SUPPLY CHAIN OPTIMIZATION WITH UNCERTAINTY AND HIERARCHICAL DECISION-MAKERS

by

PABLO GARCIA-HERREROS

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

at

CARNEGIE MELLON UNIVERSITY
DEPARTMENT OF CHEMICAL ENGINEERING
PITTSBURGH, PENNSYLVANIA

DECEMBER, 2015

Acknowledgments

This research quest would not have been possible without the valuable contributions and the support from my advisor Professor Ignacio E. Grossmann. I want to thank Ignacio for being a source of inspiration for my research work and a role model for my life. Besides being an outstanding researcher, it is hard to imagine someone more devoted to his students than Ignacio. His wisdom, patience, and generosity have been fundamental for the successful completion of this thesis.

I am also very grateful to my PhD thesis committee: Professor Nikolaos Sahinidis, Professor Chrysanthos Gounaris, Professor Alan Scheller-Wolf, and Dr. John M. Wassick. Their feedback after the PhD proposal has been very important to define the scope of my research. It has been an honor to be evaluated by this distinguished group of researchers.

I want to acknowledge the collaboration with The Dow Chemical Company and Air Products & Chemicals. They have been instrumental to guide the direction of my research and provide a realistic context to my academic interests. In particular, I would like to acknowledge Dr. John M. Wassick and Dr. Anshul Agarwal from The Dow Chemical Company for their dedication and their contributions to Chapters 2 and 4. I am also grateful to Dr. Pratik Misra, Dr. Erdem Arslan, and Dr. Sanjay Mehta from Air Products & Chemicals for their contributions to Chapters 5 and 6.

This thesis has been the result of intensive research collaboration. I would like to give special credit to Dr. Sumit Mitra for his initiative to develop the algorithm presented in Chapter 3 and to Carlos Florensa Campo for his enthusiastic productivity that is captured in Chapter 6. Additionally, I want to acknowledge all professors and students form the CAPD research center for providing a very stimulating environment for the development innovative ideas.

I also want to acknowledge the support of the Fulbright Program. A few years back, it was only a dream to be a PhD student at Carnegie Mellon University. The Fulbright program has been not only an important source of financial support, its people have also been extraordinary at providing orientation throughout these years.

I also want to express gratitude and love for my family. I thank my father Ricardo and my mother Pilar for their effort to provide me the best education and for teaching me to embrace new challenges. I am grateful to my brothers Pedro and Santiago for sharing their intellectual curiosity and transmitting their passion for science. I am also grateful to my grandmother Nelly for being a role model of love and generosity. Finally, I want to thank my girlfriend Alejandra for her tenderness and support during the last years of my PhD.

Abstract

Supply chain models describe the activities carried out in the process industry. They are used to design and operate complex sequences of tasks that transform raw materials and deliver final products to markets. Many optimization models have been developed for supply chain planning because they offer the possibility of finding strategies that lead to greater economic benefits. The traditional models have focused on finding the optimal decisions of the supply chain planner in a deterministic context. However, it is widely recognized that uncertainty and external decision-makers play a fundamental role in the economic success of industrial supply chains.

This thesis proposes mathematical programming models for supply chain optimization that consider uncertainty and external decision-makers in a variety of industrial settings. Chapter 1 provides the motivation and the necessary background for our models. In Chapter 2, we study the design of resilient supply chains with risk of disruptions. Disruptions is a type of uncertainty that has not received much attention for supply chain planning, but it is known to have a significant effect in the performance of supply chains. We develop a stochastic programming model for supply chain design that includes disruptions at distribution centers, and a tailor-made solution method to address industrial instances of the problem. In Chapter 3, we present a novel cross-decomposition algorithm for investment planning under uncertainty. The algorithm integrates Benders and Lagrangean decomposition for two-stage stochastic programming formulations. Our computational experiments on instances of the resilient supply chain design problem show the superior performance of the cross-decomposition algorithm over Benders decomposition and direct solution with commercial MILP solvers. In Chapter 4, we propose a new approach for production planning and inventory management in process networks. Inventory management in these networks is a very challenging task because of the close interaction between production activities and the presence of diverse sources of uncertainty. Our planning strategy is based on implementing basestock policies to control production rates and inventory levels. Our results show the benefits of using a policy-based approach for inventory planning in comparison to other stochastic programming approaches. In Chapter 5, we address the capacity planning problem with rational markets. Our model considers potential customers as rational decision-makers in a bilevel optimization formulation. We propose two reformulation techniques that transform the bilevel model into a single-level problem by replacing the lower level with constraints that guarantee its optimality. The reformulations are based on the Karush-Kuhn-Tucker conditions and the strong duality property of the lower-level linear program; the examples show better computational performance for the duality-based reformulation. The results also demonstrate the benefits of considering markets as rational decision-makers for capacity expansion planning, since it allows developing expansion plans according to the needs of the consumers. In Chapter 6, we extend the capacity planning model to include competitors that optimize their own capacity expansion plans. The resulting trilevel formulation considers as rational all decision-makers present in a competitive environment. We analyze the properties of the trilevel formulation and develop two algorithms to solve this challenging problem. The results reveal the complex interactions that take place in decision-making problems with multiple players and show the importance of considering them in the model.

Finally, in Chapter 7 we present the conclusions of this thesis. We demonstrate that uncertainty and external decision-makers have significant impact in supply chain operations, and that our models can be used to anticipate their influence in supply chain performance. The application of these models for industrial supply chain planning has a remarkable potential to increase efficiency.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Modeling framework for supply chain optimization 2

1.2.1 Time representation 3

1.2.2 Flow constraints 4

1.2.3 Sources of uncertainty 5

1.2.4 Response to demand 6

1.2.5 Service level 7

1.2.6 Stockout models 7

1.2.7 Cost structure 7

1.3 Classical models for inventory management 9

1.3.1 Deterministic inventory models 10

1.3.2 Stochastic inventory models 11

1.4 Overview of stochastic programming 14

1.4.1 Decomposition methods for stochastic programming problems 17

1.5 Overview of bilevel optimization 19

1.6 Outline of the thesis 22

2 Design of Resilient Supply Chains with Risk of Disruptions 25

2.1 Motivation 25

2.2 Literature review 27

2.3 Problem statement 29

2.4 Mathematical model 30

2.5 Illustrative example 32

2.6 Solution method 34

2.6.1 Tightening the formulation 35

2.6.2 Multi-cut Benders decomposition 37

2.6.3 Strengthening the Benders master problem 38

2.6.4 Pareto-optimal cuts 39

2.6.5 Bounding the impact of scenario subsets 40

2.7 Implementation 43

2.8 Large-scale example 45

2.9 Industrial supply chain design 46

2.10 Summary 49

3 Implementation of Cross-decomposition for Two-stage Stochastic Programming 51

3.1 Motivation 51

3.2 Literature review 53

3.3 Mathematical model 54

3.4 Ingredients of the cross-decomposition algorithm 56

3.5 Computational experiments on the resilient supply chain design problem 60

3.5.1 Description of the implementation 60

3.5.2 Comparison of solution methods 61

3.6 Summary 65

4 Optimizing Inventory Policies in Process Networks under Uncertainty 66

4.1 Motivation . . . 66

4.2 Literature review . . . 68

4.3 Problem description . . . 72

4.4 Outline of solution and result evaluation methods . . . 72

4.5 Illustrative example . . . 73

4.6 Capacitated single-echelon basestock policy . . . 77

4.7 Illustrative example revisited . . . 79

4.8 Mathematical model for stochastic inventory planning in process networks . . . 81

4.8.1 Supply balances . . . 82

4.8.2 Production capacity . . . 82

4.8.3 Consumption balance . . . 82

4.8.4 Production balance . . . 83

4.8.5 Inventory balance . . . 83

4.8.6 Demand balance . . . 84

4.8.7 Objective function . . . 84

4.9 Policy for inventories in parallel . . . 84

4.9.1 Logic-based formulation . . . 85

4.9.2 An MILP reformulation . . . 87

4.10 Policy for inventories in series . . . 89

4.10.1 Logic-based formulation . . . 90

4.10.2 An MILP reformulation . . . 91

4.11 Evaluating inventory planning strategies with closed-loop Monte Carlo simulations 92

4.12 Example with a parallel arrangement of inventories . . . 95

4.13 Example with a multi-echelon arrangement of inventories . . . 101

4.14 Summary . . . 105

5 Bilevel Optimization for Capacity Planning with Rational Markets 108

5.1 Motivation 108

5.2 Literature review 110

5.3 Problem statement 112

5.4 Single-level capacity planning with captive markets 113

5.5 Bilevel capacity planning with rational markets 114

5.6 Reformulation as a single-level optimization problem 117

5.6.1 KKT reformulation 117

5.6.2 Duality-based reformulation 120

5.7 Illustrative example 122

5.8 Middle-size instances 124

5.9 Solution strategies for large-scale problems 129

5.9.1 Strengthened duality-based reformulation 129

5.9.2 Domain reduction for the duality-based reformulation 129

5.10 Industrial example 132

5.11 Summary 134

6 Capacity Planning with Competitive Decision-makers 136

6.1 Motivation 136

6.2 Problem statement 139

6.3 Capacity planning with competitive decision-makers: trilevel formulation 141

6.4 Capacity planning with competitive decision-makers: bilevel reformulation 143

6.5 Multilevel programming and degeneracy 145

6.5.1 Degeneracy in bilevel programming 145

6.5.2 Degeneracy in trilevel programming 146

6.6 Properties of the trilevel capacity planning formulation 150

6.6.1 Stability regions 150
6.6.2 A cut to eliminate solutions in $R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y})$ 153
6.6.3 Equations to tighten the HP relaxation 155
6.7 Algorithm 1: Constraint-directed exploration 156
6.7.1 Reaching the sequentially optimistic solution 157
6.7.2 Reaching the hierarchically optimistic solution 158
6.8 Algorithm 2: Column-and-constraint generation algorithm 160
6.8.1 Reaching the strategically optimistic solution 160
6.9 Capacity planning instances 163
6.10 Summary 169

7 Conclusions **171**

7.1 Review of the thesis 171
7.1.1 Design of resilient supply chains with risk of disruptions 171
7.1.2 Implementation of a novel cross-decomposition algorithm for two-stage stochastic programming 173
7.1.3 Optimizing inventory policies in process networks under uncertainty 174
7.1.4 Bilevel optimization for capacity planning with rational markets 176
7.1.5 Capacity planning with competitive decision-makers 178
7.2 Contributions of this thesis 180
7.3 Publications related to this thesis 182
7.3.1 Full-length articles 182
7.3.2 Review articles 182
7.3.3 Conference articles 183
7.4 Directions for future research 183
7.4.1 Include other sources of uncertainty in the resilient supply chain design . . 183

7.4.2 Automate the implementation of cross-decomposition for two-stage stochastic programming investment planning problems 184

7.4.3 Develop efficient solution methods for the optimization of inventory policies 184

7.4.4 Include uncertainty in the bilevel capacity planning model 185

7.4.5 Include contracts, mergers, and acquisitions in the capacity planning models 185

7.4.6 Develop general domain reduction strategies for hierarchical optimization . 186

A Data for the large-scale example of the resilient supply chain design problem 187

B Data for the random instances of the resilient supply chain design problem 191

C Data for the illustrative example of the bilevel capacity planning problem 193

D Capacity planning with rational markets: proof of Proposition 5.1 198

List of Tables

2.1 Parameters of the illustrative example. 32
2.2 Transportation costs in the illustrative example. 32
2.3 Availability matrix and scenario probabilities in the illustrative example. 33
2.4 Results of the illustrative example. 34
2.5 Expected costs of the large-scale example obtained from full-space and reduced instances. 45
2.6 Instance sizes and solution times for the large-scale example. 46
2.7 Instances of the industrial supply chain example. 48
2.8 Upper and lower bounds for instances of the industrial supply chain example. . . . 48

3.1 Computational statistics of full-space instances. 61
3.2 Comparison of mean solution times. 64

4.1 Optimal basestock levels and costs for the illustrative example. 76
4.2 Consumption coefficients in the example with a parallel arrangement of inventories. 96
4.3 Cost parameters in the example with a parallel arrangement of inventories. 96
4.4 Production capacities according to their DTMC state in the example with a parallel arrangement of inventories. 97
4.5 Computational statistics of the two-stage SP and the logic-based SP formulations in the example with a parallel arrangement of inventories. 98
4.6 Performance of the planning models in the example with a parallel arrangement of inventories. 100
4.7 Production capacities according to the state of units in the example with a multi-echelon arrangement of inventories. 102

4.8 Consumption coefficients in the example with a multi-echelon arrangement of inventories. . . . 103
4.9 Cost parameters in the example with a multi-echelon arrangement of inventories. . 103
4.10 Computational statistics of the two-stage SP and the logic-based SP formulations in the example with a multi-echelon arrangement of inventories. 103
4.11 Performance of the planning models in the example with a multi-echelon arrangement of inventories. 105

5.1 Model statistics of the illustrative example. 123
5.2 Results of the single-level and bilevel expansion plans in the illustrative example. . 124
5.3 Model statistics of middle-size instances. 127
5.4 Results of the bilevel expansion plans for the middle-size instances. 128
5.5 Model statistics of industrial instances. 132
5.6 Results of the bilevel expansion plans for the industrial instances. 133

6.1 Market demands in Instance 1. 164
6.2 Selling prices in Instance 1. 164
6.3 Cost parameters and initial capacities in Instance 1. 164
6.4 Optimal objective values for Instance 1. 165
6.5 Computational statistics of Algorithms 1 and 2 in Instance 1. 166
6.6 Computational statistics of Algorithms 1 and 2 in Instance 2. 167
6.7 Optimal objective values for Instance 2. 168

A.1 Probability of disruption at DC candidate locations in the large-scale example of the resilient supply chain problem design. 187
A.2 Cost coefficients for the large-scale example of the resilient supply chain design problem. 188
A.3 Transportation cost from plant to DCs in the large-scale example of the resilient supply chain design problem. 188
A.4 Customer demands in the large-scale example of the resilient supply chain design problem. 189
A.5 Transportation cost from DCs to customers in the large-scale example of the resilient supply chain design problem. 190

C.1 Summary of datasets used in capacity planning examples. 193

C.2 Initial capacity of plants in the illustrative example of the bilevel capacity planning problem. 194
C.3 Market demands in the illustrative example of the bilevel capacity planning problem. 194
C.4 Investment cost of the leader in the illustrative example of the bilevel capacity planning problem. 195
C.5 Maintenance costs in the illustrative example of the bilevel capacity planning problem. 195
C.6 Expansion cost in the illustrative example of the bilevel capacity planning problem. 195
C.7 Production cost in the illustrative example of the bilevel capacity planning problem. 196
C.8 Initial transportation cost in the illustrative example of the bilevel capacity planning problem. 196
C.9 Initial selling prices in the illustrative example of the capacity planning problem. . 197
C.10 Growth rates for transportation costs and selling prices in the illustrative example of the capacity planning problem. 197

List of Figures

1.1 Capacity constraints in a network with a source node, a storage node, and a sink node. 4
1.2 Lead time in a network with a source node, a storage node, and a sink node. 5
1.3 Tree representation of scenarios in a stochastic program with three stages. 15
1.4 Graphical outline of the thesis. 22

2.1 Optimal deterministic design for the illustrative example. 33
2.2 Optimal stochastic design for the illustrative example. 33
2.3 Subsets of scenarios according to disruptions. 41
2.4 Implementation sequence of the solution method. 44
2.5 Convergence of Benders algorithms for the full-space instance of the large-scale example. 47

3.1 Flowchart of the cross-decomposition algorithm. 57
3.2 Performance curves for the solution methods in instances with 639 scenarios. 62
3.3 Performance curves for the solution methods in instances with 1,025 scenarios. 63
3.4 Performance curves for the solution methods in instances with 1,587 scenarios. 63

4.1 Schematic representation of the illustrative example. 74
4.2 Scenario trees for the illustrative example. 80
4.3 Residual expected value as a function of the sample size. 81
4.4 Parallel arrangement with a shared resource for the replenishment of inventories. 85
4.5 A multi-echelon arrangement with 3 echelons. 89
4.6 Trajectory of uncertain parameters in a closed-loop Monte Carlo simulation. 94

4.7 Structure of the example with a parallel arrangement of inventories. 95

4.8 Deterministic demands in the example with a parallel arrangement of inventories. . 96

4.9 Trajectories of holding and stockout costs obtained from the two-stage SP formulation for the example with a parallel arrangement of inventories. 99

4.10 Trajectories of holding and stockout costs obtained from the logic-based SP formulation for the example with a parallel arrangement of inventories. 99

4.11 Comparison of replication costs for the example with a parallel arrangement of inventories. 100

4.12 Structure of the example with a multi-echelon arrangement of inventories. 101

4.13 Normally distributed supply and demand in the example with a multi-echelon arrangement of inventories. 102

4.14 Trajectories of holding and stockout costs obtained from the two-stage SP formulation for the example with a multi-echelon arrangement of inventories. 104

4.15 Trajectories of holding and stockout costs obtained from the logic-based SP formulation for the example with a multi-echelon arrangement of inventories. 104

4.16 Comparison of replication costs for the example with a multi-echelon arrangement of inventories. 106

5.1 Superstructure of regional gas markets. 112

5.2 Optimal demand assignments obtained in the first time period of the illustrative example using the bilevel formulation. 124

5.3 Evolution of market demands in the middle-size instances. 126

5.4 Evolution of selling prices in the middle-size instances. 126

5.5 Evolution of production costs in the middle-size instances. 127

5.6 Capacity and production of the leader in the first instance of the middle-size example.128

5.7 Capacity and production of commodity 1 at the plants controlled by the leader in the first instance of the industrial example. 134

5.8 Capacity and production of commodity 2 at the plants controlled by the leader in the first instance of the industrial example. 134

6.1 Example of different degeneracy resolution models. 149

6.2 Algorithm 1 towards the hierarchically optimistic solution. 159

6.3 Algorithm 2 towards the strategically optimistic solution. 162

6.4 Network of plants and markets in Instance 1. 163

6.5 Optimal demand assignments for Instance 1. 164
6.6 Convergence of Algorithms 1 and 2 in Instance 1. 165
6.7 Convergence of Algorithm 1 and Algorithm 2 in Instance 2. 167
6.8 Capacity and production of the leader in Instance 2. 168

Chapter 1

Introduction

1.1 Motivation

A major goal of all companies in the process industry is to manufacture products to satisfy market demands. In order to accomplish this purpose, they must execute complex sequences of tasks in which raw materials are transformed to final products that are delivered to markets. The coordination of the activities involved in this process is the focus of supply chain management.

The optimization of production and logistic networks has been one of the main drivers for the advance of mathematical programming since the development of the simplex algorithm [47]. Systematic decision-making methods are essential for the design and planning of large-scale supply chains because the complexity of their operations often conceals the best decisions from the intuition. In the highly globalized market place in which most modern companies compete, supply chain efficiency has become a requirement for economic success.

The goal of this thesis is to propose mathematical programming formulations and solution methods for the optimization of supply chains in realistic industrial environments. We focus on developing models that mitigate the risks faced by supply chain planners with respect to the future performance of the system. We address two of the most pressing risks for supply chain optimization: the presence of uncertainty and external decision-makers.

Decision-making strategies in uncertain and competitive environments are an important tool for supply chain management because they allow generating plans that are robust against business conditions that cannot be controlled. We approach the challenge of decision-making under uncertainty by formulating stochastic programming models; they offer the advantage of optimizing an expected metric of performance, which is a desirable objective for design and planning problems [96]. We consider competition in the supply chain environment by assuming that all decision-makers behave rationally; under this assumption, their actions can be modeled in the framework of hierarchical optimization [4].

The motivation for our research arises from the significant impact that supply chain management has in the economic performance of the process industry. Even though there has been extensive research on supply chain optimization in the last decades, some of the challenges that are most relevant to the chemical industry have not been addressed before; we consider the presence of diverse sources of uncertainty, the complexity of chemical production networks, and the role of competition in supply chain planning. In order to include these elements in the supply chain models, we develop novel mathematical programming formulations and specialized solution methods for large-scale problems.

1.2 Modeling framework for supply chain optimization

The interrelations among material flows in supply chains are represented mathematically as networks. A network is a directed graph with three types of nodes: sources, intermediates, and sinks [20]. The simplest network links a source node and a sink node through an arc. Additionally, supply chain networks often include storage nodes that act as intermediate nodes with the purpose of transferring the availability of material through time. For networks modeling production and transportation of commodities, the successful completion of the process implies the flow of material from source to sink nodes, according to economic criteria. In the following subsections, we provide some background on operations management and discuss the most important concepts for supply chain models.

1.2.1 Time representation

A realistic representation of supply chains considers their performance as a function of time. The dynamic conditions that govern production and logistic networks imply modeling time according to the attributes of material flows. A flow can be described either as a continuous-time event characterized by its flowrate and time span, or as a discrete event characterized by its magnitude and time of incidence [140].

The effect of continuous flows in storage nodes is the gradual change in the inventory level; in mathematical terms, the trajectory of the inventory level in time is said to be twice continuously differentiable [102]. On the other hand, discrete flows produce instantaneous changes in inventory levels. The trajectory of the inventory level subject to discrete flows exhibits non-smooth profiles.

The operation of real production and logistic networks generally involves a combination of continuous and discrete flows. Therefore, the representation of time is an important modeling aspect that determines when supply chain decisions can be implemented (e.g. one time, continuously, or periodically). Network attributes such as resource availability at source nodes, flow constraints, and material requirements at sink nodes are good indicators of the appropriate model for time.

Additionally, the time frame in which the network performance is evaluated also plays an important role in the development of the right model for supply chain planning. The supply chain operations might require decisions to be implemented in different time horizons. Based on the length of the planning horizon, supply chain models can be classified according to the following categories.

- **Single period:** problems for which decisions are implemented only once. Even though supply chain problems often involve long time horizons, many problems can be reduced to single period problems. This is the case when successive time periods are identical, for which repetitive implementation of the optimal single-period decision yields the optimal sequence of decisions.

- **Finite horizon:** problems for which supply chain decisions are implemented over a limited time span. These problems often require the coordination of a decision sequence or involve parameters that change over time. The most intuitive performance measure for these systems is the total cost during the finite horizon.

- **Infinite horizon:** problems for which supply chain decisions can be implemented an infinite number of times in a cyclic manner. These problems consider the impact of the decisions in an infinitely long time span. The most appropriate performance measure for these systems is the average cost per unit time.

1.2.2 Flow constraints

There are two basic restrictions to flows in production and logistic networks: capacity constraints and lead times. Capacity constraints are maximum flowrate restrictions in the processing activities. They are represented in the network model with capacitated arcs. Figure 1.1 shows a simple network made up by three nodes: a source node (1), a storage node (2), and a sink node (3). The availability of resources (supply) is modeled with the capacity (S) of the arc leaving the source node. Similarly, the availability of inventory is modeled with the capacity (inv) of the arc leaving the storage node. Requirements at the sink node (demand) are modeled with the capacity (D) of the arcs arriving to it.

Lead time (L) is the time required to transfer a flow between two nodes. It is represented in the network model by connecting the origin and destination nodes in different time periods. In Figure 1.2, storage node 2 plays the same role as in Figure 1.1. However, the flow constraint between nodes 1 and 2 is not determined by a capacitated arc but by the absence of a link connecting them in the same time period.

The distinction between capacity constraints and lead time is very important for the network representation of supply chains. Even though real supply chains include both types of flow constraints, most models assume that one of them governs the dynamic of the network. In general, production networks are considered to be constrained by capacities and logistic networks by lead times. A good example of networks in which both types of constraints are important is given by batch production systems.

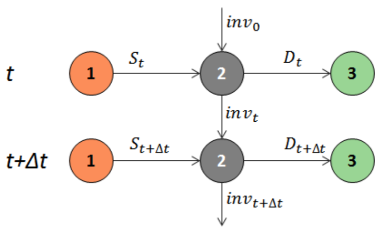


Figure 1.1: Capacity constraints in a network with a source node (1), a storage node (2), and a sink node (3).

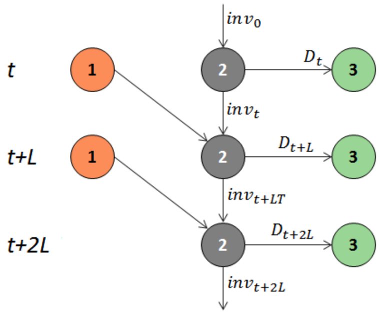


Figure 1.2: Lead time in a network with a source node (1), a storage node (2), and a sink node (3).

1.2.3 Sources of uncertainty

Uncertainty is present in all industrial supply chains. It can be characterized mathematically by modeling the parameters of the networks with random coefficients. The sources of uncertainty in production and logistic networks can be related to variability in supply, lead time, processing capacity, or demand. A detailed description of these sources of uncertainty is presented below.

- **Supply availability:** the availability of resources at source nodes has an important impact on network performance. Some authors have studied strategies to mitigate uncertainty in supply availability in logistic networks [230, 175, 14, 181, 182]. The standard approach to hedge against the risk of supply stockout is to increase the level of raw material inventory beyond the expected needs.

- **Lead time:** responsiveness of logistic networks is closely related to lead times. Lead time uncertainty affects network performance according to their duration and frequency. Longer expected lead times force operational decisions to be taken earlier, which decreases the reactive capacity of the system. Higher lead time variability implies less confidence in predictions, which requires more conservative decisions. Lead time uncertainty has been addressed in several research studies, mostly for logistic networks [145, 173, 11, 104, 241, 36].

- **Processing capacity:** responsiveness of production networks is also related to their capacity constraints. Complex networks with capacitated arcs often exhibit the formation of bottlenecks. Continuous-time models in which arc capacities are given by random quantities and flows can be temporarily stored are classified as fluid queuing networks. A good example of a supply chain with this type of uncertainty is presented by Mitra [159].

- **Demand:** demand variability is the most widely studied type of uncertainty in production and logistic networks. Demand is frequently characterized by a normal distribution but many authors have also study the impact of other distributions for supply chain management. Williams & Tokar [245] present a comprehensive review of papers related to inventory management, and classify them according with their assumptions regarding demand. Demand uncertainty can be efficiently hedged by increasing process responsiveness and by building up inventories. Demand is the main driver of production and logistic networks; therefore, the adequate handling of demand uncertainty has a deep impact in supply chain performance.

1.2.4 Response to demand

Supply chains use different strategies to deal with demand. The distinction between these strategies is based on the timing relationship between the moment of demand realization and the response of the network. Networks with negligible flow constraints (i.e. high processing capacity and short lead times) might prefer to wait until orders are placed to react. Other networks might need to anticipate future demands to be able to satisfy them when they occur. Most supply chains try to find a balance between these two strategies for their response to demand [1]. The basic modes of operation are presented below.

- **Make-to-order:** this reactive mode does not anticipate future demands for the supply chain planning. Networks operating in a *make-to-order* strategy function as pull systems. The *make-to-order* operation mode is most suitable for networks providing highly specialized products, networks with large demand variability, or networks high processing capacity. The main advantage of this mode is that it does not use inventory for demand satisfaction.

- **Make-to-stock:** this mode executes operations in anticipation of future demands. Networks operating in a *make-to-stock* strategy function as push systems. The *make-to-stock* operation mode promotes high utilization of the available processing capacity and leverages inventories to hedge against uncertainty. It is suitable for networks with limited reactive capacity that can improve their responsiveness with the accumulation of inventory.

1.2.5 Service level

One of the most important measures of performance for supply chains is the service level. Service level is an indicator that evaluates the capability of networks to satisfy demands. There are two main types of service levels as presented below.

- **In-stock probability (α):** network performance can be measured from the probability of satisfying demand realizations on time. The *in-stock probability* indicates the frequency with which demand is expected to be satisfied. This performance measure is also known as *Type-1 service level* [217].
- **Fill rate: (β):** the expected fraction of demand that can be satisfied by a network can also be used as a performance measure. The *fill rate* measures the average volume of demand that is satisfied on time with respect to the total volume of demand. This performance measure is also known as *Type-2 service level* [217].

1.2.6 Stockout models

The goal of production and logistic networks is to achieve high levels of service. However, it is not always possible to fully satisfy demands. The assumptions about the behavior of unsatisfied demands are often described by two models [27].

- **Backorders:** this model assumes that whenever stockouts occur, demand satisfaction can be postponed to the next time period. This behavior is usually assumed for products with no substitutes, companies with contracts, or highly consistent customers.
- **Lost sales:** this model assumes that late demand satisfaction is unacceptable and stockouts lead to lower sales. This model is appropriate for commodities or retail businesses in which products are suitable for substitution.

1.2.7 Cost structure

The efficiency of supply chains can be measured according to the costs associated to their design and operation. In order to balance investment and operating costs, it is common to consider them in a single cost function. Investment costs refer to the capital required to set up the supply chain. Operating costs include, among others, flow costs, inventory costs, and stockout costs. A detailed description of these elements is presented below.

- **Investment costs:** supply chain operations require infrastructure that is built with capital investments; these investments do not depend on the intensity of the supply chain operations. Investment planning is often the main goal of supply chain design. The cost structure of investments can consider fixed and variable costs, according to the discrete and continuous nature of the design decisions.

- **Flow costs:** production and transportation costs are the result of the transit of material through nodes and arcs. These operations are intended to add value to the materials. A linear function of flows is generally used to model the cost structure of production and transportation. Fixed charges can be included to model economies of scale, equipment setups, or transportation of discrete flows.

- **Inventory costs:** inventory costs are calculated from the economic value of the material and from operating expenses such as taxes, obsolescence, depreciation, insurance, and warehousing. The cost function of inventory is usually assumed to be proportional to the sum of inventory value and related expenses. There are two main approaches to calculate the value of inventory: First-In-First-Out (FIFO) and Last-In-First-Out (LIFO). The FIFO approach assumes that stored inventory corresponds to the last fraction of material that entered the system; therefore, its value is assessed according to the cost of recent purchases. The LIFO approach assumes that inventory corresponds to material left in the system from the earliest purchases. Both approaches can yield significant differences when monetizing inventory on long time horizons or during periods of high inflation.

- **Stockout costs:** depending on the stockout model, stockout costs might be described by different functions. Most frequently, backorders are modeled with a cost function that is linear in volume and lateness. Lost sales are usually considered proportional to the volume of stockouts. The coefficients of these linear cost functions represent the impact of stockouts in the long term performance of the company. They include tangible factors such as delay charges or additional labor, and intangible factors such as lost of reputation or goodwill.

1.3 Classical models for inventory management

The classical objective of inventory management is to minimize the cost associated to replenishment, holding, and stockouts. In this context, the best inventory management decisions are those that solve the corresponding minimization problem. Inventory management problems can be formulated as optimization problems, and the optimal solution for many of them can be characterized by inventory policies. The policies are rules that establish the replenishment decisions as functions of the inventory availability. Based on the method used to monitor the inventory and the time in which decisions can be implemented, inventory management strategies can be classified in two groups [217].

- **Continuous review policies:** monitor the trajectory of inventory at all times and implement replenishment decisions whenever a condition on inventory availability is met.
- **Periodic review policies:** monitor the inventory level and implement replenishment decisions only in periodic time intervals.

The replenishment decisions for both types of strategies are made according to the inventory level and the pipeline inventory. The inventory level (IL) represents the amount of material available in the storage node; it can be negative in the case of backorders. The inventory position (IP) additionally considers the replenishment orders that have been placed but are still to arrive; it is the sum of the inventory level and the pipeline inventory. Inventory level is important to calculate holding and stockout cost, whereas the replenishment decisions are usually based on the inventory position.

The optimal inventory policies for the most recurrent inventory management problems have been studied for decades [256]. They all deal with the question of determining the replenishment strategies for a single inventory system subject to demand. These models consider linear cost functions for ordering, holding, and stockouts with or without fixed charges. Some models include lead times, but none of them has capacity constraints. The only source of uncertainty considered in the classical models is given by stochastic demand.

1.3.1 Deterministic inventory models

Inventory models for which all parameters are known with certainty are designated as deterministic. For these models, the purpose of inventory is related to coordination of flows and cost efficiency. The absence of uncertainty in these systems implies that future inventory and stockout levels are established precisely from the planning model.

The Economic Order Quantity (EOQ) model

The EOQ is the basic inventory model. It was originally developed by Harris [101] in 1914 to determine the economic size of manufacturing lots. Its purpose is to balance average ordering and holding costs in an infinite horizon. The problem assumes a deterministic demand rate (D) and allows no stockouts. The inventory policy in the EOQ model is characterized by the order quantity (Q), which is the size of the replenishment that should be received the moment that the inventory becomes empty.

The original problem does not consider lead time nor capacity constraints. However, a deterministic lead time (L) can be easily included just by placing the optimal order quantity (Q^*) in advance. In order to obtain Q^*, it is necessary to define the cost function. The elements contributing to the average cost are the ordering and the holding costs. If the ordering cost is assumed to be a fixed charge (K), its contribution to the average cost is related to the frequency of orders (D/Q). The holding cost for the EOQ model is proportional to the average inventory with constant (h). The average cost as a function of the order quantity (Q) is presented in Equation (1.1),

$$g(Q) = \frac{KD}{Q} + \frac{hQ}{2} \tag{1.1}$$

where the first term represents the average fixed ordering cost and the second term the average holding cost.

