-sizing with lead times for fuel supply optimization

Pesquisa Operacional (2019) 39(1): 37-55

Printed version ISSN 0101-7438 / Online version ISSN 1678-5142
www.scielo.br/pope
doi: 10.1590/0101-7438.2019.039.01.0037

STOCHASTIC DISCRETE LOT-SIZING WITH LEAD TIMES FOR FUEL SUPPLY OPTIMIZATION

Carlos E. Testuri[1*], Héctor Cancela[1] and Víctor M. Albornoz[2]

Received November 30, 2017 / Accepted October 31, 2018

ABSTRACT. We address the problem of expected cost minimization of meeting the uncertain fuel demand during a time planning horizon, where supply is provided by selecting discrete shipments with lead times. Due to uncertainty and the passage of time, corrective actions can be taken such as cancellation and postponement on supply of shipments with associated costs and delays. This problem is modeled as a stochastic multi-stage capacitated discrete lot-sizing problem with lead times. Computational experiments were performed on the resolution of various instances of the model for four information structures of uncertainty. The experimental optimal values and stochastic rating measures obtained show the validity and interest of the stochastic model, as well as the benefits that can be obtained with respect to a deterministic variant of the model that considers average demand.

Keywords: oil and gas procurement, oil supply chain, stochastic lot-sizing, multi-stage stochastic integer programming, postponement.

1 INTRODUCTION

Distribution and storage of primary products are downstream oil supply chain activities. These involve complex logistic planning under uncertainty of product features and resources. In most non-oil producing countries, or where the production is not sufficient to cover the internal demand, it is necessary to import either crude oil, or even refined products, to cover demand. One of the most important and cheapest transportation modes is by ship. This has an important impact in the supply chain, as it is necessary to negotiate not only volume and price, but also how the delivery will be carried out; shipments are of fixed sizes, ship routes are complex and travel times are usually long. This means that supply contracts must be fixed much in advance of the actual times where the fuel will be needed as stated by Moraes and Faria [20]. Additionally, the demands may vary significantly from the best forecasts; this stochastic component introduces an additional complexity and is a source of costs.

*Corresponding author – https://orcid.org/0000-0002-7736-5062
[1]Departamento de Investigación Operativa, Instituto de Computación, Universidad de la República, Uruguay.
[2]Departamento de Industrias, Campus Santiago Vitacura, Universidad Técnica Federico Santa María, Chile.
E-mails: ctesturi@fing.edu.uy; / cancela@fing.edu.uy; / victor.albornoz@usm.cl

This paper focuses on the minimization of the expectation of costs incurred in decisions made to meet the uncertain demand of fuel over a finite discrete time planning horizon. The fuel is purchased through distinguishable optional shipments with a given size and a time of delivery delay. Shipment delivery delay is relevant in the planning horizon. Due to the passage of time and the unveil of demand uncertainty, it is possible that an acquired shipment that has not been received is no longer necessary. In this case, it could be decided to cancel its acquisition or postpone its delivery; decisions, which require minimum accomplishment times and associated costs. The motivation for this model is a real problem arising in a state-owned Uruguayan oil company that deals with fuel acquisition under contractual and logistic conditions for the energy sector. The demand that the company faces is uncertain, given that thermal electricity generation, as a complement in an electrical system, is highly dependent on renewable sources [24, 32]. While this particular situation of the company focuses on the application of fuel procurement for thermal generation, the present work discusses the formulation and solution of a more general variant of the problem, which can represent all situations in which oil procurement is carried out by selecting discrete cargo options in the context of uncertain demand.

Among different approaches to assist planning decisions in the oil industry, stochastic programming has been shown to be a very successful technique, as demonstrated by Dempster et al. [10] and Pinto et al. [22]. Another successful application is presented by Al-Othman et al. [3], who developed a multi-period stochastic supply chain planning model under uncertain market demands and prices. Christiansen et al. [8] described a comprehensive review on ship routing and scheduling. A stochastic problem of shipping of oil products among ports and the management of product storage was described by Agra et al. [1]. Mixed-integer programming models are used for scheduling refined-oil shipping as shown by Ye, Liang, and Zhu [31]. Xu et al. [30] incorporated the uncertainty of crude shipping delays within crude oil planning and scheduling operations. Oliveira and Hamacher [21] considered an integrated fuel distribution network design and binary capacity expansion problem under a multi-product and multi-period setting.

The fuel-supply problem tackled in this paper can be modeled as an extension of the lot-sizing formulation of Wagner and Whitin [27]; and particularly of the variant with variable capacity or discrete lot-sizing of Fleischmann [12]. For the case in which the parameters are known with certainty (deterministic case), the size of the lot is continuous, and without capacity or with constant capacity, the problem has efficient resolution through dynamic programming as shown by Wagner and Whitin [27] and Wagelmans et al. [26]. Bitran and Yanasse [6] established that the variant with discrete sizing is a generalization of the binary knapsack problem, and belongs to the NP-hard complexity class.

In the case that the parameters are random (stochastic case), the problem can be formulated by stochastic programming [5]. Stochastic programming model decisions are over time interspersed with random events. These events are assumed discrete establishing, by the time dynamics, a tree of scenarios. The objective of the model is to determine an optimal solution that provides coverage for all the scenarios. Haugen, Løkketangen, and Woodruff [16] describe the scenario-based stochastic lot-sizing problem. Ahmed, King and Parija [2] establish an adjusted extended formulation of the non-capacitated problem and showed that the Wagner-Whitin conditions are not satisfied for the stochastic variant. Guan et al. [13, 14] propose valid inequalities and a branch-and-cut algorithm for the non-capacitated variant.

Shipping delay or delivery time of lots, due to production, transport or capacity restrictions, is modeled in some deterministic continuous no-capacitated lot-sizing problems. Lee et al. [19] present a variant in which demands have a compliance interval that has efficient resolution by dynamic programming. Brahimi and Dauzère-Pérès et al. [7, 9] present two variants according to whether the lots are or are not distinguishable with respect to delivery times. These authors propose efficient algorithms of dynamic programming for the distinguishable case and for the undistinguishable case when the order-delivery windows are not inclusive. For these variants, Wolsey [28] sets tight extended formulations. Van den Heuvel and Wagelmans [25] show the equivalence of the lot-sizing problem with production time windows. For the stochastic scenario-based case, Huang and Küçükyavuz [17] establish that the problem can be efficiently solved in the scenario tree size when delivery windows do not intersect in time. Furthermore, Jiang and Guan improve the efficiency of this procedure [18]. The peculiar aspects of the treated problem that were not found in the literature are the corrective decisions of cancellation and postponement with time delay in a stochastic modeling.

The content of the present work is described below. In Section 2 the algebraic model of the problem is presented in two subsections, first a deterministic variant is described, together with the entities of the problem, and then the stochastic variant is presented. In Section 3 experiments are established to determine validity of the model. In Section 4, conclusions and future work are discussed.

2 MODEL

The modelled problem deals with the minimization of the cost expectation incurred in decisions made to meet the uncertain demand of fuel over a finite discrete time planning horizon. The problem is reduced to a single fuel in order to simplify the proposal without loss of generality. To satisfy the demand, there are optional distinguishable shipments, denoted as cargoes, with a non-splittable quantity of the acquired product. The cargoes have relevant delivery times in the planning horizon; so that a significant amount of time elapses between the purchase decision and when the cargo is received. Due to the passage of time and the unveil of demand uncertainty, it could happen that at any given time a cargo, that was previously acquired and not received is no longer necessary. In this case it could be decided to cancel its acquisition or postpone its delivery; decisions, which, in turn, have minimum execution times in relation to the time of delivery and associated costs. After attending the demand in a given period, the amount of remaining fuel is stored up to a certain capacity, to satisfy future demand in a flexible manner.

The mathematical formulation of the problem is presented below. First, a deterministic model of the problem is developed, where definitions and general structure are established. After defining the information structure of the uncertainty from a discrete stochastic process, the stochastic variant of the model is presented.

2.1 Deterministic model

The following is a deterministic formulation of the problem, where the data are assumed to be known with certainty. A discrete time sequential decision process is considered, in which the decisions taken at a given period depend only on the information available up to that period. Main entities of the problem are described as index sets in Table 1. The planning time is represented by a set of discrete time periods, T. The set of cargoes, C, is categorized into already acquired cargoes, A, and possible cargoes to be acquired, P. Decisions are made to purchase possible cargoes, or to cancel and postpone already acquired cargoes for different periods. Due to the nature of the problem, these decisions have relevant compliance times in the planning horizon. Each cargo has a minimum delivery time, between the time the purchase decision is made and the cargo is received. Decisions to cancel and postpone an already purchased cargo must be made before a certain minimum time, prior to the receipt of the cargo. In addition, when deciding to postpone a previously purchased cargo, the time elapsed between the original reception period and the postponement period can not be less than a given minimum postponement time. All these constraints cause some latency in the decision making process.

Table 1 – Index sets.

T	periods, $t \in T := \{1,...,H\}$
A	already acquired cargoes
P	possible cargoes to be acquired
C	cargoes, $c \in C := A \cup P$

Parameters

Parameters are described in Table 2. As mentioned before, we consider a single product (or multiple products which are interchangeable). The demand for the product in each period, d_t, is known. Due to store constraints, the inventory of the product at the end of each period is restricted between a minimum volume, $\underline{s}$, and a maximum volume, $\overline{s}$, and there is an initial storage volume, s_0, at the beginning of the planning horizon.

The period in which an already acquired cargo is received, τ^c, is fixed, and it is decided in previous acquisitions (i.e. previous model resolutions). Each cargo has a given volume, q^c. Decisions on each cargo have latency times measured in periods. The delivery time of a cargo, γ^c, establishes the length of the wait time (measured in periods) between the acquisition decision and the actual arrival of the cargo. The minimum time for cancellation of a cargo, δ^c, establishes the minimum number of periods prior to the delivery period at which the cargo may be cancelled. The minimum postponement time for a cargo, ε^c, establishes the minimum number of periods after the delivery period in which the postponed cargo can be received. The achievement period of decisions on acquisition, cancellation and postponement must take place within the planning horizon.

For each cargo there are unit costs per volume associated with the decisions to acquire, ca^c, cancel, cc^c, and postpone, cp^c, it. In addition, there is a unit cost associated with storage in each period, h_t.

The already acquired volume that is scheduled to be received in each period is determined by the sum of the cargoes that are received in that period as

$$a_t := \sum_{\{c \in A | \tau^c = t\}} q^c, \quad \forall t \in T;$$

this is an auxiliary summary parameter.

Table 2 – Parameters.

d_t	demand volume in period $t \in T$
s_0	initial storage volume
$\underline{s}, \bar{s}$	minimum and maximum storage capacities by period
τ^c	period in which already acquired cargo $c \in A$ is received
q^c	volume of cargo $c \in C$
γ^c	delivery time of cargo $c \in P$, such that $0 \leq \gamma^c \leq H - 1$
δ^c	cancellation minimum time of already acquired cargo $c \in A$, such that $0 \leq \delta^c \leq \tau^c - 1$
ε^c	postponement minimum time of already acquired cargo $c \in A$, such that $0 \leq \varepsilon^c \leq H - \tau^c$
ca^c	acquisition unit cost of cargo $c \in C$
cc^c	cancellation unit cost of cargo $c \in C$
cp^c	postponement unit cost of cargo $c \in C$
h_t	storage unit cost in period $t \in T$
a_t	already acquired volume that is received in period $t \in T$

Variables

Variables are summarized in Table 3. The variables s_t represent the fuel inventory at the end of each period t. The acquisition decision of each cargo c to be acquired in period t, subject to its delivery period, is modeled by the binary variables v_t^c. Decisions to cancel each cargo c in period t, prior to their minimum cancellation time, are established using binary variables x_t^c. When postponing a cargo reception, decisions must be made about when and until when it is done. Since both decisions are independent, the decision to postpone a cargo c is modeled with a prior cancellation decision and a decision whether to delay its receipt to another period t after its minimum delay time, represented by binary variables z_t^c. In addition, there are auxiliary variables for totals per period t of acquired, u_t, cancelled, w_t, and postponed volume, y_t. These totals variables facilitate the representation of inventory balance constraints.

Table 3 – Variables.

s_t	storage volume at the end of period $t \in T$
u_t	acquired volume into period $t \in T$
v_t^c	if cargo $c \in P$ is acquired in period $t \in \{1,...,H-\gamma^c\}$, (binary)
w_t	cancelled volume out of period $t \in T$
x_t^c	if already acquired cargo $c \in A$ is cancelled in period $t \in \{1,...,\tau^c-\delta^c\}$ (binary)
y_t	postponed volume into period $t \in T$
z_t^c	if already acquired cargo $c \in A$ is postponed to period $t \in \{\tau^c+\varepsilon^c,...,H\}$ (binary)

Objective function

Our aim is to determine a minimum cost scheme of inventory, acquisition, cancellation and postponement of fuel cargoes that satisfy demand during the planning horizon. The costs of acquisitions, cancellations and postponements accrue at the time of decision making. The objective function includes budget costs of acquisition, cancellation and postponement, and inventory costs,

$$\min \sum_{t\in T} \Big[\sum_{\{c\in P|t\leq H-\gamma^c\}} ca^c q^c v_t^c \tag{1}$$

$$+ \sum_{\{c\in A|t\leq \tau^c-\delta^c\}} (cc^c - ca^c) q^c x_t^c \tag{2}$$

$$+ \sum_{\{c\in A|\tau^c+\varepsilon^c\leq t\}} (ca^c - cc^c + cp^c) q^c z_t^c \tag{3}$$

$$+ h_t s_t \Big]. \tag{4}$$

where: the expression (1) represents the costs of acquiring possible cargoes, the expression (2) represents the costs of cancellation minus the budgeted acquisition costs of already acquired cargoes that are cancelled, the expression (3) represents the acquisition costs minus the costs of cancellation plus the costs of postponement of the already acquired cargoes that are postponed, and the expression (4) represents inventory costs. The postponement cost includes the subtraction of cancellation costs, since a postponement is represented by a prior cancellation, but it does not incur cancellation costs.

Constraints

The main requirement is to satisfy demand while maintaining the inventory balance with contributions of the fuel previously stored, what was already acquired, what is acquired, and what is cancelled, postponed and available in storage,

$$s_{t-1} + a_t + u_t + y_t = d_t + w_t + s_t, \quad \forall t \in T, \tag{5}$$

where s_0 is the given initial storage.

The amount stored in each period is constrained between lower and upper bounds,

$$\underline{s} \leq s_t \leq \overline{s}, \quad \forall t \in T. \tag{6}$$

The amount of acquired fuel that is received in each period is determined by the sum of the cargo acquisitions in the possible range of the corresponding acquisition periods,

$$u_t = \sum_{\{c \in P | \gamma^c \leq t-1\}} q^c v^c_{t-\gamma^c}, \quad \forall t \in T. \tag{7}$$

If a cargo is acquired, the acquisition has been decided in a single period before or equal to its possible receiving period less its delivery time γ^c,

$$\sum_{t=1}^{H-\gamma^c} v^c_t \leq 1, \quad \forall c \in P. \tag{8}$$

The already acquired volume that is cancelled in each period is determined by the cancellations of the cargoes in the possible range of the corresponding cancellation periods,

$$w_t = \sum_{\{c \in A | \tau^c = t\}} \left(q^c \sum_{t'=1}^{\tau^c - \delta^c} x^c_{t'} \right), \quad \forall t \in T. \tag{9}$$

If a cargo is cancelled, the cancellation is decided in a single period before or equal to its receiving period τ^c minus its cancellation time δ^c,

$$\sum_{t=1}^{\tau^c - \delta^c} x^c_t \leq 1, \quad \forall c \in A. \tag{10}$$

The postponement of a cargo is modeled by the use of cancellation, that is to say, it is only possible to postpone cargoes that are cancelled. The already acquired volume that is postponed to a certain period is determined by the postponements of the cargoes in the possible range of the corresponding postponement periods,

$$y_t = \sum_{\{c \in A | \tau^c + \varepsilon^c \leq t\}} q^c z^c_t, \quad \forall t \in T. \tag{11}$$

If a cargo is postponed, it is to be received in a single period subsequent to or equal to its original receiving period τ^c plus its delay time ε^c,

$$\sum_{t=\tau^c+\varepsilon^c}^{H} z_t^c \leq 1, \quad \forall c \in A. \tag{12}$$

A cargo can be postponed, if its original arrival decision has been cancelled,

$$\sum_{t=\tau^c+\varepsilon^c}^{H} z_t^c \leq \sum_{t=1}^{\tau^c-\delta^c} x_t^c, \quad \forall c \in A. \tag{13}$$

Finally, there are domain constraints of the variables

$$\begin{aligned} & s_t, u_t, w_t, y_t \geq 0, \quad \forall t \in T, \\ & v_t^c \in \{0,1\}, \quad \forall c \in P, t \in \{1,...,H-\gamma^c\}, \\ & x_t^c \in \{0,1\}, \quad \forall c \in A, t \in \{1,...,\tau^c-\delta^c\}, \\ & z_t^c \in \{0,1\}, \quad \forall c \in A, t \in \{\tau^c+\varepsilon^c,...,H\}. \end{aligned} \tag{14}$$

This formulation is a generalization of the discrete lot-sizing problem (DLS-C) described by Wolsey [29]. The problem DLS-C is a special case of the formulation in which the acquisition variables are selected among a special ordered set of type one (cf. constraint (8)), the delivery time of each cargo is zero, and there are no decisions of cancellation or postponement. This formulation belongs to the NP-hard class, since DLS-C belongs to the same class [6].

The feasibility of this formulation is conditioned to the timely availability of fuel acquisition to satisfy its demand,

$$s_0 + \sum_{\{c \in A | \tau^c \leq t\}} q^c + \sum_{\{c \in P | \gamma^c \leq t-1\}} q^c \geq \sum_{t'=1}^{t} d_{t'}, \quad \forall t \in T.$$

2.2 Stochastic model with uncertain demand

The following model is a stochastic extension of the deterministic model in which demand is a random parameter. Previously, the uncertain information structure was established through stochastic optimization [5]. Subsequently, the entities and formulation of the stochastic model are reported.

Uncertain information structure

The uncertain demand is represented by a discrete-time stochastic process indexed in the planning periods; in such a way that each stage of the stochastic process is associated to a period. The process is defined in a finite probability space. It is assumed that the demand of the first period is deterministic, and that the demands of the remaining periods are random with known distribution function. The decisions of a period only depend on the outcomes of the random parameters of previous periods. This process is non-anticipatory of the future decisions or the realizations of the random event outcomes. This information structure can be represented by a tree structure with H levels or stages called *tree of scenarios* [23]. This is a perfect directed tree, with the root node in period $t = 1$ and with leaf nodes in period $t = H$ (identifying the scenarios).

Each node of the scenario tree describes the state of the process and is identified by a period and a scenario. An alternative abbreviated notation is to identify the nodes by a single index n in a numerable set of nodes, N. For the first period, $t = 1$, there is a unique node, called r, that represents the root of the tree. Each node $n \in N$ has an immediate predecessor $p(n)$ node; the auxiliary node 0 is defined as the predecessor of the root node, $0 := p(r)$, such that $0 \notin N$. The period corresponding to each node n is defined as $t(n)$. The probability of the state of each node n is defined as $\pi(n)$, such that $\sum_{n \in N|t(n)=t} \pi(n) = 1$, for all $t = 1,...,H$. The t-th predecessor of node n is defined as $p(n,t)$ and the nodes of the path from the root node to n as $P(n)$. The successors of node n are defined as $S(n) := \{n' \in N, t = 1,...,H - t(n) | n = p(n',t)\}$. The nodes of the path from a given node n_1 to a successor node $n_2 \in S(n_1)$ is defined as $P(n_1,n_2)$. Leaf nodes are defined as $L := \{n \in N | t(n) = H\}$.