The optimal ordering quantity (Q^*) can be found by differentiating with respect to Q and equating to zero. The expression for Q^* is presented in Equation (1.2).

$$Q^* = \sqrt{\frac{2KD}{h}} \tag{1.2}$$

Other deterministic models

Many deterministic models have been developed as modifications of the basic EOQ model [216]. The main problem with this policy in complex logistic networks is that coordination among multiple inventories can become very difficult. In order to simplify coordination, ordering decisions can be restricted only to some time intervals. This additional restriction is used in the *power-of-two policy*, for which orders are only allowed on a power of two multiple of the base time period. Other interesting modifications of EOQ models arise from including quantity discounts in the cost structure or allowing planned backorders. Snyder & Shen [216] describe cost functions and optimal policies for these systems.

The inventory management problem with finite horizon and deterministic time-varying demand is known as the *Wagner-Whitin model* [240]. The problem considers fixed ordering cost, linear holding cost, and does not allow stockouts. The *Wagner-Whitin model* is formulated as a Mixed-Integer Linear Programing (MILP) problem and can be solved using dynamic programming. In a general framework, deterministic inventory management problems with finite time horizons can be seen as scheduling problems in which decisions are related to inventory replenishment.

1.3.2 Stochastic inventory models

Classical inventory models considering uncertainty are focused on stochastic demands. The traditional objective function of these models is to minimize the expected cost of replenishment and stockouts; however, stochastic inventory models are also subject to risk assessments [34, 22]. Whether stockouts are assessed from a risk neutral or a risk averse perspective, inventory is used to coordinate flows, increase cost efficiency, and reduce stockouts. The risk reduction function of inventory is carried out by safety stock. Inventory policies for continuous and periodic review strategies have been developed according to demand patterns and the timing of replenishment decisions. The stochastic version of the EOQ model is the fundamental approach for continuous review policies. Periodic review strategies are based on basestock policies; the *newsvendor model* describes the standard basestock policy for the single-period case.

(*r, Q*) policy

The continuous review inventory policy with lead time (L) and stochastic demand rate (D) is characterize by the reorder point (r) and the reorder quantity (Q). The policy places a replenishment order of size Q when the inventory position reaches level r. This replenishment strategy is known to be optimal to minimize expected cost functions that consider fixed ordering charges (K), linear holding costs with constant h, and linear stockout costs with constant p. The model requires the assumptions that holding costs are calculated from the inventory level, whether it is positive or negative, and that stockout costs are only charged once per unit demand when they are backordered. The cost function for the (*r,Q*) policy represents the expected cost per unit of time as given by Equation (1.3),

$$g(r,Q) = \frac{KD}{Q} + h\left(r - DL + \frac{Q}{2}\right) + p\frac{D\,n(r)}{Q} \tag{1.3}$$

where the first term is the expected ordering cost, the second term the expected holding cost, and the third term the expected stockout cost. In the case of normally distributed demands, the expected number of stockouts per inventory cycle is given by the loss function ($n(r)$) of demand during lead time. There are no closed form solutions for Equation (1.3), but the minimum expected cost solution satisfies Equations (1.5)-(1.4),

$$Q^* = \sqrt{\frac{2D\left[K + p\,n(r)\right]}{h}} \tag{1.4}$$

$$r = F^{-1}\left(1 - \frac{Qh}{pD}\right) \tag{1.5}$$

where F^{-1} is the inverse cumulative distribution function (cdf) of demand during the lead time.

Basestock policies

Basestock policies are characterized by the inventory position that is restored every time a replenishment is order is placed [217]. Such level is called the basestock level (S). This class of policies is known to minimize the expected cost of several periodic review models with finite and infinite horizons, whether they include fixed charges or not. The most prominent example of basestock policies is the newsvendor model.

Newsvendor model The single period problem with zero lead time, no fixed ordering charges, and lost-sales can be illustrated with the case of a newspaper vendor. The perishable attribute of newspapers does not allow leftover inventory to be used in the future, eliminating any connection between consecutive time periods. The cost function for this formulation includes a linear holding cost for the end-of-period inventory and a linear stockout cost; the cost coefficients are h and p, respectively. Equation (1.6) presents the cost function for a system subject to random demand D,

$$g(S) = h\,(S-D)^{+} + p\,(D-S)^{+} \tag{1.6}$$

where $(S-D)^{+}$ represents the expected end-of-period inventory and $(D-S)^{+}$ the expected stockouts. The optimal basestock level (S^{*}) for the newsvendor model is given by Equation (1.7),

$$S^{*} = F^{-1}\left(\frac{p}{h+p}\right) \tag{1.7}$$

where F^{-1} is the inverse cdf of the single-period demand, and the fraction $\frac{p}{h+p}$ is known as the critical ratio. The critical ratio corresponds to the optimal in-stock probability (service level α).

The optimal basestock and the optimal cost for the case in which demand is normally distributed with mean μ and standard deviation σ can be calculated from Equations (1.8)-(1.9),

$$S^{*} = \mu + \sigma\phi^{-1}\left(\frac{p}{h+p}\right) \tag{1.8}$$

$$g^{*}(S) = h\,[z + \mathcal{L}(z)]\,\sigma + p\,\mathcal{L}(z)\,\sigma \tag{1.9}$$

where ϕ^{-1} is the cdf of the normal demand, $z = \frac{S-\mu}{\sigma}$, and $\mathcal{L}(z)$ is the standard normal loss function.

Multi-period basestock policies The implementation of basestock policies in multi-period problems modifies the newsvendor model to allow inventory to be carried over time periods. In the finite time horizon case, the cost function includes the linear holding and stockout costs, and a terminal cost related to the final inventory. If the terminal cost is convex, then a basestock policy is optimal for every time period [217]; for time-varying stochastic demands, the optimal basestock level varies with time. In the infinite horizon case with independent and identically distributed (iid) demands, basestock policies have also been shown to be optimal. The optimality proof for basestock policies in the infinite horizon are presented by Zipkin [256].

1.4 Overview of stochastic programming

Stochastic programming is the framework that models mathematical programs with uncertainty by optimizing the expected value over the possible realizations [19]. Other approaches to model mathematical programs with uncertainty, such as robust optimization [12, 147] and chance constraint programming [142], focus on the feasibility of the solutions over the uncertainty sets. Modeling approaches focusing on feasibility tend to be appropriate for short-term problems with little room for recourse decisions, whereas stochastic programming is considered better suited for long-term planning [96]. A comprehensive review on optimization under uncertainty is presented by Sahinidis [193].

In stochastic programming models, the expected value is generally computed by integrating over the set of uncertain parameters, which might be a challenging task. In the case of discrete uncertainty sets with finite support, the realizations can be characterized with a finite number of scenarios, which simplifies the calculation of the expected value. Therefore, stochastic programming is often regarded as the scenario-based approach for optimization under uncertainty.

Stochastic programming formulations can accommodate decision making at different stages according to the sequence in which uncertainty is revealed. The stages imply a discrete time representation of the problem and establish the information about the uncertain parameters available at that time. The potential paths in which discrete uncertain parameters might evolve are represented in a scenario tree as shown in Figure 1.3. In these trees, each node is a decision-making

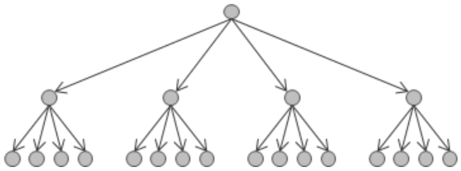


Figure 1.3: Tree representation of scenarios in a stochastic program with three stages.

instance with known realization of the uncertain parameters up to the current state; potential future realizations are represented with branches from the given node.

The simplest stochastic programming formulation only considers decisions that are made before uncertainty reveals. These models are called single-stage stochastic programs or stochastic programs without recourse. Among the stochastic programs that consider recourse, the most widely used formulation is the Mixed-Integer Linear Program (MILP) with continuous recourse in a second stage [96]. This two-stage stochastic programming formulation divides the decisions into two sets: *here-and-now* decisions that are made before uncertainty reveals, and *wait-and-see* decisions that are independent for each scenario. The typical formulation of a linear two-stage stochastic programming problem is presented in Equations (1.10)-(1.13),

$$\min_{x \in X} \quad c^T x + \mathop{\mathbb{E}}_{s} \left[Q_s \left(x \right) \right] \tag{1.10}$$

$$\text{s.t.} \quad Ax \leq b \tag{1.11}$$

$$Q_s \left(x \right) = \min_{y_s \in Y_s} \quad d_s^T y_s \tag{1.12}$$

$$\text{s.t.} \quad W_s y_s \leq h_s - T_s x \tag{1.13}$$

where x is the vector of first-stage decisions in polyhedral set X, y_s are vectors of second-stage (recourse) decisions in polyhedral sets Y_s, and s is the index for scenarios. The objective function of the first-stage is given by Equation (1.10), and the constraints of the first-stage problem are represented by Equation (1.11). Similarly, the objective function of the second-stage is given by Equation (1.12), and the corresponding constraints are presented in Equation (1.13).

An important property of the linear two-stage stochastic programming formulation is that the first-stage cost is given by a convex function [19]. Based on this property and assuming that the second-stage problems ($Q_s(x)$) are bounded, a model that explicitly calculates the expected value can be derived by including all second-stage problems in the formulation. This reformulation, known as the *deterministic equivalent* of the stochastic programming problem, is presented in Equations (1.14)-(1.17),

$$\min \quad c^T x + \sum_{s \in S} \mathbb{P}_s \, d_s^T y_s \tag{1.14}$$

$$\text{s.t.} \quad Ax \leq b \tag{1.15}$$

$$W_s y_s \leq h_s - T_s x \qquad \forall \ s \in S \tag{1.16}$$

$$x \in X, \ \ y_s \in Y_s \qquad \forall \ s \in S \tag{1.17}$$

where $\mathbb{P}_s$ denotes the probability of scenario s.

The benefits of using a stochastic programming model can be quantified by the *Value of the Stochastic Solution* (VSS). The VSS is the difference between the expected values obtained from implementing the solution predicted by the stochastic formulation and the solution obtained by a deterministic formulation that substitutes the uncertain parameters with their nominal values. The expected value of the deterministic formulation is calculated by solving the deterministic problem, implementing the *here-and-now* decisions, and evaluating the scenarios with their optimal recourse.

The model presented in Equations (1.10)-(1.13) can also be extended to a multistage stochastic programming model. The tree corresponding to a three-stage problem has the structure shown in Figure 1.3. Solving multistage stochastic programming problems can be considerably harder, and special care must be taken to model the sequence in which uncertainty is revealed. The general formulation of a linear three-stage stochastic program is presented in Equations (1.18)-(1.23),

$$\min_{x \in X} \quad c^T x + \mathop{\mathbb{E}}_{s} \left[Q_s \left(x \right) \right] \tag{1.18}$$

$$\text{s.t.} \quad Ax \leq b \tag{1.19}$$

$$Q_s \left(x \right) = \min_{y_s \in Y_s} \quad d_s^T y_s + \mathop{\mathbb{E}}_{s|s_2} \left[P_s \left(x, y_s \right) \right] \tag{1.20}$$

$$\text{s.t.} \quad W_s y_s \leq h_s - T_s x \tag{1.21}$$

$$P_s \left(x, y_s \right) = \min_{z_s \in Z_s} \quad f_s^T z_s \tag{1.22}$$

$$\text{s.t.} \quad V_s z_s \leq g_s - U_s x - R_s y_s \tag{1.23}$$

where z_s are vectors of third-stage variables defined in polyhedral sets Z_s. The objective function of the third-stage is given by Equation (1.22), and its constraints are presented in Equation (1.23). It is important to remark that the expectation in the second-stage objective function is conditional on the realization of the second level uncertainty; this conditional expectation over the scenarios is denoted by $\mathbb{E}_{s|s_2}$.

One way of transforming the multistage formulation into its deterministic equivalent is to generate a set of copied variables for each path that goes from the root node to the branches, and to introduce non-anticipativity constraints that force copied variables to have the same values according to uncertainty revealed at each stage [189]. This alternative representation has the advantage of being relatively easy to implement. Solution methods for large stochastic programming problems is still an active area of research due to the large number of scenarios needed for industrial applications and the rapid growth of scenarios in multistage stochastic programs.

1.4.1 Decomposition methods for stochastic programming problems

Methods used to solve large stochastic programming problems leverage the scenario structure that produces a block-diagonal shape in the constraints. Similarly to the application of decomposition methods to other types of mathematical programs, the strategies used for stochastic programming find the optimal solution by iteratively solving problems of lower complexity. The scenario structure of the stochastic programming problem implies that most of the variables and constraints in the formulation have no direct interaction with each others. In particular, it is easy to identify a relatively small set of complicating variables and complicating constraints connecting the scenarios. This observation gives rise to the main decomposition methods for stochastic programming: Lagrangean decomposition and the L-shaped method. Other decomposition strategies include Progressive Hedging and Nested Decomposition procedures [44].

Lagrangean decomposition

Given that a small set of constraints is responsible for linking decisions across scenarios in stochastic programs, relaxing these complicating constraints allows finding the optimal solution for each scenario independently. The Lagrangean decomposition method [97] uses Lagrangean relaxation over the alternative representation of a stochastic program to dualize the non-anticipativity constraints. An implementation of this dual decomposition method to stochastic programs was presented by Carøe & Schültz [29]. The solution of the relaxed subproblems yields a lower bound for the minimization of the original problem. This relaxation is improved iteratively by updating the Lagrange multipliers associated with the non-anticipativity constraints; the multipliers can be updated using different strategies, including subgradient and cutting planes methods. The main difficulty with the application of Lagrangean decomposition is that the solution of the Lagrangean relaxation is likely to violate the relaxed constraints and it might be difficult to find feasible solutions to the original problem. Additionally, the optimal solution of the Lagrangean dual only provides a lower bound to the original problem and the duality gap might not be satisfactory to assess the quality of the solution. Therefore, Lagrangean decomposition is often implemented in a Branch-&-Bound algorithm that closes the duality gap and with a heuristic that provides feasible solutions. The main advantage of Lagrangean decomposition is that it can be used to solve two-stage and multistage stochastic programming problems with mixed-integer variables in all stages.

L-shaped method

The L-shaped method is the implementation of Benders decomposition to linear two-stage stochastic programming problems with continuous recourse [238]. The method considers first-stage variables as complicating because the subproblem corresponding to each scenario can be solved independently once they are fixed. Every iteration of the L-shaped method solves the scenario subproblems to evaluate first-stage solutions and a relaxed master problem that generates new candidate solutions for the first-stage variables. The optimal primal and dual solutions of the subproblems provide an upper bound on the original problem, a candidate solution (if feasible), and dual information that can be used to generate Benders cuts that approximate the feasible region of the original problem in the space of the first-stage variables. These cutting planes are included iteratively in the master problem to improve the description of the feasible region, predict better lower bounds, and obtain new first-stage solutions. The algorithm is guaranteed to converge to the optimal solution of the original problem in a finite number of iterations. The rate of convergence of the L-shaped method strongly depends on the linear programming relaxation of the problem [152, 194]; in subproblems with multiple optima, judicious selection of the alternative Benders cuts can greatly improve the convergence of the algorithm [152]. Additionally, the scenario structure of the stochastic program can be exploited in the L-shaped method to generate independent cuts for each subproblem in every iteration. The addition of multi-cuts to the master problem is likely to produce a major reduction in the number of iterations required for convergence [18].

1.5 Overview of bilevel optimization

Bilevel optimization models are mathematical programs that include an optimization problem in the constraints [23]. They are suitable to model problems in which two independent decision-makers optimize their own objective functions sequentially [28, 9]. In the bilevel optimization literature, the upper-level decision-maker is designated as the *leader* and the lower-level decision-maker as the *follower*. Bilevel formulations are often used in game theory to model Stackelberg competitions [239].

The generic mixed-integer formulation of a bilevel programming problem is given by Equations (1.24)-(1.28).

$$\max_{w,x} \ F\left(w,x,y,z\right) \tag{1.24}$$

$$\text{s.t.} \ G\left(w,x\right) \leq 0 \tag{1.25}$$

$$\max_{y,z} \ f\left(y,z\right) \tag{1.26}$$

$$\text{s.t.} \ \ g\left(w,x,y,z\right) \leq 0 \tag{1.27}$$

$$\left(w,y\right) \in \mathbb{Z} \quad \left(x,z\right) \in \mathbb{R} \tag{1.28}$$

The variables in the bilevel optimization problem are divided into two groups: upper-level variables (w, x) that are controlled by the leader with the goal of maximizing the upper-level objective (1.24), and lower-level variables (y, z) that are controlled by the follower to maximize the lower-level objective (1.26). The most common formulation of the bilevel optimization problem assumes that the upper-level constraints presented in Equation (1.25) only depend on the upper-level variables. Lower-level constraints presented in Equation (1.27) depend on both, upper- and lower-level variables.

The bilevel formulation implies a hierarchy between the decisions of the leader and the follower. The leader makes its decisions first, and then the follower reacts rationally to optimize its objective function; once the leader has established its solution, the variables controlled by the leader are treated as parameters in the lower-level optimization problem. The bilevel optimization formulation assumes that the leader has perfect information about the decision criterion of the follower and that the decisions made by the leader are visible for the follower. In order to facilitate the understanding of the problem, we provide the following definitions for the bilevel problem.

The bilevel constraint region is given by:

$$\Omega = \{(w,x,y,z):\ G\left(w,x\right) \leq 0,\ g\left(w,x,y,z\right) \leq 0,\ (w,y) \in \mathbb{Z};\ (x,z) \in \mathbb{R}\} \tag{1.29}$$

The projection of Ω onto the first-level decision space is given by:

$$\Omega_{w,x} = \{(w,x):\ \exists\,(y,z)\ \text{ s.t. } (w,x,y,z) \in \Omega\} \tag{1.30}$$

The lower-level feasible set for upper-level variables $(\bar{w},\bar{x})$ is given by:

$$\Omega(\bar{w},\bar{x}) = \{(y,z):\ g\left(\bar{w},\bar{x},y,z\right) \leq 0,\ y \in \mathbb{Z};\ z \in \mathbb{R}\} \tag{1.31}$$

The second-level rational reaction set for an upper-level solution $(\bar{w},\bar{x})$ is given by:

$$\Psi(\bar{w},\bar{x}) = \{(y,z) : \arg\max\left[f\left(\bar{w},\bar{x},y,z\right)\ \text{ s.t. } (y,z) \in \Omega(\bar{w},\bar{x})\right]\} \tag{1.32}$$

The inducible region (IR) is given by:

$$IR = \{(w,x,y,z) : (w,x) \in \Omega_{w,x}, (y,z) \in \Psi(w,x)\} \tag{1.33}$$

It is important to remark that the inducible region (1.33) corresponds to the set of feasible solutions of the bilevel optimization problem. In contrast to the single level optimization problems, a relaxation of the bilevel problem is not obtained by relaxing the integrality conditions in Equation (1.28) [55]. This counterintuitive observation can be explained by the fact that relaxing the lower-level integrality increases the size of the follower's rational reaction set (1.32).

It should also be noted that checking for bilevel feasibility is not a trivial task. Bilevel feasibility implies that the solution belongs to the inducible region; however, as pointed out by Candler & Townsley [28], this region might be nonconvex even for purely linear bilevel problems. The simplest way to find a bilevel feasible solution is to follow a two-step procedure: first, obtain feasible values for upper-level variables ($(w, x) \in \Omega_{w,x}$), and then find the corresponding values for the lower-level variables in the rational reaction set (1.32). Feasible values for the upper-level variables can be found by solving the *high-point problem* presented in Equations (1.34)-(1.37).

$$
\begin{aligned}
\max \quad & F(w, x, y, z) && (1.34)\\
\text{s.t.} \quad & G(w, x) \leq 0 && (1.35)\\
& g(w, x, y, z) \leq 0 && (1.36)\\
& (w, y) \in \mathbb{Z} \quad (x, z) \in \mathbb{R} && (1.37)
\end{aligned}
$$

The *high-point problem* is obtained by removing the follower's objective function from the bilevel formulation. The *high-point* solution $(\hat{w}, \hat{x}, \hat{y}, \hat{z})$ yields an upper bound on the upper-level maximization because the *high-point problem* is a relaxation of the original bilevel optimization problem [9]. The rational response of the follower can be found by fixing the upper-level variables and solving the lower-level problem presented in Equations (1.38)-(1.40).

$$
\begin{aligned}
\max \quad & f(\hat{w}, \hat{x}, y, z) && (1.38)\\
\text{s.t.} \quad & g(\hat{w}, \hat{x}, y, z) \leq 0 && (1.39)\\
& y \in \mathbb{Z} \quad z \in \mathbb{R} && (1.40)
\end{aligned}
$$

Most solution methods for bilevel optimizatin problems have focused on mixed-integer linear problems with only continuous variables in the lower-level. The approaches for bilevel programs with lower-level Linear Programs (LPs) leverage the fact that optimal solutions occur at vertices of the region described by upper- and lower-level constraints. They rely on vertex enumeration, directional derivatives, penalty terms, or optimality conditions [192]. The most direct approach is to reformulate the bilevel optimization as a single-level problem using the optimality conditions of the lower-level LP. The classic reformulation using Karush-Kuhn-Tucker (KKT) conditions main-

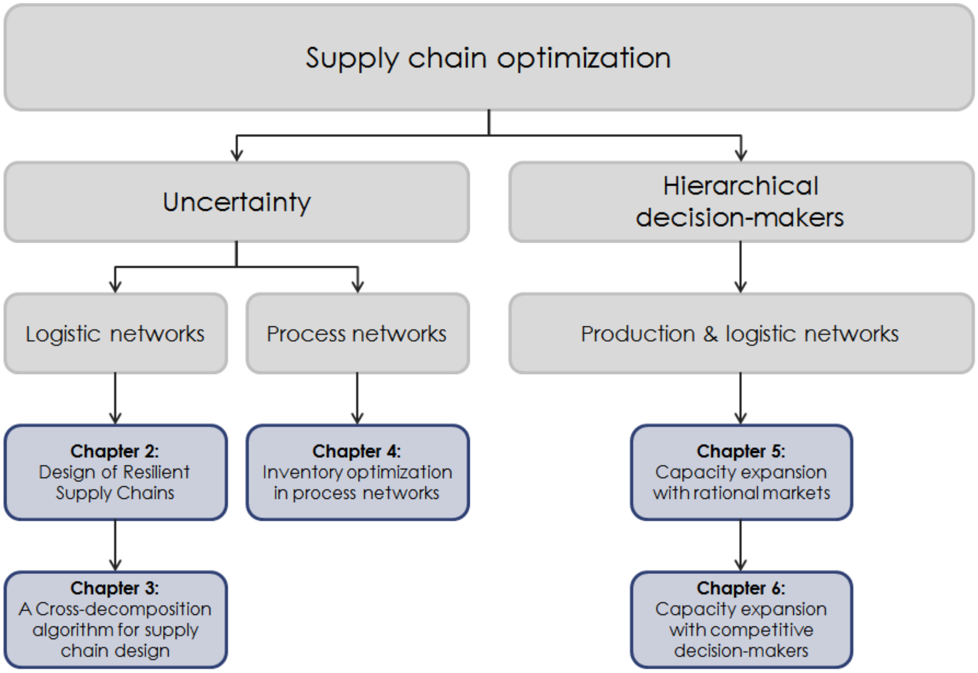

Supply chain optimization
Uncertainty
Hierarchical decision-makers
Logistic networks
Process networks
Production & logistic networks
Chapter 2: Design of Resilient Supply Chains
Chapter 4: Inventory optimization in process networks
Chapter 5: Capacity expansion with rational markets
Chapter 3: A Cross-decomposition algorithm for supply chain design
Chapter 6: Capacity expansion with competitive decision-makers

Figure 1.4: Graphical outline of the thesis.

tains linearity of the problem except for the introduction of complementarity constraints [72, 8, 16]. An equivalent reformulation replaces the lower-level problem by its primal and dual constraints, and guarantees optimality by enforcing strong duality [164, 75].

1.6 Outline of the thesis

This thesis addresses supply chain optimization with uncertainty and hierarchical decision-makers in five chapters that focus on different elements of the problem. A graphical overview of the thesis is presented in Figure 1.4. Chapters 2 to 6 address supply chain problems that are relevant to the process industry, propose mathematical programming formulations to formalize the decision-making problems, and develop methods yielding the corresponding optimal solutions. The general aim of our models is to serve as tools that facilitate the decision-making process in the design and operation of industrial supply chains.

Chapter 2 addresses the design of resilient supply chains under the risk of disruptions at candidate distribution centers (DCs). The problem is formulated as a two-stage stochastic programing model; the solution establishes optimal DC locations, storage capacities, and demand assignments in scenarios describing disruptions at potential DCs. The objective is to minimize the sum of investment cost and expected distribution cost during a finite time horizon. The rapid growth in the number of scenarios requires a specialized method to solve large-scale problems. We develop a method based on multi-cut Benders decomposition that offers deterministic bounds on the cost function for problems with a very large number of scenarios. The results demonstrate the importance of including DC capacity in the design problem and anticipating the distribution strategy in adverse scenarios.

In Chapter 3, we present a cross-decomposition algorithm that combines Benders and scenario-based Lagrangean decomposition for two-stage stochastic programming problems with complete recourse. We implement the cross-decomposition algorithm for planning problems where the first-stage variables are mixed-integer and the second-stage variables are continuous. The algorithm fully integrates primal and dual information in terms of multi-cuts added to the Benders and the Lagrangean master problems for each scenario. The benefits of the cross-decomposition scheme are demonstrated with instances of the resilient supply chain design with risk of disruptions. The computational experiments show performance improvements in comparison to commercial MILP solvers and multi-cut Benders decomposition.

Chapter 4 presents an approach to optimize inventory policies in process networks under uncertainty. In this chapter, the multiperiod inventory planning problem is formulated through a stochastic programming model that includes the logic of inventory policies. The proposed logic-based formulation is an approximation of the multistage stochastic programming model that replaces non-anticipativity constraints with decision rules that are shared across all scenarios. The logic-based formulation yields the parameters specifying the optimal inventory planning strategy in chemical process networks. We propose policies for inventory planning in process networks with arrangements of inventories in parallel and in series. The implementation of the logic-based planning strategy in Monte Carlo simulations shows significant advantages in comparison to the equivalent two-stage stochastic programming model.

In Chapter 5, we formulate the capacity planning problem as a bilevel optimization with the goal of modeling the hierarchical decision structure involving industrial producers and consumers. The formulation is a mixed-integer bilevel linear program in which the upper level maximizes the profit of a producer and the lower level minimizes the cost paid by markets. The upper-level problem includes mixed-integer variables that establish the expansion plan; the lower level problem is an LP that decides demands assignments. We reformulate the bilevel optimization as a single-level problem using two different approaches: KKT reformulation and duality-based reformulation. We analyze the performance of the reformulations and compare their results with the expansion plans obtained from the traditional single-level model. We also propose improvements on the duality-based reformulation for the solution of large-scale problems.

Chapter 6 extends the capacity planning model presented in Chapter 5 to consider competitors that are allowed to expand their capacity. We propose a mathematical programming formulation with three levels of decision-makers to fully capture the dynamic of competitive markets. The trilevel model is transformed into a bilevel optimization problem with mixed-integer variables in both levels by replacing the third-level LP with its optimality conditions. We introduce new definitions required for the analysis of degeneracy in multilevel models, and develop two novel algorithms to solve these challenging problems. The computational experiments show the value of considering competitors and markets as rational decision-makers.

Finally, Chapter 7 provides a critical review of this thesis. We present the major contributions of our research and we outline directions for future research.

Chapter 2

Design of Resilient Supply Chains with Risk of Disruptions

2.1 Motivation

Supply chain resilience has recently become one of the main concerns for major companies. The increasing complexity and interdependency of logistic networks have contributed to enhance the interest on this topic. A recent report presented by the World Economic Forum indicates that supply chain disruptions reduce the share price of impacted companies by 7% on average [15]. One interesting case of supply chain resilience happened in 2000 when a fire at the Philips microchip plant in Albuquerque (NM) cut off the supply of a key component for cellphone manufacturers Nokia and Ericsson. Nokia's production lines were able to adapt quickly by using alternative suppliers and accepting similar components. In contrast, the supply disruption had a significant impact in Ericsson's production, causing an estimated revenue loss of $400 million [137]. Similarly, the disruptions caused by hurricane Katrina in 2005 [169] and the earthquake that hit Japan in 2011 [134] exposed the vulnerabilities of centralized supply chain strategies in the process industry.

The importance of building resilient supply chain networks and quantifying the effect of unexpected events in their operation has been recognized by several studies [108, 185, 36, 45]. They advocate for the inclusion of risk reduction strategies into the supply chain design. However, disruptions are often neglected from the supply chain analysis because of their unpredictable and infrequent nature.

Disruptions comprise a wide variety of events that prevent supply chains from their normal operation. Regardless of their nature, disruptions produce undesirable effects: they shut down parts of the network and force rearrangements of the logistic strategy that can be very expensive. Furthermore, the current paradigm of lean inventory management leads to reduced supply chain flexibility and increased vulnerability to disruptions. In order to implement reliable networks that consistently deliver high performance, the value of supply chain resilience must be considered during their design [230, 200].

Traditionally, the mathematical formulation of the supply chain design has been based on the facility location problem (FLP) [242, 84]. The FLP implies selecting among a set of candidate locations the facilities that offer the best balance between investment and transportation cost to a given set of demand points. The supply chain design problem has a broader scope. It also includes the role of suppliers, inventory management, and timing of deliveries.

This chapter addresses the design of multi-commodity supply chains subject to disruptions risk at the distribution centers (DCs). The problem involves selecting DC locations, establishing their storage capacity, and determining a distribution strategy that anticipates potential disruptions. The goal is to obtain the supply chain with minimum cost from a risk neutral perspective. The cost of the supply chain is calculated as the sum of investment cost and expected distribution cost over a finite time horizon.

The benefits of flexibility in capacitated manufacturing networks with uncertain demand have been recognized in previous research studies [121]. Similar benefits can be expected in distribution networks with disruptions, but their assessment requires the consideration of capacity constraints. Unlike previous work, this research considers DC storage capacities as design variables that impact investment cost and inventory availability. This approach follows from the intuitive notion that supply chain resilience requires backup capacity. The goal is to demonstrate that significant increases in network reliability can be obtained with reasonable increases in investment cost through appropriate capacity selection and allocation of inventories.

In order to establish the optimal amount of inventories at DCs, demand assignments under the possible realizations of disruptions must be anticipated. Therefore, the problem is formulated in the context of two-stage stochastic programming with full recourse [19]. The first-stage decisions comprise the supply chain design: DC selection and their capacities. The second-stage decisions model the distribution strategy in the scenarios given by the potential combinations of active and disrupted locations. The solution of large-scale problems requires the development of specialized algorithms given the exponential growth in the number of scenarios with the increase in candidate DCs. We present different versions of Benders decomposition [13] that exploit the structure of the problem.

The remaining of this chapter is organized as follows. Section 2.2 reviews the relevant contributions to the design of resilient supply chains. Section 2.3 formalizes the problem statement. In Section 2.4, we describe the mathematical model of the problem. Section 2.5 illustrates the model with a small example. In section 2.6, we present the solution method for the design of large-scale resilient supply chains. Section 2.7 discusses some issues related to the implementation. Section 2.8 demonstrates the implementation of the solution strategy in a large-scale example. In Section 2.9, we formulate the design problem for a resilient supply chain from the process industry and present its results. Finally, we present a summary of this chapter in Section 2.10.

2.2 Literature review

Facility location problems have received significant attention since the theory of the location of industries was introduced by Weber & Friedrich [242]. In the context of supply chains, Geoffrion & Graves [84] proposed a Mixed-Integer Linear Programming (MILP) formulation that contains the essence of subsequent developments. Several authors have continued proposing different versions of this formulation. Owen & Daskin [172], Meixell & Gargeya [155], and Shen [207] offer comprehensive reviews on facility location and supply chain design. The main developments in supply chains design and planning for the process industry are reviewed by Shah [203] and by Laínez & Puigjaner [136]. A review of the FLP under uncertainty is presented by Snyder [213]. Additionally, the design of robust supply chains under uncertainty is reviewed by Klibi et al. [133].

Most recent efforts have included inventory management under demand uncertainty into the design of supply chains [232, 49, 208, 249]. These formulations exploit the variance reduction that is achieved when uncertain demands are centralized at few DCs, according to the risk pooling effect demonstrated by Eppen [58]. The benefits of centralization contrast with the risk diversification effect that becomes apparent when supply availability is considered uncertain. Snyder & Shen [215] demonstrate that centralized supply chains are more vulnerable to the effect of supply uncertainty.

The effect of unreliable supply in inventory management has been studied by several authors [230, 175, 14, 181]. Qi et al. [182] integrate inventory decisions into the supply chain design with unreliable supply. The main approach to address uncertainty in supply availability is to allocate safety stock at DCs to mitigate the risk of stockouts.

The FLP under the risk of disruptions was originally studied by Snyder & Daskin [214]. They formulate a problem in which all candidate DCs have unlimited capacities and the same disruption probability. The model avoids the generation of scenarios by establishing customer assignments according to DC availability and levels of preference. The objective is to minimize the investment cost in DCs and the expected cost of transportation. Similar formulations that allow site-dependent disruption probabilities have also been developed [46, 146, 209], together with approximation algorithms to solve them [144]. An extension that allows facility fortification decisions to improve their reliability was introduced by Li et al. [143]. An alternative design criterion (*p-robustness*) that minimizes nominal cost and reduces the risk of disruptions was presented by Peng et al. [176].

Recently, inventory management has been considered in the design of supply chains with risk of facility disruptions. Chen et al. [33] include the expected cost of holding inventory into the FLP. This formulation, like all previous work, considers the capacity of the candidate DCs to be unlimited. A capacitated version of the FLP with disruptions that includes inventory management is formulated by Jeon [119] as a two-stage stochastic programming problem. This formulation considers a fixed capacity for the candidate DCs.

Stochastic programming has been used to address different types of uncertainty in supply chain design. Tsiakis et al. [232] address the design of multi-echelon supply chains under demand uncertainty using stochastic programming. Salema et al. [197] propose a stochastic programming formulation for the design of reverse logistic networks with capacitated facilities. Some authors have resorted to Sample Average Approximation (SAA) [206, 130] to estimate the optimal design of supply chains with large numbers of scenarios. Santoso et al. [199] propose the use of SAA to estimate the optimal design of supply chains with uncertainty in costs, supply, capacity, and demand. Schültz et al. [201] distinguish between short and long-term uncertainty in their stochastic programming formulation; the problem is solved by using SAA. Klibi & Martel [131] propose various models for the design of resilient supply chains considering disruptions and other types of uncertainties. Their formulation approximates the optimal response strategy to disruptions; the solution of the supply chain design problem is estimated using SAA.

The main contribution of this research for the design of resilient supply chains in comparison to the published literature is to include DC capacities as design decisions. This extension allows detailed modeling of the inventory management, its availability and cost. Additionally, the solution strategy that we develop can be used to obtain deterministic bounds on the optimal solution of large-scale supply chains.

2.3 Problem statement

The proposed supply chain design problem involves selecting DCs among a set of candidate locations, determining their storage capacity for multiple commodities, and establishing the distribution strategy. The objective is to minimize the sum of investment costs and expected distribution cost; distribution costs include transportation from plant to DCs, storage of inventory at DCs, transportation from DCs to customers, and penalties for unsatisfied demands. These costs are incurred during a finite time horizon that is modeled as a sequence of time periods.

The DC candidate locations are assumed to have an associated risk of disruption. The risk is characterized by a probability that represents the fraction of time that the potential DC is expected to be disrupted. Disruption probabilities of individual candidate locations are assumed to be known. For the potential DC locations, the possible combinations of active and disrupted locations give rise to a discrete set of scenarios regardless of the investment decisions. The scenario probabilities are established during the problem formulation according to the probability of individual facility disruptions, which are assumed to be independent. However, the formulation easily accommodates correlation among disruption probabilities and more sophisticated approaches for the scenario generation [132].

The scenarios determine the potential availability of DCs. Actual availability depends on the realization of scenarios and the investment decisions. This property can be interpreted as an expression of endogenous uncertainty [120, 88, 99] in which the selection of DC locations renders some of the scenarios indistinguishable. Fortunately, for the case of two-stage stochastic programs, the optimal cost of indistinguishable scenarios always turns out to be the same. In contrast to multistage stochastic programming formulations [88], two-stage problems do not require conditional non-anticipativity constraints because there are no decisions to anticipate after the second stage.

The distribution strategy implies establishing demand assignments in all possible scenarios. Assignments are modeled with continuous variables to allow customers to be served from different DCs simultaneously. Customer demands must be satisfied from active DCs according to the availability of inventory. Unsatisfied demands are subject to penalty costs. The expected cost of distribution is calculated from the distribution cost of each scenario according to its associated probability.

DCs are assumed to follow a periodic review basestock inventory policy with zero lead time [212]. With this policy, DCs place a replenishment order at the beginning of every time-period; the size of the order is adjusted to bring the inventory to the base-stock level. Therefore, the inventory at DCs is always found at the base-stock level at the beginning of time periods. This policy implies that consecutive time-periods are identical and the distribution decisions are time independent. The optimal base-stock level for each DC is equal to its storage capacity, which is an optimization variable. All cost coefficients are assumed to be known and deterministic. The investment costs in DCs are given by a linear function of capacity with fixed-charges. Transportation costs are given by linear functions of volume without fixed-charges. Storage costs are given by a linear function of the mean inventory. Penalties for unsatisfied demand are given by a linear function of volume.

2.4 Mathematical model

The design of a supply chain with risk of disruptions has the structure of a two-stage stochastic program. First-stage decisions are related to the DCs location and their capacity. Second stage decisions involve assignments of customer demands according to the availability of DCs that is determined by the discrete set of scenarios. The penalties for unsatisfied demand render the recourse of the problem to be complete. Penalties are considered in the model by including an additional DC with infinite capacity, zero investment cost, and zero probability of being disrupted. This fictitious DC is labeled with subindex $|I|$.

The following indices are used in the formulation: the set of scenarios is denoted by S, the set of candidate locations for DCs is denoted by I, the set of customers is denoted by J, and the set of commodities is denoted by K. The parameters of the problem are: the number of time-periods in the design horizon (N), the customer demands per time period ($D_{j,k}$), the unit holding cost of inventory per time period (H_k), the fixed investment cost for DCs (F_i), the unit capacity cost for commodities at DCs ($V_{i,k}$), the unit transportation cost from plant to DCs (A_i), the unit transportation cost from DCs to customers ($B_{i,j}$), the probability of scenarios (π_s), the maximum capacity of DCs (C^{max}), and the matrix indicating the availability of DCs in the scenarios ($T_{s,i}$). The binary variable deciding if a DC is selected at candidate location i is denoted by x_i, the storage capacity for commodity k at location i is denoted by $c_{i,k}$, and $y_{s,i,j,k}$ represents the fraction of demand $D_{j,k}$ satisfied from location i in scenario s.

The objective function (2.1) minimizes the sum of investment at DCs, the expected cost of transportation from plant to DCs, the expected cost of transportation from DCs to customers, and the expected cost of storage at DCs. It should be noted that the cost of penalties is considered in the transportation terms indexed by $|I|$ and that all time-periods (N) are assumed to be identical.

$$\begin{aligned}\min \quad & \sum_{i\in I\setminus|I|}\left(F_i x_i + \sum_{k\in K} V_{i,k}c_{i,k}\right) + N\sum_{s\in S}\pi_s\sum_{i\in I}\sum_{k\in K}\left[\sum_{j\in J}\left(A_{i,k}+B_{i,j,k}\right)D_{j,k}y_{s,i,j,k}\right]\\ & + N\sum_{s\in S}\pi_s\sum_{i\in I}\sum_{k\in K}H_k\left(c_{i,k}-\frac{1}{2}\sum_{j\in J}D_{j,k}y_{s,i,j,k}\right) \qquad (2.1)\end{aligned}$$

The optimization problem is subject to the following constraints:

$$\begin{aligned}\text{s.t.} \quad & \sum_{i\in I} y_{s,i,j,k} = 1 && \forall\, s\in S,\ j\in J,\ k\in K \quad (2.2)\\ & \sum_{j\in J} D_{j,k}y_{s,i,j,k} - T_{s,i}c_{i,k} \le 0 && \forall\, s\in S,\ i\in I,\ k\in K \quad (2.3)\\ & c_{i,k} - C^{max}x_i \le 0 && \forall\, i\in I,\ k\in K \quad (2.4)\\ & x_i\in\{0,1\},\ \ 0\le c_{i,k}\le C^{max},\ \ 0\le y_{s,i,j,k}\le T_{s,i} && \forall\, s\in S,\ i\in I,\ j\in J,\ k\in K \quad (2.5)\end{aligned}$$

Constraints (2.2) determine demand assignments for each scenario. Constraints (2.3) ensure that customer assignments in every scenario are restricted by the inventory available at DCs; inventory availability at DCs depends on their capacity and the binary matrix ($T_{s,i}$) that indicates the realization of disruptions ($T_{s,i} = 0$) in the scenarios. Constraints (2.4) bound the storage capacity of DCs according to the selection of locations.

2.5 Illustrative example

We implement the proposed formulation to design a small supply chain with risk of disruptions at the candidate locations for DCs. We also find the optimal deterministic design based on the main-scenario (no disruptions), and we calculate its expected cost under the risk of disruptions. The implementations are based on the illustrative example presented by You & Grossmann [249]. The example includes 1 production plant, 3 candidate DCs, 6 customers, and a single commodity; there are 8 scenarios representing all possible combinations of disruptions at the DC candidate locations. The parameters for the example are shown in Tables 2.1 and 2.2. The availability matrix ($T_{s,i}$) and the scenario probabilities are shown in Table 2.3.

Parameter	**Value**	**Units**
N	365	days
D_1	95	ton/day
D_2	157	ton/day
D_3	46	ton/day
D_4	234	ton/day
D_5	75	ton/day
D_6	192	ton/day
H	0.01	\$/(ton day)
F	100,000	\$/DC
V	100	\$/ton
A_1	0.24	\$/ton
A_2	0.20	\$/ton
A_3	0.28	\$/ton
$A_{\|I\|}$	15	\$/ton

Table 2.1: Parameters of the illustrative example.

DC	**Customers**					
	1	**2**	**3**	**4**	**5**	**6**
1	0.04	0.08	0.36	0.88	1.52	3.36
2	2.00	1.36	0.08	0.10	1.80	2.28
3	2.88	1.32	1.04	0.52	0.12	0.08
$\|I\|$	10.0	10.0	10.0	10.0	10.0	10.0

Table 2.2: Transportation costs $B_{i,j}$ (\$/ton) in the illustrative example.

Scenario	**DC availability**				**Probability** π_s
	1	**2**	**3**	$\lvert I \rvert$	
1	1	1	1	1	0.795
2	0	1	1	1	0.069
3	1	0	1	1	0.033
4	1	1	0	1	0.088
5	0	0	1	1	0.003
6	1	0	0	1	0.004
7	0	1	0	1	0.008
8	0	0	0	1	$3.200 * 10^{-4}$

Table 2.3: Availability matrix ($T_{s,i}$) and scenario probabilities (π_s) in the illustrative example.

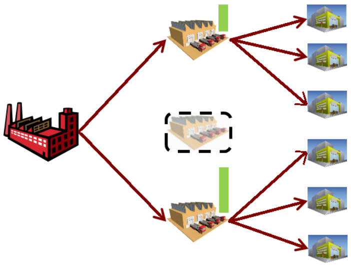


Figure 2.1: Optimal deterministic design for the illustrative example.

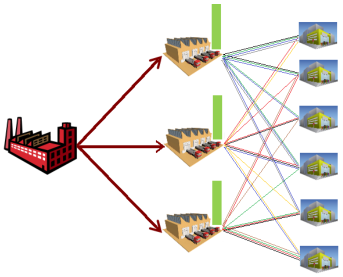


Figure 2.2: Optimal stochastic design for the illustrative example.

The optimal designs obtained are presented in Figures 2.1 and 2.2. It can be observed that the deterministic and stochastic models yield different designs. The deterministic design only selects two DC candidate locations, whereas the stochastic design selects all three candidate locations.

A detailed comparison of the deterministic and stochastic formulations and their corresponding results can be found in Table 2.4. The expected costs under the risk of disruptions are calculated by fixing the design variables to the optimal values obtained from each formulation and minimizing the distribution cost over the set of scenarios. Table 2.4 shows that the stochastic design requires a significantly higher investment cost. The investment is compensated by lower transportation cost and most importantly by lower penalties. The deterministic design has a very poor performance in the scenarios with disruptions. This is caused by its lack of flexibility: it has no slack capacity to serve demands when disruptions occur. On the other hand, the stochastic design has enough slack capacity to reallocate demands in the scenarios with disruptions; this strategy greatly decreases the expected cost of penalties. The comparison of the optimal costs obtained from both designs shows

		Deterministic formulation	**Stochastic formulation**
Expected costs under risk of disruptions	Investment ($)	279,900	419,850
	Transportation to DCs ($)	70,098	68,971
	Transportation to customers ($)	59,029	54,683
	Storage ($)	1,593	2,927
	Penalties ($)	674,703	54,244
	Total ($):	1,085,323	600,675
Storage capacity	Working inventory (ton)	298 / - / 501	400 / 400 / 400
Computational statistics	Problem type	MILP	MILP
	No. of constraints	13	76
	No. of continuous variables	31	199
	No. of binary variables	3	3
	Solution time	0.058 s	0.127 s

Table 2.4: Results of the illustrative example.

a difference of $484,648 when their performance is evaluated under the risk of disruptions; this comparative performance measure is the *Value of the Stochastic Solution* (VSS) [19]. The size and complexity of the formulations are also quite different. The number of variables and constraints grow linearly with the number of scenarios. The size of the formulations has an impact on the solution times. However, both formulations are linear and they only have a few binary variables. Therefore, the problems can be solved in short CPU times.

2.6 Solution method

The main challenge for the design of large-scale supply chains is posed by the number of scenarios; the possible combinations of disruptions grow exponentially with the number of candidate DCs. The total number of scenarios for our formulation is $2^{|I|-1}$, considering the fictitious DC that is always available. In this context, problems with a modest number candidate DCs become intractable. In order to design large-scale supply chains, we develop a number of different solution strategies.

Initially, we include in the model a set of redundant constraints that facilitates the solution of the Mixed-Integer Linear Programming (MILP) problem; this set of tightening constraints is intended to improve the Linear Programing (LP) relaxation of the formulation. Additionally, we present a Benders decomposition algorithm that leverages problem structure. Finally, we develop a strategy to bound the cost of arbitrary subsets of scenarios. This strategy is useful to evaluate the relevance of scenario sets and quantify their worst-case impact in the objective function.

2.6.1 Tightening the formulation

The proposed formulation for resilient supply chain design has a poor LP relaxation. For instance, the LP relaxation of the illustrative example presented in the previous section yields a lower bound of $420,525, whereas the optimal MILP solution is $600,675. The computational effort required to solve MILP problems strongly depends on the tightness of the LP relaxation. In particular, large-scale MILPs with poor LP relaxation can take quite a long time since a large number of nodes has to be analyzed with the state-of-the-art branch-and-cut algorithms. In order to improve the LP relaxation of our formulation, we add a new set of constraints. In fact, Proposition 2.1 states that by adding the tightening constraints, we can obtain the convex-hull of a subset of the constraints.

Proposition 2.1. *The convex hull of constraints* (2.3), (2.4)*, and* (2.5) *is obtained by adding the constraint presented in Equation* (2.6).

$$y_{s,i,j,k} - T_{s,i}x_i \leq 0 \qquad \forall\ s \in S,\ \ i \in I,\ \ j \in J,\ \ k \in K \tag{2.6}$$

Proof. According to the argument from Geoffrion & McBride [85] and decomposing the problem by DCs, constraints (2.3), (2.4), and (2.5) can be expressed in disjunctive form as follows:

$$\begin{bmatrix} x_i = 0 \\ c_{i,k} = 0 \\ y_{s,i,j,k} = 0 \end{bmatrix} \vee \begin{bmatrix} x_i = 1 \\ 0 \leq c_{i,k} \leq C^{max} \\ 0 \leq y_{s,i,j,k} \leq T_{s,i} \\ \sum\limits_{j \in J} D_{j,k} y_{s,i,j,k} \leq T_{s,i} c_{i,k} \end{bmatrix} \tag{2.7}$$

The hull reformulation is obtained by disaggregating variables x_i, $c_{i,k}$, and $y_{s,i,j,k}$ to obtain the following constraints:

$$\begin{aligned} & x_i^1 = 0 && x_i^2 = 1 \\ & c_{i,k}^1 = 0 && 0 \leq c_{i,k}^2 \leq C^{max} \\ & y_{s,i,j,k}^1 = 0 && 0 \leq y_{s,i,j,k}^2 \leq T_{s,i} \\ & && \sum_{j \in J} D_{j,k} y_{s,i,j,k}^2 \leq T_{s,i} c_{i,k}^2 \end{aligned} \tag{2.8}$$

The convex-hull is obtained from the convex combination of the disaggregated variables:

$$\begin{aligned} & x_i = (1-\alpha)\, x_i^1 + \alpha x_i^2 \\ & c_{i,k} = (1-\alpha)\, c_{i,k}^1 + \alpha c_{i,k}^2 \\ & y_{s,i,j,k} = (1-\alpha)\, y_{s,i,j,k}^1 + \alpha y_{s,i,j,k}^2 \\ & 0 \leq \alpha \leq 1 \end{aligned} \tag{2.9}$$

Fixing values of $x_i^1 = 0$, $c_{i,k}^1 = 0$, and $y_{s,i,j,k}^1 = 0$ yields:

$$\begin{aligned} & x_i = \alpha \\ & c_{i,k} = x_j c_{i,k}^2 \\ & y_{s,i,j,k} = x_i y_{s,i,j,k}^2 \end{aligned} \tag{2.10}$$

Substitution in the disaggregated constraints (2.8) yields:

$$0 \leq c_{i,k} \leq C^{max} x_i \tag{2.11}$$

$$0 \leq y_{s,i,j,k} \leq T_{s,i} x_i \tag{2.12}$$

$$\sum_{j \in J} D_{j,k} y_{s,i,j,k} \leq T_{i,k} c_{i,k} \tag{2.13}$$

$$0 \leq x_i \leq 1 \tag{2.14}$$

Constraints (2.11)-(2.14) correspond exactly to constraints (2.3), (2.4), (2.6), and the continuous relaxation of Equation (2.5). □

This MILP reformulation is known to yield the convex hull of the disjunctions [6, 183]. We illustrate the improvement in the tightness of the LP relaxation with the illustrative example presented in Section 2.5. The addition of the set of tightening constraints (2.6) to the formulation increases the lower bound of the LP relaxation from \$420,525 to \$589,403; this represents a significant improvement in a problem in which the optimal solution is \$600,675.

The addition of tightening constraints is important not only for a better LP relaxation of the full problem. The main benefit of this new set of constraints for our solution method is that it allow us generating stronger cuts in the implementation of Benders decomposition [152].

2.6.2 Multi-cut Benders decomposition

Benders decomposition, also known as the L-Shaped method for stochastic programming [238], is used to avoid the need of solving extremely large problems. This decomposition method finds the optimal value of the objective function by iteratively improving upper and lower bounds on the optimal cost. Upper bounds are found by fixing the first-stage variables and optimizing the second-stage decisions for the scenarios. Lower bounds are found in a master problem that approximates the cost of scenarios in the space of the first-stage variables. The convergence of the algorithm is achieved by improving the lower bounding approximation used in the master problem with the information obtained from the upper bounding subproblems.

The flow of information from subproblems to the master problem is determined by the dual multipliers of the subproblems. The classical implementation generates one cut in every iteration, but some authors have proposed multi-cut implementations [18, 191, 249]. Given the structure of the resilient supply chain design problem, there are several possibilities to derive cuts. After different computational experiments, we found that the most efficient strategy is to transfer as much information as possible from the subproblems to the master problem. Therefore, our implementation adds individual cuts per scenario and per commodity in every iteration.

In the multi-cut framework, the subproblems that can be decomposed by scenario ($s \in S$) and commodity ($k \in K$), are formulated according to Equations (2.15)-(2.19),

$$\min \quad N\sum_{s\in S}\pi_s\left\{\sum_{i\in I}\sum_{k\in K}\left[\sum_{j\in J}\left(A_{i,k}+B_{i,j,k}\right)D_{j,k}y_{s,i,j,k}-\frac{H_k}{2}\sum_{j\in J}D_{j,k}y_{s,i,j,k}\right]\right\} \tag{2.15}$$

$$\text{s.t.} \quad \sum_{i\in I}y_{s,i,j,k}=1 \qquad \forall\ s\in S,\ \ j\in J,\ \ k\in K \tag{2.16}$$

$$\sum_{j\in J}D_{j,k}y_{s,i,j,k}-T_{s,i}\bar{c}_{i,k}\le 0 \qquad \forall\ s\in S,\ \ i\in I,\ \ k\in K \tag{2.17}$$

$$y_{s,i,j,k}-T_{s,i}\bar{x}_i\le 0 \qquad \forall\ s\in S,\ \ i\in I,\ \ j\in J,\ \ k\in K \tag{2.18}$$

$$y_{s,i,j,k}\ge 0 \qquad \forall\ s\in S,\ \ i\in I,\ \ j\in J,\ \ k\in K \tag{2.19}$$

where $\bar{x}_i$ and $\bar{c}_{i,k}$ are candidate first-stage solutions found in the master problem.

The formulation of the multi-cut master problem is given by Equations (2.20)-(2.23),

$$\min \quad \sum_{i\in I\setminus|I|}\left(F_ix_i+\sum_{k\in K}V_{i,k}c_{i,k}\right)+N\sum_{i\in I\setminus|I|}\sum_{k\in K}H_kc_{i,k}+\sum_{s\in S}\sum_{k\in K}\theta_{s,k} \tag{2.20}$$

$$\text{s.t.} \quad \theta_{s,k}\ge\sum_{j\in J}\lambda^{iter}_{s,j,k}-\sum_{i\in I}T_{s,i}\mu^{iter}_{s,i,k}c_{i,k}-\sum_{i\in I}\sum_{j\in J}T_{s,i}\gamma^{iter}_{s,i,j,k}x_i \qquad \forall\ s\in S,\ \ k\in K \tag{2.21}$$

$$c_{i,k}-C^{max}x_i\le 0 \qquad \forall\ i\in I,\ \ k\in K \tag{2.22}$$

$$x_i\in\{0,1\}\,,\quad 0\le c_{i,k}\le C^{max} \qquad \forall\ s\in S,\ \ i\in I,\ \ j\in J,\ \ k\in K \tag{2.23}$$

where $\lambda^{iter}_{s,j,k}$, $\mu^{iter}_{s,i,k}$, and $\gamma^{iter}_{s,i,j,k}$ are the optimal multipliers associated with constraints (2.16), (2.17), and (2.18), in iteration $iter$. Constraint (2.21) provides the lower bounding approximation ($\theta_{s,k}$) for the cost of satisfying demands of commodity k in scenario s. It should be noted that no feasibility cuts are considered since the problem has complete recourse.

2.6.3 Strengthening the Benders master problem

The multi-cut approach for Benders decomposition can be very effective to obtain a good approximation of the feasible region in the master problem. However, depending on the number of scenarios and commodities in the instance, the master problem can become a hard MILP because of the large number of cuts. In order to improve the lower bounds and guide the selection of first-stage variables, the decisions of the main scenario (scenario with no disruptions) can be included in the master problem. This formulation of the master problem leverages the significant impact of the main scenario in the final design given its comparatively high probability. The increase in the size of the master problem when main-scenario decisions are included is modest for problems with a large number of scenarios. The strengthened master problem minimizes the objective function (2.20) subject to constraints (2.21), (2.22), and (2.23) from the original master problem, and constraints (2.16), (2.17), (2.18), and (2.19) corresponding to the main-scenario.

The constraints of the main-scenario subproblem are connected to the objective function (2.20) through constraint (2.24).

$$\theta_{1,k} \geq N\pi_1 \left\{ \sum_{i\in I} \sum_{k\in K} \left[\sum_{j\in J} (A_{i,k} + B_{i,j,k})\, D_{j,k} y_{1,i,j,k} - \frac{H_k}{2} \sum_{j\in J} D_{j,k} y_{1,i,j,k} \right] \right\} \tag{2.24}$$

2.6.4 Pareto-optimal cuts

The Benders subproblems that result from fixing the first-stage decisions are classical transportation problems. These problems are relatively easy to solve but their dual solution is known to be highly degenerate [244]. Therefore, it is very important to select in every iteration a set of optimal multipliers ($\lambda_{s,j,k}^{iter}$, $\mu_{s,i,k}^{iter}$, $\gamma_{s,i,j,k}^{iter}$) that produce strong Benders cuts. According to Magnanti & Wong [152], the best multipliers for the implementation of Benders decomposition are those that produce non-dominated cuts among the set of optimal multiplies. These cuts are said to be pareto optimal. Pareto-optimal cuts produce the smallest deviation in the dual objective function value when evaluated at a point (x_i^0,$c_{i,k}^0$) in the relative interior of the convex hull of the first-stage variables. Such cuts can be obtained by solving the LP problem presented in Equations (2.25)-(2.29),

$$\max \sum_{s\in S} \sum_{k\in K} \left(\sum_{j\in J} \lambda_{s,j,k} - \sum_{i\in I} T_{s,i} c_{i,k}^0 \mu_{s,i,k} - \sum_{i\in I} \sum_{j\in J} T_{s,i} x_i^0 \gamma_{s,i,j,k} \right) \tag{2.25}$$

$$\text{s.t. } v^*(\bar{x}_i^{iter}, \bar{c}_{i,k}^{iter}) = \sum_{s\in S} \sum_{k\in K} \left(\sum_{j\in J} \lambda_{s,j,k} - \sum_{i\in I} T_{s,i} \bar{c}_{i,k}^{iter} \mu_{s,i,k} - \sum_{i\in I} \sum_{j\in J} T_{s,i} \bar{x}_i^{iter} \gamma_{s,i,j,k} \right) \tag{2.26}$$

$$\begin{aligned}&\lambda_{s,j,k} - D_{j,k}\mu_{s,i,k} - \gamma_{s,i,j,k} \\ &\qquad \leq N\pi_s \left(A_{i,k} + B_{i,j,k} - \tfrac{H_k}{2}\right) D_{j,k} \end{aligned} \qquad \forall\, s \in S,\ i < |I|,\ j \in J,\ k \in K \tag{2.27}$$

$$\lambda_{s,j,k} \leq N\pi_s \left(A_{|I|,k} + B_{|I|,j,k}\right) D_{j,k} \qquad \forall\, s \in S,\ i = \{|I|\}\,,\ j \in J,\ k \in K \tag{2.28}$$

$$\lambda_{s,j,k} \geq 0,\ \ \mu_{s,i,k} \geq 0,\ \ \gamma_{s,i,j,k} \geq 0 \qquad \forall\, s \in S,\ i \in I,\ j \in J,\ k \in K \tag{2.29}$$

where $v^*(\bar{x}_i^{iter}, \bar{c}_{i,k}^{iter})$ is the optimal objective cost of the subproblems at iteration $iter$, and the point (x_i^0,$c_{i,k}^0$) satisfies Equation (2.30).

$$\left(x_i^0, c_{i,k}^0\right) \in \{(x_i, c_{i,k}) :\ 0 < x_i < 1;\ 0 < c_{i,k} < C^{max} x_i\} \tag{2.30}$$

It is worth noticing that Equation (2.26) constrains the multipliers to the set of optimizers of the subproblems; inequalities (2.27), (2.28), and (2.29) are the constraints of the dual formulation of the Benders subproblems.

2.6.5 Bounding the impact of scenario subsets

An important observation regarding the problem structure refers to the order of magnitude among different scenario probabilities. Scenarios with increasing number of disrupted locations have smaller probabilities, but scenarios with the same number of disruptions occurring simultaneously have probabilities on the same order of magnitude. Therefore, the most intuitive way to divide the scenarios is to group them according to the number of simultaneous disruptions.

For problems with a large number of scenarios, it is reasonable to select a subset of relevant scenarios ($\hat{S}$) for which the optimization problem can be solved, neglecting the effect of the scenarios with very small probabilities. However, solving this reduced problem does not provide much information about the optimal value of the objective function for the cases in which the cost of penalties is very high. Therefore, it is of interest to derive deterministic bounds on the cost of the neglected scenarios.

The calculation of the upper bound for the subset of neglected scenarios ($\tilde{S}$) is based on the implementation of an assignment policy that is always feasible. The proposed policy works as follows. In any given scenario, the main-scenario assignment is attempted for each demand ($D_{j,k}$): if the assignment is feasible (because the corresponding DC is active) the cost of satisfying the demand

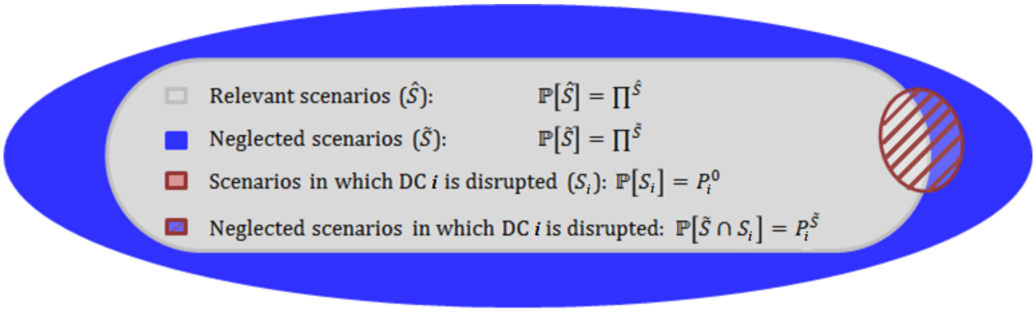

Relevant scenarios ($\hat{S}$): $\mathbb{P}[\hat{S}] = \Pi^{\hat{S}}$
Neglected scenarios ($\tilde{S}$): $\mathbb{P}[\tilde{S}] = \Pi^{\tilde{S}}$
Scenarios in which DC i is disrupted (S_i): $\mathbb{P}[S_i] = P_i^0$
Neglected scenarios in which DC i is disrupted: $\mathbb{P}[\tilde{S} \cap S_i] = P_i^{\tilde{S}}$

Figure 2.3: Subsets of scenarios according to disruptions.

equals its cost in the main-scenario, otherwise the demand is assumed to be penalized. The proportion in which these two costs are incurred depends on the conditional disruption probabilities of DCs ($P_i^{\tilde{S}}$) in the neglected scenarios. According to this policy, an upper bound for the cost of the neglected scenarios subset ($\tilde{S}$) can be calculated from equation (2.31).

$$\begin{aligned} UB^{\tilde{S}} = N\Pi^{\tilde{S}} \sum_{i \in I} \left(1 - P_i^{\tilde{S}}\right) \sum_{k \in K} \left\{ \sum_{j \in J} \left(A_{i,k} + B_{i,j,k}\right) D_{j,k} y_{1,i,j,k} + H_k c_{i,k} - \frac{H_k}{2} \sum_{j \in J} D_{j,k} y_{1,i,j,k} \right\} \\ + N\Pi^{\tilde{S}} \sum_{i \in I} P_i^{\tilde{S}} \sum_{k \in K} \left\{ \sum_{j \in J} \left(A_{|I|,k} + B_{|I|,j,k}\right) D_{j,k} y_{1,i,j,k} + H_k c_{i,k} \right\} \qquad (2.31) \end{aligned}$$

where $\Pi^{\tilde{S}} = \mathbb{P}(\tilde{S})$ is the probability of the subset of neglected scenarios $\tilde{S}$, $P_i^{\tilde{S}}$ is the conditional probability of disruption at DC i in subset of scenarios $\tilde{S}$, $y_{1,i,j,k}$ are the main-scenario assignments, and $(A_{|I|,k} + B_{|I|,j,k})$ determine the unit cost for unsatisfied demand. Therefore, the first term in (2.31) corresponds to the cost of the feasible main-scenario assignments in subset $\tilde{S}$ and the second term corresponds to penalties for infeasible assignments in subset $\tilde{S}$.

The calculation of the conditional probability of disruption in scenario subset $\tilde{S}$ is based on the assumption that disruptions at DCs are independent from each other. Figure 2.3 shows the scenario subsets and the relationship between their probabilities. Proposition 2.2 formalizes the procedure to calculate the conditional probability of disruption ($P_i^{\tilde{S}}$) in the subset of neglected scenarios.

Proposition 2.2. *The conditional probability, $\mathbb{P}(S_i|\tilde{S})$, of finding DC i disrupted in subset of scenarios $\tilde{S} \subset S$ can be calculated according to Equation* (2.32),

$$\mathbb{P}(S_i|\tilde{S}) = \frac{\mathbb{P}(S_i) - \mathbb{P}(\tilde{S}^C)\ \mathbb{P}(S_i|\tilde{S}^C)}{\mathbb{P}(\tilde{S})} \tag{2.32}$$

where $S_i \subset S$ denotes all scenarios in which DC i is disrupted, and $\tilde{S}^C$ is the complement of $\tilde{S}$.