Stochastic model formulation

From the scenario tree notation, the formulation of the deterministic model (cf. Section 2.1) is extended into a multi-stage stochastic optimization scheme considering the uncertainty in demand.

New index sets are established according to Table 4. First of all, the set of nodes and their subset of leaf nodes are incorporated into the formulation. Subsets of the set of nodes are established to abbreviate the denomination of nodes where it is possible to acquire each cargo, N^c_γ, and where it is possible to cancel and postpone each cargo, N^c_δ. In addition, subsets of periods to where it is possible to postpone each cargo are established, T^c_ε. With the purpose of disambiguation with respect to the original sets, these subsets are named with the parameters that define them as suffixes.

Table 4 – Stochastic model index sets.

N	nodes of the scenario tree
L	leaf nodes of the scenario tree, $L := \{n \in N \| t(n) = H\}$
N^c_γ	nodes where it is posible to acquire cargo $c \in P$
	$N^c_\gamma := \{n \in N \| t(n) \leq H - \gamma^c\}$
N^c_δ	nodes where it is posible to cancel and postpone cargo $c \in A$
	$N^c_\delta := \{n \in N \| t(n) \leq \tau^c - \delta^c\}$
T^c_ε	periods to where it is possible to postpone cargo $c \in A$
	$T^c_\varepsilon := \{t \in T \| t \geq \tau^c + \varepsilon^c\}$

The random parameter and mapping operators on the nodes of the stochastic model are established in Table 5. The demand parameter, d_n, which in the deterministic model depends on the periods, in the stochastic model depends on the nodes of the tree.

Table 5 – Parameters and operators of the stochastic model.

d_n	demanded volume in node $n \in N$
$t(n)$	period of node $n \in N$
$p(n)$	immediate predecessor node of node $n \in N$ in the tree
$p(n,t)$	t-th predecesor node of node $n \in N$ in the tree
$\pi(n)$	probability of node $n \in N$
$P(n)$	nodes in the path from root node to node $n \in N$
$S(n)$	successor nodes of node $n \in N$ in the tree
$P(n_1,n_2)$	nodes in the path from node n_1 to a successor node $n_2 \in S(n_1)$

In the stochastic model, most decisions depend on the nodes of the tree according to Table 6. In contrast to the deterministic model, the decision to postpone a cargo, into a given period, could be taken in different nodes; this is modeled by variable z^c_{nt}.

Table 6 – Variables of the stochastic model.

s_n	stored volume at the end of period in node $n \in N$
u_n	acquired volume into node $n \in N$
v^c_n	if cargo $c \in P$ is acquired in node $n \in N^c_\gamma$, (binary)
w_n	cancelled volume out of node $n \in N$
x^c_n	if already acquired cargo $c \in A$ is cancelled in node $n \in N^c_\delta$ (binary)
y_n	postponed volume into node $n \in N$
z^c_{nt}	if already acquired cargo $c \in A$ is postponed in node $n \in N^c_\delta$
	to period $t \in T^c_\varepsilon$ (binary)

The indexes of periods in deterministic parameters or variables are referred to the temporary realization of a node n by $t(n)$. This is the case for the already acquired volume parameter, which is indexed as $a_{t(n)}$.

From the previous definitions, the formulation of the multi-stage stochastic optimization model is

$$\min \sum_{n\in N} \pi(n)\Big[\sum_{\{c\in P|n\in N_\gamma^c\}} ca^c q^c v_n^c \tag{15}$$

$$+ \sum_{\{c\in A|n\in N_\delta^c\}} (cc^c - ca^c) q^c x_n^c \tag{16}$$

$$+ \sum_{\{c\in A, t\in T_\varepsilon^c|n\in N_\delta^c\}} (cp^c + ca^c - cc^c) q^c z_{nt}^c \tag{17}$$

$$+ h_{t(n)} s_n \Big], \tag{18}$$

s.a.

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n, \quad \forall n \in N, \tag{19}$$

$$\underline{s} \le s_n \le \bar{s}, \quad \forall n \in N, \tag{20}$$

$$u_n = \sum_{\{c\in P|\gamma^c+1\le t(n)\}} q^c v_{p(n,\gamma^c)}^c, \quad \forall n \in N, \tag{21}$$

$$\sum_{n'\in P(n)} v_{n'}^c \le 1, \quad \forall c \in P, \forall n \in N | t(n) = H - \gamma^c, \tag{22}$$

$$w_n = \sum_{\{c\in A|t(n)=\tau^c\}} \left(q^c \sum_{\{n'\in P(n)|t(n')\le \tau^c-\delta^c\}} x_{n'}^c \right), \quad \forall n \in N, \tag{23}$$

$$\sum_{n'\in P(n)} x_{n'}^c \le 1, \quad \forall c \in A, \forall n \in N | t(n) = \tau^c - \delta^c, \tag{24}$$

$$y_n = \sum_{\{c\in A|t(n)\ge \tau^c+\varepsilon^c\}} \left(q^c \sum_{\{n'\in P(n)\cap N_\delta^c\}} z_{n't}^c \right), \quad \forall n \in N, \tag{25}$$

$$\sum_{\{n'\in P(n), t\in T_\varepsilon^c\}} z_{n't}^c \le 1, \quad \forall c \in A, \forall n \in N_\delta^c, \tag{26}$$

$$x_n^c \ge z_{nt}^c, \quad \forall c \in A, \forall n \in N_\delta^c, \forall t \in T_\varepsilon^c, \tag{27}$$

$$s_n, u_n, w_n, y_n \ge 0, \quad \forall n \in N,$$

$$v_n^c \in \{0,1\}, \quad \forall c \in P, \forall n \in N_\gamma^c,$$

$$x_n^c, z_{nt}^c \in \{0,1\}, \quad \forall c \in A, \forall n \in N_\delta^c, \forall t \in T_\varepsilon^c.$$

This formulation takes into account the same structural properties of the deterministic model extended with the information structure of the scenario tree. It minimizes the expectation of acquisition costs (15), cancellation costs minus acquisition costs in case of cancellation (16), postponement costs plus acquisition costs minus postponement costs (a postponement is modeled in conjunction with a cancellation) and storage costs (18).

The constraint (19) sets the volume balance for each node. The lower and upper storage bounds at each node are determined by (20). The amount of acquired fuel that is received in each node is determined by acquisitions of cargoes in the possible range of the corresponding acquisition periods according to (21). The constraint (22) states that each cargo is acquired at a single node in each path from the root node to a node whose period coincides with the receiving period minus the delivery time of the cargo. The fuel previously acquired that is cancelled at each node is determined by the cancellations of the nodes in the path from the root node to the node, whose cancellation periods are less than the delivery period less the cancellation time, according to (23). The constraint (24) states that each cargo to be cancelled is at a single node in each path from the root node to a node whose period coincides with the receiving period minus the cancellation time of the cargo. The postponement of the cargoes is modeled in conjunction with the cancellation, i.e. only cancelled cargoes can be postponed, (27). The already acquired volume that is postponed in a node is determined by the postponements of the cargoes in the nodes in the path from the root to the node for all periods superior to the period of reception plus the delay time of the node, according to (25). The constraint (26) states that each cargo to be postponed is at a single node in each path from the root node to a node in some period greater than the receiving period plus the time of postponement of the node. The formulation does not need nonanticipativity constraints because the direct indexing of the decisions on the scenario tree implicitly establishes this condition.

The stochastic formulation does not guarantee that all solutions that satisfy constraints at initial stages are also feasible in the remaining stages; that is, it does not have *complete recourse* [5]. This formulation is a generalization of the deterministic one. The deterministic formulation is a special case of the stochastic for the case of a unique scenario in which the tree of scenarios is reduced to a path. Therefore, the feasibility of the stochastic formulation is conditioned to the timely availability of fuel to satisfy its demand along the path of each node of the tree,

$$s_0 + \sum_{\{c \in A | \tau^c \leq t(n)\}} q^c + \sum_{\{c \in P | \gamma^c + 1 \leq t(n)\}} q^c \geq \sum_{n' \in P(n)} d_{n'}, \quad \forall n \in N.$$

3 EMPYRICAL ANALYSIS

In order to generate a number of diverse test instances, four scenario tree structures were considered. Each structure, depicted in Table 7, is determined by the number of direct descendants of each node (tree arity) and the number of periods of the planning horizon. The time size of the periods is measured in weeks. For each tree structure with arity g and H periods there are g^{H-1} escenarios and $(g^H - 1)/(g - 1)$ nodes.

Table 7 – Size of scenario tree structures.

Arity	Periods	Scenarios	Nodes
2	5	16	31
2	6	32	63
3	5	81	121
3	6	243	364

In order to show how the stochastic process is considered, the scenario tree structure with arity $g = 2$ and $H = 5$ periods is depicted in Figure 1.

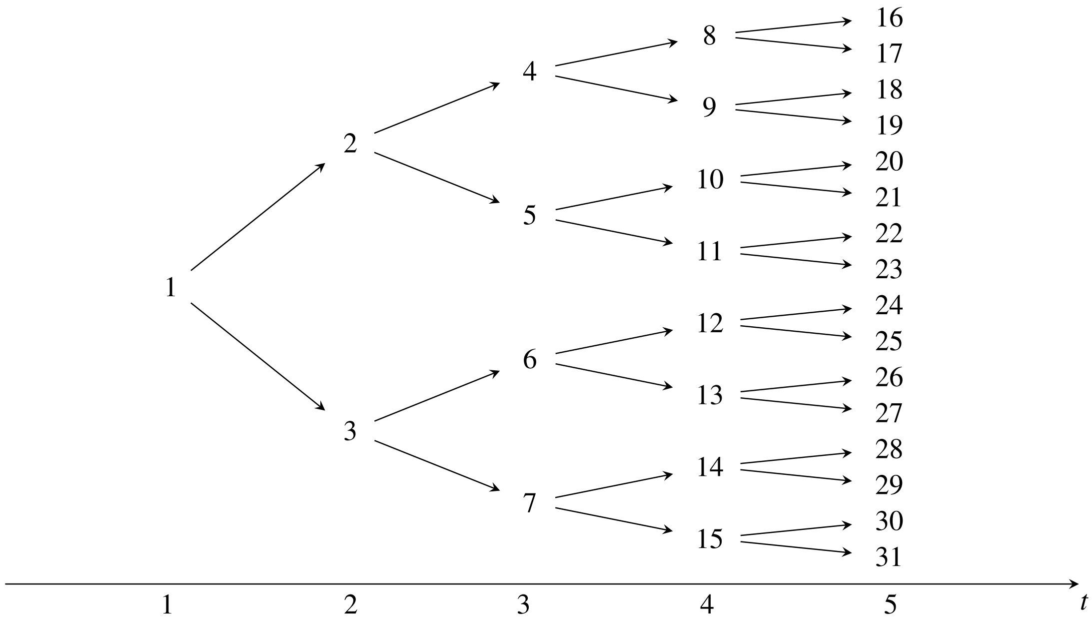


Figure 1 – Tree structure with arity $g = 2$ and $H = 5$ periods where the set of 31 nodes is numbered sequentially by stage increase and the 16 scenarios are defined as the paths from root node 1 to each leaf node 16 to 31.

The size of each tree structure model instance (number of equations and variables) for a given distribution of already acquired cargoes (A) and possible cargoes to be acquired (P) is shown in Table 8.

Table 8 – Instance size of scenario tree structures by cargo distribution.

Arity	Periods	Cargoes $[\lvert C\rvert\ (\lvert A\rvert + \lvert P\rvert)]$	Equations	Variables	(binary)
2	5	10 (2+8)	225	249	(124)
2	6	12 (3+9)	480	549	(296)
3	5	10 (3+7)	827	809	(324)
3	6	12 (4+8)	2542	2485	(1028)

Three data instances were generated for each tree structure and cargo distribution, totaling 12 instances. Each instance has an initial storage, $s_0 = 20$, and a lower and an upper bound storage, $\underline{s} = 0$ and $\overline{s} = 80$, respectively. For each cargo $c \in C$ there is an uniformly distributed volume, $q^c \sim U[10,50]$, and there are costs evenly distributed according to the operations of acquisition, $ca^c \sim U[150,250]$, cancellation, $cc^c \sim U[30,50]$, and postponement, $cp^c \sim U[5,12]$. Each already acquired cargo $c \in A$ has delivery period $\tau^c = 1$ or 2 with equal probability. Each cargo $c \in C$ has delivery time $\gamma^c = 1$, cancellation time $\delta^c = 1$ and delay time $\varepsilon^c = 1$. The unit storage cost in each period t is $h_t = 1$. For each scenario $n \in L$ (leaf node), a probability of state $\pi(n)$ is established from a distribution $Beta(\alpha = 2, \beta = 2)$; the probability of the remaining nodes is obtained from the sum of the probabilities of their corresponding immediate successor nodes. Finally, the demand for each node is evenly distributed, $d_n \sim U[10,50]$.

The computational implementation was performed using AMPL [4] for the algebraic coding of the stochastic model, and GUROBI 6.5 [15] for the resolution of the instances through its branch and cut solver. The calculations were carried out on an Intel Core i7 5960X 3.5GHz computer with 20MB cache and 64GB RAM, operating with CentOS-7 Linux system.

Optimal values of the instances are shown in Table 9. The optimal value and time of resolution of the stochastic model, denoted as recourse problem, is depicted at attributes RP and Time, the optimal value of the "wait and see" approach is represented with attribute WS, and the uncertainty measure "expected value of perfect information" is shown at attribute EVPI [5]. WS represent the lower bound of the optimum in the idealized case of knowing the future with certainty; and EVPI is defined as the difference between RP and WS. It can be seen that EVPI shows a good performance, since large values validate the use of the stochastic programming approach. Unsurprisingly, instances of larger problems take longer execution time. Particularly the instances 10 and 11 reach the time limit of 900 s. Their best feasible solutions do not change after 48 and 294 s, respectively, and both optimality MIP gap's marginal rate reduction, at the time limit, are 5×10^{-6} per second.

Table 9 – Optimal values of recourse problem (RP) and "wait and see" approach (WS), and expected value of perfect information measure (EVPI) of the instances.

Instance	Arity	Periods	Cargoes	RP	Time(s)	WS	EVPI
1				10,254	0.07	8,993	1,261
2	2	5	10	11,699	0.32	10,428	1,271
3				12,755	0.66	10,601	2,154
4				8,091	6.65	6,961	1,130
5	2	6	12	11,760	14.76	10,415	1,345
6				17,014	31.59	15,561	1,453
7				8,186	1.14	6,177	2,009
8	3	5	10	13,284	20.76	10,945	2,339
9				16,719	17.05	14,294	1,632
10				12,371	(2.34%)†	10,008	2,363
11	3	6	12	6,346	202.68	4,468	1,878
12				11,580	(1.60%)†	9,003	2,577

(†) MIP gap for instances that reach the time limit of 900 s.

The "expectation of the expected value problem" for each period t, EEV_t, is obtained by solving the problem while fixing the solution of its variables up to time t with the solution of a variant of the problem in which the uncertain parameter is substituted by the mean. Then the "values of the stochastic solution" measure corresponding to period t, VSS_t, is defined as the difference between each EEV_t and RP. The values of VSS_t for the periods of the first two instance are reported in Table 10. These values measures the importance of using the distribution of the uncertain outcomes, and they are calculated according to the proposal of Escudero et al. [11]. The VSS_t values depicted confirm the model time-step advantage with respect to the solution of the expected value problem. For the case of Instance 1 an $t = 3$, the value of VSS_3 indicates that the stochastic model obtains a solution with an expected reduced cost of 3,489 with respect to a deterministic variant of the model that consider expected value demand. For larger values of t, the tightening of the solution of the expected value problems turn the EEV_t values infeasible, since the formulation does not have complete recourse and the settling of the variable makes the EEV_t problems increasingly restrictive as t increases.

Table 10 – Optimal value of expected result of using the expected value problem (EEV) and value of the stochastic solution measure (VSS) for the periods by selected instances.

Instance	Arity	Periods	Cargoes	t	EEV_t	VSS_t
				1	10,254	
				2	10,524	270
1	2	5	10	3	13,743	3,489
				4	infeas.	∞
				5	infeas.	∞
				1	11,699	
				2	11,699	0
2	2	5	10	3	12,125	426
				4	12,146	21
				5	infeas.	∞

Note that EEV_1 = RP, therefore VSS_1 = 0.

Table 11 shows a summary of the optimal solution of the instances. For each instance, it depicts the number of cargoes that are acquired, $\#\text{AcqCrg} = |\{c \in P : n \in N_\gamma^c, v_n^c = 1\}|$, among the available ones that total between 7 and 8 depending upon the instance (cf. Table 8). The number of decisions on acquisition made is depicted in $\#\text{Acq} = |\{c \in A, n \in N_\gamma^c : v_n^c = 1\}|$. The results show a high granularity of the acquisition process, because the optimal acquisition decisions are made at many different nodes; and the number of nodes where acquisition decisions are made grows as the scenario tree size increases. The number of decisions on cancellation and postponement, among the already acquired cargoes that total between 2 and 4 depending upon the instance (cf. Table 8), made are depicted in $\#\text{Can} = |\{c \in A, n \in N_\delta^c : x_n^c = 1\}|$ and $\#\text{Pos} = |\{c \in A, n \in N_\delta^c, t \in T_\varepsilon^c : z_{nt}^c = 1\}|$, respectively, for each instance. As can be seen that these decisions are sporadic, which is valuable from the commercial and logistical point of view. Furthermore, it may be difficult to detect them in advance without the support of a model. The model's advantage of detecting all these decisions in advance allows an early probabilistic consideration of their execution.

Table 11 – Optimal solution summary with cardinality of selected cargos and cardinality of acquisition, cancellation and postponement decisions by instances.

Inst.	Arity	Periods	Cargoes	#AcqCrg	#Acq	#Can	#Pos
1				4	8	0	0
2	2	5	10	4	12	0	0
3				5	11	0	0
4				7	32	1	0
5	2	6	12	6	36	1	0
6				7	30	0	0
7				4	36	0	0
8	3	5	10	6	36	1	0
9				6	31	0	0
10†				7	74	3	0
11	3	6	12	6	87	1	1
12†				6	81	2	1

(†) Instances that reach the time limit of 900 s.

4 CONCLUSIONS

In this paper, we proposed a stochastic multi-stage capacitated discrete lot-sizing model formulation for a discrete cargo fuel supply with lead times problem. The decisions of the problem were represented in detail with their delay time, aspect that for cancellation and postponement decisions is not covered in previous literature. The structure of the uncertain information was modeled by a discrete time stochastic process with finite probability, summarized in a scenario tree. Stochastic programming methodology with entities indexed by nodes of the scenario tree was used to formulate the model. The model extends deterministic models of the literature, which implied the revision of the definitions of the variables and the restrictions to take into account the structure of the scenario tree.

Computational experiments where carried out for several instances with scenarios of a variety of sizes and characteristics. Most of these experiments were solved to optimality for the medium-size generated instances. The experimental results have shown the validity of the model. Results shows a considerable advantage over the expected value of corresponding deterministic models. Although medium-sized instances could be solved, computational times grew significantly, and for scenarios with more stages or more decisions per stage it is important to improve computational efficiency. Therefore, for future work, it is recommended to study reformulations of the stochastic model with tight high level relaxations, and develop resolution techniques according to them.

ACKNOWLEDGEMENTS

The authors thank the anonymous reviewers for their helpful comments. This work has been supported by the Programa de Desarrollo de las Ciencias Básicas (PEDECIBA), Uruguay.

REFERENCES

[1] AGRA A, CHRISTIANSEN M, DELGADO A & HVATTUM LM. 2015. A maritime inventory routing problem with stochastic sailing and port times. *Computers and Operations Research*, **61**: 18–30, http://www.sciencedirect.com/science/article/pii/S0305054815000210.

[2] AHMED S, KING AJ & PARIJA G. 2003. A Multi-Stage Stochastic Integer Programming Approach for Capacity Expansion under Uncertainty. *Journal of Global Optimization*, **26**(1): 3–24, http://dx.doi.org/10.1023/A:1023062915106.

[3] AL-OTHMAN WB, LABABIDI HM, ALATIQI IM & AL-SHAYJI K. 2008. Supply chain optimization of petroleum organization under uncertainty in market demands and prices. *European Journal of Operational Research*, **189**(3): 822–840, http://www.sciencedirect.com/science/article/pii/S0377221707006595.

[4] AMPL. AMPL Optimization Inc. http://www.ampl.com, online; accessed July, 2018.

[5] BIRGE JR & LOUVEAUX F. 1997. *Introduction to Stochastic Programming*. Springer, New York, springer Series in Operations Research and Financial Engineering.

[6] BITRAN GR & YANASSE HH. 1982. Computational Complexity of the Capacitated Lot Size Problem. *Management Science*, **28**(10): 1174–1186, http://www.jstor.org/stable/2630946.

[7] BRAHIMI N, DAUZÈRE-PÉRÈS S & NAJID N. 2005. Capacitated multi-item lot-sizing problems with time windows. Tech. rep., Ecole des Mines de Nantes.

[8] CHRISTIANSEN M, FAGERHOLT K, NYGREEN B & RONEN D. 2013. Ship routing and scheduling in the new millennium. *European Journal of Operational Research*, **228**(3): 467–483, http://www.sciencedirect.com/science/article/pii/S0377221712009125.

[9] DAUZÈRE-PÉRÈS S, BRAHIMI N, NAJID N & NORDLI A. 2005. Uncapacitated lot-sizing problems with time windows. Tech. rep., Ecole des Mines de Saint-Etienne.

[10] DEMPSTER MAH, HICKS PEDRÓN N, MEDOVA EA, SCOTT JE & SEMBOS A. 2000. Planning logistics operations in the oil industry. *Journal of the Operational Research Society*, **51**(11): 1271–1288, https://doi.org/10.1057/palgrave.jors.2601043.

[11] ESCUDERO LF, GARÍN A, MERINO M & PÉREZ G. 2007. The value of the stochastic solution in multistage problems. *TOP*, **15**(1): 48–64, https://doi.org/10.1007/s11750-007-0005-4.

[12] FLEISCHMANN B. 1990. The discrete lot-sizing and scheduling problem. *European Journal of Operational Research*, **44**(3): 337–348, http://www.sciencedirect.com/science/article/pii/0377221790902457.

[13] GUAN Y, AHMED S, MILLER AJ & NEMHAUSER GL. 2006. On formulations of the stochastic uncapacitated lot-sizing problem. *Operations Research Letters*, **34**(3): 241–250, http://www.sciencedirect.com/science/article/pii/S016763770500057X.

[14] GUAN Y, AHMED S, NEMHAUSER GL & MILLER AJ. 2006. A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem. *Mathematical Programming*, **105**(1): 55–84, http://dx.doi.org/10.1007/s10107-005-0572-9.

[15] GUROBI. Gurobi Optimization Inc. http://www.gurobi.com/, online; accessed July, 2018.

[16] HAUGEN KK, LØKKETANGEN A & WOODRUFF DL. 2001. Progressive hedging as a meta-heuristic applied to stochastic lot-sizing. *European Journal of Operational Research*, **132**(1): 116–122, http://www.sciencedirect.com/science/article/pii/S0377221700001168.

[17] HUANG K & KÜÇÜKYAVUZ S. 2008. On stochastic lot-sizing problems with random lead times. *Operations Research Letters*, **36**(3): 303–308, http://www.sciencedirect.com/science/article/pii/S0167637707001460.

[18] JIANG R & GUAN Y. 2011. An $\mathcal{O}(N^2)$-time algorithm for the stochastic uncapacitated lot-sizing problem with random lead times. *Operations Research Letters*, **39**(1): 74–77, http://www.sciencedirect.com/science/article/pii/S0167637710001422.

[19] LEE CY, ÇETINKAYA S & WAGELMANS APM. 2001. A Dynamic Lot-Sizing Model with Demand Time Windows. *Management Science*, **47**(10): 1384–1395, http://www.jstor.org/stable/822493.

[20] MORAES LA & FARIA LF. 2016. A stochastic programming approach to liquified natural gas planning. *Pesquisa Operacional*, **36**: 151–165, http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0101-74382016000100151&nrm=iso.

[21] OLIVEIRA F & HAMACHER S. 2012. Stochastic Benders decomposition for the supply chain investment planning problem under demand uncertainty. *Pesquisa Operacional*, **32**: 663–678, http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0101-74382012000300010&nrm=iso.

[22] PINTO J, JOLY M & MORO L. 2000. Planning and scheduling models for refinery operations. *Computers and Chemical Engineering*, **24**(9): 2259–2276, http://www.sciencedirect.com/science/article/pii/S0098135400005718.

[23] RÖMISCH W & SCHULTZ R. 2001. Multistage Stochastic Integer Programs: An Introduction. In: GRÖTSCHEL M, KRUMKE SO & RAMBAU J (Eds.), *Online Optimization of Large Scale Systems*, Springer Berlin Heidelberg, pp. 581–600, http://dx.doi.org/10.1007/978-3-662-04331-8_29.

[24] TESTURI CE, ZIMBERG B & FERRARI G. 2012. Modelado estocástico múltiple etapa de adquisición de combustible para la generación de electricidad bajo demanda incierta. Tech. Rep. INCO RT 12-07, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, https://www.fing.edu.uy/inco/pedeciba/bibliote/reptec/TR1207.pdf.

[25] VAN DEN HEUVEL W & WAGELMANS AP. 2008. Four equivalent lot-sizing models. *Operations Research Letters*, **36**(4): 465–470, http://www.sciencedirect.com/science/article/pii/S0167637708000035.

[26] WAGELMANS A, HOESEL SV & KOLEN A. 1992. Economic Lot Sizing: An $\mathcal{O}(n \log n)$ Algorithm That Runs in Linear Time in the Wagner-Whitin Case. *Operations Research*, **40**: 145–156, http://www.jstor.org/stable/3840844.

[27] WAGNER HM & WHITIN TM. 1958. Dynamic Version of the Economic Lot Size Model. *Management Science*, **5**(1): 89–96, http://mansci.journal.informs.org/content/5/1/89.abstract.

[28] WOLSEY AL. 2006. Lot-sizing with production and delivery time windows. *Mathematical Programming*, **107**(3): 471–489, http://dx.doi.org/10.1007/s10107-005-0675-3.

[29] WOLSEY LA. 2002. Solving Multi-Item Lot-Sizing Problems with an MIP Solver Using Classification and Reformulation. *Management Science*, **48**(12): 1587–1602, http://www.jstor.org/stable/822525.

[30] XU J, QU H, WANG S & XU Q. 2017. A New Proactive Scheduling Methodology for Front-End Crude Oil and Refinery Operations under Uncertainty of Shipping Delay. *Industrial and Engineering Chemistry Research*, **56**(28): 8041–8053, https://doi.org/10.1021/acs.iecr.7b01496.

[31] YE Y, LIANG S & ZHU Y. 2017. A mixed-integer linear programming-based scheduling model for refined-oil shipping. *Computers and Chemical Engineering*, **99**: 106–116, http://www.sciencedirect.com/science/article/pii/S0098135417300315.

[32] ZIMBERG B, TESTURI CE & FERRARI G. 2019. Stochastic modeling of fuel procurement for electricity generation with contractual terms and logistics constraints. *Computers and Chemical Engineering*, **123**: 49–63, https://doi.org/10.1016/j.compchemeng.2018.12.021.

A.2 Valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement

Intl. Trans. in Op. Res. xx (2019) 1–21

Valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement

Carlos E. Testuri[a,*], Héctor Cancela[a] and Víctor M. Albornoz[b]

[a]*Depto. de Investigación Operativa, Instituto de Computación, Universidad de la República, J.Herrera y Reissig 565, 11.300 Montevideo, Uruguay*
[b]*Depto. de Industrias, Campus Santiago Vitacura, Universidad Técnica Federico Santa María, Avda. Santa María 6400, Vitacura, Santiago, Chile*
E-mail: ctesturi@fing.edu.uy; cancela@fing.edu.uy; victor.albornoz@usm.cl

Submitted 17 June 2019; under review process

Abstract

A stochastic capacitated discrete procurement problem with lead times, cancellation and postponement is addressed. The problem determines the expected cost minimization of meeting the uncertain demand of a product during a time planning horizon. The supply of the product is made through the purchase of optional distinguishable orders of fixed size with lead time. Due to the uncertainty of demand, corrective actions, such as order cancellation and postponement, may be taken with associated costs and time limits. The problem is modeled as an extension of a capacitated discrete lot-sizing problem with uncertain demand and lead times through a multi-stage stochastic mixed-integer programming approach. To improve the resolution of the model by tightening its formulation, valid inequalities are generated based on the conventional $(\ell, \mathcal{S})$ inequalities approach. Computational experiments are performed for several instances with different uncertainty information structure. The experimental results allow to conclude that the inclusion of a subset of the generated valid inequalities enable a more efficient resolution of the model.

Keywords: stochastic lot-sizing; multi-stage stochastic mixed-integer programming; valid inequalities, lead time

1. Introduction

This work deals with the resolution of a capacitated discrete procurement problem with lead times, cancellation and postponement. The problem is about the minimization of the expectation of the costs incurred in decisions taken to meet the uncertain demand of a product over a finite discrete time planning horizon. To meet the demand, there are optional distinguishable orders with an indivisible amount of the product. The orders can be acquired at most once with an associated cost and a delivery time. After

*Author to whom all correspondence should be addressed (e-mail: ctesturi@fing.edu.uy).

meeting the demand of the product in a given period, the remaining quantity is stored up to a certain capacity, to satisfy future demand in subsequent periods. The orders have significant delivery times within the planning horizon; so that a considerable amount of time elapses between the purchase decision and the moment when the product is received. As the time passes, the uncertainty of the demand is revealed. Then it can happen that, at a given time, a purchase order which has not yet been received is no longer necessary. In this case, it could be decided to cancel its acquisition or postpone its delivery; decisions that in turn, have minimum execution times in relation to the time of delivery and associated costs.

The problem can be modeled as an extension of the lot-sizing formulation of Wagner and Whitin (1958) and particularly of the variant with variable capacity and discrete dimensioning of Nemhauser and Wolsey (1988). For the case where the parameters are known with certainty (deterministic case), the dimensioning is continuous, and without capacity constraints or with constant capacity, the problem has efficient resolution through dynamic programming as shown by Wagner and Whitin (1958) and Wagelmans et al. (1992). In addition, there are known formulations which determine the convex hull of the feasible region: the extended facility location formulation of Krarup and Bilde (1977) and the $(\ell, \mathcal{S})$ valid inequalities formulation of Barany et al. (1984). Bitran and Yanasse (1982) established that the deterministic variant with discrete sizing is a generalization of the binary knapsack problem, and that it belongs to the $\mathcal{NP}$-hard complexity class.

In the case that the parameters are random variables (stochastic variant) the problem can be formulated by stochastic programming (Birge and Louveaux, 2011). Ahmed et al. (2003) established an adjusted extended formulation of the stochastic continuos non-capacitated problem and showed that the Wagner-Whitin conditions are not satisfied for the stochastic variant. Guan et al. (2006) showed that the $(\ell, \mathcal{S})$ inequalities are also valid for the stochastic continuous variant, and they extend the inequalities to a general class that allow to define facets of the feasible set.

Other variants of the deterministic continuous non-capacitated lot-sizing problem model delivery time of the lots (e.g. due to production time). Lee et al. (2001) present a variant in which demands have a compliance interval that has efficient resolution by dynamic programming. Brahimi et al. (2006) present two variants according to whether the lots are or are not distinguishable with respect to delivery times. These authors propose efficient algorithms based on dynamic programming for the distinguishable case and for the undistinguishable case when the order-delivery windows are not inclusive. For these variants, Wolsey (2006) sets tight extended formulations. For the stochastic case, Huang and Küçükyavuz (2008) establish that the problem with random lead times can be efficiently solved when delivery windows do not intersect in time and Jiang and Guan (2011) establish an quadratic polinomial time algorithm. Liu and Küçükyavuz (2018) propose valid inequalities for the static probabilistic lot-sizing problem. Hosseini and MirHassani (2017) generate valid inequalities for tightening a refueling station location model. A stronger formulation based on valid inequalities for simple assembly line balancing is proposed by Ritt and Costa (2018). Testuri et al. (2019) present a specific example of the proposed new problem, which details the complexity of its resolution and motivates the present work.

In this work the problem under study is modeled by a new extension of the stochastic capacitated discrete lot-sizing problem with corrective actions and lead times. The model is formulated through a multi-stage stochastic mixed-integer programming approach. The formulation is tightened by valid inequalities derived of the conventional $(\ell, \mathcal{S})$ inequalities approach. Computational experiments show that the derived inequalities are adequate to improve the resolution of the model.

The work is organized as follows. In Section 2 an algebraic model of the problem is presented. Valid inequalities for the model are presented in Section 3. In Section 4 experiments are established to determine utility of the valid inequalities formulation. Work is completed with Section 5, where conclusions and future work are discussed.

2. Stochastic model formulation

The model formulation is based on the mathematical programming approach. Basic entities represented by index sets are described as:

T	periods, $\{1, ..., H\}$ (ordered set),
A	already acquired orders,
F	possible (future) orders to be acquired,
O	orders, $A \cup F$,
N	nodes of the scenario tree.

The planning time is represented by the set T of discrete time periods, from initial period 1 up to horizon period H. The set O of orders is partitioned in two sets: the set A of already acquired orders –orders established in previous time execution of the model– that are pending reception, and the set F of possible (future) orders to be acquired from now on. Acquisition decisions are made on orders in set F, and cancellation or postponement decisions are taken on orders in set A.

The uncertain demand is represented by a discrete-time stochastic process indexed in the planning periods. The process is defined in a finite probability space. It is assumed that the demand of the first period is deterministic, and that the demands of the remaining periods are random with known distribution functions. The decisions made in a period can not anticipate the revelation of the uncertainty of the next period. These decisions must simultaneously take into account the entire distribution of uncertain demand in the following periods. This information structure can be represented by an arborescence structure called *tree of scenarios* (Römisch and Schultz, 2001). The structure is a directed rooted tree, with the root node in period $t = 1$ and with leaf nodes in period $t = H$ (identifying the scenarios).

Each node of the scenario tree describes the state of the process and is identified by a period and a scenario. An useful abbreviated notation is to identify the nodes by a single index n in a numerable set of nodes, N. For the first period, $t = 1$, there is a unique node, denoted by 1, that represents the root of the tree.

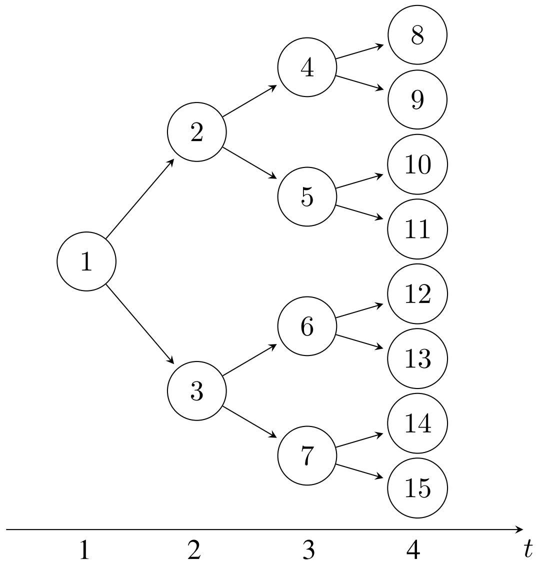
Set-valued functions on the nodes of the scenario tree are described as:

$t(n)$	period corresponding to node $n \in N$,
$p(n)$	immediate time predecessor node of node $n \in N$; the auxiliary node 0 is defined as the predecessor of the root node, such that $0 \notin N$,
$p(n, k)$	k-th time predecessor of node $n \in N$; defined as $p(n, k) := p(p(n, k-1))$ for $k = 2, ..., t(n) - 1$, such that $p(n, 1) := p(n)$,
$P(n)$	set of ordered nodes on the path from the root node to node $n \in N$. It is defined as sequence $\big(p(n, t(n) - 1), p(n, t(n) - 2), ..., p(n, 1), n\big)$,

$S(n)$ set of nodes successors of node $n \in N$; defined as $S(n) := \{n' \in N, k = 1, ..., H - t(n) | n = p(n', k)\}$,

L set of leaf nodes of the tree; defined as $L := \{n \in N | t(n) = H\}$.

A perfect binary scenario tree with four periods in conjunction with set-value functions is depicted as an example in Figure 1.



(a) Indexed scenario tree

$N = \{1, 2, ..., 15\}$
$L = \{8, 9, ..., 15\}$
$t(4) = 3$
$p(4) = 2$
$p(4, 2) = 1$
$P(4) = (1, 2, 4)$
$S(4) = \{8, 9\}$

(b) Sets and set-value functions on nodes

Fig. 1: Example of a scenario tree with four periods and set-value functions on nodes

The parameters of the model are described as:

d_n demand volume at node $n \in N$,
π_n probability of node $n \in N$,
s_0 initial inventory volume,
$\underline{s}, \overline{s}$ minimum and maximum storage capacities by period,
τ^i period in which already acquired order $i \in A$ is received,
q^i volume of order $i \in O$,
γ^i delivery time of order $i \in F$, such that $0 \leq \gamma^i \leq H - 1$,
δ^i cancellation minimum time of already acquired order $i \in A$, such that $0 \leq \delta^i \leq \tau^i - 1$,
ϵ^i postponement minimum time of already acquired order $i \in A$, such that $0 \leq \epsilon^i \leq H - \tau^i$,
ca^i acquisition unit cost of order $i \in O$,
cc^i cancellation unit cost of order $i \in O$,
cp^i postponement unit cost of order $i \in O$,
h_t storage unit cost in period $t \in T$,
a_t already acquired volume that is received in period $t \in T$.

In the stochastic setting the product demand at each node n is defined as d_n. The probability of the state on each node n is denoted as π_n, such that $\pi_n \geq 0$ and $\sum_{n\in N|t(n)=t} \pi_n = 1$, for each $t \in T$. The demand distribution for each period $t \in T$ is represented by (d_n, π_n) such that $n \in N$ and $t(n) = t$. Due to storage constraints, the inventory of the product at the end of each period is restricted between a minimum volume, $\underline{s}$, and a maximum volume, $\overline{s}$, and there is an initial inventory volume, s_0, at the beginning of the planning horizon.