Proof. By definition:

$$\mathbb{P}\left(S_i|\tilde{S}_i\right) = \frac{\mathbb{P}\left(S_i \cap \tilde{S}\right)}{\mathbb{P}\left(\tilde{S}\right)} \tag{2.33}$$

$$\mathbb{P}\left(S_i|\tilde{S}_i^C\right) = \frac{\mathbb{P}\left(S_i \cap \tilde{S}^C\right)}{\mathbb{P}\left(\tilde{S}^C\right)} \tag{2.34}$$

Since $\tilde{S}$ and $\tilde{S}^C$ are the complements of each other, then:

$$\mathbb{P}\left(S_i\right) = \mathbb{P}\left(\left(S_i \cap \tilde{S}\right) \cup \left(S_i \cap \tilde{S}^C\right)\right) \tag{2.35}$$

$$\mathbb{P}\left(S_i\right) = \mathbb{P}\left(\tilde{S}\right)\mathbb{P}\left(S_i|\tilde{S}\right) + \mathbb{P}\left(\tilde{S}^C\right)\mathbb{P}\left(S_i|\tilde{S}^C\right) \tag{2.36}$$

The proof is completed by noticing that Equation (2.36) is equivalent to Equation (2.32). □

Analogously, a lower bound on the cost of scenarios subset $\tilde{S}$ can be calculated by assuming that all demands can be satisfied from the DC assigned in the main-scenario as presented in Equation (2.37).

$$LB^{\tilde{S}} = N\Pi^{\tilde{S}} \sum_{i \in I} \sum_{k \in K} \left\{ \sum_{j \in J} \left(A_{i,k} + B_{i,j,k}\right) D_{j,k} y_{1,i,j,k} + H_k c_{i,k} - \frac{H_k}{2} \sum_{j \in J} D_{j,k} y_{1,i,j,k} \right\} \tag{2.37}$$

2.7 Implementation

The proposed solution method is implemented in GAMS 24.1.1 for a large-scale case study. All problems are solved using GUROBI 5.5.0 in an Intel Core i7 CPU (8 cores) 2.93 GHz with 4 GB of RAM. In order to speed-up the solution time, we leverage a number of problem specific properties.

- **Indistinguishability:** the upper bound for a particular design is evaluated in the Benders subproblems. Scenarios that are only different from each other because of disruptions at locations that are not selected ($\bar{x}_i^{iter} = 0$) become indistinguishable. All the scenarios in these sets have the same optimal solution. Therefore, it is enough to solve one of the indistinguishable scenarios and use the solution for all the scenarios in the set.

- **Parallelization:** the upper bounding subproblems are completely independent of each other with respect to scenarios and commodities. They can be solved in parallel using GAMS grid computing. The degree of parallelization must balance the time required to start the executions, solve the subproblems, and read the solutions. For our large-scale instance, the highest efficiency was found by solving for all commodities at the same time in individual scenarios.

- **Relevance of scenarios:** the total number of scenarios grows exponentially with the number of candidate DCs. If all scenarios are considered, it is impossible to find the optimal design of industrial supply chains with the current computational tools. However, most of the scenarios that can be generated have very small probabilities. The magnitude of the scenario probabilities are directly related to the number of disruptions occurring at the same time. Therefore, it is easy to identify a reduced subset of relevant scenarios whose optimal solution is a good approximation of the full-space solution.

- **Full-space bounds:** bounds on the cost of scenarios excluded from the optimization problem can be calculated from Equations (2.31) and (2.37). Upper and lower bounds on the full-space problem can be calculated by adding the bounds obtained for the relevant set of scenarios through Benders decomposition to the bounds obtained from Equations (2.31) and (2.37) for scenarios excluded from the optimization problem.

The sequence in which the proposed solution method is implemented is presented in Figure 2.4.

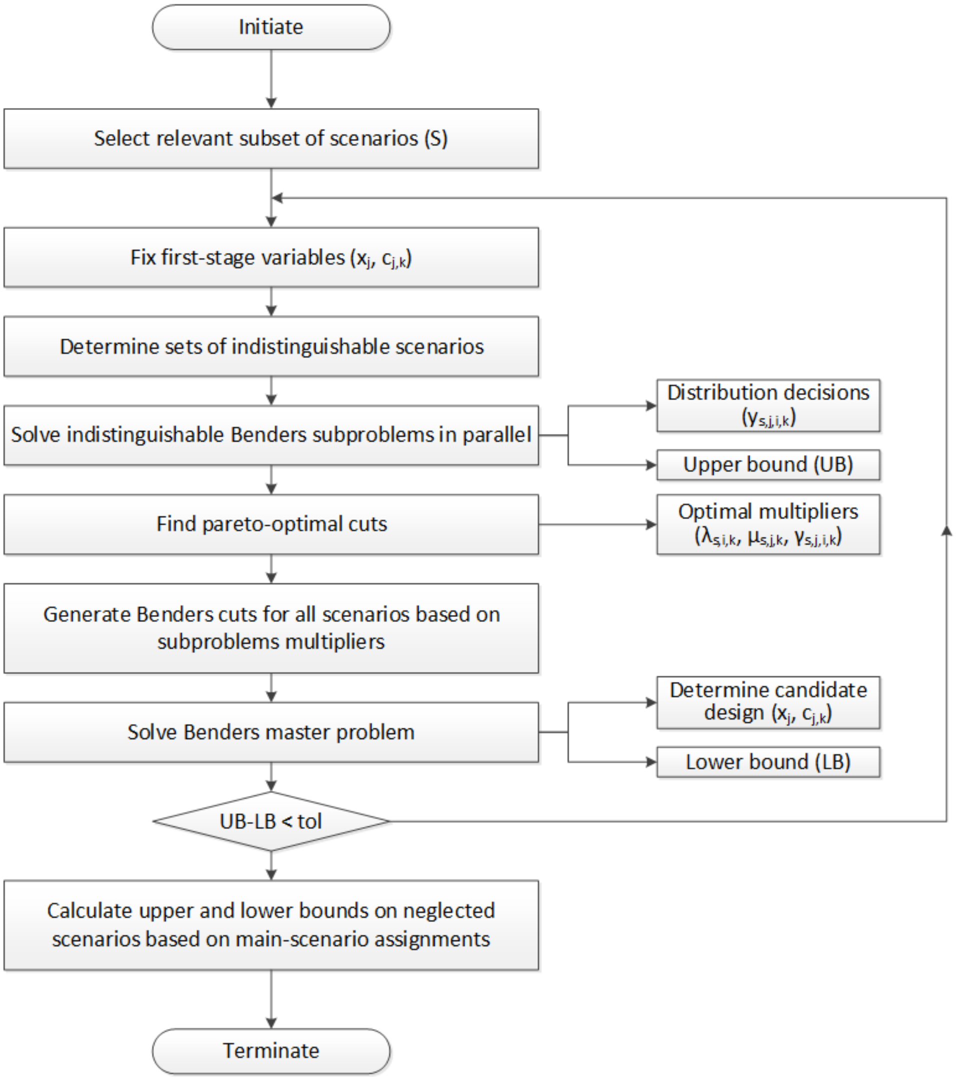

Initiate
Select relevant subset of scenarios (S)
Fix first-stage variables (x_j, $c_{j,k}$)
Determine sets of indistinguishable scenarios
Solve indistinguishable Benders subproblems in parallel
Distribution decisions ($y_{s,j,i,k}$)
Upper bound (UB)
Find pareto-optimal cuts
Optimal multipliers ($\lambda_{s,i,k}$, $\mu_{s,j,k}$, $\gamma_{s,j,i,k}$)
Generate Benders cuts for all scenarios based on subproblems multipliers
Solve Benders master problem
Determine candidate design (x_j, $c_{j,k}$)
Lower bound (LB)
UB-LB < tol
Calculate upper and lower bounds on neglected scenarios based on main-scenario assignments
Terminate

Figure 2.4: Implementation sequence of the solution method.

2.8 Large-scale example

We use the solution strategy developed in the previous sections to solve a large-scale supply chain design problem with risk of disruptions at candidate DC locations. The parameters of the problem were generated randomly; they are presented in Appendix A. The problem includes: 1 production plant, 9 candidate locations for DCs, and 30 customers with demands for 2 commodities. The candidate DCs have disruption probabilities between 2% and 10%. The number of scenarios in the full-space problem is (2^9) 512. The design is based on a time-horizon (N) of 365 days; in this time-scale, investment cost can be interpreted as an annualized cost.

The instance is used to illustrate the implementation of Benders decomposition, the benefits of strengthening the master problem, and the impact of solving a reduced subset of relevant scenarios. The selected relevant subset of scenarios includes scenarios with up to 4 simultaneous disruptions, for a total of 256 scenarios with probability ($\Pi^{\hat{S}}$) equal to 99.99%. A comparison of the results for the full-space problem and the reduced problem is shown in Table 2.5.

The results show that solutions obtained for the full-space and the reduced problem are very similar. In particular, their optimal solutions imply the same design decisions, and therefore the same investment cost. It is interesting to note that the largest difference in the results appears in the expected cost of penalties. This result is reasonable because the scenarios that are ignored in the reduced problem are expected to be expensive in terms of penalties. Both problems were solved to 0% optimality tolerance using the proposed Benders decomposition algorithm. The solution obtained for the full-space problem establishes with certainty the optimal design because all the

Expected cost	**Full-space instance**	**Reduced instance**
Investment ($):	2,194,100	2,194,100
Transportation to DCs ($):	936,260	936,238
Transportation to customers ($):	3,615,300	3,615,209
Storage ($):	319,440	319,429
Penalties ($):	160,347	159,615
Total ($):	7,225,447	7,224,591
Full-space upper bound:	7,225,447	7,225,898
Full-space lower bound:	7,225,447	7,224,728

Table 2.5: Expected costs of the large-scale example obtained from full-space and reduced instances.

Model statistic	**Full-space instance**	**Reduced instance**
Number of constraints:	318,479	159,247
Number of continuous variables:	309,263	154,639
Number of binary variables:	9	9
Number of multi-cut Benders iterations:	15	15
Multi-cut Benders solution time [s]:	281	151
Strengthened multi-cut Benders solution time [s]:	176	89
Number of strengthened multi-cut Benders iterations:	8	8

Table 2.6: Instance sizes and solution times for the large-scale example.

scenarios are included in the optimization problem. The solution obtained for the reduced problem establishes a lower bound on the full-space optimum because it neglects the effect of some scenarios. When the bounds on the full-space problem are calculated from equations (2.31) and (2.37), it can be observed that they yield a very tight approximation of the full-space solution with an optimality gap less than 0.1%.

The size of the optimization instances and their solution times are shown in Table 2.6. It can be observed that the reduced problem is almost half the size of the full-space problem in terms of constraints and continuous variables. This is explained by the reduction in the number of scenarios. The solution time for both instances decreases in a smaller proportion because the algorithm spends most of the time solving the MILP master problems. The use of the strengthened multi-cut master problem further reduces the solution time because it implies fewer iterations as shown in Figure 2.5. The solution times for the full-space and the reduced instances without any decomposition strategy using GUROBI 5.5.0 are 3,349 s and 1,684 s, respectively. The much smaller solution times presented in Table 2.6 demonstrate that the proposed methodology is effective to solve large-scale instances of high computational complexity.

2.9 Industrial supply chain design

We use the proposed model and solution method for the optimal design of an industrial supply chain with risk of disruption at candidate DCs. The problem includes: 1 production plant, 29 candidate locations for DCs, 110 customers, and 61 different commodities. Not all customers have demand

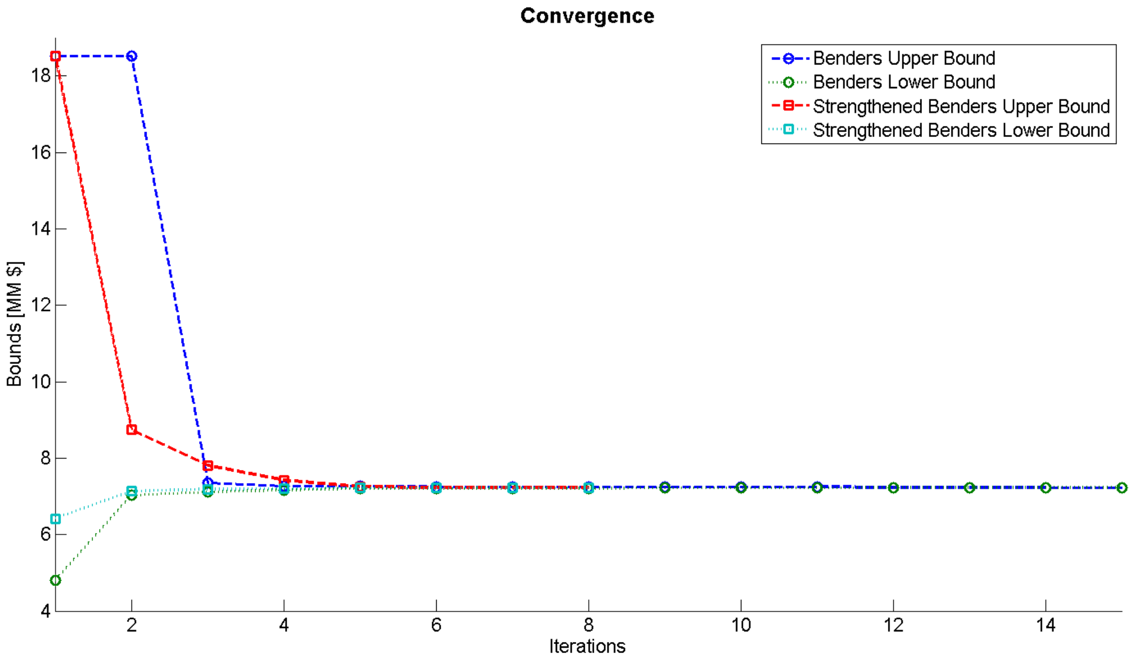


Figure 2.5: Convergence of Benders algorithms for the full-space instance of the large-scale example.

for all commodities; there are a total of 277 demands for commodities in every time-period. The DC candidate locations have independent probabilities of being disrupted between 0.5% and 3%. The number of scenarios in the full-space problem is 22^9, approximately 537 million. The magnitude of penalties for unsatisfied demand is around 10 times the highest distribution cost among all possible assignments. The design is based on a time-horizon (N) of 60 months. The specific data for this instance is not disclosed for confidentiality reasons.

Three subsets of scenarios are used for the design of the industrial supply chain. The first reduced problem only includes the main scenario, and is equivalent to the deterministic formulation of the supply chain design problem. The second reduced problem considers the main scenario and the scenarios with one disruption, giving rise to a subset of 30 scenarios. The third reduced problem includes the scenarios with up to 2 simultaneous disruptions, giving rise to a larger subset of 436 scenarios. Table 2.7 shows the problem sizes for the different instances. It can be observed that the third instance, in which the scenarios comprise 98.5% of the possible realizations, is a very challenging problem in terms of size.

The first instance was solved directly without decomposition. The second and third instances were solved using the strengthened multi-cut Benders decomposition. The results obtained are shown in Table 2.8.

From Table 2.8, we observe that the investment cost has a modest increase when the model includes a larger number of adverse scenarios (1, 30, 436). In this case, the formulation leverages the complexity of the supply chain network by decentralizing inventories at a relative low cost. This strategy avoids costly demand penalties and improves supply chain resilience. In contrast with the instance bounds, which are obtained only with subsets of scenarios, the full-space bounds obtained using Equations (2.31) and (2.37) show the importance of considering a relevant subset of scenarios that provides a good representation of the full-space problem. The design obtained from Reduced problem 0 can only guarantee a full-space expected cost of $57.41 million, that is 16.13% higher than the corresponding lower bound. On the other hand, the design obtained from Reduced problem 2 yields a full-space upper bound of $55.59 million that in the worst case is 3.5% higher than the optimal cost.

Maximum simultaneous disruptions	**Number of scenarios in subset**	**Probability of subset**	**Number of constraints**	**Number of continuous variables**	**Number of binary variables**
0	1	0.590	11,854	10,085	29
1	30	0.905	304,261	251,191	29
2	436	0.985	4,397,989	3,626,675	29

Table 2.7: Instances of the industrial supply chain with increasing number of maximum simultaneous disruptions.

Expected cost	**Reduced instance 0**	**Reduced instance 1**	**Reduced instance 2**
Optimal investment [M$]:	18.47	18.77	21.01
Number of selected DCs:	1	4	12
Instance upper bound [M$]:	34.09	48.68	53.66
Instance lower bound [M$]:	34.09	48.30	53.25
Instance optimality gap:	0%	0.78%	0.77%
Number of Benders iterations:	-	4	6
Solution time [min]:	0.1	84	1,762
Full-space upper bound [M$]:	57.41	56.31	55.59
Full-space lower bound [M$]:	48.15	52.87	53.67

Table 2.8: Upper and lower bounds for instances of the industrial supply chain example.

The computational effort to solve the larger instances of the industrial supply chain example is very significant. Even after applying the methodology developed in the previous sections, it takes a long time to find satisfactory solutions. In this example, the number of scenarios and commodities implies a large number of cuts in the Benders master problem. Therefore, the complexity of the MILP master problem increases very rapidly with iterations. Fortunately, the algorithm converges after few iterations.

2.10 Summary

We formulated the design of resilient supply chains as a two-stage stochastic programming problem to include the risk of disruptions at DCs. The model allows finding the design decisions that minimize investment and expected distribution cost over a finite time-horizon by anticipating the distribution strategy in the scenarios with disruptions. The allocation of inventory at DCs plays a critical role in supply chain resilience since it allows flexibility for the satisfaction of customers' demands in different scenarios. This strategy contradicts the trend to centralize distribution centers and reduce inventories. The examples show that resilient supply chain designs can be obtained with reasonable increases in investment costs. These increased investments are compensated by lower transportation costs and better performance in adverse scenarios.

The main challenge for the design of resilient large-scale supply chains originates from the exponential growth in the number of scenarios as a function of the number of DC candidate locations. We developed different strategies to exploit the structure of the problem. The importance of a tight MILP formulation was demonstrated in our examples. We adapted the multi-cut version of Benders decomposition to leverage the particular problem structure. In order to reduce the number of iterations, we generated pareto-optimal cuts that were added to the master problem for every commodity in each scenario. Additionally, we found that including the assignment decisions of the main scenario in the Benders master problem reduces the number of iterations and the computational time. For large-scale problems, the optimization over reduced number scenarios yielded good approximations of the optimal design. Furthermore, the implementation of a distribution policy in the scenarios with very small probabilities allowed finding deterministic bounds on the performance of the supply chain.

We used the proposed solution method to design a multi-commodity industrial supply chain. The results demonstrated the economic benefits of considering resilience in supply chain design. The implementation of resilient designs has a significant potential to improve supply chain performance and reduce their vulnerability to unexpected events.

Chapter 3

Implementation of a Novel Cross-decomposition Algorithm for Two-stage Stochastic Programming Investment Planning

3.1 Motivation

Two-stage stochastic programming investment planning problems can be difficult to solve because the formulation of their deterministic equivalent programs often leads to large-scale problems. There are three main approaches that to address the resulting computational challenge: sampling methods, scenario reduction techniques, and decomposition methods. Sampling methods [148] and scenario reduction techniques [105] are used to limit the number of scenarios in the formulation, in the attempt of obtaining a good approximation of the original problem. Decomposition schemes, such as the Benders decomposition [13, 83, 238] or Lagrangean decomposition [97, 29], aim at solving the exact problem by exploiting its decomposable structure.

Only a few research efforts have tried to combine the complementary strengths of Benders and Lagrangean decomposition is a single scheme. Originally proposed by Van Roy [237], the cross-decomposition algorithm is a framework that unifies the concepts behind Benders and Lagrangean decomposition. The original cross decomposition iterates between the Benders and Lagrangean subproblems, where each of them yields input for the other. One of the main motivations for the development of cross-decomposition algorithms was to avoid solving master problems since their solution, potentially Mixed-Integer Linear Programs (MILPs), is regarded as a difficult task. Some variations of the method, like mean value cross decomposition [110, 111], even eliminate the use of master problems completely at the cost of potentially slow convergence.

However, two paradigm shifts that influence our perception of cross decomposition have occurred over the last 20 years. First, the advance of solvers and computational resources now allows solving large MILP problems in reasonable CPU times; therefore, we no longer need to avoid solving master problems in a cross-decomposition scheme. Second, the growing grid computing infrastructure is leading to more parallelization; hence, it is desirable to develop novel algorithms that leverage the decomposable structure of stochastic programming problems.

This chapter presents an updated version of the cross-decomposition algorithm proposed by Mitra et al. [161] for linear two-stage stochastic investment planning problems with complete recourse; we focus on the implementation of the algorithm and the computational experiments testing its performance. In the framework of the two-stage stochastic program, we model investment decisions in the first stage with mixed-integer variables and operational decisions in the second stage with continuous variables. The cross-decomposition algorithm integrates Benders and Lagrangean decomposition by strengthening both master problems with cuts generated using dual information from the subproblems. The primal search is guided by a multi-cut Benders master problem, while the dual search is guided by the Lagrangean master problem. The motivation for this information exchange is to improve the bounds predicted by the original decomposition methods.

The cross-decomposition algorithm is implemented for the design of resilient supply chains, based on the model presented in Chapter 2. Several instances of the resilient supply chain design problem are used to test the cross-decomposition algorithm, and to compare its performance with commercial solvers and with multi-cut Benders decomposition. The results show the benefits of the cross-decomposition scheme for stochastic investment planning problems.

The remaining of this chapter is organized as follows. Section 3.2 reviews the published literature related to cross decomposition. In Section 3.3, we describe the mathematical model of the investment planning problem that we address. Section 3.4 presents the formulations used in the cross-decomposition scheme and the structure of the algorithm. In Section 3.4, we evaluate the performance of the cross-decomposition algorithm on several instances of the reliable supply chain design problem. Finally, in Section 3.6 we present the analysis of results.

3.2 Literature review

There is extensive literature on decomposition methods for stochastic programming problems, but few publications have focused on the implementation cross-decomposition algorithms since it was proposed by Van Roy [237]. Significant contributions to the original algorithm were developed by Holmberg [109], who generalized the ideas of cross decomposition and introduced a set of enhanced convergence tests. However, we have found only two publications combining Benders and Lagrangean decomposition in an approach similar to the algorithm that we present. The scheme developed by Cerisola et al. [30] uses dual information from a component-based Lagrangean relaxation in a nested Benders approach that is implemented for the unit commitment problem. Sohn et al. [218] implement a mean value cross-decomposition approach for two-stage stochastic programming problems based on the work by Holmberg [110]; their algorithm avoids solving master problems. Sohn et al. [218] apply their algorithm to a set of random instances, from which they claim to outperform Benders and ordinary cross decomposition in CPU time.

Benders decomposition, also known as the L-shaped method for stochastic programming, has found a wide range of applications in stochastic planning problems [64, 17, 63, 158, 199]. The L-shaped method was originally implemented by Van Slyke & Wets [238] to solve stochastic programming and optimal control problems. The algorithm iterates between a master problem that approximates the original problem in the space of the first-stage variables and subproblems that find feasible solutions after fixing the first-stage variables; its convergence rate usually has a strong dependence on the linear programming (LP) relaxation of the problem [152, 194]. Therefore, the bound initially provided by the master problem might be weak and a large number of iterations are potentially required. Several strategies have been developed to speed-up convergence of Benders algorithms; the most relevant include the multi-cut version of Benders decomposition developed by Birge & Louveaux [18], the pareto-optimal cuts presented by Magnanti & Wong [152], and cut bundle generation methods [191].

The Lagrangean decomposition approach developed by Guignard & Kim [97] has also been used to solve stochastic programming problems [56, 100, 156]. Most implementations are based on the alternative representation of the stochastic program that introduces copied variables to disaggregate the decisions by scenarios and non-anticipativity constraints to maintain the information structure [189]. The algorithm presented by Carøe & Schültz [29] decomposes the stochastic programming model by relaxing the non-anticipativity constraints enforcing unique decisions across indistinguishable scenarios; this Lagrangean relaxation is known to provide sharp bounds on the original problem [70]. However, the classical Lagrangean decomposition approach has two weaknesses. First, it might be difficult to generate good first-stage candidate solutions from the Lagrangean dual subproblems. Second, the update of the multipliers by subgradient optimization [106, 107, 70] or cutting planes [35, 126] can be a bottleneck that slows down the overall convergence of the algorithm. Techniques to speed up convergence of the multipliers update include the bundle method [141, 257, 128], the volume algorithm [7], and the analytic center cutting plane method [89]. Other strategies that combine the bounds obtained from subgradients and cutting planes methods have also been developed [165, 170], as well as multiplier update methods based on dual sensitivity analysis [225].

3.3 Mathematical model

We consider the two-stage Stochastic Program (SP) presented in Equations (3.1)-(3.6),

$$(SP) \qquad \min \quad TC = c^T x + \sum_{s \in S} \tau_s d_s^T y_s \tag{3.1}$$

$$\text{s.t.} \quad A_0 x \leq b_0 \tag{3.2}$$

$$A_1 x + B_1 y_s \leq b_1 \qquad \forall \;\; s \in S \tag{3.3}$$

$$B_s y_s \leq b_s \qquad \forall \;\; s \in S \tag{3.4}$$

$$x \in X \tag{3.5}$$

$$y_s \geq 0 \qquad \forall \;\; s \in S \tag{3.6}$$

where first-stage variables are denoted by x, second-stage variables are denoted by y_s, and s is the subindex for the set of scenarios ($s \in S$).

The objective function (3.1) minimizes the total expected cost (TC), which includes investment cost ($c^T x$) and expected operational cost ($\sum\limits_{s \in S} \tau_s d_s^T y_s$); the probability of scenario s is denoted by τ_s. The first-stage decisions (x) are mixed-integer and correspond to discrete choices for investments and continuous capacities, as presented in Equation (3.7).

$$X = \left\{ x = (x_1, x_2)^T : x_1 \in \{0,1\}^n, x_2 \in \mathbb{R}_+^m \right\} \tag{3.7}$$

All second-stage decisions (y_s) are continuous and correspond to operational decisions in scenario s. The constraints on the investment decisions are modeled with Equation (3.2). The constraints linking investment and operational decisions are given by Equation (3.3). Purely operational constraints are enforced with Equation (3.4).

The problem naturally decomposes into scenarios once the investment decisions x are fixed. Furthermore, we can develop an equivalent formulation by creating copied variables for the investment decisions in each scenario (x_s) and including non-anticipativity constrains that force the copied variables to have the same value for all scenarios. We use the approach proposed by Carøe & Schültz [29] to formulate the alternative representation of problem (SP) according to Equations (3.8)-(3.14),

$$(SPNAC) \qquad \min \quad TC = c^T x_s + \sum_{s \in S} \tau_s d_s^T y_s \tag{3.8}$$

$$\text{s.t.} \quad A_0 x_s \leq b_0 \qquad \forall \; s \in S \tag{3.9}$$

$$A_1 x_s + B_1 y_s \leq b_1 \qquad \forall \; s \in S \tag{3.10}$$

$$B_s y_s \leq b_s \qquad \forall \; s \in S \tag{3.11}$$

$$\sum_{s \in S} H_s x_s = 0 \tag{3.12}$$

$$x_s \in X_s \qquad \forall \; s \in S \tag{3.13}$$

$$y_s \geq 0 \qquad \forall \; s \in S \tag{3.14}$$

where Equation (3.12) models the non-anticipativity conditions ($x_1 = x_2 = ... = x_n$) with a suitable matrix $H = (H_1, ..., H_{|S|})$, and Equation (3.15) defines X_s analogously to X for the first-stage copied variables.

$$X_s = \left\{ x_s = (x_{1,s}, x_{2,s})^T : x_{1,s} \in \{0,1\}^n, x_{2,s} \in \mathbb{R}_+^m \right\} \quad \forall\ s \in S \tag{3.15}$$

Proposition 3.1. *Problems (SP) and ($SPNAC$) are equivalent.*

Proof. This trivially follows by substituting the non-anticipativity constraints (3.12). □

3.4 Ingredients of the cross-decomposition algorithm

Our cross-decomposition scheme is based on Benders and Lagrangean decomposition, which both exploit the decomposable problem structure of the two-stage stochastic programming problem. We assume complete recourse, which means that all scenarios are feasible regardless of the first-stage decisions. This assumption can be relaxed, but it would require including primal feasibility cuts and addressing dual unboundedness.

The proposed algorithm implies solving four problems per iteration: the primal Benders subproblems that yield upper bounds, the dual Lagrangean subproblems, the dual Lagrangean master problem, and the primal Benders master problem that yields lower bounds. The sequence in which these problems are solved and the information flow is presented in Figure 3.1. The basic idea is to use Benders decomposition in an outer loop and Lagrangean decomposition inside it. The objective is twofold: on one hand we strengthen the master problems with cuts derived from both subproblems (hence cross decomposition), and on the other hand we avoid the need to rely on a heuristic to generate feasible solutions from the Lagrangean subproblems. The properties of the two-stage MILP model that allow implementing the cross-decomposition scheme are presented in detail by Mitra et al. [161].

The Benders subproblems based on ($SPNAC$) can be obtained by fixing the first-stage variables at iteration k to a candidate solution $\hat{x}^k$. The formulation for the resulting Benders subproblems (BSP_s^k) corresponding to each scenario s is presented in Equations (3.16)-(3.19).

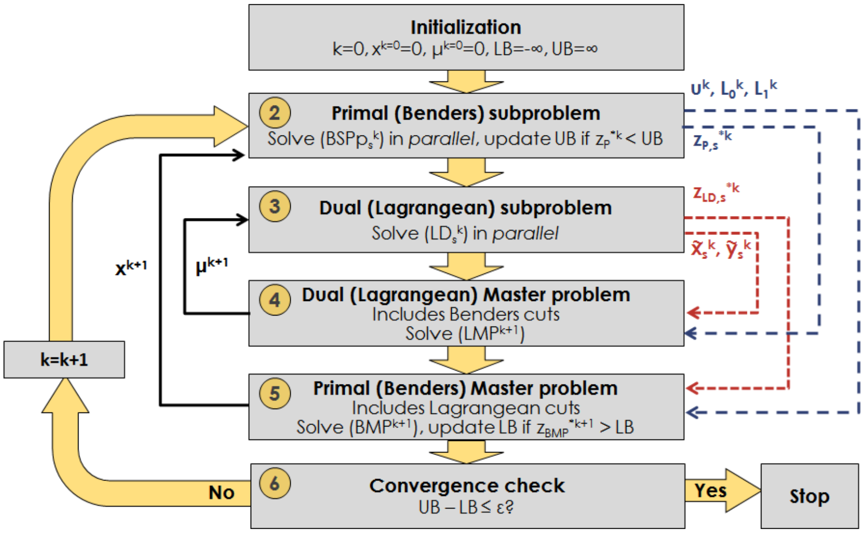

Initialization
k=0, x^k=0=0, μ^k=0=0, LB=-∞, UB=∞
2 Primal (Benders) subproblem
Solve (BSPp_s^k) in parallel, update UB if z_P^*k < UB
3 Dual (Lagrangean) subproblem
Solve (LD_s^k) in parallel
4 Dual (Lagrangean) Master problem
Includes Benders cuts
Solve (LMP^k+1)
5 Primal (Benders) Master problem
Includes Lagrangean cuts
Solve (BMP^k+1), update LB if z_BMP^*k+1 > LB
6 Convergence check
UB – LB ≤ ε?
Yes
Stop
No
k=k+1
x^k+1
μ^k+1
U^k, L_0^k, L_1^k
z_P,s^*k
z_LD,s^*k
x̃_s^k, ỹ_s^k

Figure 3.1: Flowchart of the cross-decomposition algorithm.

$$
\begin{aligned}
(BSP_s^k) \qquad & \min \quad z_{B,s}^k = \tau_s(c^T\hat{x}^k + d_s^T y_s) && (3.16)\\
& \text{s.t.} \quad B_1 y_s \le b_1 - A_1\hat{x}^k && (3.17)\\
& \qquad\quad B_s y_s \le b_s && (3.18)\\
& \qquad\quad y_s \ge 0 && (3.19)
\end{aligned}
$$

A valid relaxation of $(SPNAC)$ can be obtained by formulating its Lagrangean dual, in which the non-anticipativity constraints (3.12) are dualized [97, 29]. The Lagrangean dual subproblems (LD_s^k) are formulated for a fixed set of Lagrange multipliers μ^k. The MILP formulation of the Lagrangean subproblems can be decomposed by scenario s as shown in Equations (3.20)-(3.25).

$$
\begin{aligned}
(LD_s^k) \qquad & \min \quad z_{LD,s}^k = \tau_s(c^T x_s + d_s^T y_s) + \mu^k H_s x_s && (3.20)\\
& \text{s.t.} \quad A_0 x_s \qquad\quad \le b_0 && (3.21)\\
& \qquad\quad A_1 x_s + B_1 y_s \le b_1 && (3.22)\\
& \qquad\qquad\qquad B_s y_s \le b_s && (3.23)\\
& \qquad\quad x_s \in X_s && (3.24)\\
& \qquad\quad y_s \ge 0 && (3.25)
\end{aligned}
$$

The Lagrangean subproblems (LD_s^k) yield a solution in which some of the original non-anticipativity constraints (3.12) are most likely violated. Therefore, it is necessary to obtain a primal solution that provides a valid upper bound. In the framework of Lagrangean decomposition, a heuristic is applied to generate first-stage feasible solutions (x^k) that can be used in $(SPNAC)$ in order to obtain the primal solution. However, in our cross-decomposition scheme the heuristic can be avoided because the solutions obtained from the Benders subproblems (BSP_s^k) are feasible in $(SPNAC)$.

The Lagrangean master problem (LMP^{k+1}) is based on the cutting planes problem that is often used to update the multipliers of the Lagrangean dual [126]. We adopt the multi-cut version of the cutting planes problem and strengthen it with constraints (3.29). The cuts given by Equation (3.29) are generated with the dual information from the Benders subproblems (BSP_s^k); they strengthen the Lagrangean master problem by providing upper bounds on its objective value. The detailed derivation of these cuts is presented by Mitra et al. [161]. The Lagrangean master problem yields the next set of Lagrangean multipliers (μ^{k+1}). The mixed-integer quadratic formulation of the Lagrangean master problem (LMP^{k+1}) is presented in Equations (3.26)-(3.30),

$$(LMP^{k+1}) \qquad \max \; z_{LMP}^{k+1} = \eta_{LMP} + \frac{\delta}{2}\|\mu - \bar{\mu}\|_2^2 \tag{3.26}$$

$$\text{s.t.} \; \eta_{LMP} \leq \sum_{s \in S} \kappa_s \tag{3.27}$$

$$\kappa_s \leq \tau_s(c^T \tilde{x}_s^k + d_s^T \tilde{y}_s^k) + \mu H_s \tilde{x}_s^k \qquad \forall \; s \in S, \; k \in K \tag{3.28}$$

$$\kappa_s \leq z_{B,s}^{*k} + \mu H_s \hat{x}^k \qquad \forall \; s \in S, k \in K \tag{3.29}$$

$$\eta_{LMP} \in \mathbb{R}^1, \quad \mu \in \mathbb{R}^{(|S|-1)\times n}, \quad \kappa_s \in \mathbb{R}^{|S|} \qquad \forall \; s \in S \tag{3.30}$$

where $(\tilde{x}_s^k$,$\tilde{y}_s^k)$ is the solutions obtained from the Lagrangean subproblem in iteration k, $\hat{x}^k$ is the solution obtained from the Benders master problem in iteration k, and $z_{B,s}^{*k}$ is the optimal cost of scenario s in the Benders subproblem corresponding to iteration k.

The objective function (3.26) contains the additional quadratic stabilization term $\frac{\delta}{2}\|\mu - \bar{\mu}\|_2^2$ that defines a trust-region for the update of the Lagrangean multipliers [141, 257, 128, 73]. The stabilization requires initial values and update strategies for the penalty δ defining the size of the trust region and the stabilization center $\bar{\mu}$. It is worth noticing that Equation (3.29) not only makes the Lagrangean master problem bounded, but it also guarantees a bound at least as tight as the best known primal upper bound obtained from the Benders subproblems.

We formulate the Benders master problem (BMP^{k+1}) with disaggregated cuts for each scenario and strengthen it with constraints (3.34). The cuts represented in Equation (3.34) are generated from the solution of the Lagrangean dual subproblems; their derivation is presented by Mitra et al. [161]. The Benders master problem (BMP^{k+1}) yielding the primal vector for the next iteration $(\hat{x}^{k+1})$ is given by Equations (3.31)-(3.36),

$$(BMP^{k+1}) \qquad \min \; z_{BMP}^{k+1} = \eta_{BMP} \tag{3.31}$$

$$\text{s.t. } \eta_{BMP} \geq \sum_{s \in S} \theta_s \tag{3.32}$$

$$\theta_s \geq \tau_s^k c^T x + (A_1 x - b_1)^T u_s^k - b_s^T v_s^k \qquad \forall \; s \in S, \; k \in K \tag{3.33}$$

$$\theta_s \geq z_{LD,s}^{*k} - \mu^k H_s x \qquad \forall \; s \in S, \; k \in K \tag{3.34}$$

$$A_0 x \leq b_0 \tag{3.35}$$

$$x \in X, \;\; \eta_{BMP} \in \mathbb{R}^1, \;\; \theta_s \in \mathbb{R}^1 \qquad \forall \; s \in S \tag{3.36}$$

where u_s^k and v_s^k are the optimal Lagrange multipliers associated with constraints (3.17) and (3.18) of (BSP_s^k) in iteration k, and $z_{LD,s}^{*k}$ is the optimal cost of scenario s in the Lagrangean subproblem (LD_s^k) corresponding to iteration k.

Since the Lagrangean cuts presented in Equation (3.34) are valid inequalities for problem $(SPNAC)$, the optimal objective value (z_{BMP}^{*k+1}) obtain from problem (BMP^{k+1}) is a lower bound for the cost of original two-stage stochastic program. This condition is formalized in Equation (3.37).

$$z_{BMP}^{*k+1} \leq TC \qquad \forall \; k \in K \tag{3.37}$$

It is worth noticing that the Lagrangean cuts given in Equation (3.34) strengthen the Benders master problem. Furthermore, they guarantee that the lower bound obtained in the Benders master problem is at least as tight as the best known solution from the Lagrangean dual subproblems.

3.5 Computational experiments on the resilient supply chain design problem

We test the proposed cross-decomposition algorithm on a number of instances of the resilient supply chain design problem presented in Chapter 2. In our experiments, we use two models: the original formulation given by Equations (2.1)-(2.5) and the improved formulation that includes the tightening constraint presented in Equation (2.6). The idea is to analyze the impact of the LP relaxation in the performance of the solution methods. We refer to the original formulation as the Resilient Supply Chain Design (RSCD) model and to the formulation improved with the tightening constraint as the tightened Resilient Supply Chain Design (t-RSCD).

We compare on 30 random instances the performance of the full-space models, multi-cut Benders decomposition, and our cross decomposition; the decomposition strategies are implemented for both models RSCD and t-RSCD. The instances are divided into three groups with different number of candidate DCs and scenarios. The instances are generated based on the dataset presented by Snyder & Daskin [214] by sampling transportation costs and demands from uniform distributions bounded between 80% and 120% of the original values. We use random transportation costs to include different cost structures in the objective function and random demands to analyze different shapes of the feasible region; the evaluation of the solution methods on a large number of instances with different values for key parameters allows establishing the variability of their performance. Details of the methodology used to generate the random instances can be found in Appendix B.

3.5.1 Description of the implementation

We implement the full-space model, multi-cut Benders decomposition [18], and our cross decomposition in GAMS 24.3.1 [24]. The decomposition schemes employ parallel computing in two different ways: Benders and Lagrangean subproblems are solved in parallel as groups of 50 scenarios using the GAMS grid computing capabilities [26]; Lagrangean and Benders master problems are solved by allowing GUROBI 5.6.3 to use the processors as parallel threads. The full-space implementations also use GUROBI with parallel threads. The problems are solved using the 12 processors of an Intel Xeon (2.67 GHz) machine with 16 GB RAM.

The numerical challenges that arise from small scenario probabilities are addressed by setting tolerances for reduced costs to 10^{-7}. Instances are considered solved when their relative optimality gap is below 10^{-4}; the MILPs inside the cross-decomposition loop are solved using an integrality tolerance of 10^{-6}. The wall-clock time limit for all instances is set 10,000 s.

The update strategies for the penalty term δ (initial value $\delta = 1$) and for the stability center $\bar{\mu}$ are rather simple. In each iteration, the trust-region parameter δ is updated according to the following rule: $\delta^{k+1} = \max\{\frac{1}{2}\delta^k, 10^{-10}\}$. The stability center is kept constant for all iterations at $\bar{\mu} = 0$.

3.5.2 Comparison of solution methods

The computational statistics of the full-space models for the three groups of instances are presented in Table 3.1. It can be observed that an increasing number of candidate DCs implies more scenarios, constraints, and variables. The mean optimal solutions for the 10 instances together with their mean LP relaxation gap are also presented in Table 3.1. The statistics show a significant increase in the number of constraints for the t-RSCD model as a result of the addition of tightening constraints; these constraints are also responsible for the improvement in the LP relaxation. Despite the size of the formulations, the number of discrete variables remains small because there is only one binary variable per candidate DC.

The performance of the solution methods on the instances are presented in Figures 3.2, 3.3, and 3.4. These performance curves show the percentage of instances that can be solved to optimality within the time limit shown in the horizontal axis. The solution methods with higher percentages of problems solved in shorter times offer better computational performance.

Figure 3.2 compares the wall-clock time required by the methods to solve the instances with 639 scenarios. It can be observed that cross decomposition on the t-RSCD formulation outperforms all other methods by solving all instances in less than 800 s. The full-space RSCD formulation

Model	**DCs** (N)	**Scenarios**	**Constraints**	**Variables**	**Binary variables**	**Mean objective [$]**	**Mean LP relaxation gap**
RSCD	10	639	38,992	345,084	10	970,182	49.68%
t-RSCD	10	639	383,413	345,084	10	970,182	0.55%
RSCD	11	1,025	63,564	603,751	11	978,036	51.89%
t-RSCD	11	1,025	666,264	603,751	11	978,036	0.67%
RSCD	12	1,587	99,996	1,012,534	12	981,636	54.09%
t-RSCD	12	1,587	1,110,915	1,012,534	12	981,636	0.79%

Table 3.1: Computational statistics of full-space instances.

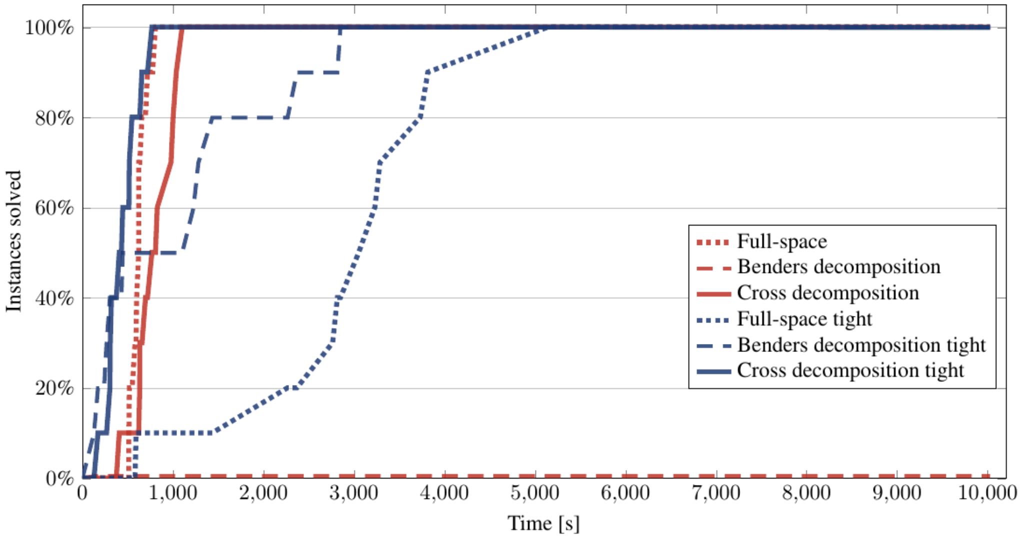


Figure 3.2: Performance curves for the solution methods in instances with 639 scenarios.

also solves all instances in less than 800 s, but it takes more than 600 s for most of them. Benders decomposition on the t-RSCD formulation is very efficient for some instances (5 instances require less than 500 s), but it takes a long time to solve others. Benders decomposition on the RSCD formulation does not solve any instance within the time limit in the case with 639 scenarios.

Similar trends can be observed in Figure 3.3 for the performance of the solution methods on the case with 1,025 scenarios. In this case, cross decomposition on the t-RSCD formulation clearly outperforms all methods by solving all instances in less than 3,250 s. Cross decomposition on the RSCD formulation also presents a relatively good performance by solving 7 instances in less than 3,000 and all instances in less than 5,200 s. The full-space RSCD model solves all instances in less than 5,100 s, but it requires over 4,000 s for most of them. Poor performance is observed for the full-space t-RSCD model and Benders decomposition on both RSCD and t-RSCD models.

For the case with 1,587 scenarios, Figure 3.4 shows that cross decomposition is the only solution method capable of solving some instances. In this case, cross decomposition on the RSCD formulation solves 3 instances and cross decomposition on the t-RSCD formulation solves 8 instances within the time limit. This results are in line with the trends observed in the previous cases: the cross-decomposition method outperforms all other methods in large-scale problems.

The mean solution time for each method over the 10 instances is presented in Table 3.2. We observe

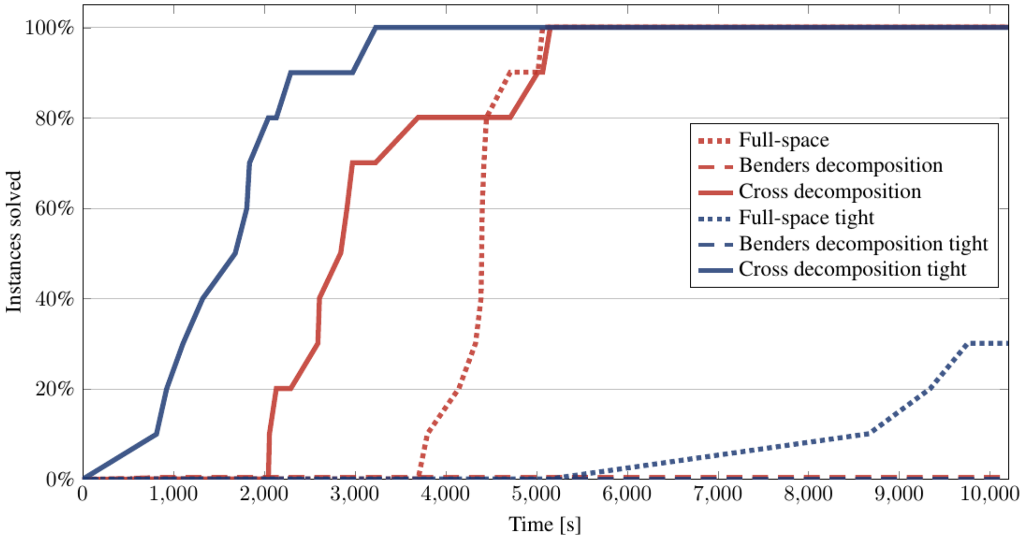


Figure 3.3: Performance curves for the solution methods in instances with 1,025 scenarios.

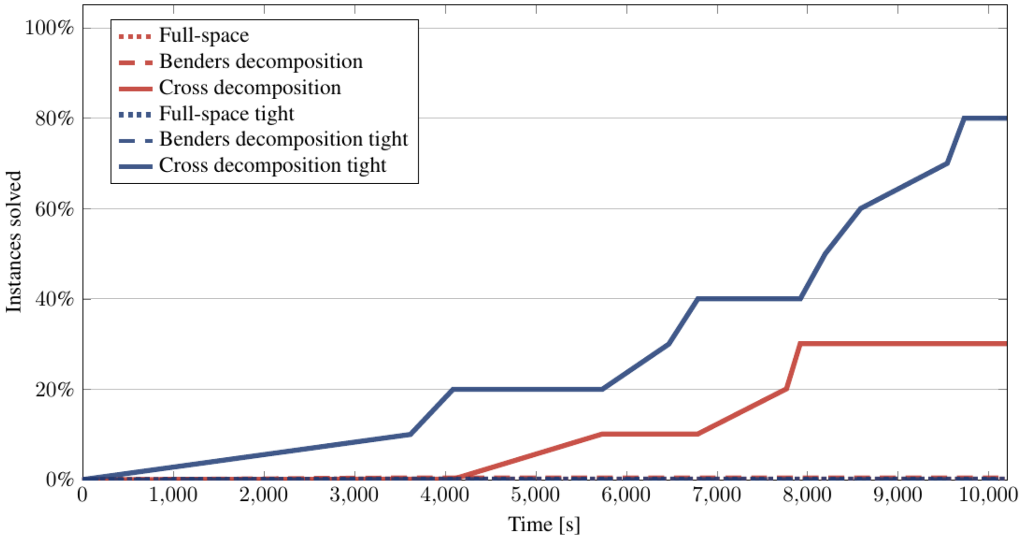


Figure 3.4: Performance curves for the solution methods in instances with 1,587 scenarios.

Scenarios	Fullspace RSCD [s]	Benders RSCD [s]	Cross RSCD [s]	Fullspace t-RSCD [s]	Benders t-RSCD [s]	Cross t-RSCD [s]
639	626	10,000*	809	3,070	1,048	472
1,025	4,413	10,000*	3,200	9,783	10,000*	1,707
1,587	10,000*	10,000*	9,947	10,000*	10,000*	7,752

*Time limit reached

Table 3.2: Comparison of mean solution times.

that cross decomposition on the t-RSCD formulation is on average the fasted method for all cases. It is interesting to note that Benders decomposition on the t-RSCD formulation shows good results for the instances with 639 scenarios but not for the instances with 1,025 and 1,587 scenarios. There is also a progressive deterioration of the performance of the fullspace RSCD formulation from the case with 639 scenarios to the case with 1,587 scenarios. This performance confirms that cross decomposition is the best alternative for large-scale problems. The computational experiments also show that both Benders and cross decomposition are affected by the underlying LP relaxation of the models. However, if we compare both decomposition strategies on the RSCD and t-RSCD formulations, it is clear that the performance of the cross-decomposition method is less affected by the weakness of the formulation. Hence, we conclude that cross decomposition is less sensitive to weak formulations due to the presence of strong cuts that originate from the Lagrangean dual.

The time spent by the decomposition methods is used to solve subproblems and master problems. For both implementations (RSCD and t-RSCD) of Benders decomposition, the wall-clock time spent in the Benders master problems is over 90% of the total time. We also conducted a separate analysis to estimate the impact of parallelization for the Benders and Lagrangean subproblems. As a result of parallelization, the time spent in Benders decomposition is significantly reduced: the speed-up for solving the Benders subproblems of the RSCD model in 12 parallel threads is around 8.5 in the case with 639 scenarios and around 7.0 for the Benders subproblems of the t-RSCD model with 1,587 scenarios. A similar situation occurs in the implementation of the cross-decomposition method. Most of the wall-clock time is spent in solving the Benders master problems: on average, around 58% of the time in the case with 639 scenarios and over 75% in the case with 1,587 scenarios. The Lagrangean master problems also require a considerable amount of time: approximately 24% of the total time in the case with 639 scenarios and 12% in the case with 1,587 scenarios. There is also a significant decrease in the solution time required to solve the MILP Lagrangean subproblems as a result of parallelization: the speed-up is approximately 5.9 for the Lagrangean subproblems of the RSCD model with 639 scenarios and 6.3 for the Lagrangean subproblems of the t-RSCD model with 1,587 scenarios. The results suggest that parallelization of the subproblems yields better speed-ups with LP subproblems that can be solved efficiently by individual threads.

3.6 Summary

We have described a cross-decomposition algorithm that combines Benders and Lagrangean decomposition for two-stage stochastic MILP problems with complete recourse, where the first-stage variables are mixed-integer and the second-stage variables are continuous. The algorithm fully integrates primal and dual information with multi-cuts that are added to the Benders and the Lagrangean master problems for each scenario. Computational results for several instances of the resilient supply chain design problem show evidence of the benefits of the cross-decomposition scheme with respect to the reduction in the number of iterations and stronger lower bounds compared to multi-cut Benders decomposition. While the computational times per iteration increase because of the solution of additional problems, cross decomposition seems to be especially advantageous compared to Benders decomposition if the underlying LP relaxation is weak. Aside from the unique integration of primal-dual multi-cuts in the master problems, the proposed cross-decomposition scheme is also unique with respect to previous work because it has been specially developed to solve two-stage stochastic programming problems.

Chapter 4

Optimizing Inventory Policies in Process Networks under Uncertainty

4.1 Motivation

Inventory planning is a critical aspect of enterprise-wide optimization [92]. Inventories are used in production and logistic networks to coordinate supply cycles and to mitigate the risks associated with uncertainty. The importance of inventory management in industrial applications derives from the effect of stockouts in the levels of customer satisfaction and the impact of stock in the economic balance of companies. Remarkably, the value of U.S. inventories was estimated to be over $1,707 billion in December 2013 [229], and the opportunity cost ascribed to capital invested in inventories added up to $434 billion in 2012 [246]. Therefore, the potential savings from stockout prevention and inventory related cost offer a huge opportunity for optimization.

Many strategies have been proposed to manage inventories since Harris [101] introduced the Economic Order Quantity (EOQ) model in 1913. The EOQ model was developed to balance ordering and holding cost for problems with a deterministic demand rate. Classical models for inventory management with uncertain demand include continuous-review (r,Q) policies and periodic-review basestock policies; the main purpose of these models is to minimize the expected cost of replenishment and stockouts, since complete satisfaction of uncertain demand might be too expensive or impossible.

One of the main advantages of the classical models is that they prescribe a policy for inventory management that is easy to implement. In fact, these policies are often optimal under assumptions satisfied by simple inventory management problems. Therefore, policies are in practice the method of choice to plan inventories in most industrial applications. However, the complexity of production networks limits the applicability of the classical models for inventory management. The main complications for inventory planning in process networks arise from the network topology, the limitations in production capacity, and multiple sources of uncertainty.

It is common practice in industry to allocate storage units at different stages of the network in order to decouple the production of successive sections. The role of inventory is to buffer temporal mismatches among supply availability, processing rates, and demand. In addition to the raw material and final product inventories that are used to hedge against external uncertainties, production networks also include intermediate inventories that protect against the variability in processing rates. The importance of intermediate inventories resides in their ability to reduce the interdependence of processing units, to delay the formation of bottlenecks, and to increase capacity utilization.

The interest in the control of intermediate inventories in production processes can be traced back to the work by Simpson [210] in the 1950's. However, few methodologies have been proposed for inventory planning under uncertainty in continuous process networks. In this chapter, we focus on developing stochastic programming formulations that leverage the nature of the inventory planning problem. We propose a new approach that includes the logic of inventory policies in a mathematical programming framework with the purpose of finding optimal policy parameters. The idea is to combine the advantages of logic-based mathematical programming with the pragmatism derived from inventory management theory. This approach for inventory optimization is novel and offers significant benefits for production planning in complex networks. We show that using policies for inventory management in process networks has advantages over multistage or two-stage stochastic programing techniques. From the modeling perspective, policies offer an alternative way to avoid anticipativity that can be used on arbitrary sets of scenarios. From the industrial perspective, policies are attractive because they are intuitive and easy to implement.

The remaining of this chapter is organized as follows. In Section 4.2, we review the publications that are most relevant to our work. Section 4.3 introduces the inventory planning problem that we address. The method that we propose to solve the problem and to evaluate the solutions is outlined in Section 4.4. In Section 4.5, we present a small example that illustrates the particularities of the stochastic inventory planning problem. Section 4.6 presents the optimization model for single-echelon basestock policies. In Section 4.7, we revisit the illustrative example to compare the inventory plans obtained from different stochastic programming models. Section 4.8 presents a general model for stochastic inventory planning in process networks. Sections 4.9 and 4.10 propose policies for inventory planning in process networks with inventories in parallel and in series, respectively. In Section 4.11, we present a simulation approach to evaluate the performance of inventory planning strategies. Sections 4.12 and 4.13 implement the proposed inventory planning models in two different examples. Finally, Section 4.14 presents a summary of this work.

4.2 Literature review

Management of intermediate inventories has been addressed in the literature of multi-echelon supply chains, which was initiated with the seminal work of Clark & Scarf [41]. They proved that basestock policies are optimal for the average cost of multi-echelon serial systems with stationary stochastic demand, convex cost function, and finite horizon. Later, Federgruen & Zipkin [66] demonstrated the optimality of basestock policies for the infinite horizon case. A recursive algorithm to calculate optimal basestock levels in serial and assembly multi-echelon systems with linear costs was developed by van Houtum & Zijm [234]. A simpler procedure yielding lower and upper bounds on the echelon cost functions was developed by Shang & Song [205]; they also present a heuristic for approximating optimal basestock levels that performs well in practice.

The study of multi-echelon systems has been expanded to find solutions for systems with less restrictive assumptions. The relationship between cost minimization models and service level models was studied by van Houtum & Zijm [235]; they conclude that the optimal solution for many of the models with probabilistic service constraints can be obtained from a corresponding cost minimization model. Chen [32] studied multi-echelon serial systems with fixed-batch sizes for the orders between installations. They show that the optimal policies are described with reorder points and discrete batch-quantities (r, nQ). A generalization of the multi-echelon inventory model that considers fixed replenishment intervals was presented by Graves [90]; van Houtum et al. [236] proved the optimality of basestock policies in these systems and presented the corresponding newsvendor formulas.

The derivation of optimal policies for inventory management in networks with general topologies is a challenging task. The analysis of multi-echelon assembly systems presented by Rosling [187] showed that their optimal basestock policies can be obtained from an equivalent serial system. For multi-echelon distribution systems, basestock policies have only been proved to be optimal under the assumption that stockouts occur with equal probability at the downstream installations [59]. Under this *balancing assumption*, optimality of basestock policies has been proved for two-echelon systems [65, 66] and for multi-echelon systems [57].

The application of multi-echelon basestock policies to supply chain design is based on two models: guaranteed service-time and stochastic service-level. The guaranteed service-time model strategically locates safety inventories to satisfy the maximum product requirements that installations are committed to satisfy during their net lead time. The model was initially developed by Kimball [127] for a single-stage system, and implemented in serial systems by Simpson [210]. Extensions of the guaranteed service-time model for safety stock placement in assembly and distribution networks have been developed for bounded demands [90, 91] and for normally distributed demands [116, 117]. The alternative stochastic service-level model developed by Lee & Billington [138] locates inventories to offer prescribed service levels at the installations of a decentralized supply chain; basestock levels are obtained from the characterization of random delays experienced by installations as a result of shortages in the upstream stages.

Ettl et al. [61] developed expressions for the actual lead time in a multi-echelon supply chain by approximating the dynamics of inventory levels with queuing models; they also included their inventory model in an optimization framework to minimize the total inventory cost. The use of queuing models to characterize production and distribution networks started with the work of Jackson [118]. The advantage of queuing networks is that they allow modeling the dynamics of inventories in networks with finite processing capacity. The most influential queuing models of manufacturing systems characterize them with product-form solutions that can be found for a restrictive class of networks, from which Jackson networks are representative. An exceptional model capturing the dynamics of basestock policies in serial servers was developed by Lee & Zipkin [139]; they showed that the serial system can be described exactly for some special cases and they developed approximations for the general case.

The characterization of optimal policies for capacitated production-inventory systems with stationary demand was presented by Federgruen & Zipkin [67, 68]. They showed that under the usual assumptions, a modified basestock policy is optimal in the infinite horizon for the average and discounted cost criteria, and also for the discounted cost criterion in a finite horizon. The modification of the classical basestock policy accounts for the capacity limitation by truncating the replenishment when the order-up-to quantity cannot be fulfilled. An algorithm to calculate optimal basestock levels and the corresponding costs for capacitated multi-echelon systems in the infinite horizon was developed by Tayur [226] using a sequence of uncapacitated models that converge to the capacitated system. A more general simulation-based method to find optimal basestock levels in capacitated multi-echelon systems was presented by Glasserman & Tayur [86]; their *Infinitesimal Perturbation Analysis* (IPA) estimates the sensitivity of the cost function with respect to the policy parameters and use them to recursively improve the basestock levels.

Most of the literature about inventory management in chemical process networks is related to deterministic systems. Karimi & Reklaitis [125] recognized the importance of intermediate storage for batch and semicontinuous processes, and derived expressions to find optimal storage capacities according to the periodicity of the production processes. Other models for multiproduct batch plants have included uncertainty in the design problem [243, 204, 115], but they have not considered inventory management in their formulations. The integration of batch plant design and scheduling was addressed by Subrahmanyam et al. [221] using a decomposition approach that iterates between a design superproblem and scheduling subproblems. Petkov & Maranas [179] addressed the optimal design and operation of batch plants with normally distributed demand for multiple products assuming a single-product campaign production mode; they exploited the properties of normal distributions to find the optimal operating policy corresponding to the potential designs.

Multi-echelon policies have also been applied for inventory management in the process industry. Jung et al. [123] developed a simulation-optimization approach in which safety stock levels are determined in a linear program and evaluated using discrete-event simulation. The proposed approach can accommodate diverse network structures and uncertainty characterizations. Recently, Chu et al. [39, 40] presented a similar approach that uses agent-based simulations to generate linear inequalities that are added to the LP planning problem to enforce the service level constraint; this approach has been used for reactive scheduling and multi-echelon inventory planning.

The guaranteed service-time model was implemented by You & Grossmann [250] for the design of chemical supply chains with uncertain demand; they extended the guaranteed service-time methodology for production planning and inventory management in dedicated chemical networks that include capacity constraints [251]. In a subsequent publication, dedicated and flexible processes are considered simultaneously by including a cyclic scheduling model that determines the sequence and duration of the flexible processes [253]. An MILP formulation for the optimal design of chemical networks with uncertainty in supply, demand, and random failures was developed by Terrazas-Moreno et al. [227, 228]. Their analysis considers the impact of slack production capacity and the effect of intermediate inventories in the reliability of the production network. The formulation proposed by Terrazas-Moreno et al. [227] allows including diverse characterizations of uncertainty as exemplified by their description of random failures using a Markov process.

An alternative approach for inventory management in production and distribution networks has leveraged control theory for sequential decision-making. Bose & Pekny [21] proposed using Model Predictive Control (MPC) for planning and scheduling of supply chain activities; their framework included forecasting, optimization, and simulation modules. Perea-López et al. [177, 178] modeled the dynamics of supply chains by considering flows of material and information. In a first article [177], they implemented site-dependent control laws to simulate the behavior of decentralized supply chains in closed loop. In a second article [178], they developed a discrete-time model of the supply chain dynamics and used MPC to plan production and distribution in a rolling horizon. The integration of scheduling and control for coordination of production and distribution has been recently addressed by Subramanian et al. [223]; their model characterizes the state of the system according to inventory levels and compare three MPC approaches that manipulate orders and shipments. In a related article, Subramanian et al. [222] proposed a state-space model for scheduling that describes the system with the levels of inventory, the tasks in progress, and their starting time; shipments, yield variations, delays, and unit breakdown are considered disturbances in the model.

Another body of literature related to our research advocates for the use of stochastic programming in supply chain design and operation. Tsiakis et al. [232] proposed an MILP formulation for the design of multi-echelon supply chains considering scenarios with uncertain demand. You et al. [252] developed a two-stage stochastic programming model for supply chain planning under uncertainty with risk management. Jung et al. [122] proposed a multistage stochastic programming formulation for multiperiod supply chain planning; their solution method iterates between a rolling horizon simulation and an outer loop that improves the safety stock targets using a gradient-based search.

Stochastic programming problems with a very large number of scenarios have been successfully solved through Sample Average Approximation (SAA) [206, 130]. SAA is a framework to approximate the optimal expected value of a stochastic program based on the solution of smaller problems with randomly sampled scenarios; the method provides statistical bounds on the expectation of the optimal objective value. Santoso et al. [199] implemented SAA for the optimal design of a supply chain with uncertain supply, capacity, cost structure, and demand. The minimum-cost design of a supply chain with a complex topology was formulated as a two-stage stochastic program by Schütz et al. [202]; in their formulation, the design is decided in the first stage and the operation is modeled in the second stage. An implementation of SAA for the design of resilient supply chains was presented by Klibi & Martel [131]; their stochastic programming formulation considers disruptions and other types of uncertainty in the scenarios.

4.3 Problem description

Inventory management involves decisions related to the replenishment and depletion of inventories. In continuous process networks, inventory decisions are closely related to production planning because most units are simultaneously internal suppliers and consumers. The complexity of chemical production networks requires storage of raw materials to guard against supply variability, intermediates to avoid the formation of bottlenecks, and final products to hedge against demand uncertainty. The role of intermediate inventories is widely understood in industrial applications but no methodologies have been proposed to optimize their management strategies in continuous process networks with complex topologies and capacity constraints.

This work addresses the inventory planning problem in continuous process networks with uncertainty in supply, available production capacity, and demand. We impose no restrictions on the characterization of the uncertain parameters other than the availability of discrete-time forecasts. Then, given a process network with known structure, our goal is to propose planning strategies that minimize the expected costs of inventory holding and stockouts in a finite horizon.