The period at which an already acquired order i, τ^i, is received is fixed, and it is decided in previous acquisitions (i.e. previous model resolutions). Each order i has a given volume, q^i. Decisions on each order have a delay time of achievement of its results measured in periods. The delivery time of order i, γ^i, establishes the length of the wait time (measured in periods) between the acquisition decision and the actual arrival of the order. The minimum time for cancellation of order i, δ^i, establishes the minimum number of periods prior to the delivery period at which the order may be cancelled. The minimum postponement time of order i, ϵ^i, establishes the minimum number of periods after the delivery period in which the posponed order can be received. The achievement period of decisions on acquisition, cancellation and postponement must take place within the planning horizon.

For each order i there are unit costs per volume associated with the decisions to acquire, ca^i, cancel, cc^i, and postpone, cp^i. In addition, there is a unit cost associated with storage at each period t, h_t. The already acquired volume that is scheduled to be received at each period is determined by the sum of the volume of the orders that are received in that period,

$$a_t := \sum_{\{i\in A|\tau^i=t\}} q^i, \quad t \in T; \tag{1}$$

this is an auxiliary summary parameter.

In order to facilitate the formulation, derived subsets of the sets of nodes and periods that are indexed in the parameters are established as

N^i_γ	nodes where it is possible to acquire order $i \in F$, $\{n \in N|t(n) \leq H - \gamma^i\}$,
N^i_δ	nodes where it is possible to cancel and postpone order $i \in A$, $\{n \in N|t(n) \leq \tau^i - \delta^i\}$,
T^i_ϵ	periods to where it is possible to postpone order $i \in A$, $\{t \in T|t \geq \tau^i + \epsilon^i\}$.

These subsets abbreviate the denomination of nodes where, for each order i, it is possible to acquire it, N^i_γ, and where it is possible to cancel and postpone it, N^i_δ. In addition, subsets of periods to where it is possible to postpone each order i are established, T^i_ϵ. The subscripts of this subsets are part of their denomination.

In the stochastic model all decisions depend on the nodes of the tree according to the following definitions of the variables:

s_n	inventory volume at the end of the period of node $n \in N$
u_n	acquired volume incoming at node $n \in N$
v^i_n	if order $i \in F$ is acquired at node $n \in N^i_\gamma$ (binary)
w_n	cancelled volume outgoing from node $n \in N$
x^i_n	if an already acquired order $i \in A$ is cancelled in node $n \in N^i_\delta$ (binary)
y_n	postponed volume incoming at node $n \in N$
z^i_{nt}	if an already acquired order $i \in A$ is postponed in node $n \in N^i_\delta$ to period $t \in T^i_\epsilon$

(binary)

There are binary variables associated with decisions on order i taken at node n for acquisition, v_n^i, cancellation, x_n^i, and postponement z_{nt}^i towards period t. There are three continuous auxiliary variables that consolidate the volume, at each node n, by type of decision: s_n integrates the inventory volume at the end of the period of the node, u_n unifies the acquired volume incoming at the node, w_n unites the cancelled volumen outgoing of the node, and y_n combines the postponed volume incoming at the node.

The indexes of periods in deterministic parameters or variables are reduced to the temporary realization of a node n by $t(n)$. This is the case for parameters corresponding to the already acquired volume and storage unit cost.

Based on the previous definitions of index sets, parameters and variables a multi-stage stochastic mixed-integer programming formulation of the problem is

$$\text{(SCS)} : \min \sum_{n \in N} \pi_n \Big[\sum_{\{i \in F | n \in N_\gamma^i\}} ca^i q^i v_n^i \tag{2}$$

$$+ \sum_{\{i \in A | n \in N_\delta^i\}} (cc^i - ca^i) q^i x_n^i \tag{3}$$

$$+ \sum_{\{i \in A, t \in T_\epsilon^i | n \in N_\delta^i\}} (cp^i + ca^i - cc^i) q^i z_{nt}^i \tag{4}$$

$$+ \ h_{t(n)} s_n \Big], \tag{5}$$

s.t.

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n, \quad n \in N, \tag{6}$$

$$\underline{s} \leq s_n \leq \overline{s}, \quad n \in N, \tag{7}$$

$$u_n = \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} q^i v_{p(n,\gamma^i)}^i, \quad n \in N, \tag{8}$$

$$\sum_{n' \in P(n)} v_{n'}^i \leq 1, \quad i \in F, n \in N, t(n) = H - \gamma^i, \tag{9}$$

$$w_n = \sum_{\{i \in A | t(n) = \tau^i\}} \left(q^i \sum_{\{n' \in P(n) | t(n') \leq \tau^i - \delta^i\}} x_{n'}^i \right), \quad n \in N, \tag{10}$$

$$\sum_{n' \in P(n)} x_{n'}^i \leq 1, \quad i \in A, n \in N, t(n) = \tau^i - \delta^i, \tag{11}$$

$$x_n^i \geq z_{nt}^i, \quad i \in A, n \in N_\delta^i, t \in T_\epsilon^i, \tag{12}$$

$$y_n = \sum_{\{i \in A | t(n) \geq \tau^i + \epsilon^i\}} \left(q^i \sum_{\{n' \in P(n) \cap N_\delta^i\}} z_{n', t(n)}^i \right), \quad n \in N, \tag{13}$$

$$\sum_{\{n' \in P(n), t \in T^i_\epsilon\}} z^i_{n't} \leq 1, \quad i \in A, n \in N^i_\delta, \tag{14}$$

$$s_n, u_n, w_n, y_n \geq 0, \quad n \in N, \tag{15}$$

$$v^i_n \in \{0,1\}, \quad i \in F, n \in N^i_\gamma, \tag{16}$$

$$x^i_n, z^i_{nt} \in \{0,1\}, \quad i \in A, n \in N^i_\delta, t \in T^i_\epsilon. \tag{17}$$

This formulation takes into account the information structure of the scenario tree. It minimizes the expectation of acquisition costs (2), cancellation costs less acquisition costs in case of cancellation (3), postponement costs plus acquisition costs minus cancellation costs (4) –a postponement is modeled in conjunction with a cancellation–, and storage costs (5).

Constraints (6) set the product volumen flow conservation over time for each node, where the left and right expressions represent the incoming and outgoing flow, respectively. The lower and upper storage bounds at each node are determined by constraints (7). The amount of acquired product that is received at each node is determined by acquisitions of orders in the possible range of the corresponding acquisition periods according to (8). Constraints (9) state that each order is acquired at a single node at most in each path from the root node to a node whose period coincides with the receiving period minus the delivery time of the order. The product previously acquired that is cancelled at each node is determined by the cancellations of the nodes in the path from the root node to the node, whose cancellation periods are less than the delivery period less the cancellation time, according to (10). Constraints (11) state that each order to be cancelled is at a single node in each path from the root node to a node whose period coincides with the receiving period minus the cancellation time of the order. The postponement of the orders is modeled in conjunction with the cancellation, i.e. only cancelled orders can be postponed, (12). The already acquired volume that is postponed in a node is determined by the postponements of the orders in the nodes in the path from the root to the node for all periods superior to the period of reception plus the delay time of the node, according to (13). Constraints (14) state that each order to be postponed is at a single node in each path from the root node to a node in some period greater than the receiving period plus the time of postponement of the node. Constraints (15)–(17) state the domain of the variables. The set of feasible solutions of (SCS) is denoted by X_{SCS}.

3. Valid inequalities for the stochastic model

The problem belongs to the time complexity class $\mathcal{NP}$-hard, since it is an extension of the discrete lot-sizing problem. Therefore, there is no known polyhedral description of the convex hull of X_{SCS}. It is nevertheless interesting to derive valid inequalities which can be used to strengthen the formulation. In some cases adding these inequalities can directly improve the capability of solvers to find solutions for larger instances in shorter times. Even when this is not the case, they may be used within a more sophisticated solving strategy, such as branch and cut methods relying on constraint separation.

Valid inequalities for X_{SCS} are derived from the $(\ell, \mathcal{S})$ valid inequalities formulation for the deterministic uncapacitated lot-sizing problem of Barany et al. (1984) while considering the extension for the stochastic case of Guan et al. (2006). The derived inequalities establish bounds on decision variables for the nodes of possible paths in the scenario tree.

Two sets of valid inequalities are derived:

(i) from equations (10) and inequalities (11) it is satisfied that

$$w_n \leq \sum_{\{i \in A | t(n) = \tau^i\}} q^i, \quad n \in N.$$

From these inequalities and definition of a_t (cf. (1)) it holds that

$$w_n \leq a_{t(n)}, \quad n \in N. \tag{18}$$

Consider the material balance equation (6) of model (SCS), for all $n \in N$

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n,$$

given that $s_{p(n)}, a_{t(n)}, y_n \geq 0$ and inequalities (18) it follows that

$$u_n \leq d_n + s_n. \tag{19}$$

From inequalities (19) the following valid inequalities can be established

$$u_n \leq d_n \beta(n) + s_n, \quad \text{for all } n \in N, \tag{20}$$

where $\beta(n) := \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} v^i_{p(n,\gamma^i)}$, since for all $n \in N$, if $\beta(n) = 0$, then from (8) $u_n = \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} q^i v^i_{p(n,\gamma^i)} = 0$. Otherwise, if $\beta(n) \geq 1$, then (20) holds.

In general, from the sum of material balance equation (6) between $n \in N$ and $\ell \in S(n)$, the following condition holds

$$u_n \leq d_{n\ell} + s_\ell, \tag{21}$$

where $d_{n\ell} := \sum_{n' \in P(\ell) \setminus P(p(n))} d_{n'}$ is the accumulated demand in the nodes in the path from n to ℓ. From (21) the following valid inequalities can be established

$$u_n \leq d_{n\ell} \beta(n) + s_\ell, \quad \text{for all } n \in N, \ell \in S(n), \tag{22}$$

similar to the provision for (20).

The valid inequalities (25) are obtained by adding the inequalities (22) for each subset of the set of nodes of the path from the root node to the ℓ node.

(ii) Given the material balance equation (6) of model (SCS), for all $n \in N$

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n,$$

given that $s_{p(n)}, a_{t(n)}, y_n \geq 0$ it follows that

$$u_n \leq d_n + w_n + s_n. \tag{23}$$

From inequalities (23) and following similar provisions than case (i) the following valid inequalities can be established

$$u_n \leq d_{n\ell}\beta(n) + w_{n\ell} + s_\ell, \quad \text{for all } n \in N, \ell \in S(n). \tag{24}$$

The valid inequalities (26) are obtained by adding the inequalities (24) for each subset of the set of nodes of the path from the root node to the ℓ node.

Theorem 1. *Let $\ell \in N$ and $\mathcal{S} \subseteq P(\ell)$ then the SCS-$(\ell, \mathcal{S})$ inequalities*

$$\text{(i)} \quad \sum_{n \in \mathcal{S}} u_n \leq \sum_{n \in \mathcal{S}} d_{n\ell}\beta(n) + s_\ell, \tag{25}$$

$$\text{(ii)} \quad \sum_{n \in \mathcal{S}} u_n \leq \sum_{n \in \mathcal{S}} d_{n\ell}\beta(n) + \sum_{n \in \mathcal{S}} w_{n\ell} + s_\ell, \tag{26}$$

are valid for X_{SCS}.

Proof. The proof of (25) is based on the deterministic case presented by Barany et al. (1984). Given a point $(s, v, x, z) \in X_{\text{SCS}}$ there are two cases.

1) If $\beta(n) = \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} v^i_{p(n,\gamma^i)} = 0$ for all $n \in \mathcal{S}$, then $u_n = \sum_{\{i \in F | t(n) \geq \gamma^i + 1\}} q^i v^i_{p(n,\gamma^i)} = 0$ for all $n \in \mathcal{S}$ and $s_\ell \geq 0$, therefore the inequality holds.

2) Otherwise, there exists $n \in \mathcal{S}$ such that $\beta(n) = 1$. Let $n' = \text{argmin}\{t(n) | n \in N, \beta(n) = 1\}$. Then $\beta(n) = 0$ and $u_n = 0$ for all $n \in \mathcal{S} \cap P(p(n'))$. Thus $\sum_{n \in \mathcal{S}} u_n \leq \sum_{n \in P(\ell) \setminus P(p(n'))} u_n \leq d_{n'\ell} + s_\ell \leq \sum_{n \in \mathcal{S}} d_{n\ell}\beta(n) + s_\ell$.

The proof of (26) is similar considering that $\sum_{n \in \mathcal{S}} w_{n\ell} \geq 0$. □

Lemma 1. *The SCS-$(\ell, \mathcal{S})$ inequalities can be written alternatively as*

$$\text{(i)} \quad \sum_{n \in P(\ell) \setminus \mathcal{S}} u_n + \sum_{n \in \mathcal{S}} d_{n\ell}\beta(n) + \sum_{n \in P(\ell)} (y_n - w_n) \geq d_{1\ell} - \sum_{n \in P(\ell)} a_{t(n)} - s_0, \quad \textit{for all } \ell \in N, \mathcal{S} \subseteq P(\ell). \tag{27}$$

$$\text{(ii)} \quad \sum_{n \in P(\ell) \setminus \mathcal{S}} u_n + \sum_{n \in \mathcal{S}} d_{n\ell}\beta(n) + \sum_{n \in P(\ell)} (y_n - w_n) + \sum_{n \in \mathcal{S}} w_{n\ell} \geq d_{1\ell} - \sum_{n \in P(\ell)} a_{t(n)} - s_0, \quad \textit{for all } \ell \in N, \mathcal{S} \subseteq P(\ell). \tag{28}$$

Proof. The sum of equations (6), for all $n \in P(\ell)$ of a given $\ell \in N$, results in

$$s_0 + \sum_{n \in P(\ell)} a_{t(n)} + \sum_{n \in P(\ell)} y_n + \sum_{n \in P(\ell)} u_n = d_{1\ell} + w_{1\ell} + s_\ell,$$

from where it is possible to solve for s_ℓ and substitute it in (25) and (26), obtaining an alternative representation of the valid inequalities without inventory variables, denoted as SCS-$(\ell, \mathcal{S})$-i and SCS-$(\ell, \mathcal{S})$-ii, respectively. □

The formulation variants in which the inequalities (27) and (28) are added to (SCS) are called (SCS-$(\ell, \mathcal{S})$-i) and (SCS-$(\ell, \mathcal{S})$-ii), respectively.

4. Computational experiments

This section explores the computational impact of adding a subset of the SCS-$(\ell, \mathcal{S})$ inequalities introduced in the previous section to the original formulation. Specifically, the original formulation and its variants where the set $\mathcal{S}$ is fixed with the root node, are tested over a set of instances, checking the quality of the obtained solutions and the computational effort invested by the solver.

In order to generate a number of diverse test instances, six scenario tree structures were considered based on rooted perfect directed trees. Each tree structure, depicted in Table 1, is determined by the number of immediate time successors of each node (tree arity) and the number of periods of the planning horizon. Each tree structure with arity g and time horizon H contains g^{H-1} scenarios and $(g^H - 1)/(g - 1)$ nodes.

Table 1: Size of scenario tree structures

Arity	Horizon	Scenarios	Nodes
2	5	16	31
2	6	32	63
2	7	64	127
3	5	81	121
3	6	243	364
3	7	729	1,093

A distribution of orders by quantity is associated to each tree structure. Each distribution of orders by quantity ($|O|$) is identified by the sum of numbers of already acquired orders ($|A|$) and possible orders to be acquired ($|F|$), as shown in column labelled "Orders" in Table 2. The 3-uple ⟨*arity, horizon, distribution of orders*⟩ identifies table rows, denominated as categories of data instances. The table depicts, for each category, the numbers of constraints, $(\ell, \mathcal{S})$ inequalities, variables and binary variables.

Thirty data instances were generated for each of the six instance categories, totaling 180 instances. Each instance has an initial storage, $s_0 = 20$, and a lower and an upper bound storage, $\underline{s} = 0$ and $\overline{s} = 80$, respectively. For each order $i \in O$ there is an uniformly distributed volume, $q^i \sim U[10, 50]$, and there are costs evenly distributed according to the operations of acquisition, $ca^i \sim U[150, 250]$, cancellation, $cc^i \sim U[30, 50]$, and postponement, $cp^i \sim U[5, 12]$. Each already acquired order $i \in A$ has delivery period $\tau^i = 1$ or 2 with equal probability. Each order $i \in O$ has delivery time $\gamma^i = 1$, cancellation time $\delta^i = 1$ and delay time $\epsilon^i = 1$. The unit storage cost at each period t is $h_t = 1$. For each scenario, identified as leaf node $n \in L$, a probability of state π_n is established from a distribution $Beta(\alpha = 2, \beta = 2)$; the probability of the remaining nodes is obtained from the sum of the probabilities of

Table 2: Size of instance categories defined by scenario tree structure and order distribution

Arity	Horizon	Orders [$\|O\|$ ($\|A\| + \|F\|$)]	(SCS)-constraints	$(\ell, \mathcal{S})$-inequalities	Variables	(binary)
2	5	10 (2+ 8)	225	31	249	(124)
2	6	12 (3+ 9)	480	63	549	(296)
2	7	14 (3+11)	1,012	127	1,223	(714)
3	5	10 (3+ 7)	827	121	809	(324)
3	6	12 (4+ 8)	2,542	364	2,485	(1,028)
3	7	14 (4+10)	7,987	1,093	8,091	(3,718)

their corresponding immediate successor nodes. Finally, the demand for each node is evenly distributed, $d_n \sim U[10, 50]$.

The computational implementation was performed using AMPL (Fourer et al., 2002) for the algebraic coding of the stochastic model, and GUROBI 6.5 (Gurobi Optimization, LLC, 2018) for the resolution of the instances through its branch and cut solver. The calculations were carried out on an Intel Core i7 5960X 3.5 GHz computer with 20 MiB cache and 64 GiB RAM, operating with CentOS-7 Linux system.

For each instance, the original model and the variants were solved within a time limit of 900 s or without gap between the objective and its lower bound (MIP-gap = 0). The instances average results of the original model and the variants by instance category are presented in Table 3, Table 4 and Table 5, respectively for formulations (SCS), (SCS-$(\ell, \mathcal{S})$-i) and (SCS-$(\ell, \mathcal{S})$-ii). Detailed result of each instance by category and formulations (SCS), (SCS-$(\ell, \mathcal{S})$-i), and (SCS-$(\ell, \mathcal{S})$-ii) are given in Tables 6–11 at the Appendix.

Table 3: Average results of 30 instances of formulation (SCS) by instance category

g	H	$\|O\|$	Time-mean(s)	Time-median(s)	MIP-gap(%)	Nodes	Cuts	LP-gap(%)
2	5	10	1.60	0.87	0	15,261	159	10.31
2	6	12	27.61	19.53	0	111,228	473	11.25
2	7	14[a]	868.58	900.80	0.73	2,752,608	855	9.89
3	5	10	21.99	12.61	0	42,727	444	12.40
3	6	12[b]	821.46	900.39	2.86	868,850	1,797	19.69
3	7	14[c]	900.20	900.14	5.81	46,457	2,031	24.43
3	*7*	*14[cd]*	*900.20*	*900.14*	*4.47*	*33,177*	*1,520*	*21.32*

[a]28 of 30 instances reach the 900 s time limit.
[b]27 of 30 instances reach the 900 s time limit.
[c]All instances reach the 900 s time limit.
[d]Median value results.