4.4 Outline of solution and result evaluation methods

The inventory planning problem under uncertainty can be formulated as a Stochastic Programming (SP) problem where production and inventory decisions are optimized to obtain the plan with minimum expected cost. In multiperiod problems with a discrete number of scenarios, the optimal solution of such a problem can be obtained by solving a multistage SP formulation. However, because of the computational difficulty to solve large-scale multistage SP models, it is often necessary to approximate them with two-stage SP formulations. Two-stage SP models are significantly easier to solve, but they do not capture the sequence in which information about uncertain parameters is revealed, which might deteriorate the quality of their solutions. We propose an alternative approximation of the multistage SP model that avoids anticipating the outcomes of uncertainty by enforcing inventory policies for all scenarios.

We develop a logic-based SP formulation that integrates inventory policies in a mathematical programming framework. In order to optimize these policies, we first postulate a parametric model mapping the levels of inventory in the network to replenishment and depletion actions. This parametric model is based on the logic of basestock policies and includes additional rules according to the topology of the process network. The logic-based SP formulation optimizes the parameters of the inventory policy with the objective of minimizing the expected cost over the scenarios.

Each scenario describes the trajectory of all uncertain parameters throughout the planning horizon. The scenarios can be generated by reproducing all possible trajectories in problems with discrete uncertain parameters, by simulating sample-paths from stochastic processes, from historical data, or from any other forecasting method. The probability associated to scenarios depends on the method used to generate them.

The most rigorous evaluation of the quality of a stochastic solution requires comparing the expected cost obtained by implementing it with the optimal expected cost of the multistage SP model. This is the approach that we follow for the illustrative example in Sections 4.5 and 4.7. The alternative for problems with too many scenarios is to compare different decision strategies using closed-loop Monte Carlo simulations. These simulations involve a sequential decision-making process that implements the first-period decisions recursively. The simulation horizon specifies the number of times that decisions are made and implemented. Closed-loop Monte Carlo simulations yield a cost associated with the decision-making strategy, but this cost is a random outcome. Therefore, several replications are required to estimate the expected simulation cost and to compare the quality of different decision-making strategies. We use this approach to evaluate the inventory planning strategies presented in Sections 4.9 and 4.10.

4.5 Illustrative example

We present a small example to illustrate the proposed inventory planning approach. The problem considers production planning and inventory management in a production-inventory system with uncertain demand. The system includes a single processing unit with deterministic production capacity, a storage unit with unlimited capacity, and stochastic demand. The planning problem has

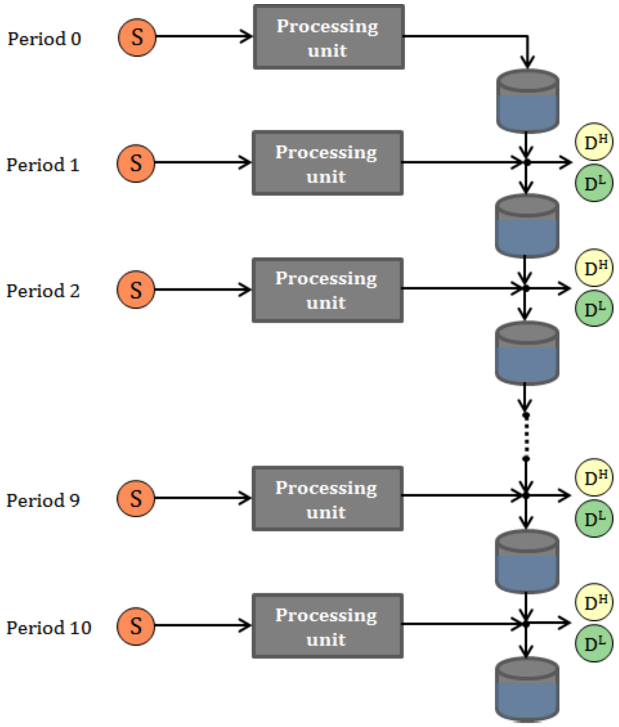


Figure 4.1: Schematic representation of the illustrative example.

a discrete time horizon with 11 periods, from period t_0 to period t_{10}. Demands are independent and identically distributed (iid) uncertain parameters characterized by a discrete uniform probability distribution in periods t_1 to t_{10}. A schematic representation of the illustrative example is presented in Figure 4.1.

We consider the case study in which supply (S) is unlimited, available production capacity (C) is 100 units of product per period, and demand can be either 110 or 90 units of product per period ($D^H = 110$, $D^L = 90$). It is worth noticing that demand can be fully satisfied by accumulating inventory in the initial time period (t_0), even if the outcome of uncertain demand is high in periods t_1 to t_{10}.

The objective of the planning problem is to minimize the expected costs associated to inventory holding and stockouts. Stockouts are calculated according to the backorders model that carries out unsatisfied demands to the next time period. We use a unit holding cost (H) of \$5/unit-period and a unit backorder cost (P) of \$15/unit-period.

The planning problem entails a sequential decision-making process in which new information becomes available as uncertainty is revealed with time. The problem has a discrete representation of time and a finite support for the uncertainty space; therefore, we can formulate it as a multistage SP problem. The multistage SP model is presented in Equations (4.1)-(4.6).

$$\min \quad Hx_{t_0} + \mathop{\mathbb{E}}_{\xi\in\Xi}\left[\sum_{t\in T\setminus\{t_0\}} Hx_{\xi,t} + Pb_{\xi,t}\right] \tag{4.1}$$

$$\text{s.t.} \quad q_{\xi,t} + u_{\xi,t} = C \qquad \forall\ t \in T,\ \xi \in \Xi \tag{4.2}$$

$$[x_{\xi,t} - b_{\xi,t}] = [x_{\xi,t-1} - b_{\xi,t-1}] + q_{\xi,t} - D_{\xi,t} \qquad \forall\ t \in T,\ \xi \in \Xi \tag{4.3}$$

$$x_{\xi,t} = x_{\xi',t},\ \ b_{\xi,t} = b_{\xi',t},\ \ q_{\xi,t} = q_{\xi',t},\ \ u_{\xi,t} = u_{\xi',t} \qquad \forall\ t = \{t_0\},\ (\xi,\xi') \in \Xi\times\Xi \tag{4.4}$$

$$x_{\xi,t} = x_{\xi',t},\ \ b_{\xi,t} = b_{\xi',t},\ \ q_{\xi,t} = q_{\xi',t},\ \ u_{\xi,t} = u_{\xi',t} \qquad \forall\ t \in T\setminus\{t_0\},\ (\xi,\xi') \in \Gamma_t \tag{4.5}$$

$$x_{\xi,t},\ b_{\xi,t},\ q_{\xi,t}, u_{\xi,t} \in \mathbb{R}^+ \qquad \forall\ t \in T,\ \xi \in \Xi \tag{4.6}$$

where T is the set of time periods (t), Ξ is the set of scenarios (ξ), and Γ_t is the set of scenario pairs with the same outcomes of the uncertain parameters up to time t. This set is used to enforce that decisions can only be based on the outcomes of past stages, which is the non-anticipativity condition. The formal definition of Γ_t for the example is given by Equation (4.7).

$$\Gamma_t := \{(\xi,\xi') : \ \ (\xi,\xi') \in \Xi\times\Xi,\ \ (D_{\xi,t_1}, D_{\xi,t_2}, ..., D_{\xi,t}) = (D_{\xi',t_1}, D_{\xi',t_2}, ..., D_{\xi',t})\} \tag{4.7}$$

The multistage SP formulation given by Equations (4.1)-(4.6) is known as the *explicit representation* because it includes copied variables for each scenario and Non-Anticipativity Constraints (NAC) relating them. The model denotes end-of-period inventory level with variables $x_{\xi,t}$ and end-of-period stockouts with variables $b_{\xi,t}$; processing rate is denoted with variables $q_{\xi,t}$ and underutilization with variables $u_{\xi,t}$. The objective function is given by Equation (4.1). In the first period, t_0, only holding cost is considered because there is no demand; in subsequent periods, holding and backorder costs are incurred. Equation (4.2) represents the capacity constraint of the processing unit, with underutilization ($u_{\xi,t}$) as a slack variable. The mass balance in the storage unit is modeled with Equation (4.3); production in a period is considered instantaneous. Non-anticipativity of the decisions is enforced with Equations (4.5)-(4.4). The domains of the variables are presented in Equation (4.6).

The multistage SP formulation for this illustrative example describes in scenarios the possible trajectories of demand; there are 1,024 scenarios corresponding to the sequences of demand from period t_1 to t_{10}. Despite the large number of scenarios, this multistage SP model is a Linear Program (LP) that can be solved with any commercial solver. The optimal solution specifies the value of 45,056 variables in the explicit representation or 8,188 variables in an implicit formulation without copied variables. However, the same optimal solution can be described in much simpler form using a basestock policy.

The capacitated single-echelon basestock policy establishes rules to operate the system according to the inventory level. The basestock level indicates the ideal level of inventory in a given period. Following the basestock policy, production capacity and inventory are first used to satisfy demand; surplus capacity is used to intend replenishing inventory up to the basestock, but no inventory in excess of the basestock level is hold. The optimal basestocks for the illustrative example and the corresponding expected costs are presented in Table 4.1. Section 4.6 describes the methodology to find these optimal basestock levels.

Period	**Basestock**	**Expected costs [$]**		
		Holding	**Backorder**	**Total**
t_0	30	150.00	0.00	150.00
t_1	20	100.00	0.00	100.00
t_2	20	75.00	0.00	75.00
t_3	20	62.50	0.00	62.50
t_4	20	56.25	18.75	75.00
t_5	20	50.00	28.13	78.13
t_6	20	46.88	46.88	93.76
t_7	10	27.34	58.59	85.93
t_8	10	19.53	76.17	95.70
t_9	10	17.77	100.19	117.98
t_{10}	0	0.00	121.00	121.00
Total:		605.27	449.71	1,054.98

Table 4.1: Optimal basestock levels and costs for the illustrative example.

4.6 Capacitated single-echelon basestock policy

It is not always easy to infer the optimal basestock levels from the solution of the multistage SP formulation. Nevertheless, the simplicity and intuitive appeal of inventory policies advocates for a general framework to obtain optimal basestock levels. Let us denote by y_t the basestock level of the single-echelon system at time t. Then, the sequence of events involved in the implementation of the basestock policy can be described as follows:

1. Random demand ($D_{\xi,t}$) is realized.
2. Production capacity (C) and carried over inventory ($x_{\xi,t-1}$) are used to satisfy demand ($D_{\xi,t}$) and backorders ($b_{\xi,t-1}$).
3. Surplus capacity is used to replenish inventory up to the basestock level (y_t).
4. Inventory level ($x_{\xi,t}$) and backorders ($b_{\xi,t}$) are updated.
5. Holding or stockout cost is calculated.

The logic describing the operation of the basestock policy in a capacitated single-echelon systems is simple. It can be characterized with the conditions given by Equations (4.8)-(4.9).

- Backorders ($b_{\xi,t}$) are allowed if there is no inventory:

$$b_{\xi,t} = \begin{cases} 0, & \text{if } x_{\xi,t} > 0 \\ D_{\xi,t} + b_{\xi,t-1} - x_{\xi,t-1} - C_t, & \text{if } x_{\xi,t} = 0 \end{cases} \tag{4.8}$$

- Underutilization ($u_{\xi,t}$) is allowed if inventory is at basestock level:

$$u_{\xi,t} = \begin{cases} 0, & \text{if } x_{\xi,t} < y_t \\ x_{\xi,t-1} + C - x_{\xi,t} - D_{\xi,t} - b_{\xi,t-1}, & \text{if } x_{\xi,t} = y_t \end{cases} \tag{4.9}$$

In order to include the basestock policy in a mathematical programming formulation, we divide the state-space of the system in three discrete states: empty inventory, intermediate level, and full inventory. In each state, the logic dictates the processing rate and inventory management plan according to a different rule. This logic can be modeled with the disjunctions presented in Equation

(4.10),

$$\begin{bmatrix} x_{\xi,t} = 0 \\ b_{\xi,t} \geq 0 \\ u_{\xi,t} = 0 \end{bmatrix} \vee \begin{bmatrix} 0 < x_{\xi,t} < y_t \\ b_{\xi,t} = 0 \\ u_{\xi,t} = 0 \end{bmatrix} \vee \begin{bmatrix} x_{\xi,t} = y_t \\ b_{\xi,t} = 0 \\ u_{\xi,t} \geq 0 \end{bmatrix} \qquad \forall \; t \in T, \; \xi \in \Xi \tag{4.10}$$

where the term on the left models the basestock policy with an empty inventory, the term on the center with an intermediate level, and the term on the right with a full inventory. Strict inequalities modeling intermediate levels ($0 < x_{\xi,t} < y_t$) can be implemented in the mathematical programming environment with epsilon precision ($\epsilon \leq x_{\xi,t} \leq y_t - \epsilon$).

The formulation enforcing a basestock policy for inventory management in the illustrative example is obtained by replacing NAC constraints (4.5) with the logic presented in Equation (4.10). The most obvious advantage of this logic-based SP formulation is that its solution can be easily characterized with the basestock levels (y_t). The formulation is a Generalized Disjunctive Program (GDP) that can be seen as a multiperiod SP formulation with piece-wise linear decision rules for inventory management [135].

The GDP model can be reformulated as a Mixed-Integer Linear Program (MILP) by introducing binary variables; for notational convenience, we denote binary variables with a hat (ˆ) throughout the chapter. Binary variables $\hat{\mathrm{x}}^0_{\xi,t}$ and $\hat{\mathrm{x}}^y_{\xi,t}$ indicate if the inventory is empty or at the basestock level, respectively. The conditions defining these variables are presented in Equations (4.11)-(4.14),

$$x_{\xi,t} \leq M \left(1 - \hat{\mathrm{x}}^0_{\xi,t}\right) \qquad \forall \, \xi \in \Xi, \; t \in T \tag{4.11}$$

$$x_{\xi,t} \leq y_t \qquad \forall \, \xi \in \Xi, \; t \in T \tag{4.12}$$

$$x_{\xi,t} \geq y_t - M \left(1 - \hat{\mathrm{x}}^y_{\xi,t}\right) \qquad \forall \, \xi \in \Xi, \; t \in T \tag{4.13}$$

$$\hat{\mathrm{x}}^0_{\xi,t} + \hat{\mathrm{x}}^y_{\xi,t} \leq 1 \qquad \forall \, \xi \in \Xi, \; t \in T \tag{4.14}$$

where the parameter M is an upper bound on the basestock level, Equation (4.11) forces the inventory to be empty if variable $\hat{\mathrm{x}}^0_{\xi,t}$ equals one, Equations (4.12)-(4.13) forces the inventory to be at basestock level if $\hat{\mathrm{x}}^y_{\xi,t}$ equals one, and Equation (4.14) allows selecting only one of these states per scenario and time period.

The logic presented in Equation (4.10) is completed with Equations (4.15)-(4.16),

$$b_{\xi,t} \leq M\hat{\mathrm{x}}^{0}_{\xi,t} \qquad \forall\, \xi \in \Xi,\ t \in T \tag{4.15}$$

$$u_{\xi,t} \leq M\hat{\mathrm{x}}^{y}_{\xi,t} \qquad \forall\, \xi \in \Xi,\ t \in T \tag{4.16}$$

where Equation (4.15) allows stockouts only when the inventory is empty, and Equation (4.16) allows underutilization only when the inventory is at basestock level.

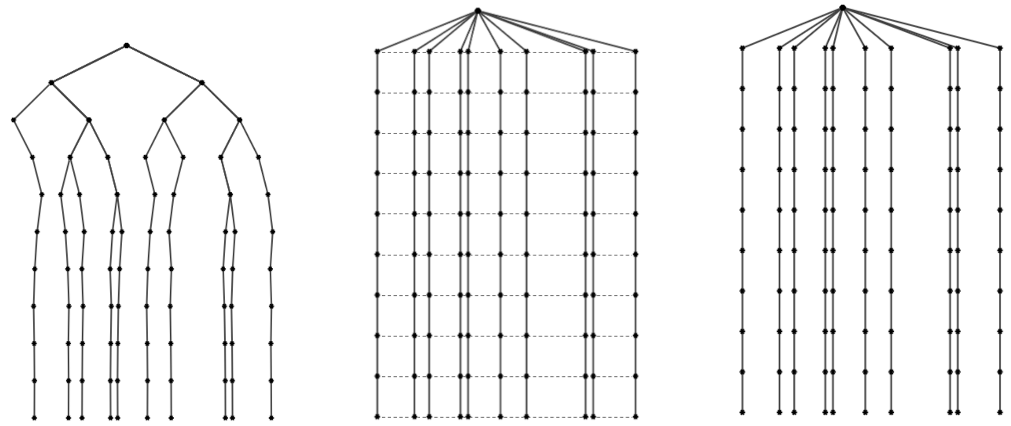
The MILP reformulation of the logic-based SP model is obtained by replacing the NAC constraints (4.5) in the multistage SP model with Equations (4.11)-(4.16). The resulting model can be solved using any available MILP solver.

4.7 Illustrative example revisited

Despite the convenience of establishing production and inventory management plans according to a policy, solving the logic-based SP formulation can be significantly harder than solving an LP model. Additionally, there is no guarantee that the optimal policy obtained from the logic-based SP formulation yields an expected value as good as the optimal multistage SP solution. However, large-scale multistage SP problems are also difficult to solve and often the multistage model is only an approximation of the real problem. The most common approximation is to restrict the number of scenarios in problems with a large number of discrete uncertain parameters or in problems with continuous support.

In order to asses the quality of the solutions obtained from different approximations of multistage stochastic programs, we propose a new performance metric called the *Residual Expected Value* (REV). The REV of a solution is the optimal expected value of the multistage SP problem after implementing the here-and-now decisions. We calculate the REV of a decision-making strategy by fixing the first-stage variables to the values that it dictates, and solving the remaining multistage SP problem. The REV generalizes the multistage *Value of the Stochastic Solution* (VSS) to allow comparing the quality of different decision-making strategies, since VSS only compares the SP solution with the solution of the expected value problem [60].

We evaluate the performance of a decision-making strategy by measuring how much the REV deviates from the expected value obtained from the exact multistage SP formulation. Our analysis



(a) Sampled multistage tree. (b) Sampled policy-constraint tree. (c) Sampled two-stage tree.

Figure 4.2: Scenario trees for the illustrative example.

considers three decision-making strategies for the production and inventory planning problem presented in Section 4.5. All formulations approximate the multistage SP solution based on a model with a reduced number of sampled scenarios. In addition to the multistage SP and the logic-based SP problem, we include in our analysis the results from the two-stage SP problem. The two-stage SP problem is obtained by relaxing NAC constraints (4.5) of the multistage SP problem in all stages after the first. Instances of the scenario trees generated using the sampling technique are presented in Figure 4.2.

The trees presented in Figure 4.2 are generated by randomly sampling 10 scenarios out of the 1,024 possible scenarios. The multistage structure in Figure 4.2a can only be recognized in the first few periods. After period 5, the sampled multistage tree does not have indistinguishable scenarios, which makes it identical to the two-stage tree in Figure 4.2c. On the other hand, the policy-constraint tree maintains non-anticipativity by implementing a single decision logic for all scenarios, which is represented by the doted lines connecting the scenarios in Figure 4.2b.

We compare the REV for the three SP models using different sample sizes. Each point presented in Figure 4.3 was estimated from 200 replications of sample trees generated using Latin-Hypercubes Sampling; the same 200 sample trees were used to evaluate all SP models. Figure 4.3 shows that a relatively low number of scenarios is needed to obtain a good first-stage solution with the multistage and logic-based SP formulations; with 100 sampled scenarios, both formulations produce a REV that is within 1% of the expected value of the full multistage SP model. The two-stage SP

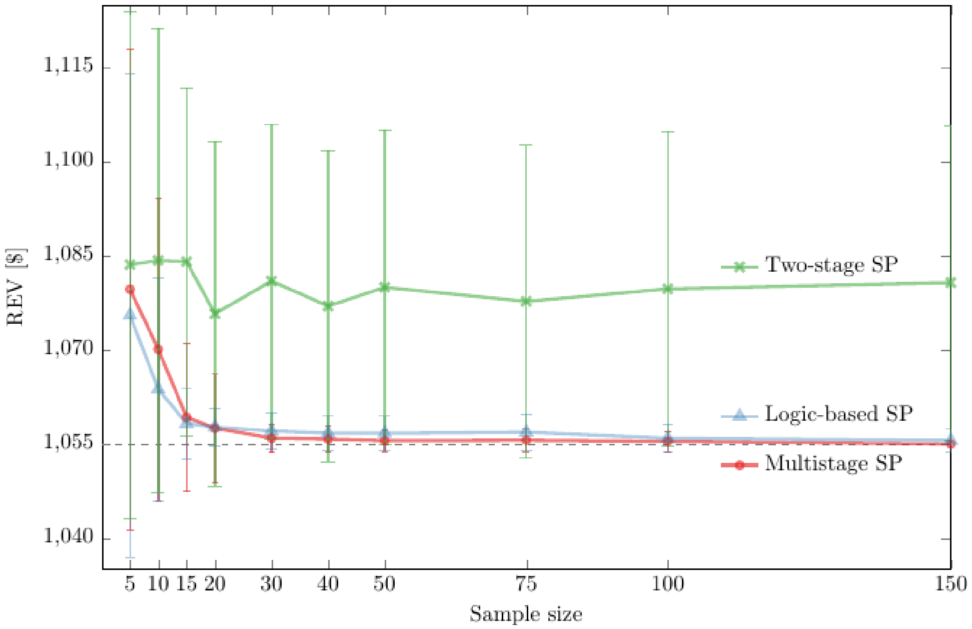


Figure 4.3: Residual expected value as a function of the sample size.

formulation on the other hand, does not seem to provide better solutions even with a larger number of scenarios; furthermore, the error bars indicate a high variability in its results. One of the most interesting observations from Figure 4.3 is that the logic-based SP formulation outperforms the multistage SP formulation when small sample sizes are used. This might be specially relevant for stochastic programming problems with a large number of scenarios or for stochastic problems with continuous random parameters.

4.8 Mathematical model for stochastic inventory planning in process networks

Our model to formulate the inventory planning problem considers process networks of general topology. The transformation of raw materials into final products is achieved with a sequence of steps that are carried out in specific processing units. We denote the set of materials by M and the set of processing units by I. Three sets of parameters are considered uncertain in the formulation: available supply, available production capacity, and demand. For ease of notation, we use capitalized letters for parameters and sets, and lower-case letters for variables and indices; all variables in this section are defined in the positive real domain. The equations describing the mathematical model are presented in the remainder of this section.

4.8.1 Supply balances

The availability of supply is modeled with Equation (4.17). The subset of materials that are externally supplied is denoted by M^S. The amount of material m that is available as external supply at time t and scenario ξ is given by parameter $S_{\xi,t,m}$. I_m^S is the subset of processing units that receive external supply of material m. The flow of supply consumed in unit i is denoted by $f_{\xi,t,i,m}^S$, the flow that is stored as inventory by $r_{\xi,t,m}^S$, and the underutilization of supply by $v_{\xi,t,m}$.

$$S_{\xi,t,m} = \sum_{i \in I_m^S} f_{\xi,t,i,m}^S + r_{\xi,t,m}^S + v_{\xi,t,m} \qquad \forall\, \xi \in \Xi,\ t \in T,\ m \in M^S \tag{4.17}$$

4.8.2 Production capacity

The capacity of processing units is modeled with Equation (4.18). We define the available production capacity ($C_{\xi,t,i}$) as an uncertain parameter to model random variations impacting the potential throughput of processing units; the maximum capacity of a unit is always greater than its available production capacity. The difference between maximum capacity and available production capacity arises due to various sources of uncertainty, such as equipment breakdown, instrumentation failure, personnel issues, or utility shortages. In Equation (4.18), the processing rate is denoted by $q_{\xi,t,i}$ and the underutilization by $u_{\xi,t,i}$.

$$C_{\xi,t,i} = q_{\xi,t,i} + u_{\xi,t,i} \qquad \forall\, \xi \in \Xi,\ t \in T,\ i \in I \tag{4.18}$$

4.8.3 Consumption balance

The consumption of material m in processing unit i is modeled with Equation (4.19). The subset of materials that are consumed in unit i is denoted by M_i^{in}; the mass balance coefficient indicating the amount of material m that is consumed per unit production rate is given by parameter $A_{i,m}$.

The subset of processing units feeding material m to unit i is denoted by $I^{up}_{i,m}$. The flow of material m from unit i' to unit i is $f_{\xi,t,i',i,m}$, and the amount of inventory depleted to feed unit i is modeled with variable $d_{\xi,t,i,m}$.

$$A_{i,m} q_{\xi,t,i} = f^S_{\xi,t,i,m} + \sum_{i' \in I^{up}_{i,m}} f_{\xi,t,i',i,m} + d_{\xi,t,i,m} \qquad \forall\, \xi \in \Xi,\ t \in T,\ i \in I,\ m \in M^{in}_i \tag{4.19}$$

4.8.4 Production balance

The production of material m in processing unit i is modeled with Equation (4.20). The subset of materials that are produced in unit i is denoted by M^{out}_i; the mass balance coefficient indicating the amount of material i that is produced per unit production rate is given by parameter $B_{i,m}$. The subset of processing units receiving material m from unit i is denoted by $I^{down}_{i,m}$. The amount of inventory replenished by unit i is $r_{\xi,t,i,m}$, and the production flow that is used to satisfy demand is $f^D_{\xi,t,i,m}$.

$$B_{i,m} q_{\xi,t,i} = \sum_{i' \in I^{down}_{i,m}} f_{\xi,t,i,i',m} + r_{\xi,t,i,m} + f^D_{\xi,t,i,m} \qquad \forall\, \xi \in \Xi,\ t \in T,\ i \in I,\ m \in M^{out}_i \tag{4.20}$$

4.8.5 Inventory balance

The inventory of material m is modeled with Equation (4.21). The subset of materials that can be stored is denoted by M^x. The balance includes the inventory carried-over from the last period ($x_{\xi,t-1,m}$), the replenishment from supply ($r^S_{\xi,t,m}$), the replenishment from processing units ($r_{\xi,t,i,m}$), the inventory used to feed processing units ($d_{\xi,t,i,m}$), and the inventory used to satisfy external demand ($d^D_{\xi,t,m}$). The set of units allowed to replenish the inventory of material m is denoted by I^r_m, and the set of units that can deplete inventory of material m is denoted by I^d_m.

$$x_{\xi,t,m} = x_{\xi,t-1,m} + r^S_{\xi,t,m} + \sum_{i \in I^r_m} r_{\xi,t,i,m} - \sum_{i \in I^d_m} d_{\xi,t,i,m} - d^D_{\xi,t,i,m}$$
$$\forall\, \xi \in \Xi,\ t \in T,\ m \in M^x \tag{4.21}$$

4.8.6 Demand balance

Demand satisfaction is modeled with Equation (4.22). The subset of materials with external demand is denoted by M^D. Demand ($D_{\xi,t,m}$) and carried-over backorders ($b_{\xi,t-1,m}$) are equal to the production flow that is used satisfy demand ($f^D_{\xi,t,i,m}$), the inventory that is depleted to satisfy demand ($d^D_{\xi,t,m}$), and the end-of-period backorders ($b_{\xi,t,m}$).

$$D_{\xi,t,m} + b_{\xi,t-1,m} = \sum_{i \in I^D_m} f^D_{\xi,t,i,m} + d^D_{\xi,t,m} + b_{\xi,t,m} \qquad \forall\, \xi \in \Xi,\ t \in T,\ m \in M^D \tag{4.22}$$

4.8.7 Objective function

Different objective functions can be used in the inventory planning problem. In our formulation, we minimize the sum of expected holding and stockout costs as presented in Equation (4.23). The probability of scenario ξ is denoted by $\mathbb{P}_\xi$. The holding cost of material m at period t is denoted by $H_{t,m}$, and the penalty per unit backorder of material m at period t is denoted by $P_{t,m}$.

$$\min \quad \sum_{\xi \in \Xi} \mathbb{P}_\xi \sum_{t \in T} \left(\sum_{m \in M^x} H_{t,m} x_{\xi,t,m} + \sum_{m \in M^D} P_{t,m} b_{\xi,t,m} \right) \tag{4.23}$$

4.9 Policy for inventories in parallel

We propose a priority-based policy for storable materials that compete for the same replenishment resources. The basic condition is that the parameters specifying the policy must be the same across scenarios. The goal of the model is to establish the optimal priorities and basestock levels for inventories in a parallel arrangement. An illustration of a parallel arrangement with three storable materials is presented in Figure 4.4.

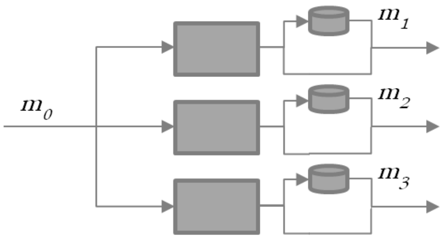


Figure 4.4: Parallel arrangement with m_0 as a shared resource for the replenishment of inventories m_1, m_2, and m_3.

4.9.1 Logic-based formulation

We denote by N the set of parallel arrangements in the process network, by $\tilde{M}_n$ the subset of storable materials that belong to parallel arrangement n, and by $R_n \subset M$ the materials that are considered shared resources for the production of $m \in \tilde{M}_n$. The set of priority levels (l) in parallel arrangement n is L_n. The number of priorities and the number of storable materials in a parallel arrangement are set equal ($|\tilde{M}_n| = |L_n|$) with the purpose of assigning unique priority levels.

The binary variables indicating the ordering of priorities for the storable materials in a parallel arrangement are defined according to Equation (4.24).

$$\hat{z}_{n,l,m} = \begin{cases} 1, & \text{if material } m \text{ has priority level } l \text{ in parallel arrangment } n \\ 0, & \text{otherwise} \end{cases} \tag{4.24}$$

In order to ensure that each storable material in a parallel arrangement is assigned a unique priority level, we use the exclusive -or- conditions presented in Equation (4.25)-(4.26),

$$\underline{\bigvee}_{m \in \tilde{M}_n} [\hat{z}_{n,l,m} = 1] \qquad \forall\, n \in N,\ l \in L_n \tag{4.25}$$

$$\underline{\bigvee}_{l \in L_n} [\hat{z}_{n,l,m} = 1] \qquad \forall\, n \in N,\ m \in \tilde{M}_n \tag{4.26}$$

where we express the disjunctions in terms of binary variables for notational convenience. The boolean logic can be obtained by establishing the following correspondence between binary ($\hat{z}_{n,l,m}$) and boolean ($Z_{n,l,m}$) variables:

$\hat{z}_{n,l,m} = 1 \;\Leftrightarrow\; Z_{n,l,m} = \text{true}$

$\hat{\mathrm{z}}_{n,l,m} = 0 \;\Leftrightarrow\; Z_{n,l,m} = \text{false}$

The priorities established by variables $\hat{\mathrm{z}}_{n,l,m}$ specify the order in which inventories in the arrangement are replenished. In particular, the material that is assigned priority $l+1$ can only be replenished if the replenishment of material with priority l is complete. Binary variable $\hat{\mathrm{w}}_{\xi,t,n,l}$ indicates that the replenishment of material with priority level l is complete in a given scenario (ξ) and time period (t). The definition of variable $\hat{\mathrm{w}}_{\xi,t,n,l}$ is given by Equation (4.27).

$$\hat{\mathrm{w}}_{\xi,t,n,l} = \begin{cases} 1, & \text{if replenishment of material with priority } l \text{ is complete} \\ 0, & \text{otherwise} \end{cases} \tag{4.27}$$

The completion of replenishment for material with priority level l implies that no additional upstream materials shared in the parallel arrangement are needed to replenish this inventory. If we denote by I_m^r the set of units that can replenish the inventory of material m, variable $\hat{\mathrm{w}}_{\xi,t,n,l}$ must satisfy the condition given by Equation (4.28).

$$\begin{bmatrix} \hat{\mathrm{z}}_{n,l,m} = 1 \\ x_{\xi,t,m} < y_{t,m} \\ \bigvee_{i \in I_m^r} \begin{bmatrix} u_{\xi,t,i} > 0 \\ \hat{\mathrm{g}}_{\xi,t,i,m} = 0 \quad \forall\, m \in M_i^{in} \backslash \{R_n\} \end{bmatrix} \end{bmatrix}$$
$$\Longrightarrow \quad \hat{\mathrm{w}}_{\xi,t,n,l} = 0 \qquad\qquad \forall\, \xi \in \Xi,\; t \in T,\; n \in N,\; l \in L_n,\; m \in \tilde{M}_n \tag{4.28}$$

The implication presented in Equation (4.28) states that the replenishment of inventory with priority level l cannot be considered complete if the inventory level ($x_{\xi,t,m}$) is below the basestock ($y_{t,m}$), and there is available capacity ($u_{\xi,t,i} > 0$) and upstream materials ($\hat{\mathrm{g}}_{\xi,t,i,m} = 0$) for the units ($i \in I_m^r$) that can replenish it; we exclude the shared resource (R_n) from the set of upstream materials (M_i^{in}) required for the replenishment because their shortage does not relax the implication. Binary variables $\hat{\mathrm{g}}_{\xi,t,i,m}$ indicate if there is an upstream shortage of material m that does not allow increasing the processing rate in unit i. The logic establishing material shortage is given by Equation (4.29).

$$[x_{\xi,t,m} > 0] \;\vee\; [v_{\xi,t,m} > 0] \bigvee_{i' \in I^{up}_{i,m}} \begin{bmatrix} u_{\xi,t,i'} > 0 \\ \hat{\mathrm{g}}_{\xi,t,i',m} = 0 \quad \forall\, m \in M^{in}_{i'} \end{bmatrix}$$
$$\implies \hat{\mathrm{g}}_{\xi,t,i,m} = 0 \qquad \forall\ \xi \in \Xi,\ t \in T,\ i \in I^{cons}_m,\ m \in M \tag{4.29}$$

Expression (4.29) does not allow indicating shortage of material m for the units that consume it ($i \in I^{cons}_m$), if there is available inventory, supply underutilization, or the upstream units capable of producing it ($i' \in I^{up}_{i,m}$) are not fully utilized nor in shortage of the materials they consume (M^{in}_i).