These summary tables show, for each instance category average results of the 30 instances of the model (SCS) and its variants (SCS-$(\ell, \mathcal{S})$-i) and (SCS-$(\ell, \mathcal{S})$-ii). The metric results by instance category, depicted by columns labeled g, H and $|O|$, are solver mean and median elapsed time at columns "Time-mean" and "Time-median", solver mean relative mixed-integer programming gap for instances that reach the time limit of 900 s at column "MIP-gap(%)", mean number of nodes of solver branch and cut method

Table 4: Average results of 30 instances of formulation (SCS-$(\ell, \mathcal{S})$-i) by instance category

g	H	$\|O\|$	Time-mean(s)	Time-median(s)	MIP-gap(%)	Nodes	Cuts	LP-gap(%)
2	5	10	0.67	0.36	0	3,835	108	7.48
2	6	12	17.65	6.10	0	61,852	431	9.51
2	7	14[a]	542.96	727.28	0.48	1,311,339	1,218	7.01
3	5	10	7.94	3.17	0	18,532	318	9.74
3	6	12[b]	728.13	900.23	2.46	495,200	2,211	17.26
3	7	14[c]	900.32	900.16	7.15	55,149	2,048	26.08
3	*7*	*14*[cd]	*900.32*	*900.16*	*4.75*	*31,722*	*1,716*	*17.08*

[a]15 of 30 instances reach the 900 s time limit.
[b]22 of 30 instances reach the 900 s time limit.
[c]All instances reach the 900 s time limit.
[d]Median value results.

Table 5: Average results of 30 instances of formulation (SCS-$(\ell, \mathcal{S})$-ii) by instance category

g	H	$\|O\|$	Time-mean(s)	Time-median(s)	MIP-gap(%)	Nodes	Cuts	LP-gap(%)
2	5	10	1.13	0.88	0	11,346	159	9.51
2	6	12	18.19	12.78	0	56,467	483	11.00
2	7	14[a]	743.11	900.68	0.56	1,766,353	1,349	8.89
3	5	10	18.13	5.81	0	30,371	476	11.77
3	6	12[b]	819.11	900.32	2.46	593,253	2,387	19.47
3	7	14[c]	900.27	900.17	6.52	39,807	1,712	25.18
3	*7*	*14*[cd]	*900.27*	*900.17*	*4.80*	*30,776*	*1,260*	*21.24*

[a]23 of 30 instances reach the 900 s time limit.
[b]27 of 30 instances reach the 900 s time limit.
[c]All instances reach the 900 s time limit.

at column "Nodes", mean number of cuts added by solver's branch and cut method at column "Cuts", and mean relative gap of the best objective with respect to the corresponding one under linear programming relaxation at column "LP-gap(%)". The median elapsed time is shown as a summary statistic since most of the instance categories present outliers for the metric. Due that most of the metrics are bounded below, the median tends to be smaller than the mean. The types of cuts added by the solver are by descending frequency of occurrence: mixed integer rounding, flow cover, Gomory passes, implied bound, mod-k, zero-half, generalized upper bound cover, implied bound, and cover. There are almost no cases of cut types: strong Chvátal-Gomory, infinity proof, network, and lift and project. The distribution of cut type frequency of the original model and its variants are similar.

The results of formulation (SCS-$(\ell, \mathcal{S})$-i) are better than those of formulation (SCS) except for instance category (3,7,14) that are equally hard for both. Formulation (SCS-$(\ell, \mathcal{S})$-i) has smaller mean metric values than formulation (SCS) for the remaining instance categories. For all instance categories where some of its instances reach the 900 s time limit, the formulation (SCS-$(\ell, \mathcal{S})$-i) obtains a lower or equal number of these instances and a lower MIP-gap average than formulation (SCS). Specifically in the case of instance category (2,7,14), while formulation (SCS) is solved to optimality for 2 instances in the allotted time, formulation (SCS-$(\ell, \mathcal{S})$-i) is solved for 15 instances. Nevertheless formulation (SCS-$(\ell, \mathcal{S})$-ii) presents mixed results with respect to formulation (SCS) depending on instance categories, and overall worse results than formulation (SCS-$(\ell, \mathcal{S})$-i).

5. Conclusions

A stochastic multi-stage capacitated discrete lot-sizing model formulation of the provision with lead times of the uncertain demand of a product has been proposed. The decisions on product orders are modeled with delay time, aspect that for cancellation and postponement decisions is not covered in the previous literature of the problem. A discrete time stochastic process with finite probability, summarized in a scenario tree, is used to model the information structure of the uncertain demand. The model is formulated by stochastic mixed-integer programming with entities indexed by nodes of the scenario tree. The model incorporates decisions to cancel and postpone orders with delay time, which implied the revision of the definitions of the variables and the restrictions to take into account the structure of the scenario tree. To tighten the formulation, two types of valid inequalities were generated based in the $(\ell, \mathcal{S})$ inequalities approach.

Computational experiments where carried out for several instances of six tree structures and order distributions of different sizes. All of the computational experiments were solved at optimum for the structures of small and medium sized instance categories. The (SCS-$(\ell, \mathcal{S})$-i) formulation shows improved performance over the (SCS) original formulation for execution time, MIP-gap, number of branch and bound nodes, and number of solver cuts. The experimental results shows the interest of the (SCS-$(\ell, \mathcal{S})$-i) formulation, which can be used to solve larger instances in shorter computational time. As future work, the development of reduction and separation algorithms, the determination of dominated and violated valid inequalities, and the study of formulations that combine the derived $(\ell, \mathcal{S})$ valid inequalities for different node paths in the scenario tree are considered.

Acknowledgments

This work was partially supported by the Sectorial Commission of Scientific Research, CSIC (Universidad de la República, Uruguay), and the Uruguayan Basic Sciences Development Program, PEDECIBA.

References

Ahmed, S., King, A.J., Parija, G., 2003. A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization* 26, 1, 3–24.

Barany, I., Roy, T., Wolsey, L.A., 1984. Mathematical Programming at Oberwolfach II, Springer Berlin Heidelberg, Berlin, Heidelberg, chapter Uncapacitated lot-sizing: The convex hull of solutions. pp. 32–43.

Birge, J.R., Louveaux, F., 2011. *Introduction to Stochastic Programming (2nd. ed.)*. Springer Series in Operations Research and Financial Engineering. Springer, New York.

Bitran, G.R., Yanasse, H.H., 1982. Computational complexity of the capacitated lot size problem. *Management Science* 28, 10, 1174–1186.

Brahimi, N., Dauzere-Peres, S., Najid, N.M., Nordli, A., 2006. Single item lot sizing problems. *European Journal of Operational Research* 168, 1, 1–16.

Fourer, R., Gay, D.M., Kernighan, B.W., 2002. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press.

Guan, Y., Ahmed, S., Nemhauser, G.L., Miller, A.J., 2006. A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem. *Mathematical Programming* 105, 1, 55–84.

Gurobi Optimization, LLC, 2018. Gurobi optimizer reference manual.

Hosseini, M., MirHassani, S., 2017. A heuristic algorithm for optimal location of flow-refueling capacitated stations. *International Transactions in Operational Research* 24, 6, 1377–1403.

Huang, K., Küçükyavuz, S., 2008. On stochastic lot-sizing problems with random lead times. *Operations Research Letters* 36, 3, 303–308.

Jiang, R., Guan, Y., 2011. An $O(n^2)$-time algorithm for the stochastic uncapacitated lot-sizing problem with random lead times. *Operations Research Letters* 39, 1, 74 – 77.

Krarup, J., Bilde, O., 1977. Plant location, Set Covering and Economic Lot Size: An $O(mn)$-Algorithm for Structured Problems, Birkhäuser Basel, Basel. pp. 155–180.

Lee, C.Y., Çetinkaya, S., Wagelmans, A.P.M., 2001. A dynamic lot-sizing model with demand time windows. *Management Science* 47, 10, 1384–1395.

Liu, X., Küçükyavuz, S., 2018. A polyhedral study of the static probabilistic lot-sizing problem. *Annals of Operations Research* 261, 1, 233–254.

Nemhauser, G., Wolsey, L., 1988. *Integer and Combinatorial Optimization*. John Wiley & Sons, Inc.

Ritt, M., Costa, A., 2018. Improved integer programming models for simple assembly line balancing and related problems. *International Transactions in Operational Research* 25, 4, 1345–1359.

Römisch, W., Schultz, R., 2001. Multistage stochastic integer programs: An introduction. In Grötschel, M., Krumke, S.O. and Rambau, J. (eds), *Online Optimization of Large Scale Systems*. Springer Berlin Heidelberg, pp. 581–600.

Testuri, C.E., Cancela, H., Albornoz, V.M., 2019. Stochastic discrete lot-sizing with lead times for fuel supply optimization. *Pesquisa Operacional* 39, 37 – 55.

Wagelmans, A., Hoesel, S.V., Kolen, A., 1992. Economic lot sizing: An $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research* 40, S145–S156.

Wagner, H.M., Whitin, T.M., 1958. Dynamic version of the economic lot size model. *Management Science* 5, 1, 89–96.

Wolsey, A.L., 2006. Lot-sizing with production and delivery time windows. *Mathematical Programming* 107, 3, 471–489.

Appendix

Table 6: Results of 30 instances for $(g, H, |O|) =$ **(2,5,10)** instance category by formulations

	(SCS)						(SCS-$(\ell, \mathcal{S})$-i)						(SCS-$(\ell, \mathcal{S})$-ii)					
Instance	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap
1	16117	2.81	0	35139	272	0.0721	16117	1.58	0	15684	225	0.0721	16117	3.07	0	45955	278	0.0721
2	10254	0.07	0	88	22	0.2155	10254	0.08	0	0	0	0.1966	10254	0.15	0	96	32	0.2155
3	11699	0.35	0	7045	117	0.1235	11699	0.21	0	0	14	0.0907	11699	0.36	0	2072	146	0.1125
4	12755	0.71	0	2924	33	0.1856	12755	0.18	0	0	24	0.1005	12755	0.15	0	721	86	0.1705
5	21232	3.19	0	32636	338	0.0927	21232	1.17	0	5132	114	0.0436	21232	1.16	0	11395	213	0.0781
6	12877	0.51	0	2996	106	0.1263	12877	0.38	0	3065	118	0.1263	12877	0.23	0	3696	121	0.1263
7	21445	1.72	0	8483	155	0.0945	21445	1.28	0	1764	73	0.0728	21445	2.2	0	13428	276	0.0945
8	19249	11.31	0	44771	221	0.0512	19249	2.29	0	17356	239	0.0456	19249	4.99	0	68732	395	0.0490
9	14811	0.86	0	7450	186	0.1610	14811	0.31	0	3181	197	0.0728	14811	0.87	0	2282	90	0.1338
10	15772	1.18	0	5977	179	0.0867	15772	1.20	0	5757	167	0.0867	15772	1.02	0	9687	203	0.0867
11	15407	0.88	0	4474	142	0.0784	15407	0.90	0	9261	108	0.0784	15407	1.25	0	3608	112	0.0784
12	19838	2.79	0	56968	231	0.1219	19838	0.72	0	1538	41	0.1053	19838	1.3	0	9197	186	0.1201
13	8987	0.76	0	7716	149	0.0854	8987	1.26	0	7132	172	0.0854	8987	1.06	0	5626	161	0.0854
14	16745	0.11	0	0	30	0.0464	16745	0.30	0	3580	169	0.0464	16745	0.2	0	1921	102	0.0464
15	18621	1.49	0	18691	240	0.1122	18621	0.29	0	2499	91	0.0736	18621	0.69	0	3184	120	0.0978
16	16488	0.94	0	3830	172	0.0800	16488	0.94	0	2067	67	0.0453	16488	0.89	0	3188	150	0.0713
17	20002	0.62	0	6043	127	0.0654	20002	0.30	0	1613	122	0.0654	20002	0.86	0	3664	111	0.0654
18	19150	3.54	0	66603	307	0.1002	19150	1.23	0	12041	230	0.0891	19150	3.16	0	48701	303	0.0945
19	19036	0.50	0	4223	76	0.1389	19036	0.19	0	741	90	0.0766	19036	0.24	0	3166	118	0.1198
20	15028	0.14	0	357	69	0.0751	15028	0.21	0	0	9	0.0454	15028	0.47	0	7801	152	0.0653
21	17702	1.46	0	9990	309	0.1674	17702	1.80	0	1910	40	0.0695	17702	1.06	0	4337	167	0.1357
22	21816	4.08	0	57284	361	0.0820	21816	0.27	0	1150	137	0.0512	21816	2.73	0	41866	263	0.0729
23	17552	0.14	0	1663	101	0.1028	17552	0.19	0	52	35	0.0595	17552	0.17	0	2630	115	0.0891
24	16526	3.37	0	40264	229	0.0637	16526	0.40	0	4856	168	0.0637	16526	1.19	0	14991	173	0.0637
25	11129	0.10	0	45	21	0.1871	11129	0.12	0	0	2	0.0844	11129	0.11	0	0	31	0.1555
26	21238	1.98	0	20763	256	0.1398	21238	0.33	0	3892	99	0.0984	21238	1.41	0	9390	189	0.1278
27	18169	0.56	0	2993	134	0.0358	18169	0.45	0	3408	208	0.0358	18169	0.86	0	4544	124	0.0358
28	16351	0.68	0	3069	102	0.0550	16351	1.09	0	6096	132	0.0550	16351	1.06	0	6829	162	0.0550
29	13472	0.15	0	1878	67	0.0537	13472	0.21	0	55	22	0.0443	13472	0.14	0	162	34	0.0495
30	15545	1.01	0	3454	22	0.0936	15545	0.27	0	1226	114	0.0628	15545	0.82	0	7520	149	0.0845
Total	495013	48.01	0	457817	3003	3.0939	495013	20.15	0	115056	3227	2.2433	495013	33.87	0	340389	4762	2.8531
Median		0.87	0	6010	146	0.0932		0.36	0	2283	111	0.0725		0.88	0	4441	150	0.0861
Mean		1.60	0	15261	159	0.1031		0.67	0	3835	108	0.0748		1.13	0	11346	159	0.0951
Maximum		11.31	0	66603	361	0.2155		2.29	0	17356	239	0.1966		4.99	0	68732	395	0.2155

Table 7: Results of 30 instances for $(g, H, |O|) =$ **(2,6,12)** instance category by formulations

	(SCS)						(SCS-$(\ell, \mathcal{S})$-i)						(SCS-$(\ell, \mathcal{S})$-ii)					
Instance	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap
1	8091	6.78	0	48106	428	0.2056	8091	2.76	0	6687	312	0.2056	8091	9.96	0	57228	452	0.2056
2	11760	14.81	0	76127	498	0.1403	11760	9.34	0	66511	491	0.1352	11760	12.67	0	64247	527	0.1402
3	17014	31.33	0	45509	249	0.0792	17014	1.28	0	5453	199	0.0698	17014	12.20	0	65005	624	0.0774
4	10895	1.36	0	5375	162	0.2989	10895	1.35	0	3298	157	0.2407	10895	0.65	0	8930	183	0.2909
5	14198	2.52	0	7604	219	0.1800	14198	5.01	0	22548	352	0.1741	14198	4.40	0	11050	292	0.1800
6	15234	1.82	0	12849	362	0.1279	15234	0.38	0	3103	142	0.0929	15234	1.37	0	8638	239	0.1234
7	11462	14.20	0	46751	383	0.1810	11462	2.51	0	12780	246	0.1317	11462	4.93	0	11687	308	0.1810
8	14742	40.02	0	73061	648	0.1746	14742	outlier	outlier	outlier	outlier	0.1746	14742	34.76	0	84926	786	0.1746
9	12075	25.73	0	46813	338	0.1899	12075	1.84	0	2179	144	0.0884	12075	12.88	0	67719	612	0.1657
10	22733	42.36	0	85613	724	0.0435	22733	6.10	0	21336	312	0.0362	22733	38.57	0	59626	674	0.0407
11	11928	62.02	0	403777	840	0.2169	11928	35.90	0	88001	716	0.2169	11928	49.40	0	177655	1059	0.2169
12	8741	2.35	0	2178	55	0.3968	8741	0.90	0	6321	117	0.3963	8741	1.21	0	5387	182	0.3968
13	16204	27.71	0	94167	554	0.1536	16204	36.97	0	71112	688	0.1536	16204	33.63	0	66162	350	0.1536
14	11043	5.26	0	11927	423	0.1316	11043	16.87	0	60143	702	0.1316	11043	7.10	0	2820	188	0.1316
15	16868	17.96	0	117128	573	0.1501	16868	2.54	0	11226	210	0.0836	16868	5.43	0	21724	435	0.1400
16	18497	1.46	0	5757	226	0.0993	18497	1.95	0	6557	232	0.0867	18497	1.23	0	2469	136	0.0964
17	20697	94.37	0	646716	831	0.0810	20697	7.40	0	27499	380	0.0578	20697	44.07	0	79107	1051	0.0766
18	14827	20.35	0	43185	86	0.0733	14827	7.02	0	7479	222	0.0733	14827	16.07	0	42166	67	0.0733
19	11139	54.12	0	104203	754	0.0953	11139	61.09	0	135826	971	0.0953	11139	46.17	0	139957	1000	0.0953
20	151	18.71	0	106300	726	-1.1411	151	13.19	0	59565	517	-1.1411	151	8.33	0	39518	350	-1.1411
21	9049	40.67	0	94220	839	0.3309	9049	33.96	0	117377	830	0.3109	9049	32.66	0	71814	672	0.3309
22	15268	15.06	0	79360	564	0.1679	15268	0.71	0	5525	204	0.0968	15268	2.84	0	4650	252	0.1578
23	14065	23.34	0	58417	603	0.0883	14065	23.36	0	57485	515	0.0883	14065	24.09	0	47508	433	0.0883
24	5902	26.46	0	48555	411	0.1634	5902	23.70	0	51274	469	0.1634	5902	22.19	0	94833	653	0.1634
25	13951	35.42	0	46810	221	0.1757	13951	26.30	0	126225	789	0.1749	13951	19.72	0	77823	645	0.1757
26	11692	15.80	0	44239	328	0.1147	11692	5.38	0	29990	361	0.1074	11692	18.94	0	44339	223	0.1147
27	13254	62.68	0	232988	753	0.0632	13254	43.76	0	166833	912	0.0632	13254	23.83	0	63176	667	0.0632
28	10012	6.86	0	36830	419	0.0999	10012	4.28	0	2978	177	0.0526	10012	2.44	0	4291	252	0.0940
29	6741	1.74	0	4172	231	0.1963	6741	2.67	0	3272	179	0.1963	6741	1.40	0	5215	240	0.1963
30	16334	115.01	0	708098	737	0.0957	16334	133.43	0	607306	959	0.0957	16334	52.59	0	174352	947	0.0957
Total	384567	828.28	0	3336835	14185	3.3737	384567	511.95	0	1785889	12505	2.8528	384567	545.73	0	1604022	14499	3.2989
Median		19.53	0	48331	426	0.1452		6.10	0	22548	352	0.1021		12.78	0	52368	434	0.1401
Mean		27.61	0	111228	473	0.1125		17.65	0	61582	431	0.0951		18.19	0	53467	483	0.1100
Maximum		115.01	0	708098	840	0.3968		133.43	0	607306	971	0.3963		52.59	0	177655	1059	0.3968

Table 8: Results of 30 instances of $(g, H, |O|) =$ **(2,7,14)** instance category by formulations

	(SCS)						(SCS-$(\ell, \mathcal{S})$-i)						(SCS-$(\ell, \mathcal{S})$-ii)					
Instance	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap
1	15366	900.96	0.0020	3137444	918	0.1025	15366	270.83	0	786948	1340	0.1025	15366	900.41	0.0008	3954405	1547	0.1025
2	20402	900.37	0.0009	3424678	739	0.1357	20402	17.21	0	58670	517	0.0544	20402	111.63	0.0000	142243	1187	0.1078
3	19329	900.74	0.0038	2167463	797	0.0641	19329	386.43	0	1241489	1245	0.0554	19329	756.30	0.0000	1143643	1247	0.0615
4	18164	900.68	0.0062	1725779	764	0.0862	18163	900.44	0.0037	1231979	1161	0.0660	18165	900.87	0.0072	1833474	1251	0.0863
5	18149	901.02	0.0141	2071166	1030	0.0717	18149	900.79	0.0142	1746174	1613	0.0717	18156	900.66	0.0170	1388290	1704	0.0722
6	24021	901.85	0.0045	3972141	960	0.0647	24021	232.20	0.0000	613161	1246	0.0261	24021	900.69	0.0039	2172518	1515	0.0502
7	19186	900.33	0.0012	3128238	828	0.0975	19186	485.24	0	2421332	1194	0.0784	19186	166.61	0.0000	226715	1386	0.0946
8	25987	901.49	0.0079	3345386	738	0.0566	25987	900.33	0.0031	2283690	1295	0.0532	25987	900.45	0.0058	1729862	1173	0.0548
9	16849	902.13	0.0129	3801549	789	0.0585	16849	900.90	0.0096	2216006	1511	0.0583	16848	900.86	0.0135	1868487	1261	0.0584
10	24740	900.42	0.0032	1267931	848	0.0884	24740	265.40	0.0000	298798	1258	0.0332	24740	900.50	0.0016	1609297	1356	0.0689
11	26590	900.69	0.0085	1599586	947	0.0472	26590	901.13	0.0050	2237633	1679	0.0472	26590	900.38	0.0070	953227	1573	0.0472
12	22882	901.74	0.0037	4572707	773	0.0784	22882	93.99	0	119365	859	0.0712	22882	901.20	0.0053	2630617	1363	0.0766
13	11686	903.51	0.0217	5514996	885	0.1425	11686	902.08	0.0221	3682411	1516	0.1425	11636	902.25	0.0140	4092200	1357	0.1377
14	16044	902.13	0.0092	4058014	808	0.1223	16044	900.86	0.0072	2056671	1409	0.1218	16044	901.67	0.0074	3349432	1381	0.1223
15	14858	900.75	0.0037	3106888	888	0.0973	14858	900.55	0.0011	3405142	1305	0.0973	14858	900.77	0.0017	3867310	1205	0.0973
16	22878	901.53	0.0099	3576851	937	0.0451	22851	900.44	0.0013	2990304	1644	0.0379	22851	900.80	0.0025	2432331	1567	0.0399
17	22188	900.74	0.0034	1911426	765	0.0693	22188	118.37	0	162626	1200	0.0503	22188	900.73	0.0028	2350524	1248	0.0639
18	25223	900.92	0.0080	1885349	930	0.1018	25207	554.23	0.0000	413536	1241	0.0278	25207	900.51	0.0041	1406100	1558	0.0742
19	22962	900.48	0.0006	2431644	800	0.0798	22962	88.45	0	249936	988	0.0601	22962	214.58	0.0000	234329	1268	0.0718
20	24067	900.57	0.0014	1899704	925	0.1579	24067	22.97	0	29964	567	0.0328	24067	118.45	0.0000	91065	1107	0.1091
21	19425	900.62	0.0145	1605238	830	0.0482	19425	900.74	0.0161	1496294	1381	0.0482	19425	900.59	0.0120	1224220	1304	0.0482
22	19161	901.61	0.0042	3500900	822	0.1374	19161	105.32	0	88711	1159	0.0525	19161	900.30	0.0020	1310901	1291	0.1103
23	21624	900.54	0.0013	3568223	888	0.1100	21624	106.41	0	159532	1231	0.0574	21624	900.69	0.0018	2268061	1260	0.0943
24	16377	900.84	0.0270	1684915	884	0.2278	16365	900.96	0.0248	1721591	1312	0.2269	16365	900.79	0.0193	1773505	1318	0.2269
25	27192	900.58	0.0192	1281687	856	0.0532	27061	900.97	0.0123	1693686	1456	0.0475	27046	900.81	0.0131	1453014	1396	0.0474
26	22044	901.75	0.0032	3185615	865	0.1037	22042	900.70	0.0036	1661186	1291	0.0833	22042	901.34	0.0034	2644434	1318	0.1036
27	20959	901.76	0.0079	3567392	877	0.0799	20972	901.31	0.0075	2977777	1331	0.0805	20972	901.37	0.0086	2601981	1315	0.0805
28	12557	199.46	0	1048364	791	0.1749	12557	9.47	0	10995	369	0.0980	12557	120.83	0.0000	256113	1290	0.1495
29	18298	625.87	0	2258918	777	0.2042	18298	19.58	0	23910	571	0.0618	18298	85.07	0.0000	61383	1103	0.1508
30	21024	901.25	0.0161	2278062	1000	0.0595	21013	900.54	0.0136	1260639	1664	0.0589	21013	901.06	0.0144	1920907	1613	0.0590
Total	610232	26057.33	0.2202	82578254	25659	2.9663	610045	16288.84	0.14520	39340156	36553	2.1033	609988	22293.17	0.1692	52990588	40462	2.6676
Median		900.80	0.0044	2769266	852	0.0873		727.28	0.00055	1251064	1275	0.0586		900.68	0.0037	1751684	1317	0.0786
Mean		868.58	0.0073	2752608	855	0.0989		542.96	0.00484	1311339	1218	0.0701		743.11	0.0056	1766353	1349	0.0889
Maximum		903.51	0.0270	5514996	1030	0.2278		902.08	0.02480	3682411	1679	0.2269		902.25	0.0193	4092200	1704	0.2269

Table 9: Results of 30 instances of $(g, H, |O|) =$ **(3,5,10)** instance category by formulations

	(SCS)						(SCS-$(\ell, \mathcal{S})$-i)						(SCS-$(\ell, \mathcal{S})$-ii)					
Instance	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap
1	17109	42.26	0	46270	631	0.1318	17109	3.32	0	2001	233	0.0652	17109	25.77	0	92465	872	0.1191
2	8186	1.18	0	3809	162	0.1789	8186	0.44	0	2788	213	0.1789	8186	1.06	0	2753	153	0.1789
3	13284	20.69	0	43023	117	0.1090	13284	20.00	0	43642	122	0.1090	13284	20.07	0	44858	115	0.1090
4	16719	17.04	0	73701	708	0.1420	16719	0.54	0	1345	274	0.1121	16719	5.61	0	8755	423	0.1343
5	9743	3.08	0	2957	181	0.1116	9743	1.00	0	11546	251	0.1116	9743	1.65	0	2265	121	0.1116
6	21919	73.12	0	121561	1137	0.0433	21919	15.60	0	59803	547	0.0433	21919	83.57	0	80619	1093	0.0433
7	14515	29.55	0	43728	133	0.1662	14515	4.52	0	1658	132	0.1223	14515	18.88	0	69250	793	0.1548
8	12154	2.28	0	2178	63	0.1946	12154	1.66	0	2856	157	0.1860	12154	1.27	0	1808	127	0.1895
9	23147	98.88	0	184585	1193	0.0792	23147	4.71	0	4883	351	0.0344	23147	57.51	0	62864	1127	0.0695
10	14146	3.30	0	6885	234	0.1800	14146	0.59	0	1774	216	0.1096	14146	2.20	0	4263	246	0.1583
11	10491	2.88	0	3265	260	0.1342	10491	2.90	0	2874	278	0.1342	10491	8.32	0	29868	516	0.1342
12	14169	13.39	0	59468	563	0.1109	14169	2.13	0	2317	211	0.0867	14169	4.44	0	6485	320	0.1045
13	14129	2.08	0	4638	304	0.1205	14129	0.65	0	1396	264	0.0687	14129	2.49	0	2717	244	0.1089
14	9842	7.45	0	37499	489	0.2742	9842	0.88	0	6755	215	0.1039	9842	2.42	0	2696	283	0.2332
15	17830	43.24	0	59007	839	0.0959	17830	28.52	0	72609	755	0.0959	17830	56.55	0	68648	1020	0.0959
16	12560	3.39	0	2705	196	0.1088	12560	0.61	0	2808	215	0.0897	12560	2.24	0	3059	263	0.1065
17	15102	1.87	0	2218	139	0.1990	15102	0.26	0	78	102	0.1260	15102	0.50	0	3072	209	0.1822
18	18978	42.17	0	46003	398	0.1030	18978	3.59	0	3067	245	0.0808	18978	4.71	0	11232	446	0.0947
19	10735	35.69	0	136321	918	0.0589	10735	5.02	0	17319	456	0.0551	10735	25.37	0	61793	826	0.0584
20	13696	3.09	0	7391	354	0.0917	13696	2.17	0	1836	135	0.0885	13696	3.57	0	3064	322	0.0911
21	10879	1.76	0	3097	201	0.1804	10879	4.88	0	1906	92	0.1804	10879	3.13	0	2900	177	0.1804
22	14134	4.29	0	1876	128	0.1021	14134	0.67	0	3837	274	0.1021	14134	1.17	0	7230	363	0.1021
23	15817	28.33	0	44665	245	0.2054	15817	3.43	0	3161	71	0.0912	15817	10.60	0	59610	582	0.1865
24	13947	17.90	0	42546	130	0.0858	13947	5.89	0	26038	537	0.0858	13947	33.40	0	41625	19	0.0858
25	20582	51.91	0	48613	647	0.0438	20582	22.22	0	88504	781	0.0438	20582	54.37	0	53823	1056	0.0438
26	19503	11.82	0	54850	747	0.1115	19503	0.60	0	840	216	0.0700	19503	6.00	0	25448	480	0.1015
27	17465	11.46	0	56864	641	0.0670	17465	3.16	0	6429	303	0.0577	17465	9.27	0	49100	607	0.0624
28	12699	26.83	0	44846	292	0.0765	12699	27.00	0	47059	318	0.0765	12699	30.89	0	42734	287	0.0765
29	15750	55.37	0	95374	1134	0.0639	15750	68.00	0	132477	1307	0.0639	15750	62.57	0	64343	1143	0.0639
30	11591	3.53	0	1868	138	0.1502	11591	3.17	0	2353	260	0.1502	11591	4.40	0	1789	37	0.1502
Total	440821	659.83	0	1281811	13322	3.7204	440821	238.13	0	555959	9531	2.9233	440821	544.00	0	911136	14270	3.5309
Median		12.61	0	43376	298	0.1112		3.17	0	2971	248	0.0905		5.81	0	18340	343	0.1077
Mean		21.99	0	42727	444	0.1240		7.94	0	18532	318	0.0974		18.13	0	30371	476	0.1177
Maximum		98.88	0	184585	1193	0.2742		68.00	0	132477	1307	0.1860		83.57	0	92465	1143	0.2332

Table 10: Results of 30 instances for $(g, H, |O|) =$ **(3,6,12)** instance category by formulations

	(SCS)						(SCS-$(\ell, \mathcal{S})$-i)						(SCS-$(\ell, \mathcal{S})$-ii)					
Instance	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap
1	6327	901.92	0.0177	3730778	1389	0.3979	6327	162.90	0	48584	838	0.3745	6327	900.75	0.0123	2309079	2192	0.3979
2	15960	901.43	0.0079	2348271	1888	0.1169	15960	413.26	0	72033	1619	0.0921	15960	900.48	0.0087	759479	1754	0.1163
3	12371	900.29	0.0234	513844	1911	0.2785	12371	900.68	0.0072	1525181	1888	0.2213	12371	900.36	0.0156	596614	1991	0.2785
4	5598	38.60	0	27606	555	0.2440	5598	19.66	0	1880	38	0.1798	5598	72.95	0	10368	674	0.2360
5	7739	900.14	0.0324	250541	2015	0.2006	7740	900.27	0.0368	451567	2622	0.2007	7739	900.29	0.0301	430574	2802	0.2006
6	7557	900.50	0.0492	831253	1881	0.1993	7557	900.27	0.0523	510301	2915	0.1993	7557	900.40	0.0439	684202	3037	0.1993
7	14247	900.20	0.0204	328170	2170	0.0969	14233	900.18	0.0175	277018	2662	0.0927	14228	900.25	0.016	459347	2859	0.0950
8	10178	900.78	0.0218	1371786	1794	0.1395	10176	900.67	0.0286	1082345	2752	0.1393	10176	900.59	0.027	946497	2664	0.1393
9	15810	900.25	0.0182	454755	2236	0.1211	15810	900.19	0.0194	314766	2799	0.1211	15810	900.15	0.0173	197750	2093	0.1211
10	6346	200.25	0	148102	1345	0.2514	6346	20.63	0	13155	778	0.1917	6346	156.61	0	124587	1095	0.2504
11	9207	900.19	0.0263	341574	2192	0.1852	9207	900.19	0.025	245126	2681	0.1852	9207	900.31	0.0204	485952	2621	0.1852
12	7941	900.43	0.0356	756663	1884	0.1293	7924	900.28	0.0327	260053	2797	0.1268	7924	900.21	0.019	235500	2560	0.1268
13	9062	901.11	0.0249	2008184	1589	0.2146	9062	900.61	0.0209	1253083	2486	0.2146	9062	900.37	0.0218	679235	2449	0.2146
14	-1190	90.08	0	140692	1150	-0.4374	-1190	72.04	0	16685	806	-0.4374	-1190	33.85	0	7541	645	-0.4374
15	11580	900.67	0.0160	1346720	1989	0.2216	11580	349.23	0	478517	1667	0.1752	11580	900.71	0.0159	1425838	2497	0.2164
16	10077	900.90	0.0241	1689468	1734	0.2571	10070	900.89	0.0155	1927095	2278	0.2096	10070	900.29	0.0248	526566	2698	0.2562
17	-1905	900.35	0.0849	650281	1565	-0.2538	-1905	900.50	0.0895	985314	1766	-0.2538	-1911	900.31	0.0762	553401	1758	-0.2515
18	13769	900.28	0.0191	452851	2175	0.1786	13769	900.22	0.0172	322059	2587	0.1585	13769	900.23	0.0112	415921	2495	0.1751
19	8428	900.48	0.0476	737467	1867	0.3472	8402	900.71	0.0425	1032391	2458	0.3430	8409	900.41	0.0367	669298	2501	0.3441
20	16790	900.73	0.0184	1242881	1725	0.1591	16789	900.30	0.0105	546247	2652	0.1051	16789	900.54	0.017	789884	2916	0.1520
21	14480	900.37	0.0206	556920	2080	0.2508	14459	608.47	0	137552	2222	0.0985	14460	900.20	0.0239	257901	2820	0.2325
22	7812	900.29	0.0284	476752	1932	0.3667	7802	900.24	0.0181	417212	3096	0.3260	7812	900.35	0.0272	550710	3151	0.3607
23	10093	900.63	0.0351	929580	1836	0.2553	10094	900.41	0.0376	604982	2793	0.2555	10089	900.33	0.0225	538581	3035	0.2549
24	12622	900.40	0.0383	631726	1946	0.2275	12570	900.38	0.0361	572386	2256	0.2224	12600	900.29	0.0383	477190	2043	0.2253
25	6730	900.50	0.0624	792165	1647	0.4053	6730	900.37	0.0612	576174	2641	0.4053	6732	900.21	0.0613	285404	2888	0.4057
26	10682	900.27	0.0741	399237	2398	0.2031	10742	900.15	0.0762	187211	2504	0.2098	10632	900.34	0.0608	529057	2927	0.1974
27	20108	900.48	0.0167	780356	1499	0.0900	20100	900.26	0.0137	340903	2760	0.0767	20102	900.40	0.0162	694601	2584	0.0875
28	10064	900.37	0.0057	677989	1555	0.2914	10064	900.11	0.005	247528	2581	0.2188	10064	900.21	0.0066	251075	2374	0.2914
29	4262	900.32	0.0743	550181	1876	0.5715	4232	900.19	0.0741	300996	2211	0.5605	4261	900.69	0.0603	1078921	2418	0.5712
30	8249	900.49	0.0141	898703	2073	0.1986	8249	389.51	0	107669	2175	0.1651	8249	900.35	0.0078	826508	3061	0.1972
Total	290994	24643.70	0.8576	26065496	53896	5.9079	290868	21843.77	0.7376	14856013	66328	5.1782	290822	24573.43	0.7388	17797581	71602	5.8402
Median		900.39	0.0226	664135	1878.5	0.2088		900.23	0	331481	2495	0.1885		900.32	0.0197	533819	2530.5	0.2076
Mean		821.46	0.0286	868850	1797	0.1969		728.13	0.0246	495200	2211	0.1726		819.11	0.0246	593253	2387	0.1947
Maximum		901.92	0.0849	3730778	2398	0.5715		900.89	0.0895	1927095	3096	0.5605		900.75	0.0762	2309079	3151	0.5712

Table 11: Results of 30 instances for $(g, H, |O|) =$ **(3,7,14)** instance category by formulations