Priorities for the replenishments ($r_{\xi,t,i,m}$) are enforced with Equations (4.30)-(4.31).

$$\hat{\mathrm{w}}_{\xi,t,n,l} = 0 \implies \hat{\mathrm{w}}_{\xi,t,n,l+1} = 0 \qquad \forall\, \xi \in \Xi,\ t \in T,\ n \in N,\ l \in L_n \tag{4.30}$$

$$\bigvee_{l \in L_n} \begin{bmatrix} \hat{\mathrm{z}}_{n,l,m} = 1 \\ \hat{\mathrm{w}}_{\xi,t,n,l-1} = 0 \end{bmatrix} \implies r_{\xi,t,i,m} = 0$$
$$\forall\, \xi \in \Xi,\ t \in T,\ n \in N, i \in I^r_m,\ m \in \tilde{M}_n \tag{4.31}$$

where Equation (4.30) guarantees that variables indicating the completion of replenishment ($\hat{\mathrm{w}}_{\xi,t,n,l}$) are activated following the order of priorities, and Equation (4.31) constraints replenishments according to the completion of levels that are hierarchically higher.

4.9.2 An MILP reformulation

We reformulate the logic for inventory management in parallel arrangements using mixed-integer constraints. The reformulation of constraints (4.25)-(4.26) is given by Equations (4.32)-(4.33).

$$\sum_{m \in \tilde{M}_n} \hat{\mathrm{z}}_{n,l,m} = 1 \qquad \forall\, n \in N,\ l \in L_n \tag{4.32}$$

$$\sum_{l \in L_n} \hat{\mathrm{z}}_{n,l,m} = 1 \qquad \forall\, n \in N,\ m \in \tilde{M}_n \tag{4.33}$$

The implication on replenishment completion (Equation (4.28)) can be reformulated according to

Equation (4.34).

$$(1-\hat{\mathrm{z}}_{n,l,m})+\hat{\mathrm{x}}^{y}_{t,m}+\hat{\mathrm{u}}^{0}_{\xi,t,i}+\sum_{m'\in M^{in}_{i}\setminus R_n}\hat{\mathrm{g}}_{\xi,t,i,m'}+(1-\hat{\mathrm{w}}_{\xi,t,l})\geq 1$$
$$\forall\,\xi\in\Xi,\ t\in T,\ n\in N,\ l\in L_n,\ i\in I^{r}_{m},\ m\in\tilde{M}_n \tag{4.34}$$

where binary variable $\hat{\mathrm{u}}^{0}_{\xi,t,i}$ indicates if there is underutilization of unit i in scenario ξ at time period t. We enforce the definition of $\hat{\mathrm{u}}^{0}_{\xi,t,i}$ with the big-M constraint presented in Equation (4.35).

$$u_{\xi,t,i}\leq M\left(1-\hat{\mathrm{u}}^{0}_{\xi,t,i}\right)\qquad \forall\,\xi\in\Xi,\ t\in T,\ i\in I \tag{4.35}$$

The condition (4.29) that indicates shortage of upstream material m in unit i can be reformulated with Equations (4.36)-(4.38),

$$\hat{\mathrm{x}}^{0}_{\xi,t,m}+(1-\hat{\mathrm{g}}_{\xi,t,i,m})\geq 1\qquad \forall\,\xi\in\Xi,\ t\in T,\ i\in I,\ m\in M \tag{4.36}$$

$$\hat{\mathrm{v}}^{0}_{\xi,t,m}+(1-\hat{\mathrm{g}}_{\xi,t,i,m})\geq 1\qquad \forall\,\xi\in\Xi,\ t\in T,\ i\in I,\ m\in M \tag{4.37}$$

$$\hat{\mathrm{u}}^{0}_{\xi,t,i'}+\sum_{m'\in M^{in}_{i'}}\hat{\mathrm{g}}_{\xi,t,i,m'}+(1-\hat{\mathrm{g}}_{\xi,t,i,m})\geq 1$$
$$\forall\,\xi\in\Xi,\ t\in T,\ i\in I^{cons}_{m},\ i'\in I^{up}_{i,m},\ m\in M \tag{4.38}$$

where binary variable $\hat{\mathrm{v}}^{0}_{\xi,t,m}$ indicates if there is supply underutilization of material m in scenario ξ at time period t. We enforce the definition of $\hat{\mathrm{v}}^{0}_{\xi,t,m}$ with Equation (4.39),

$$v_{\xi,t,m}\leq M\left(1-\hat{\mathrm{v}}^{0}_{\xi,t,m}\right)\qquad \forall\,\xi\in\Xi,\ t\in T,\ m\in M^{S} \tag{4.39}$$

Finally, the logic expressed in Equations (4.30)-(4.31) can be reformulated with Equations (4.40)-(4.41), respectively.

$$\hat{\mathrm{w}}_{\xi,t,n,l}\geq\hat{\mathrm{w}}_{\xi,t,n,l+1}\qquad \forall\,\xi\in\Xi,\ t\in T,\ n\in N,\ l\in L_n \tag{4.40}$$

$$\sum_{i\in I^{r}_{m}}r_{\xi,t,i,m}\leq M\,(1-\hat{\mathrm{z}}_{n,m,l})+M\hat{\mathrm{w}}_{\xi,t,n,l-1}$$
$$\forall\,\xi\in\Xi,\ t\in T,\ n\in N,\ l\in L_n,m\in\tilde{M}_n \tag{4.41}$$

where the parameter M is an upper bound for the total replenishment from all units $i \in I^r_m$.

Equations (4.32)-(4.41) yield an MILP reformulation of the logic developed for inventory management in parallel arrangements. It is important to remark that this is only one possible reformulation; other reformulations with different number of variables and constraints are possible. They might lead to stronger or weaker formulations with respect to the LP relaxation.

4.10 Policy for inventories in series

The inventory planning for materials that undergo sequential transformation is based on multi-echelon inventory theory. We identify from the network structure processing paths (k) starting at raw material nodes and finishing at end product nodes; the purpose is to coordinate inventory management for the materials in these paths. A multi-echelon arrangement is a subset of storable materials ($\bar{M}_k \subseteq M^x$) associated with a particular processing path. We define an echelon as the subset ($\bar{M}_{k,e} \subseteq \bar{M}_k$) containing a number e of the most downstream materials in multi-echelon arrangement $\bar{M}_k$; echelons are numbered from the most downstream (echelon $\bar{M}_{k,1}$) to the most upstream (echelon $\bar{M}_{k,|E_K|}$), according to the conventions from multi-echelon literature. An illustration of the echelons comprising a multi-echelon arrangement is presented in Figure 4.5.

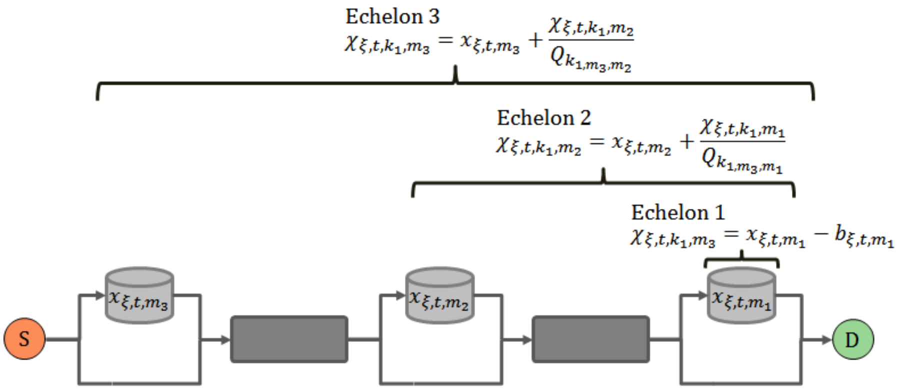

Echelon 3
$\chi_{\xi,t,k_1,m_3} = x_{\xi,t,m_3} + \frac{\chi_{\xi,t,k_1,m_2}}{Q_{k_{1,m_3,m_2}}}$
Echelon 2
$\chi_{\xi,t,k_1,m_2} = x_{\xi,t,m_2} + \frac{\chi_{\xi,t,k_1,m_1}}{Q_{k_{1,m_3,m_1}}}$
Echelon 1
$\chi_{\xi,t,k_1,m_3} = x_{\xi,t,m_1} - b_{\xi,t,m_1}$
x_{ξ,t,m_3}
x_{ξ,t,m_2}
x_{ξ,t,m_1}
S
D

Figure 4.5: A multi-echelon arrangement with 3 echelons.

4.10.1 Logic-based formulation

Formally, the subsets of materials in echelon $\bar{M}_{k,e}$ is given by Equation (4.42),

$$\bar{M}_{k,e} = \Big\{ m : m \in \{ \bar{M}_{k,e-1} \cup m_{k,e} \} \Big\} \tag{4.42}$$

where $m_{k,e}$ is the storable material preceding other $e-1$ materials in processing path k. Consequently, echelon $\bar{M}_{k,1}$ only contains one final product ($m_{k,1} \in M^D \;\; \forall\, k \in K$).

The logic of basestock policies in multi-echelon systems is based on the concept of echelon inventory level. The echelon inventory level considers the available inventory of all the materials that belong to the echelon. The challenge to define the echelon inventory level in process networks is that materials change their identity through the production process; therefore, we have to consider the mass balance coefficients ($A_{i,m}$ and $B_{i,m}$) to calculate the equivalence between one material and its downstream successor. The inventory level ($\chi_{\xi,t,k,e}$) for echelons $\bar{M}_{k,1}$ and $\bar{M}_{k,e}$ can be calculated from Equations (4.43)-(4.44), respectively.

$$\chi_{\xi,t,k,1} = x_{\xi,t,m} - b_{\xi,t,m} \qquad \forall\ \xi \in \Xi,\ t \in T,\ k \in K,\ m = m_{k,1} \tag{4.43}$$

$$\chi_{\xi,t,k,e} = \frac{1}{Q_{k,e,e-1}} \chi_{\xi,t,k,e-1} + x_{\xi,t,m}$$
$$\forall\ \xi \in \Xi,\ t \in T,\ k \in K,\ e \in E_k \backslash \{e=1\},\ m = m_{k,e} \tag{4.44}$$

where $Q_{k,e,e-1}$ is the conversion ratio in the process that transforms material $m_{k,e}$ into material $m_{k,e-1}$ following processing path k. It is worth noticing that the inventory level of echelon $\bar{M}_{k,1}$ includes backorders, and that our process does not consider *in-transit* inventory since the transportation between units is assumed to be instantaneous.

Based on the echelon inventory level, we can extend the capacitated single-echelon basestock policy for inventory planning in sequential production processes. The idea is to define basestock levels ($y_{t,k,e}$) for each echelon, such that the available downstream inventory is considered in the replenishment decisions corresponding to material $m_{k,e}$. The logic for capacity utilization of the units ($i \in I^r_m$) that can replenish inventory $x_{\xi,t,m}$ remains the same as in the single-echelon system, except that we now have to consider the case in which underutilization is forced because of upstream material shortage. In a multi-echelon arrangement, the conditions allowing backorders and underutilization are given by expressions (4.45)-(4.46), respectively.

$$\chi_{\xi,t,k,1} > 0 \implies b_{\xi,t,m} = 0 \qquad \forall\ \xi \in \Xi,\ t \in T,\ k \in K,\ m = m_{k,1} \tag{4.45}$$

$$[\chi_{\xi,t,k,e} < y_{t,k,e}] \wedge [\hat{g}_{\xi,t,m} = 0\ \forall\, m \in M_i^{in}] \implies u_{\xi,t,i} = 0$$
$$\forall\ \xi \in \Xi,\ t \in T,\ k \in K,\ e \in E_k,\ i \in I^r_{m_{k,e}} \tag{4.46}$$

The equations defining echelon inventory levels (Equations (4.43)-(4.44)) and the logic controlling production decisions (Equations (4.45)-(4.46)) can be used in a logic-based formulation to find the optimal parameters of the basestock policy. For processing networks with multi-echelon arrangements, the parameters to optimize are the basestock levels of each echelon ($y_{t,k,e}$).

4.10.2 An MILP reformulation

We reformulate the logic for inventory management in multi-echelon arrangements using mixed-integer constraints. Similarly to the capacitated single-echelon basestock policy, this reformulation requires variables that indicate the state of the inventory level. We introduce binary variable $\hat{\chi}^0_{\xi,t,k,1}$ indicating if inventory in echelon 1 is empty, and variable and $\hat{\chi}^y_{\xi,t,k,e}$ indicating if inventory of echelon e is at basestock level. The definition for these variables is enforced with Equations (4.47)-(4.50).

$$\chi_{\xi,t,k,1} \leq M\left(1 - \hat{\chi}^0_{\xi,t,k,1}\right) \qquad \forall\, \xi \in \Xi,\ t \in T,\ k \in K \tag{4.47}$$

$$\chi_{\xi,t,k,e} \leq y_{t,k,e} \qquad \forall\, \xi \in \Xi,\ t \in T,\ k \in K, e \in E_k \tag{4.48}$$

$$\chi_{\xi,t,k,e} \geq y_{t,k,e} - M\left(1 - \hat{\chi}^y_{\xi,t,k,e}\right) \qquad \forall\, \xi \in \Xi,\ t \in T,\ k \in K, e \in E_k \tag{4.49}$$

$$\hat{\chi}^0_{\xi,t,k,1} + \hat{\chi}^y_{\xi,t,k,1} \leq 1 \qquad \forall\, \xi \in \Xi,\ t \in T,\ k \in K \tag{4.50}$$

The implication presented in Equation (4.45), preventing stockouts if inventory is available, can be reformulated with big-M constraint (4.51).

$$b_{\xi,t,m} \leq M\hat{\chi}^0_{\xi,t,k,1} \qquad \forall\, \xi \in \Xi,\ t \in T,\ k \in K, m = m_{k,1} \tag{4.51}$$

Finally, condition (4.46) can be reformulated with constraint (4.52).

$$\hat{\chi}^{y}_{\xi,t,k,e} + \sum_{m \in M_i^{in}} \hat{\mathrm{g}}_{\xi,t,i,m} + \hat{\mathrm{u}}^{0}_{\xi,t,i} \geq 1 \qquad \forall\, \xi \in \Xi,\ t \in T,\ k \in K, e \in E_k, i \in I^{r}_{m_{k,e}} \tag{4.52}$$

The MILP reformulation that we propose for multi-echelon arrangements is obtained by including Equations (4.43)-(4.44) and Equations (4.47)-(4.52) in the optimization model described by Equations (4.17)-(4.23). The resulting reformulation is only one reformulation of the logic proposed for multi-echelon arrangements. Other reformulations are also possible.

4.11 Evaluating inventory planning strategies with closed-loop Monte Carlo simulations

In order to asses the potential benefits of implementing a policy-based production planning, we compare the planning decisions obtained by solving the logic-based SP formulation with the decision obtained from the equivalent two-stage SP formulation. The challenge for large-scale problems is that the number of scenarios in multiperiod models grows exponentially; therefore, we cannot calculate the REV exactly as we have done with the illustrative example in Section 4.7. The alternative is to use the planning strategies in a receding horizon with the purpose of simulating the sequential implementation of the decision-making process. The proposed closed-loop Monte Carlo simulations resemble Economic MPC [184], but our focus is on finite planning horizons and we solve a stochastic programming problem at each time period.

The scenarios for the SP formulations represent possible values of the exogenous uncertain parameters, from the current period until the end of the planning horizon. We assume to have a probabilistic description of these parameters, which allow us generating possible trajectories using sampling techniques. The multiperiod SP formulations with sampled scenarios can be considered sample-path optimization problems [186]; the purpose of solving these sample-path problems is to estimate the optimal planning strategy based on a reduced set of scenarios.

Four different parameters must be specified for the implementation of the closed-loop simulations: number of replications, length of the simulation horizon, length of the planning horizon, and sample size for the planning problem. The number of replications specifies how many closed-loop simulations we run; a large number of replications is desirable because it allows better estimation of the simulation expected cost and its variance. The simulation horizon is the length of the simulation and specifies how many optimization problems we solve in each replication. The planning horizon is the length of the sample-paths used as scenarios in the multiperiod formulations; it defines how far into the future we look when solving the planning problem. Finally, the sample size specifies how many scenarios we include in the optimization problems; a larger number of scenarios tends to produce better approximations of the full problem, but the sample size is constrained by the computational complexity of the instances.

The closed-loop simulations are used to evaluate the performance of the proposed formulations for inventory and production planning. The procedure to estimate the expected performance of these planning strategies has the following steps:

1. Establish the parameters for the closed-loop simulations.
2. Start a replication of the closed-loop simulation ($t^* = 0$).
   - 2.1. Observe the state of the simulation model at time t^*.
   - 2.2. Generate the scenarios for the optimization problem by randomly sampling paths of the exogenous uncertain parameters.
   - 2.3. Formulate and solve the stochastic optimization problem.
   - 2.4. Implement in the simulation the optimal decisions corresponding to the current simulation time period (t^*).
   - 2.5. Randomly generate the realization of all exogenous uncertain parameters for the next simulation time period ($t^* + 1$).
   - 2.6. Roll the simulation time forward ($t^* = t^* + 1$).
   - 2.7. If simulation time is less than the simulation horizon, go back to Step 2.1. Otherwise, continue to Step 3.
3. If the current number of closed-loop simulations is less than the desired number of replications, go back to Step 2. Otherwise, continue to Step 4.
4. Calculate the statistics over all replications and terminate.

Figure 4.6 shows the trajectory of uncertain parameters in a closed-loop Monte Carlo simulation, where the past is represented by a unique path and the future is represented by alternative paths indicating possible scenarios. The simulation presented in Figure 4.6 is performed over five periods ($t^* = 0$ to $t^* = 4$). In each period, a stochastic SP problem with a 4-period planning horizon is solved. Then, time moves forward and uncertainty is revealed.

In the following examples, we compare the performance of the planning decisions obtained from the logic-based SP formulation and the equivalent two-stage SP formulation. For both planning models, we use exactly the same sampled scenarios in every instance. In addition, we use the same realizations of the uncertain parameters in the implementation of the closed-loop simulations. The mathematical models and the sampling procedure were implemented in AIMMS 4.8.3; all optimization problems were solved using GUROBI 6.0.0 on an Intel Core i7 CPU 2.93 Ghz processor with 4 GB of RAM.

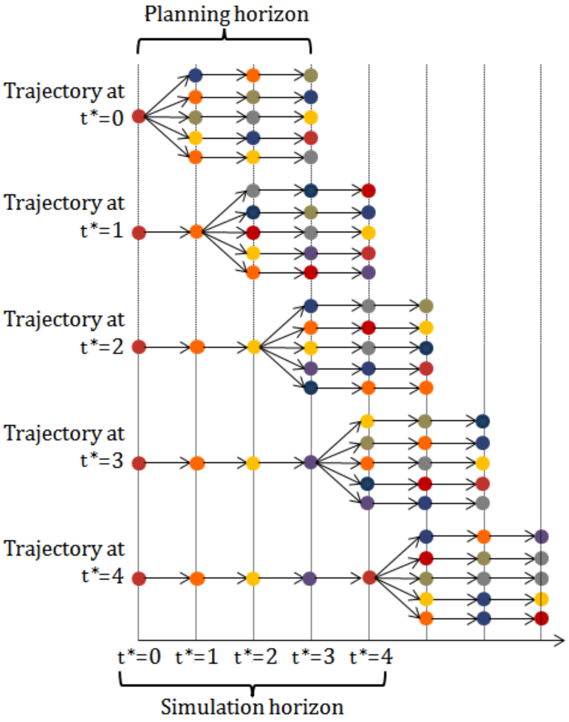


Figure 4.6: Trajectory of uncertain parameters in a closed-loop Monte Carlo simulation.

4.12 Example with a parallel arrangement of inventories

This example illustrates the implementation of the two-stage and the priority-policy approaches for inventory planning in processes networks with parallel arrangements and uncertain production capacities. The network structure of the example is shown in Figure 4.7. The purpose of the network is to transform raw material (m_0) into four products ($m_1, m_2, m_3,$ and m_4) with final demands. The units producing final products share the same raw material, which creates competition for the replenishment of inventories. The process network only allows storage of final products.

Raw material supply and final product demands are deterministic. Supply (S_{t,m_0}) is constant at 90 $ton/period$ throughout the time horizon. Demand ($D_{t,m}$) for final products is deterministic but time-varying. The demand profiles are presented in Figure 4.8.

Mass balance coefficients ($A_{i,m}$) indicating the amount of materials consumed per unit production rate are presented in Table 4.2; all mass balance coefficients for the amount of material produced per unit production rate are set equal to one ($B_{i,m} = 1$ ton of $m \ \forall \ i \in I_m^{prod}, \ m \in M_i^{out}$). Unit holding costs ($H_{t,m}$) and unit backorder costs ($P_{t,m}$) are constant in time; their values are given in Table 4.3.

Available production capacities ($C_{\xi,t,i}$) in the processing units are considered uncertain. Each uncertain parameter is modeled as an independent time-homogeneous Discrete Time Markov Chain (DTMC) with the purpose of describing the state-dependent evolution of uncertainty in industrial processes. The states of the DTMCs characterize the value of the uncertain parameters; each parameter has three states that imply different available production capacities. Table 4.4 shows the value of each uncertain parameter according to their state.

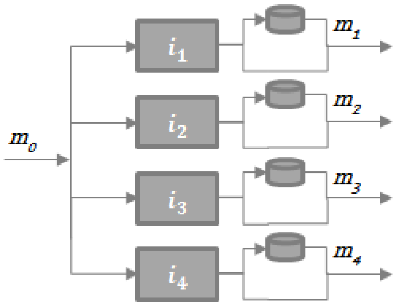


Figure 4.7: Structure of the example with a parallel arrangement of inventories.

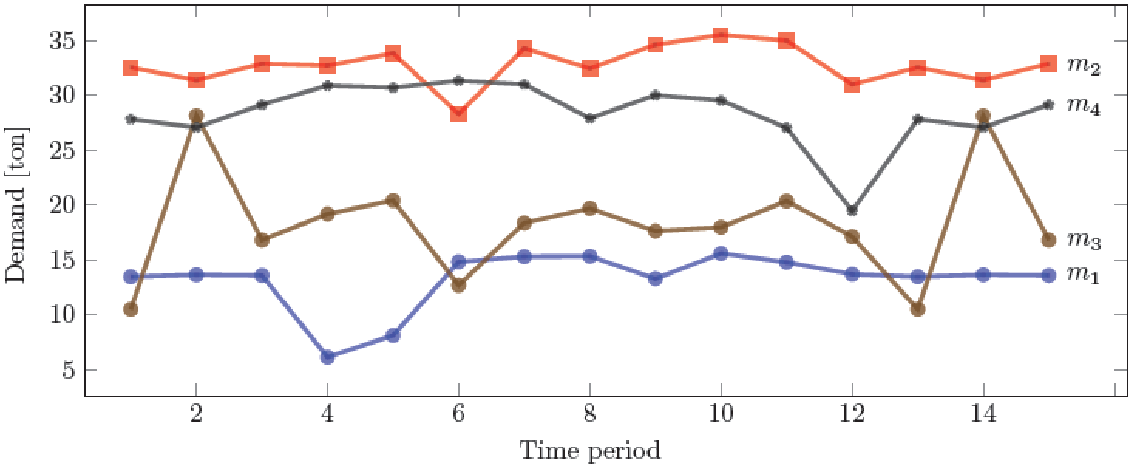


Figure 4.8: Deterministic demands in the example with a parallel arrangement of inventories.

Unit	A_{i,m_0} [ton of m_0]
i_1	1.180
i_2	1.355
i_3	0.724
i_4	0.570

Table 4.2: Consumption coefficients (A_{i,m_0}) in the example with a parallel arrangement of inventories.

Material	$H_{t,m}$ [$ ton/period]	$P_{t,m}$ [$ ton/period]
m_1	0.55	4.40
m_2	0.45	3.60
m_3	0.65	5.20
m_4	0.85	6.80

Table 4.3: Cost parameters in the example with a parallel arrangement of inventories.

Parameter	**State**		
	Low	$Nominal$	$High$
C_{ξ,t,i_1}	13.23	14.70	16.17
C_{ξ,t,i_2}	32.13	35.70	39.27
C_{ξ,t,i_3}	22.68	25.20	27.72
C_{ξ,t,i_4}	28.35	31.50	34.65

Table 4.4: Production capacities according to their DTMC state in the example with a parallel arrangement of inventories.

We assume that all uncertain parameters are initially at their nominal values. The evolution of each DTMCs is characterized with the one-step transition matrix (Π). The same transition matrix is used to model the evolution of all production capacities. The transition matrix is given by Equation (4.53).

$$\begin{array}{cc} & \begin{array}{ccc} Low & Nominal & High \end{array} \\ \Pi = & \left[\begin{array}{ccc} 0.70 & 0.25 & 0.05 \\ 0.15 & 0.70 & 0.15 \\ 0.05 & 0.25 & 0.70 \end{array}\right] \begin{array}{l} Low \\ Nominal \\ High \end{array} \end{array} \tag{4.53}$$

It is worth noticing that in a single time period, there are 4 uncertain parameters with 3 possible outcomes, giving rise to 81 possible combinations. In a multiperiod optimization problem with 6 time periods there are millions (81^6) of possible scenarios, which would result in an intractable model for any practical purpose.

We compare performance of the two-stage SP and the logic-based SP inventory planning strategies based on 25 closed-loop simulations. In each period, each strategy solves a stochastic optimization problem with 10 sampled scenarios and a planning horizon of 6 time periods. In the logic-based SP formulation, we enforce the priority policy for the parallel arrangement made up by the four final products ($\tilde{M}_1 = \{m_1, m_2, m_3, m_4\}$) in planning periods 2, 3, 4, and 5. It is unnecessary to enforce the policy in the first planning period because the uncertainty has already been revealed; enforcing the policy in the last planning period does not bring any benefit because no future periods can be anticipated. The length of the simulation horizon is set to 12 periods.

Table 4.5 presents the computational statistics for the two-stage SP formulation and the MILP reformulation of the logic-based SP model. The number of variables and constraints remain the

Statistic	**Formulation**	
	Two-stage SP	**Logic-based SP**
Constraints:	1,812	4,204
Continuous variables:	2,460	2,484
Binary variables:	0	976
Instances solved to optimality:	300	300
Mean CPU time of instances [s]:	< 1	176 (± 501)

Table 4.5: Computational statistics of the two-stage SP and the logic-based SP formulations in the example with a parallel arrangement of inventories.

same throughout the simulations because we use a receding horizon approach. All MILPs are solved to an optimality gap of 0.25%.

Table 4.5 shows a significant difference in the computational complexity of both models. It is important to remark that the two-stage SP formulation is strictly a relaxation of the logic-based SP formulation, and it has only a subset of the variables and constraints. As a consequence, the mean CPU time required to solve the instances of the two-stage SP model is less than one second; the mean CPU time for the instances of the logic-based SP model is 176 seconds, with a standard deviation of 501 seconds.

The results of the closed-loop simulations can be observed in Figures 4.9 and 4.10, where the shaded lines represent the cost trajectories for the individual replications and the solid lines are the averages over all replications. The figures show similar costs for both approaches, with a slightly higher stockout cost for the two-stage SP model that can be observed in periods 10 and 11. The trajectories presented in Figures 4.9-4.10 evidence significant variability in the results obtained from the implementation of both planning strategies. This variability is inherent to the nature of the problem, because uncertainty in production capacities constitutes a high risk for stockouts. The main performance metric for the planning strategies is the expected cost of simulations. Table 4.6 presents the mean cost for each planning model over all simulations, together with its standard deviation and the corresponding service level (type β).

The results from Table 4.6 show a 2.7% reduction in the total expected cost for the logic-based SP model in comparison to the two-stage SP model; the reduction is obtained from lower stockout costs without increasing the inventory cost significantly. Although the difference is rather small, the results suggest that the logic-based SP model is more effective at selecting the materials that are stored as inventories according to the representation of the future given by the scenarios. Despite

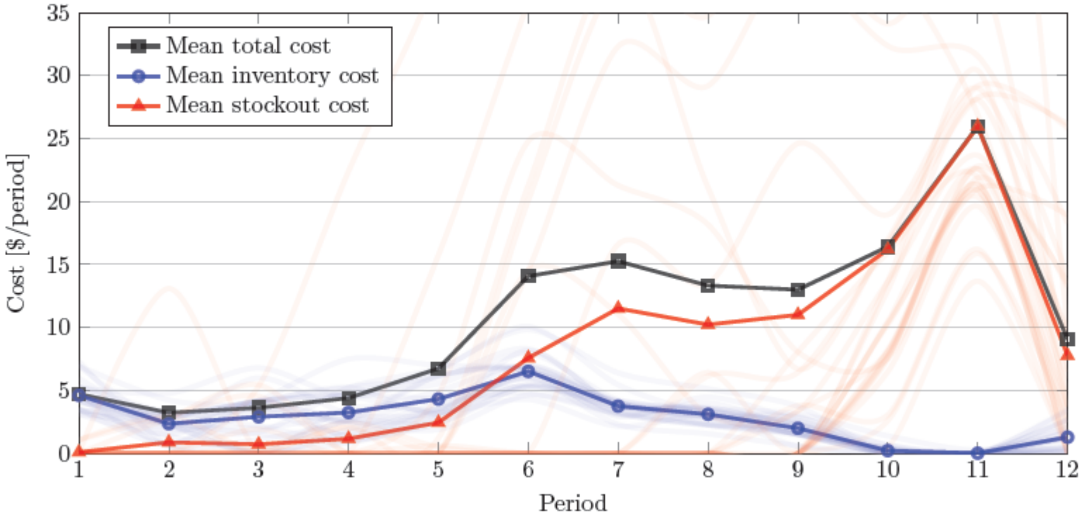


Figure 4.9: Trajectories of holding and stockout costs obtained from the two-stage SP formulation for the example with a parallel arrangement of inventories.

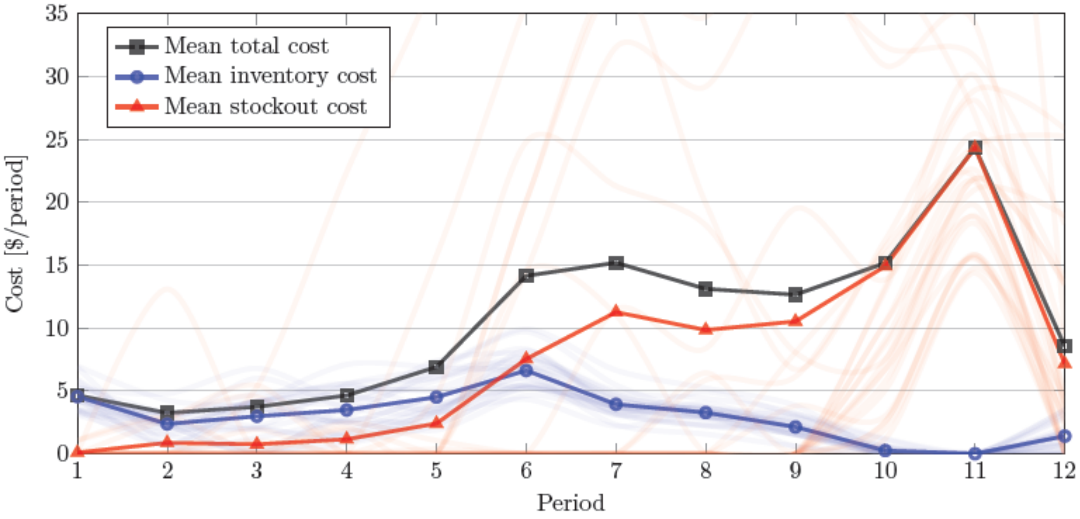


Figure 4.10: Trajectories of holding and stockout costs obtained from the logic-based SP formulation for the example with a parallel arrangement of inventories.