	(SCS)						(SCS-$(\ell, \mathcal{S})$-i)						(SCS-$(\ell, \mathcal{S})$-ii)					
Instance	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap	Objective	Time (s)	MIPGap	Nodes	Cuts	LPGap
1	11499	900.37	0.1039	29967	1000	0.1760	11459	900.32	0.0988	33894	965	0.1719	11812	900.26	0.1246	29752	1028	0.2080
2	15946	900.14	0.0201	30940	1165	0.1836	15935	900.11	0.0103	50457	2296	0.0613	15955	900.37	0.0223	29184	1255	0.1448
3	15617	900.12	0.1107	37733	1286	0.2456	15017	900.17	0.0743	30010	1269	0.1977	14745	900.12	0.0562	29946	1264	0.1760
4	8952	900.11	0.1355	43469	2817	0.3415	8854	900.13	0.1252	30733	888	0.3268	8828	900.14	0.1202	65985	2893	0.3229
5	16448	900.15	0.0221	72359	2993	0.2510	16445	900.46	0.0195	29500	1181	0.1337	16447	900.32	0.0192	28982	1160	0.2296
6	30853	900.14	0.0473	44693	2343	0.0897	29831	900.97	0.0144	30157	1276	0.0505	29876	900.12	0.0172	63628	2666	0.0552
7	17256	900.52	0.0278	33687	1177	0.2522	17269	900.27	0.0290	31654	1219	0.1658	17249	900.21	0.0293	29123	1185	0.2452
8	6164	900.25	0.0608	32666	1113	0.6178	7387	900.19	0.2185	17813	1178	0.9388	6156	900.62	0.0568	30019	1311	0.6157
9	20272	900.09	0.0270	30202	1155	0.1280	20372	900.13	0.0279	26599	3111	0.0810	21824	900.14	0.0971	28949	2382	0.2036
10	15190	901.00	0.0355	29256	917	0.2234	15179	900.10	0.0346	30769	1041	0.1644	15185	900.23	0.0373	29921	869	0.2230
11	12878	900.07	0.1179	29410	1174	0.2025	12515	900.14	0.0923	61791	3378	0.1686	14631	900.14	0.2227	29901	1190	0.3662
12	10633	900.30	0.0809	63181	3026	0.3722	11223	900.24	0.1287	30060	1260	0.4481	11100	900.28	0.1180	30874	1306	0.4324
13	8633	900.10	0.0235	115086	2844	0.2636	8631	900.21	0.0040	271268	2895	0.2105	8632	900.14	0.0235	71300	3317	0.2635
14	12155	900.14	0.0727	39341	1754	0.1844	12151	900.13	0.0716	31790	1396	0.1840	12130	900.10	0.0681	36192	2045	0.1819
15	5723	900.27	0.0865	31778	1020	0.4329	5740	900.22	0.0917	182061	3480	0.4372	5745	900.46	0.0775	30084	1031	0.4384
16	14258	900.19	0.0305	31747	1763	0.2132	14311	900.15	0.0343	29541	2496	0.1650	14257	900.14	0.0319	32557	1103	0.2132
17	14634	900.15	0.0741	30407	1100	0.1372	14374	901.22	0.0564	36244	1090	0.1169	14148	900.13	0.0399	31125	1114	0.0994
18	19048	900.11	0.0345	60787	3406	0.0814	19091	900.14	0.0377	54597	2739	0.0838	19276	900.14	0.0465	75384	3160	0.0943
19	15907	900.13	0.0861	29693	1060	0.2756	15604	900.16	0.0687	15465	1478	0.2513	15336	900.27	0.0495	30533	1109	0.2298
20	13114	900.13	0.0233	77216	4140	0.2956	13114	900.09	0.0185	30646	1175	0.1696	13160	900.26	0.0258	33301	1315	0.2798
21	13216	900.12	0.0337	66356	3747	0.1542	13229	902.62	0.0374	31177	954	0.1554	13200	900.09	0.0351	29861	990	0.1528
22	11302	900.14	0.0421	109871	3018	0.2132	11296	900.16	0.0403	149590	4267	0.1978	11298	900.15	0.0445	109117	4042	0.2116
23	11877	900.17	0.0684	50862	3763	0.2456	12248	900.12	0.0975	34553	2441	0.2845	12239	900.47	0.0948	30147	1304	0.2836
24	4044	900.12	0.1508	62962	3039	0.7424	6202	900.11	0.4477	32887	1676	1.6721	3894	900.16	0.1100	71955	4655	0.6777
25	19226	900.20	0.0596	30307	1019	0.0969	19108	900.14	0.0546	23519	1846	0.0901	20797	900.29	0.1300	31836	1019	0.1865
26	16114	900.09	0.0361	29313	640	0.1227	16686	900.12	0.0688	29663	2205	0.1625	17296	901.67	0.0993	30974	1105	0.2050
27	15947	900.12	0.0540	24467	2776	0.1784	16277	900.18	0.0734	37140	2639	0.2028	16198	900.17	0.0681	31184	1318	0.1969
28	10219	900.37	0.0275	3069	1037	0.2042	10209	900.13	0.0238	93842	2843	0.1981	10220	900.15	0.0365	32105	1748	0.2043
29	21022	900.14	0.0196	30294	1240	0.0794	20979	900.19	0.0182	45970	3166	0.0610	21227	900.18	0.0291	29598	1254	0.0862
30	7749	900.15	0.0319	92586	3412	0.3253	7742	900.18	0.0272	121068	3597	0.2729	7752	900.10	0.0251	30678	1219	0.3258
Total	415896	27006.10	1.7444	1393705	60944	7.3297	418478	27009.50	2.1453	1654458	61445	7.8245	420613	27008.02	1.9561	1194195	51357	7.5536
Median		900.14	0.0447	33177	1520	0.2132		900.16	0.0475	31722	1761	0.1708		900.17	0.0480	30776	1260	0.2124
Mean		900.20	0.0581	46457	2031	0.2443		900.32	0.0715	55149	2048	0.2608		900.27	0.0652	39807	1712	0.2518
Maximum		901.00	0.1508	115086	4140	0.7424		902.62	0.4477	271268	4267	1.6721		901.67	0.2227	109117	4655	0.6777

A.3 Undominated valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement

Undominated Valid Inequalities for a Stochastic Capacitated Discrete Lot-sizing Problem with Lead Times, Cancellation and Postponement

Carlos E. Testuri[1], Héctor Cancela[1] and Víctor M. Albornoz[2]

[1]*Depto. de Investigación Operativa, Instituto de Computación, Facultad de Ingeniería, Universidad de la República, J. Herrera y Reissig 565, 11300 Montevideo, Uruguay*

[2]*Depto. de Industrias, Campus Santiago Vitacura, Universidad Técnica Federico Santa María, Avda. Santa María 6400, Vitacura, Santiago, Chile*

Keywords: Stochastic Lot-sizing, Multi-stage Stochastic Integer Programming, Valid Inequalities.

Abstract: The problem addresses the expected cost minimization of meeting the uncertain demand of a product during a discrete time planning horizon. The product is supplied by selecting fixed quantity shipments that have lead times. Due to the uncertainty of demand, corrective actions, such as shipment cancellations and postponements, must be taken with associated costs and delays. The problem is modeled as an extension of the discrete lot-sizing problem with different capacities and uncertain demand, which belongs to the $\mathcal{NP}$-hard class. To improve the resolution of the problem by tightening its formulation, valid inequalities based on the (ℓ, S) inequalities approach are used. Given that the inequalities are highly dominated for most experimental instances, a scheme is established to determine undominated ones. Computational experiments are performed on the resolution of the model and variants that include subsets of undominated and representative valid inequalities for instances of several information structures of uncertainty. The experimental results allow to conclude that the inclusion of undominated and representative derived (ℓ, S) valid inequalities enable a more efficient resolution of the model.

1 INTRODUCTION

The studied problem is the minimization of the expectation of the costs incurred in decisions taken to meet the uncertain demand of a product over a finite discrete time planning horizon. To meet the demand, there are certain optional distinguishable shipments (denominated as cargoes) with a non-fractional quantity of the product that can be acquired at most once with an associated cost. The cargoes have meaningful delivery lead times within the planning horizon; so that a significant amount of time elapses between the purchase decision and the moment when the cargo is received. After meeting the demand in a given period, the remaining quantity of product is stored, keeping an inventory up to a certain capacity, to flexibly satisfy the future demand in subsequent periods. Due to the passage of time, while the uncertainty of the demand is revealed and changes, it could happen that at a given time a cargo, that was already acquired (ordered) and has not yet been received, is no longer necessary. In this case it could be decided to cancel its acquisition order or postpone its delivery; decisions, which, in turn, have minimum execution times in relation to the time of delivery and associated costs. These decisions hedge against the risk of excess inventory.

The problem can be modeled as an extension of the lot-sizing formulation (Wagner and Whitin, 1958) and particularly of the variant with variable capacity and discrete dimensioning (Nemhauser and Wolsey, 1988). For the case where the parameters are known with certainty (deterministic case), the dimensioning is continuous, and without capacity constraints or with constant capacity, the problem has efficient resolution through dynamic programming (Wagner and Whitin, 1958; Wagelmans et al., 1992). In addition, there are known formulations which determine the convex hull of the feasible region: the extended facility location formulation (Krarup and Bilde, 1977) and the (ℓ, S) valid inequalities formulation (Barany et al., 1984). The deterministic variant with discrete sizing is a generalization of the binary knapsack problem, and belongs to the $\mathcal{NP}$-hard complexity class (Bitran and Yanasse, 1982).

In the case that the parameters are random variables (stochastic variant) the problem can be formulated by stochastic programming (Birge and Lou-

Testuri C., Cancela H. and Albornoz V.
Undominated Valid Inequalities for a Stochastic Capacitated Discrete Lot-sizing Problem with Lead Times, Cancellation and Postponement.
DOI: 10.5220/0007395203900397
In *Proceedings of the 8th International Conference on Operations Research and Enterprise Systems (ICORES 8th International Conference on Operations Research and Enterprise Systems)*, pages 390-397
ISBN: 978-989-758-352-0

veaux, 1997). An adjusted extended formulation of the stochastic continuos non-capacitated problem with Wagner-Whitin conditions are not satisfied for the stochastic variant (Ahmed et al., 2003). The (ℓ, S) inequalities are also valid for the stochastic continuous variant, and they were extended to a general class that allow to define facets of the feasible set (Guan et al., 2006).

Other variants of the deterministic continuous non-capacitated lot-sizing problem model delivery time of the lots (e.g. due to production time). A variant in which demands have a compliance interval has efficient resolution by dynamic programming (Lee et al., 2001). There are two variants according to whether the lots are or are not distinguishable with respect to delivery times (Brahimi et al., 2006). For there are efficient algorithms based on dynamic programming for the distinguishable case and for the undistinguishable case when the order-delivery windows are not inclusive. For these variants, there are tight extended formulations (Wolsey, 2006). For the stochastic case, the problem can be efficiently solved when delivery windows do not intersect in time (Huang and Küçükyavuz, 2008). The distinctive features of the problem under study in the present work are cancellation and postponement corrective decisions with time delays in a stochastic setting; these aspects are novel and were not found in the literature review.

The present work is organized as follows. In Section 2 an algebraic model of the problem is presented. Valid inequalities for the model are presented in Section 3. In Section 4 experiments are established to determine utility of the valid inequalities formulation. Work is completed with Section 5, where conclusions and future work are discussed.

2 STOCHASTIC MODEL FORMULATION

Basic index sets are established according to Table 1. The planning time is represented by the set T of discrete time periods. The set C of cargoes is partitioned in two sets: the set A of already acquired cargoes – cargoes ordered in past resolutions of the model– that are pending reception, and the set P of possible cargoes to be acquired from now on. Acquisition decision could be made on possible cargoes to be acquired, and cancellation and postponement decisions could be made on cargoes that were acquired before the actual planning horizon.

The uncertain demand is represented by a discrete-time stochastic process indexed in the planning periods. The process is defined in a finite probability space. It is assumed that the demand of the first period is deterministic, and that the demands of the remaining periods are random with known distribution functions. The decisions made in a period can not anticipate the realization of the uncertainty of the next period. These decisions must simultaneously take into account all possible revelations of the demand uncertainty of the following periods. This information structure can be represented by a tree structure called *tree of scenarios* (Römisch and Schultz, 2001). This is a perfect directed tree, with the root node representing the present time at period $t = 1$, and with leaf nodes identifying the future scenarios at period $t = H$.

Each node of the scenario tree describes the state of the process at a given period, and it is identified by a period and a scenario. An useful abbreviated notation is to identify the nodes by a single index n in a numerable set of nodes, N. For the first period, $t = 1$, there is a unique node, denoted by 1, that represents the root of the tree. Each node $n \in N$ has an immediate time predecessor node, $p(n)$; the auxiliary node 0 is defined as the predecessor of the root node, $0 := p(1)$, such that $0 \notin N$. The period corresponding to each node n is denoted as $t(n)$. The probability of the state of each node n is denoted as π_n, such that $\sum_{n \in N|t(n)=t} \pi_n = 1$, for all $t = 1, ..., H$. The t-th time predecessor of node n is defined as $p(n,t) := p(p(n,t-1))$, such that $p(n,1) := p(n)$. The nodes of the path from the root node to node n are denoted as the ordered set $P(n) := \{p(n,t(n)-1), ..., p(n,1), n\}$. The set of successors of node n is defined as $S(n) := \{n' \in N, k = 1, ..., H - t(n) | n = p(n', k)\}$. The set of leaf nodes is $L := \{n \in N | t(n) = H\}$.

Table 1: Basic index sets.

T	periods, $\{1, ..., H\}$
A	already acquired cargoes
P	possible cargoes to be acquired
C	cargoes, $A \cup P$
N	nodes of the scenario tree
L	leaf nodes of the scenario tree

Parameters are described in Table 2. The demand volume of the product at each node n is known and denoted as d_n. The demand distribution for period $t = 1, ..., H$ is represented by (d_n, π_n) such that $n \in N$ and $t(n) = t$. Due to storage constraints, the inventory of the product at the end of each period is restricted between a minimum volume, $\underline{s}$, and a maximum volume, $\bar{s}$, and there is an initial storage volume, s_0, at the beginning of the planning horizon.

The period at which an already acquired cargo

Table 2: Parameters.

d_n	demand volume at node $n \in N$
π_n	probability of node $n \in N$
s_0	initial inventory volume
$\underline{s}, \overline{s}$	min. and max. storage capacities by period
τ^c	period in which already acquired cargo $c \in A$ is received
q^c	volume of cargo $c \in C$
γ^c	delivery time of cargo $c \in P$, such that $0 \leq \gamma^c \leq H-1$
δ^c	cancellation minimum time of already acquired cargo $c \in A$, such that $0 \leq \delta^c \leq \tau^c - 1$
ε^c	postponement minimum time of already acquired cargo $c \in A$, such that $0 \leq \varepsilon^c \leq H - \tau^c$
ca^c	acquisition unit cost of cargo $c \in C$
cc^c	cancellation unit cost of cargo $c \in A$
cp^c	postponement unit cost of cargo $c \in A$
h_t	storage unit cost in period $t \in T$
a_t	already acquired volume that is received in period $t \in T$ (auxiliary deducted parameter)

$c \in A$ is received is fixed, τ^c, and it is decided in previous acquisitions (i.e. previous model resolutions). Each cargo $c \in C$ has a given volume, q^c. The achievement of decisions on cargoes have latency times measured in periods. The delivery time of a cargo $c \in P$, γ^c, establishes the length of the wait time (measured in periods) between the acquisition decision and the actual arrival of the cargo. The minimum time for cancellation of a cargo $c \in A$, δ^c, establishes the minimum number of periods prior to the delivery period at which the cargo may be cancelled. The minimum postponement time for a cargo $c \in A$, ε^c, establishes the minimum number of periods after the initial delivery period in which the posponed cargo can be received. The achievement period of decisions on acquisition, cancellation and postponement must take place within the planning horizon.

For each cargo $c \in C$ there are unit costs per volume associated with the decisions to acquire, ca^c, cancel, cc^c, and postpone, cp^c. In addition, there is a unit cost associated with storage at each period $t \in T$, h_t.

The already acquired volume that is scheduled to be received at each period is determined by the sum of the cargoes that are received in that period,

$$a_t := \sum_{\{c \in A | \tau^c = t\}} q^c, \quad \forall t \in T;$$

this is an auxiliary parameter that summarize decisions of previous model resolutions on a rolling horizon scheme.

Table 3: Derived index sets.

N^c_γ	nodes where it is possible to acquire cargo $c \in P$, $\{n \in N \| t(n) \leq H - \gamma^c\}$
N^c_δ	nodes where it is possible to cancel and postpone cargo $c \in A$, $\{n \in N \| t(n) \leq \tau^c - \delta^c\}$
T^c_ε	periods to where it is possible to postpone cargo $c \in A$, $\{t \in T \| t \geq \tau^c + \varepsilon^c\}$

Table 4: Functions and mappings of nodes.

$t(n)$	period of node $n \in N$
$p(n)$	immediate predecessor node of node $n \in N$ in the tree
$p(n,t)$	t-th time predecessor node of node $n \in N$ in the tree
$P(n)$	nodes in the path from root node to node $n \in N$ in the tree
$S(n)$	successor nodes of node $n \in N$ in the tree

Table 5: Variables.

s_n	inventory volume at the end of the period of node $n \in N$
u_n	acquired volume incoming at node $n \in N$
v^c_n	if cargo $c \in P$ is acquired at node $n \in N^c_\gamma$, (binary)
w_n	cancelled volume outgoing of node $n \in N$
x^c_n	if already acquired cargo $c \in A$ is cancelled in node $n \in N^c_\delta$ (binary)
y_n	postponed volume incoming at node $n \in N$
z^c_{nt}	if already acquired cargo $c \in A$ is postponed in node $n \in N^c_\delta$ to period $t \in T^c_\varepsilon$ (binary)

Derived subsets of the sets of nodes and periods that are indexed in the parameters are established in Table 3 in order to facilitate the formulation. There are subsets to abbreviate the denomination of nodes where it is possible to acquire each cargo $c \in P$, N^c_γ, and where it is possible to cancel and postpone each cargo $c \in A$, N^c_δ. In addition, subsets of periods to where it is possible to postpone each cargo $c \in A$ are established, T^c_ε. The subset subscripts are part of their denomination.

Functions and mappings on the nodes of the tree are summarized in Table 4.

In the stochastic model, all decisions depend on the nodes of the tree according to Table 5. An acquisition decision on a cargo $c \in P$ at node $n \in N^c_\gamma$ is represented by binary variable v^c_n. For each of these decisions, the delivery of cargo c will be on the nodes of the subtree rooted on n with period $t(n) + \gamma^c$, $\{n' \in S(n) | t(n') = t(n) + \gamma^c\}$. A cancellation decision of an already acquired cargo $c \in A$ at node $n \in N^c_\delta$ is represented by binary variable x^c_n. For each of these decisions, the already acquired volume of cargo c that was budgeted to be delivered on the nodes of the subtree rooted at n with period τ^c, $\{n' \in S(n) | t(n') = \tau^c\}$, is cancelled. A postponement decision of an already acquired cargo $c \in A$ is modeled by a cancellation decision of the cargo in conjunction with a decision to postpone it towards period $t \in T^c_\varepsilon$, that is represented by binary variable z^c_{nt}. For each of these decisions, the already acquired volume of cargo c that was budgeted to be delivered on the nodes of the subtree rooted on n at period τ^c, $\{n' \in S(n) | t(n') = \tau^c\}$, is postponed to the nodes of the subtree rooted on n at period $t \in T^c_\varepsilon$, $\{n' \in S(n) | t(n') \in T^c_\varepsilon\}$. The amounts of inventory, acquisition, cancellation and postponement at node n are summarized and represented by the variables, s_n, u_n, w_n and y_n, respectively.

The indexes of periods in deterministic parameters or variables are reduced to the temporary realization of a node n by $t(n)$. This is the case for parameters corresponding to the already acquired volume and storage unit cost.

From the previous definitions the formulation of the multi-stage stochastic optimization model (SCS) is

$$\min \sum_{n\in N} \pi_n \Big[\sum_{\{c\in P | n\in N^c_\gamma\}} ca^c q^c v^c_n \tag{1}$$

$$+ \sum_{\{c\in A | n\in N^c_\delta\}} (cc^c - ca^c) q^c x^c_n \tag{2}$$

$$+ \sum_{\{c\in A, t\in T^c_\varepsilon | n\in N^c_\delta\}} (cp^c + ca^c - cc^c) q^c z^c_{nt} \tag{3}$$

$$+ h_{t(n)} s_n \Big], \tag{4}$$

s.t.

$$s_{p(n)} + a_{t(n)} + u_n + y_n = d_n + w_n + s_n, \quad \forall n \in N, \tag{5}$$

$$\underline{s} \le s_n \le \overline{s}, \quad \forall n \in N, \tag{6}$$

$$u_n = \sum_{\{c\in P | \gamma^c + 1 \le t(n)\}} q^c v^c_{p(n,\gamma^c)}, \quad \forall n \in N, \tag{7}$$

$$\sum_{n'\in P(n)} v^c_{n'} \le 1, \quad \forall c \in P, \forall n \in N | t(n) = H - \gamma^c, \tag{8}$$

$$w_n = \sum_{\{c\in A | t(n) = \tau^c\}} \left(q^c \sum_{\{n'\in P(n) | t(n') \le \tau^c - \delta^c\}} x^c_{n'} \right), \quad \forall n \in N, \tag{9}$$

$$\sum_{n'\in P(n)} x^c_{n'} \le 1, \quad \forall c \in A, \forall n \in N | t(n) = \tau^c - \delta^c, \tag{10}$$

$$x^c_n \ge z^c_{nt}, \quad \forall c \in A, \forall n \in N^c_\delta, \forall t \in T^c_\varepsilon, \tag{11}$$

$$y_n = \sum_{\{c\in A | t(n) \ge \tau^c + \varepsilon^c\}} \left(q^c \sum_{\{n'\in P(n) \cap N^c_\delta\}} z^c_{n',t(n)} \right), \quad \forall n \in N, \tag{12}$$

$$\sum_{\{n'\in P(n), t\in T^c_\varepsilon\}} z^c_{n't} \le 1, \quad \forall c \in A, \forall n \in N^c_\delta, \tag{13}$$

$$s_n, u_n, w_n, y_n \ge 0, \quad \forall n \in N,$$
$$v^c_n \in \{0,1\}, \quad \forall c \in P, \forall n \in N^c_\gamma,$$
$$x^c_n, z^c_{nt} \in \{0,1\}, \quad \forall c \in A, \forall n \in N^c_\delta, \forall t \in T^c_\varepsilon.$$

This formulation takes into account the information structure of the scenario tree. It minimizes the expectation of acquisition costs (1), cancellation costs less acquisition costs in case of cancellation (2), postponement costs plus acquisition costs minus postponement costs (3) (a postponement is modeled in conjunction with a cancellation) and storage costs (4).

Constraints (5) set the volume balance for each node. The lower and upper storage bounds at each node are determined by inequalities (6). The amount of product acquired that is received at each node is determined by acquisitions of cargoes in the possible range of the corresponding acquisition periods according to equalities (7). The constraints (8) state that each cargo is acquired at a single node at most in each path from the root node to a node whose period coincides with the latest acquisition period of the cargo. The product previously acquired that is cancelled at each node is determined by the cancellations of the nodes in the path from the root node to the node, whose cancellation periods are less than the delivery period less the cancellation time, according to (9). Constraints (10) state that each cargo to be cancelled is at a single node at most in each path from the root node to a node whose period coincides with the receiving period minus the cancellation time of the cargo. The postponement of the cargoes is modeled in conjunction with the cancellation, i.e. only cancelled cargoes can be postponed, (11). The already acquired volume that is postponed in a node is determined by the postponements of the cargoes in the nodes in the path from the root to the node for all periods superior to the period of reception plus the delay time of the node, according to (12). Constraint (13) state that each cargo to be postponed is at a single node at most in each path from the root node to a node in some period greater than the receiving period plus the time of postponement of the node.

3 VALID INEQUALITIES FOR THE STOCHASTIC MODEL

Since (SCS) belongs to the time complexity class $\mathcal{NP}$-hard, there is no known polyhedral description of the convex hull of its feasible solutions. It is nevertheless interesting to derive valid inequalities which can be used to strengthen the original formulation. In some cases adding these inequalities can directly improve the capacity of the solver to find solutions for larger instances in shorter times; even when this is not the case, they may be used within a more sophisticated solving strategy, such as branch and cut methods relying on constraint separation.

In this section, we discuss a variation of classic (ℓ,S) valid inequalities for the stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement, and a scheme to obtain undominated valid inequalities of the variation.

3.1 Derived (ℓ,S) Valid Inequalities

A set of valid inequalities are derived por (SCS) based on the (ℓ,S) valid inequalities formulation for the deterministic uncapacitated lot-sizing problem (Barany et al., 1984) while considering the extension for the stochastic case (Guan et al., 2006). The derived inequalities establish bounds on decision variables for the nodes of possible paths in the scenario tree (Testuri et al., 2018).

Proposition 1 . *Let $\ell \in N$ and $S \subseteq P(\ell)$ then the derived (ℓ,S) inequality*

$$\sum_{n\in S} u_n \leq \sum_{n\in S} d_{n\ell}\beta(n) + \sum_{n\in S} w_{n\ell} + s_\ell, \tag{14}$$

where

$\beta(n) := \sum_{\{c\in P|\gamma^c+1\leq t(n)\}} v^c_{p(n,\gamma^c)}$,

$d_{n\ell} := \sum_{n'\in P(\ell)\setminus P(p(n))} d_{n'}$ *and*

$w_{n\ell} := \sum_{n'\in P(\ell)\setminus P(p(n))} w_{n'}$,

is valid for the feasible region of (SCS).

As shown by the authors the derived (ℓ,S) valid inequalities has alternative and equivalent inequalities without inventory variable, s_n,

$$\sum_{n\in P(\ell)\setminus S} u_n + \sum_{n\in S} d_{n\ell}\beta(n) + \sum_{n\in S} w_{n\ell} - w_{1\ell} + \sum_{n\in P(\ell)} y_n \geq d_{1\ell} - \sum_{n\in P(\ell)} a_{t(n)} - s_0, \quad \text{for all } \ell\in N, S\subseteq P(\ell). \tag{15}$$

3.2 Undominanted (ℓ,S) Valid Inequalities

Depending on the instance values of the parameters q^c and d_n, some of the derived (ℓ,S) inequalities (15) of a given subset $S\subseteq P(\ell)$, $\ell\in N$, may be dominated by other inequalities of a different subset. Therefore, a procedure was established to determine undominanted inequalities on the power set of $P(\ell)$.

Let $\boldsymbol{\chi} := [(v^c_n)_{c\in P, n\in N^c_\gamma}, (x^c_n)_{c\in A, n\in N^c_\delta}, (z^c_{nt})_{c\in A, n\in N^c_\delta, t\in T^c_\varepsilon}]$ be the composite variable. Let $b := d_{1\ell} - \sum_{n\in P(\ell)} a_{t(n)} - s_0$, for each $\ell\in N$, be the independent term of (15). Given the power set of $P(\ell)$, $S^p_\ell := \{S_1,...,S_{K_\ell}\}$, let $\boldsymbol{\alpha}_k$ be the coefficient vector of variable $\boldsymbol{\chi}$ on the inequality (15) for subset $S_k, k\in\{1,...,K_\ell\}$. Therefore, the inequalities (15) can be established as

$$\boldsymbol{\alpha}^T_k\boldsymbol{\chi} \geq b, \quad k\in\{1,...,K_\ell\}, \ell\in N.$$

Given $i,j\in\{1,...,K_\ell\}$ and that $\boldsymbol{\chi}$ is nonnegative, it is said that $\boldsymbol{\alpha}^T_i\boldsymbol{\chi}\geq b$ dominantes $\boldsymbol{\alpha}^T_j\boldsymbol{\chi}\geq b$, if the componentwise comparison of $\boldsymbol{\alpha}_i$ and $\boldsymbol{\alpha}_j$ is such that $\alpha_{i\lambda}\leq\alpha_{j\lambda}$ for each component $\lambda\in\Lambda$, and at least for one component the inequality is strict. Let S^d_ℓ be the subset of *dominant* inequalities on S^p_ℓ.

The procedure to obtain S^d_ℓ by pairwise comparison of inequalities has and upper bound of $O(K^2_\ell|\Lambda|)$ operations. If S^d_ℓ has few elements, an efficient heuristic to obtain a promising undominanted inequality candidate,

$$i^* := \text{argmin}_{i\in K_\ell} \sum_{\lambda\in\Lambda} \alpha_{i\lambda}, \tag{16}$$

takes $\Theta(K_\ell|\Lambda|)$ operations. Lets denote $S^{d*}_\ell := \{i^*\}$.

Furthermore, lets denote S^r_ℓ the case where the *power set*, S^p_ℓ, is approximated by a *representative* subset of S^p_ℓ that contains only the root node, $n=1$.

Tree variants of the original formulation, (SCS), are generated by including to it the inequalities of the sets S^p_ℓ, S^{d*}_ℓ and S^r_ℓ, for each $\ell\in N$, establishing formulations denoted as (SCS-S^p), (SCS-S^{d*}) and (SCS-S^r), respectively.

4 COMPUTATIONAL EXPERIMENTS

This section explores the computational impact of adding three families of inequalities introduced in the previous section to the original formulation. These are the power set inequalities (S^p), the dominance reduction inequalities (S^{d*}), and the root representative inequalities (S^r). The original formulation and the three modified formulations are tested over a set of test instances, checking the quality of the obtained solutions and the computational effort invested by the solver.

In order to generate a number of diverse test instances, six scenario tree structures were considered.

Each structure, depicted in Table 6, is determined by the number of direct descendants of each node (tree arity) and the number of periods of the planning horizon. For each tree structure with arity g and H periods there are g^{H-1} escenarios and $(g^H - 1)/(g-1)$ nodes.

Table 6: Size of scenario tree structures.

Arity(g)	Periods(H)	Scenarios	Nodes
2	5	16	31
2	6	32	63
2	7	64	127
3	5	81	121
3	6	243	364
3	7	729	1093

The size of each tree structure model instance (number of equations and variables) for a given distribution of cargos (C) is shown in Table 7.

Table 7: Instance size of scenario tree structures by cargo distribution.

g	H	$\lvert C\rvert(\lvert A\rvert+\lvert P\rvert)$	Eqs.	Vars.	(binary)
2	5	10 (2+ 8)	225	249	(124)
2	6	12 (3+ 9)	480	549	(296)
2	7	14 (3+11)	1.012	1.223	(714)
3	5	10 (3+ 7)	827	809	(324)
3	6	12 (4+ 8)	2.542	2.485	(1.028)
3	7	14 (4+10)	7.987	8.091	(3.718)

Thirty data instances were randomly generated for each tree structure and cargo distribution, totaling 180 instances. Each instance has an initial storage, $s_0 = 20$, and a lower and an upper bound storage, $\underline{s} = 0$ and $\bar{s} = 80$, respectively. For each cargo $c \in C$ there is an uniformly distributed volume, $q^c \sim U[10, 50]$, and there are costs evenly distributed according to the operations of acquisition, $ca^c \sim U[150, 250]$, cancellation, $cc^c \sim U[30, 50]$, and postponement, $cp^c \sim U[5, 12]$. Each already acquired cargo $c \in A$ has delivery period $\tau^c = 1$ or 2 with equal probability. Each cargo $c \in C$ has delivery time $\gamma^c = 1$, cancellation time $\delta^c = 1$ and delay time $\varepsilon^c = 1$. The unit storage cost at each period t is $h_t = 1$. For each scenario $n \in L$ (leaf node), a probability of state π_n is established from a distribution $Beta(\alpha = 2, \beta = 2)$; the probability of the remaining nodes is obtained from the sum of the probabilities of their corresponding immediate successor nodes. Finally, the demand for each node is evenly distributed, $d_n \sim U[10, 50]$.

The computational implementation was performed using AMPL (Fourer et al., 2002) for the algebraic coding of the stochastic model, and GUROBI 7.5 (Gurobi Optimization, LLC, 2018) for the resolution of the instances through its branch and cut solver.

The calculations were carried out on an Intel Core i7 5960X 3.5GHz computer with 20MB cache and 64GB RAM, operating with CentOS-7 Linux system.

For each instance, the original model and the variants were solved. A summary of the results of the original model and each variant is presented in Table 8, Table 9, Table 10 and Table 11, respectively for (SCS), (SCS-S^p), (SCS-S^{d*}) and (SCS-S^r).

Table 8: Average results of formulation (SCS) by tree structure and cargoes.

g-H-C	Time(s)	MIP	Nodes	Cuts	LP
2-5-10	0.68	-	6,449	125	10.31
2-6-12	13.07	-	30,706	189	18.85
2-7-14	†493.45	0.26	1,093,185	713	9.84
3-5-10	13.39	-	31,260	300	12.40
3-6-12	‡758.80	1.90	733,244	1,319	19.64
3-7-14	#900.25	5.02	27,291	1,264	23.42

(†) 12 of 30 instances reach the time limit of 900 s.
(‡) 24 of 30 instances reach the time limit of 900 s.
(#) All instances reach the time limit of 900 s.

Table 9: Average results of formulation (SCS-S^p) by tree structure and cargoes.

g-H-C	Time(s)	MIP	Nodes	Cuts	LP
2-5-10	1.11	-	2,132	88	7.44
2-6-12	17.48	-	24,885	253	11.25
2-7-14	†430.79	0.25	431,172	1,184	6.97
3-5-10	7.51	-	3,189	206	9.72
3-6-12	‡692.13	1.36	174,553	1,539	17.21
3-7-14	#900.83	12.85	5,754	?	36.60

(†) 12 of 30 instances reach the time limit of 900 s.
(‡) 19 of 30 instances reach the time limit of 900 s.
(#) All instances reach the time limit of 900 s.

Table 10: Average results of formulation (SCS-S^{d*}) by tree structure and cargoes.

g-H-C	Time(s)	MIP	Nodes	Cuts	LP
2-5-10	0.48	-	2,693	98	7.46
2-6-12	13.22	-	37,603	249	11.25
2-7-14	†435.25	0.28	1,005,992	1,024	6.97
3-5-10	5.87	-	12,281	239	9.72
3-6-12	‡720.13	1.71	513,898	1,609	17.24
3-7-14	#900.19	4.66	41,495	1,367	20.53

(†) 12 of 30 instances reach the time limit of 900 s.
(‡) 21 of 30 instances reach the time limit of 900 s.
(#) All instances reach the time limit of 900 s.

The summary shows, for each tree structure defined by arity, periods and number of cargoes, depicted at column "g-H-C", the average results of the 30 instances of the model (SCS) and its variant with the corresponding valid inequalities. The average results depicted are solver elapsed time at column "Time", solver MIP gap percentage for instances that reach the time limit of 900 s at column "MIP", solver number of nodes of solver branch and cut method at column

Table 11: Average results of formulation (SCS-S^r) by tree structure and cargoes.

g-H-C	Time(s)	MIP	Nodes	Cuts	LP
2-5-10	0.45	-	3,361	94	7.48
2-6-12	11.90	-	26,885	237	11.25
2-7-14	†443.32	0.27	1,145,992	978	6.99
3-5-10	4.87	-	11,584	264	9.74
3-6-12	‡449.50	0.73	330,166	1,546	17.20
3-7-14	#900.18	4.84	29,881	1,416	20.90

(†) 12 of 30 instances reach the time limit of 900 s.
(‡) 12 of 30 instances reach the time limit of 900 s.
(#) All instances reach the time limit of 900 s.

"Nodes", number of cuts added by solver's branch and cut method at column "Cuts", and the relative ratio percentage of the objective value with respect of the objective value of the linear programming relaxation of the model, at column "LP".

In the case of formulation (SCS-S^p), it can be seen that the average Time results for the tree structures (2-5-10) and (2-6-12) are worse than the corresponding to formulation (SCS). On the other hand, the average Time and MIP-gap results of the formulation for the tree structures (2-7-14), (3-5-10) and (3-6-12) are better than the corresponding to formulation (SCS). Also, the formulation reduces to 19 the number of instances of structure (3-6-12) that reach the time limit of 900 s, compared with 24 of the (SCS) formulation. Finally, except for structure (3-7-14), the formulation obtains a reduction of the LP-gap of the remaining structures compared with formulation (SCS).

In the case of formulation (SCS-S^{d*}), only the average Time results for the tree structures (2-6-12) are slightly worse than the corresponding ones of formulation (SCS). The formulation has lower LP-gap for all tree structures compared to formulation (SCS). With regards to its comparison with formulation (SCS-S^p), the formulation obtains better Time results for the tree structures (2-5-10), (2-6-12) and (3-5-10); and it obtains equal or slightly worse MIP-gap results, except for formulation (3-7-14), where it gets better result.

Formulation (SCS-S^r) obtains better Time results than formulations (SCS) and (SCS-S^p) for all tree structures. While it get better MIP-gap results for all tree structures than the formulation (SCS), it gets slightly worse MIP-gap results than formulation (SCS-S^p), except for structure (3-7-14), where it gets better result. In comparison with formulation (SCS-S^{d*}), it has slightly better Time results, and similar MIP-gap results. It reduces to 12 the number of instances of structure (3-6-12) that reach the time limit of 900 s, compared with 21 of the (SCS-S^{d*}) formulation.

5 CONCLUSIONS

A stochastic multi-stage capacitated discrete lot-sizing model formulation of the provision with lead time of the uncertain demand of a product has been proposed. The decisions on product lots are modeled with their delay time, aspect that for cancellation and postponement decisions is not covered in the previous literature. A discrete time stochastic process with finite probability, summarized in a scenario tree, is used to model the information structure of the uncertain demand. The model is formulated by stochastic programming with entities indexed by nodes of the scenario tree. The model incorporates the cancellation and postponement decisions with delay time, which implied the revision of the definitions of the variables and the restrictions to take into account the structure of the scenario tree. To tighten the formulation valid inequalities based on the (ℓ, S) inequalities approach were used. Since the inequalities are highly dominated for most experimental instances, a scheme is established to determine undominated ones. Three variants of the formulation are obtained from the inclusion of the power-set, undominated and representative valid inequations. The original formulation and the three variants are tested over a set of test instances, checking the quality of the obtained solutions and the computational effort invested by the solver. Computational experiments where carried out for several instances within a few tree structures of different sizes. Most computational experiments could be solved to optimality for the small and medium-size tree structures. The representative and undominated formulations obtains a slightly better results than the original and power set formulations for all tree structures.

ACKNOWLEDGMENTS

This work was partially supported by the Comisión Sectorial de Investigación Científica, CSIC, and the Programa de Desarrollo de las Ciencias Básicas, PEDECIBA.

REFERENCES

Ahmed, S., King, A. J., and Parija, G. (2003). A multi-stage stochastic integer programming approach for capacity expansion under uncertainty. *Journal of Global Optimization*, 26(1):3–24.

Barany, I., Roy, T., and Wolsey, L. A. (1984). *Mathematical Programming at Oberwolfach II*, chapter Uncapacitated lot-sizing: The convex hull of solutions, pages

32–43. Springer Berlin Heidelberg, Berlin, Heidelberg.

Birge, J. R. and Louveaux, F. (1997). *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering. Springer, New York.

Bitran, G. R. and Yanasse, H. H. (1982). Computational complexity of the capacitated lot size problem. *Management Science*, 28(10):1174–1186.

Brahimi, N., Dauzere-Peres, S., Najid, N. M., and Nordli, A. (2006). Single item lot sizing problems. *European Journal of Operational Research*, 168(1):1 – 16.

Fourer, R., Gay, D. M., and Kernighan, B. W. (2002). *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press.

Guan, Y., Ahmed, S., Nemhauser, G. L., and Miller, A. J. (2006). A branch-and-cut algorithm for the stochastic uncapacitated lot-sizing problem. *Mathematical Programming*, 105(1):55–84.

Gurobi Optimization, LLC (2018). Gurobi optimizer reference manual.

Huang, K. and Küçükyavuz, S. (2008). On stochastic lot-sizing problems with random lead times. *Operations Research Letters*, 36(3):303 – 308.

Krarup, J. and Bilde, O. (1977). *Plant location, Set Covering and Economic Lot Size: An $O(mn)$-Algorithm for Structured Problems*, pages 155–180. Birkhäuser Basel, Basel.

Lee, C.-Y., Çetinkaya, S., and Wagelmans, A. P. M. (2001). A dynamic lot-sizing model with demand time windows. *Management Science*, 47(10):1384–1395.

Nemhauser, G. and Wolsey, L. (1988). *Integer and Combinatorial Optimization*. John Wiley & Sons, Inc.

Römisch, W. and Schultz, R. (2001). Multistage stochastic integer programs: An introduction. In Grötschel, M., Krumke, S. O., and Rambau, J., editors, *Online Optimization of Large Scale Systems*, pages 581–600. Springer Berlin Heidelberg.

Testuri, C. E., Cancela, H., and Albornoz, V. M. (2018). Valid inequalities for a stochastic capacitated discrete lot-sizing problem with lead times, cancellation and postponement. *(Manuscript submitted for publication)*.

Wagelmans, A., Hoesel, S. V., and Kolen, A. (1992). Economic lot sizing: An $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research*, 40:S145–S156.

Wagner, H. M. and Whitin, T. M. (1958). Dynamic version of the economic lot size model. *Management Science*, 5(1):89–96.

Wolsey, L. A. (2006). Lot-sizing with production and delivery time windows. *Mathematical Programming*, 107(3):471–489.