Metric	Model	
	Two-stage SP	**Logic-based SP**
Mean inventory cost [$]:	34.26	34.50
Mean stockout cost [$]:	95.54	90.80
Mean total cost [$]:	129.80	126.30
Standard deviation [$]:	110.34	110.69
Service level (β):	0.985	0.986

Table 4.6: Performance of the planning models in the example with a parallel arrangement of inventories.

the large variability in the simulation total costs, we can be confident in the advantages of the logic-based SP model because it consistently outperforms the two-stage SP model throughout the replications. A comparison of the total cost for each replication is presented in Figure 4.11; we observe that the two-stage SP model only outperforms the logic-based SP model by a negligible amount in 8 out 25 replications.

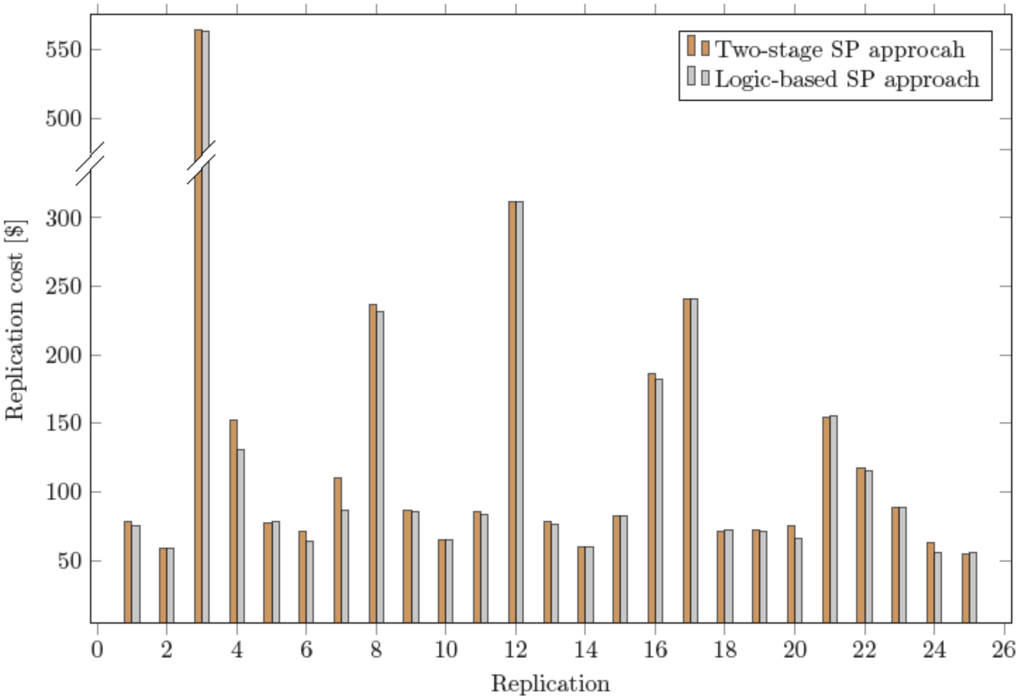


Figure 4.11: Comparison of replication costs for the example with a parallel arrangement of inventories.

4.13 Example with a multi-echelon arrangement of inventories

This example compares the inventory plan obtained from the two-stage SP formulation with the plan dictated by the logic-based SP formulation modeling the multi-echelon inventory policy. The example has been adapted from Example 1 presented by Terrazas-Moreno et al. [227], and originally proposed by Straub & Grossmann [220]. The purpose of the process network is to transform a single raw material (m_0) into one final product (m_2). The network offers three alternative processing paths, from which we identify one multi-echelon arrangement including both storable materials: $\bar{M}_1 = \{m_1, m_2\}$. The structure of the network and the echelons of $\bar{M}_1$ are shown in Figure 4.12.

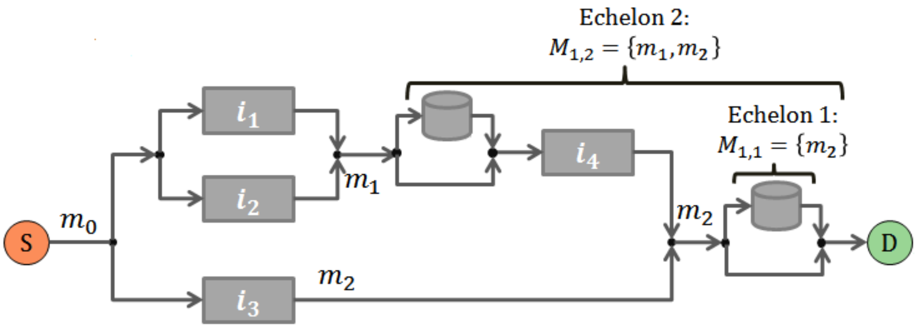


Figure 4.12: Structure of the example with a multi-echelon arrangement of inventories.

Supply availability (S_{ξ,t,m_0}), available production capacities ($C_{\xi,t,i}$), and demand (D_{ξ,t,m_2}) are considered uncertain. Supply and demand are modeled as normally distributed random variables; their mean values ($\bar{S}_{\xi,t,m_0}$, $\bar{D}_{\xi,t,m_2}$) are periodic functions presented in Figure 4.13; their coefficients of variation are set to 15%.

The capacity ($C_{\xi,t,i}$) of each processing unit is modeled as an independent time-homogeneous Discrete Time Markov Chain (DTMC) with the purpose of describing probabilistic failures. The DTMCs characterize the states of the units that can be either working normally (up) or failed ($down$). Table 4.7 shows the value of production capacities according to their state.

We assume that all units are initially at their up state. The evolution of each DTMCs is characterized with the one-step transition matrices (Π_i) given by Equation (4.54).

$$
\Pi_{i_1} = \begin{array}{cc} \begin{array}{cc} Up & Down \end{array} & \\ \left[\begin{array}{cc} 0.97 & 0.03 \\ 0.50 & 0.50 \end{array}\right] & \begin{array}{c} Up \\ Down \end{array} \end{array}
\qquad
\Pi_{i_2} = \begin{array}{cc} \begin{array}{cc} Up & Down \end{array} & \\ \left[\begin{array}{cc} 0.95 & 0.05 \\ 0.50 & 0.50 \end{array}\right] & \begin{array}{c} Up \\ Down \end{array} \end{array}
$$

$$
\Pi_{i_3} = \begin{array}{cc} \begin{array}{cc} Up & Down \end{array} & \\ \left[\begin{array}{cc} 0.96 & 0.04 \\ 0.50 & 0.50 \end{array}\right] & \begin{array}{c} Up \\ Down \end{array} \end{array}
\qquad
\Pi_{i_2} = \begin{array}{cc} \begin{array}{cc} Up & Down \end{array} & \\ \left[\begin{array}{cc} 0.93 & 0.07 \\ 0.50 & 0.50 \end{array}\right] & \begin{array}{c} Up \\ Down \end{array} \end{array}
\tag{4.54}
$$

From the individual states of the processing units, we know that there are 16 different discrete states for the entire system. The entire system could be characterize as a single DTMC, but it is unnecessary because we assume that the state transitions for each unit only depend on its own state. In addition to the discrete states characterizing production capacities, supply and demand are modeled with continuous distributions; therefore, the total number of scenarios is uncountable.

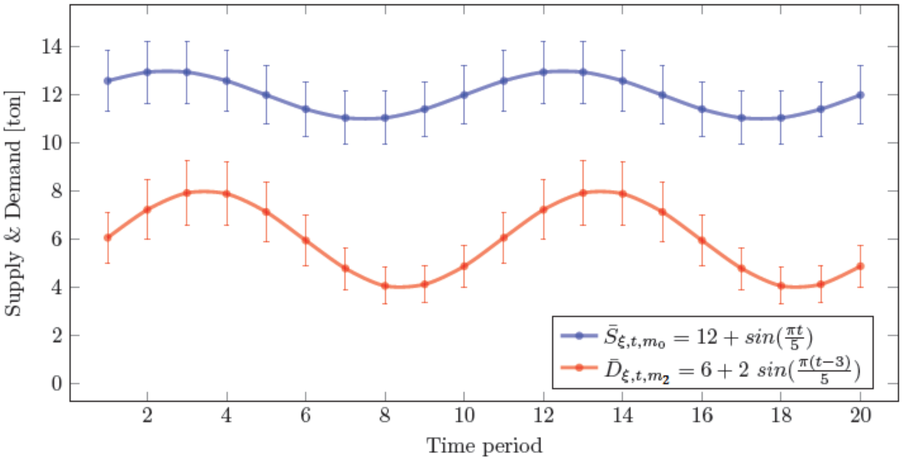


Figure 4.13: Normally distributed supply and demand in the example with a multi-echelon arrangement of inventories.

Parameter	**State**	
	Up	$Down$
C_{ξ,t,i_1}	5	0
C_{ξ,t,i_2}	5	0
C_{ξ,t,i_3}	7	0
C_{ξ,t,i_4}	9	0

Table 4.7: Production capacities according to the state of units in the example with a multi-echelon arrangement of inventories.

The remaining parameters of the example are deterministic; they are given in Tables 4.8 and 4.9. All mass balance coefficients indicating the amount of materials produced per unit production rate are set to one ($B_{i,m} = 1$ ton of $m \;\; \forall \, i \; \in \; I_m^{prod}$, $m \in M_i^{out}$). Unit holding costs ($H_{t,m}$) and unit backorder costs ($P_{t,m}$) are constant in time.

We compare the performance of the two-stage SP and the logic-based SP inventory planning strategies based on 25 closed-loop simulations. In each period, both strategies solve a stochastic optimization problem with 10 sampled scenarios and a planning horizon of 5 time periods. In the logic-based SP formulation, we enforce the multi-echelon basestock policy for arrangement $\bar{M}_1$. The length of the simulation horizon is set to 15 periods.

The computational statistics for the instances of each formulation are presented in Table 4.10. All MILPs are solved to an optimality gap of 0.25%.

Unit	A_{i,m_0} [ton of m_0]	A_{i,m_1} [ton of m_1]
i_1	1.087	-
i_2	1.111	-
i_3	1.176	-
i_4	-	1.333

Table 4.8: Consumption coefficients ($A_{i,m}$) in the example with a multi-echelon arrangement of inventories.

Material	$H_{t,m}$ [\$ ton/period]	$P_{t,m}$ [\$ ton/period]
m_1	1	-
m_3	3	10

Table 4.9: Cost parameters in the example with a multi-echelon arrangement of inventories.

Statistic	**Formulation**	
	Two-stage SP	**Logic-based SP**
Constraints:	1,304	2,204
Continuous variables:	1,750	1,760
Binary variables:	0	450
Instances solved to optimality:	300	300
Mean CPU time of instances [s]:	< 1	189 (± 487)

Table 4.10: Computational statistics of the two-stage SP and the logic-based SP formulations in the example with a multi-echelon arrangement of inventories.

The cost trajectories for the two-stage SP and the logic-based SP models are presented in Figures 4.14 and 4.15, respectively. Shaded lines represent the trajectories for individual replications and solid lines are the averages. The figures show a trend for the two-stage SP model to produce higher inventory costs; this can be observed at periods 3, 4, and 5.

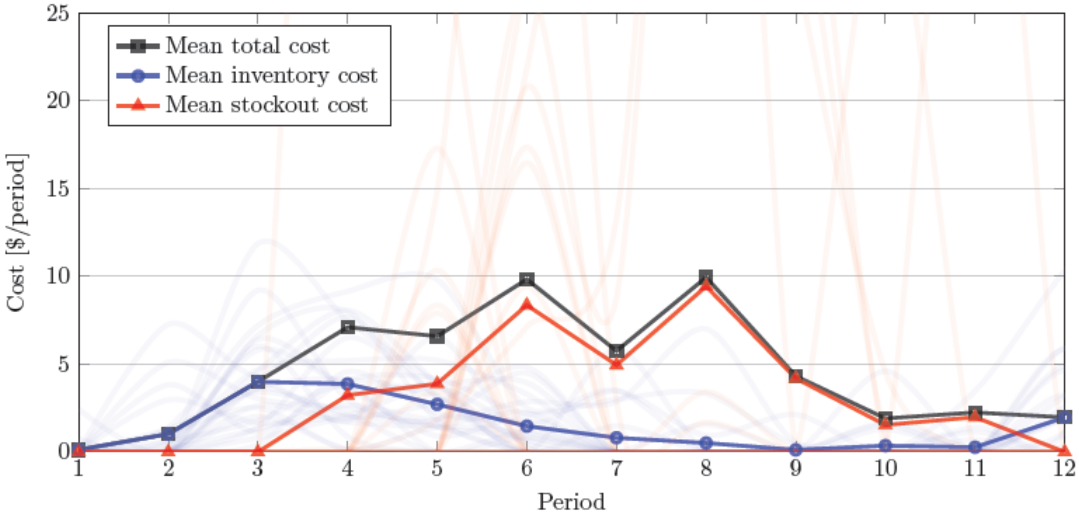


Figure 4.14: Trajectories of holding and stockout costs obtained from the two-stage SP formulation for the example with a multi-echelon arrangement of inventories.

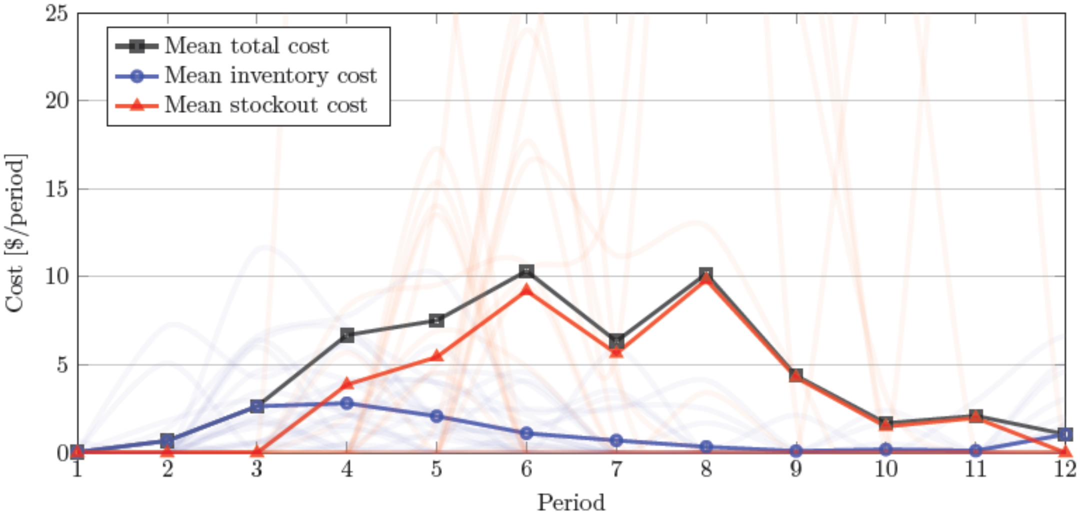


Figure 4.15: Trajectories of holding and stockout costs obtained from the logic-based SP formulation for the example with a multi-echelon arrangement of inventories.

The trajectories in Figures 4.14-4.15 show significant variability for holding and stockout cost across replications. Variability in this process network is the result of random failures that produce high stockout risk. The performance metrics for the planning models are presented in Table 4.11.

Metric	**Model**	
	Two-stage SP	**Logic-based SP**
Mean inventory cost [$]:	32.30	21.07
Mean stockout cost [$]:	42.33	48.31
Mean total cost [$]:	74.63	69.38
Standard deviation [$]:	59.10	56.34
Service level (β):	0.958	0.952

Table 4.11: Performance of the planning models in the example with a multi-echelon arrangement of inventories.

We observe from Table 4.11 a reduction in the mean total cost obtained from the logic-based SP model that corresponds to 7.0% of the cost obtained from the two-stage SP model. The reduction is the result of an inventory planning strategy that is more effective at balancing holding and backorders cost; the logic-based SP model benefits from an increased coordination between intermediate and final product inventory levels.

Finally, in Figure 4.16 we present the cost obtained for each replication using the two-stage SP and the logic-based SP planning strategies. The figure shows that the logic-based SP approach yields a lower cost than the two-stage SP approach in 19 out of 25 replications. These results clearly illustrate the advantages of the multi-echelon basestock policy for inventory planning in process networks.

4.14 Summary

In this chapter, we have proposed a policy-based approach for stochastic inventory planning in process networks. Our motivation originates from the effectiveness of policies for inventory management and their appeal for industrial implementation. Given the difficulty to obtain optimal policies analytically in process networks, we have developed the logic describing these policies with the purpose of including them into the production and inventory planning problem.

We have proposed two sets of logic rules for inventory planning in networks with parallel and sequential structures. The logic is formulated as a GDP model that avoids anticipativity in stochastic

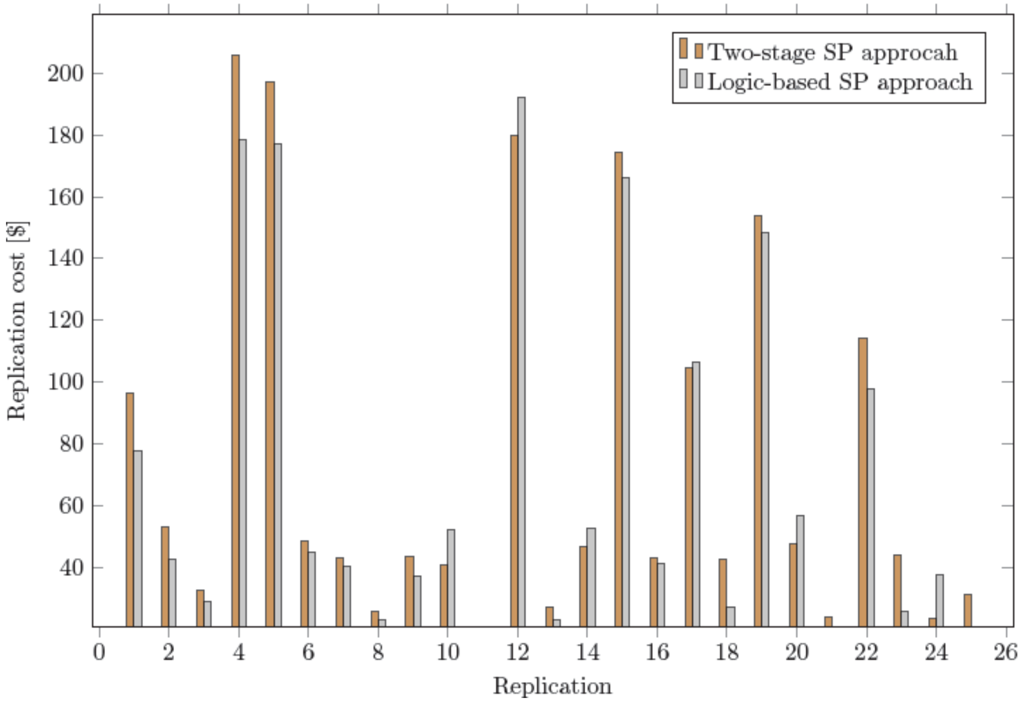


Figure 4.16: Comparison of replication costs for the example with a multi-echelon arrangement of inventories.

programing problems and yields the optimal parameters of inventory policies. We implemented the MILP reformulations of our logic-based SP models in two examples, and compared the results with the corresponding two-stage SP models. The comparisons were based on closed-loop simulations that mimic the actual implementation of these planning strategies in an industrial environment. Despite the increase in the computational complexity of the instances, the examples show a significant improvement in the inventory plans obtained from the logic-based SP model.

The proposed logic-based SP formulation has the advantage of being completely flexible with respect to the probabilistic description of the uncertain parameters. The only requirement for the model is to be able to generate scenarios describing the evolution of uncertain parameters by any forecasting method. This feature is specially important for industrial applications in which correlation and autocorrelation of the uncertain parameters is very common, and allows using historical data in the inventory planning model.

The logic developed for inventory planning in process networks with parallel and sequential structures can be extended to address networks of arbitrary topology with complex uncertainty models.

There is an extraordinary potential for inventory optimization in these networks because their complexity conceals the most effective planning strategies. This contribution offers a novel approach for a very challenging problem in the process industry.

Chapter 5

Bilevel Optimization for Capacity Planning with Rational Markets

5.1 Motivation

Capacity expansion is one of the most important strategic decisions for industrial gas companies. In this industry, most of the markets are served by local producers because of the competitive advantage given by the location of production plants. The dynamics of the industrial gas markets imply that companies must anticipate demand increases in order to plan their capacity expansion, maintain supply availability, and avoid regional incursion of new producers. The selection of the right investment and distribution plan plays a critical role for companies in this environment. A rigorous approach based on mathematical modeling and optimization offers the possibility to find the investment and distribution plan that yields the greatest economic benefit.

Since the late 1950s, capacity expansion planning has been studied to develop models and solution approaches for diverse applications in the process industries [195], communication networks [31], electric power services [167], and water resource systems [168]. Capacity planning is considered a central problem for enterprise-wide optimization, a topic for which comprehensive reviews are available [92, 93]. Despite the importance of capacity expansion in industry, the study of the problem in a competitive environment has not received much attention. Soyster & Murphy [219] formulated a capacity planning problem for a perfectly competitive market. However, perfect competition is a strong assumption. A more realistic hypothesis is to assume an oligopolistic market as presented by Murphy & Smeers [166]. Similar models based on game theory have also been used for the supply chain planning in cooperative and competitive environments [254].

The competition between two players whose decisions are made sequentially can be modeled as a Stackelberg game [239]. A Stackelberg competition is an extensive game with perfect information in which the leader chooses his actions before the follower has the opportunity to play. It is well known that the most interesting equilibria of such games correspond to the solution of a bilevel optimization problem [171].

Bilevel optimization problems are mathematical programs with optimization problems in the constraints [23]. They are suitable to model problems in which two independent decision makers try to optimize their own objective functions sequentially [28, 9]. We present a mixed-integer linear bilevel formulation for the capacity planning of an industrial gas company operating in a competitive environment. The purpose of the upper-level problem is to determine the investment and distribution plan that maximizes the Net Present Value (NPV). The response of markets that can choose among different producers is modeled in the lower-level as a Linear Programming (LP) problem. The lower-level objective function is selected to represent the rational behavior of the markets.

Solution approaches for bilevel optimization problems with lower-level LPs leverage the fact that optimal solutions occur at vertices of the region described by upper and lower level constraints. They rely on vertex enumeration, directional derivatives, penalty terms, or optimality conditions [192]. The most direct approach is to reformulate the bilevel optimization as a single-level problem using the optimality conditions of the lower-level LP. The classical reformulation using Karush-Kuhn-Tucker (KKT) conditions maintains linearity of the problem except for the introduction of complementarity constraints [72, 8, 16]. An equivalent reformulation replaces the lower level problem by its primal and dual constraints, and guarantees optimality by enforcing strong duality [164, 75].

The novelty of our research resides on the application of bilevel optimization for capacity expansion planning in a competitive environment. Bilevel programming for these kind of problems can be seen as a risk mitigation strategy given the significant influence of external decision-makers in the economic success of investment plans. In particular, we propose a mathematical model that includes a rational market behavior beyond the traditional game theoretical models. The investment plans obtained from this approach are found to be less sensitive to changes in the business environment in comparison to the single-level models.

In order to solve the challenging bilevel formulation, we implement the KKT and the duality-based reformulations. The results obtained from examples of different sizes show the advantages of the duality-based reformulation in terms of computational effort. Despite the efficiency obtained with this reformulation, we found necessary to implement two additional improvement strategies to solve large-scale instances.

The remaining of this chapter is organized as follows. Section 5.2 reviews the publications that are relevant for our capacity planning model. In Section 5.3, we describe the problem. In Section 5.4, we present the single-level capacity planning formulation. Section 5.5 presents the bilevel capacity planning problem with rational markets. In Section 5.6, we develop two reformulations that allow solving the bilevel optimization problem. Section 5.7, presents a small example that illustrates the proposed formulations. Subsequently, in Section 5.8 we evaluate the performance of the proposed reformulations with a middle-size example. In section 5.9, we elaborate on solution approaches for large-scale bilevel capacity planning problems. Section 5.10 presents an industrial example. Finally, in Section 5.11 we present our analysis and concluding remarks.

5.2 Literature review

A rather large body of literature has been published on capacity planning problems for several industries [151]. Sahinidis et al. [196] proposed a comprehensive MILP model for long range planning of process networks. Van den Heever & Grossmann [233] used disjunctive programming to extend this methodology to multiperiod design and planning of nonlinear chemical processes. An MILP formulation that integrates scheduling with capacity planning for product development was presented by Maravelias & Grossmann [153]. Sundaramoorthy et al. [224] proposed a two-stage stochastic programming formulation for the integration of capacity and operations planning.

Strategic investment planning for electric power networks has been the most prolific application of bilevel optimization models. Motto et al. [164] implemented the duality-based reformulation for the analysis of electric grid security under disruptive threats. This bilevel problem was originally formulated by Salmeron et al. [198] with the purpose of identifying the interdictions that maximize network disruptions. A bilevel formulation for the expansion of transmission networks was developed by Garces et al. [75] to maximize the average social welfare over a set of lower-level problems representing different market clearing scenarios; they implemented the duality-based reformulation. Ruiz et al. [188] modeled electricity markets as an Equilibrium Problem with Equilibrium Constraints (EPEC) in which competing producers maximize their profit in the upper level and a market operator maximizes social welfare in the lower level; they use the duality-based reformulation to guarantee optimality of the lower level problem and obtain an equilibrium solution by jointly formulating the KKT conditions of all producers. A similar strategy that includes the combination of duality-based and KKT reformulations was used by Huppmann & Egerer [114] to solve a three-level optimization problem that models the roles of independent system operators, regional planners, and supra-national coordination in the European energy system.

Another interesting application of bilevel optimization is the facility location problem in a duopolistic environment. The model presented by Fischer [69] selects facilities among a set of candidate locations and considers selling prices as optimization variables, which leads to a nonlinear bilevel formulation. The problem is simplified to a linear discrete bilevel formulation under the assumption that Nash equilibrium is reached for the prices. The solution of the discrete bilevel optimization problem is obtained using a heuristic algorithm.

Bilevel optimization models have also found application in chemical engineering. Clark & Westerberg [42] presented a nonlinear bilevel programming approach for the design of chemical processes and proposed algorithms to solve them. In their formulation, the upper level optimizes the process design and the lower level models thermodynamic equilibrium by minimizing Gibbs free energy. Burgard & Maranas [25] used bilevel optimization to test the consistency of experimental data obtained from metabolic networks with hypothesized objective functions. In the upper level, the problem minimizes the square deviation of the fluxes predicted by the metabolic model with respect to experimental data, whereas the lower level quantifies the individual importance of the fluxes. A bilevel programming model for supply chain optimization under uncertainty was developed by Ryu et al. [190]; the conflicting interests of production and distribution operations in a supply chain are modeled using separate objective functions. They reformulate the bilevel problem in a single level after finding the solution of the lower-level problem as parametric functions of the upper-level variables and the uncertain parameters. Chu & You [38] presented an integrated scheduling and dynamic optimization problem for batch processes. The scheduling problem, formulated in the upper level, is subject to the processing times and costs determined by the nonlinear dynamic lower-level problem. The bilevel formulation is transformed to a single level problem by replacing the lower-level with piece-wise linear response functions. They assert that the bilevel formulation can be used as a distributed optimization approach whose solutions can easily adapt to variation in the problem's parameters.

It should also be noted that bilevel programming for nonlinear models has been the subject of research in chemical engineering. Faisca et al. [62] presented a multi-parametric programming approach that replaces the lower-level problem by its rational reaction set parametrized on the upper-level variables. For global optimization of continuous and mixed-integer bilevel problems, Kleniati & Adjiman [129] developed the Branch-&-Sandwich algorithm, which solves bilevel programs with noncovex lower-level problems.

5.3 Problem statement

A company that produces and commercializes industrial products in a given geographic region is interested in developing an investment plan to expand its capacity in anticipation of future demand increase. The company operates some plants with limited production capacity. Existing plants are eligible for capacity expansion and other locations are candidates to open new plants. The construction and expansion of plants requires the investment of capital to develop the project and install new production lines. The potential increases in production capacity are assumed to be discrete and the corresponding investments are given by fixed costs. Based on the available capacity in the plants, the company allocates production to satisfy market demands. Figure 5.1 shows a schematic representation of a region with several production plants and gas markets.

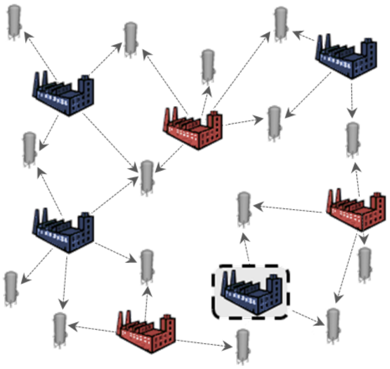


Figure 5.1: Superstructure of regional gas markets.

The company obtains revenue from selling its products at fixed prices in each market. The goal of the company is to find the investment plan that maximizes the Net Present Value (NPV) of its profit during a finite time horizon. The NPV is calculated by applying the appropriate discount factor to the income received from sales and the expenses related to investment, production, maintenance, and transportation costs.

5.4 Single-level capacity planning with captive markets

The basic model to plan the capacity expansion of a company serving industrial markets assumes that all market demands are willing to buy the products at the price offered. In this context, markets are regarded as captive. The capacity expansion planning with captive markets can be formulated as the single-level Mixed-Integer Linear Program (MILP) presented in Equations (5.1)-(5.8),

$$
\begin{aligned}
\max \quad & \sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k} \\
& -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\frac{1}{(1+R)^t}\left(A_{t,i}v_{t,i}+B_{t,i}w_{t,i}\right) \\
& -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{k\in K}\frac{1}{(1+R)^t}\left(E_{t,i,k}x_{t,i,k}+F_{t,i,k}\sum_{j\in J}y_{t,i,j,k}\right) \\
& -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}G_{t,i,j,k}y_{t,i,j,k} && (5.1)
\end{aligned}
$$

$$
\begin{aligned}
\text{s.t.} \quad & w_{t,i}=V_{0,i}+\sum_{t'\in T'_t}v_{t',i} && \forall\, t\in T, i\in I^{\mathcal{L}} && (5.2) \\
& x_{t,i,k}\leq w_{t,i} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K && (5.3) \\
& c_{t,i,k}=C_{0,i,k}+\sum_{t'\in T'_t}H_{i,k}x_{t',i,k} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K && (5.4) \\
& \sum_{j\in J}y_{t,i,j,k}\leq c_{t,i,k} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K && (5.5) \\
& \sum_{i\in I^{\mathcal{L}}}y_{t,i,j,k}\leq D_{t,j,k} && \forall\, t\in T, j\in J, k\in K && (5.6) \\
& v_{t,i},\ w_{t,i},\ x_{t,i,k}\in\{0,1\} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K && (5.7) \\
& c_{t,i,k},\ y_{t,i,j,k}\in\mathbb{R}^+ && \forall\, t\in T, i\in I^{\mathcal{L}}, j\in J, k\in K && (5.8)
\end{aligned}
$$

where T, $I^{\mathcal{L}}$, J, and K are respectively, the index sets for time (t), production plants of the decision maker ($i^{\mathcal{L}}$), markets (j), and products (k). We also define T' as the subset of time periods T in which expansions are allowed, and T'_t as the subset of time periods before t in which expansions are allowed. Formally, $T'_t = \{t' : t' \in T', t' \leq t\}$.

The first term in expression (5.1) represents the income obtained from sales. Income is proportional to demand assignments ($y_{t,i,j,k}$) according to the price paid by the markets ($P_{t,i,j,k}$). The second term includes the cost of opening new facilities and the maintenance cost of open plants. The binary variable deciding if a new plants is open at location i at time period t is $v_{t,i}$; parameter $A_{t,i}$ determines the fixed cost to build a new plant. The binary variable $w_{t,i}$ indicates if plant i is open at time period t; if the plant is open, a fixed cost $B_{t,i}$ must be paid for maintenance in period t. The third term includes expansion and production costs. The expansion of production capacity for product k in plant i at period t is decided with binary variable $x_{t,i,k}$; the cost of expansions is given by parameter $E_{t,i,k}$. Production costs are proportional to demand assignments ($y_{t,i,j,k}$) according to their unit production cost ($F_{t,i,k}$). Finally, the last term represents the transportation cost from production plants to markets. Transportation is proportional to demand assignments ($y_{t,i,j,k}$) according to the unit transportation cost ($G_{t,i,j,k}$). All terms are discounted in every time period with an interest rate (R).

Constraint set (5.2) is used to model the maintenance cost of plants during the time periods when they are open; the binary parameter V_i^0 indicates the plants that are initially open. Constraint set (5.3) requires capacity expansions to take place only at open plants. Constraint set (5.4) determines the production capacity of plants according to the expansion decisions; parameters $C_{0,i,k}$ indicate the initial capacities and H_k is the magnitude of the potential capacity expansion. Constraints (5.5) bound the demand assignments according to the production capacities. Finally, constraints set (5.6) bounds demand assignments according to market demands. The domains of the variables are given by expressions (5.7) and (5.8).

5.5 Bilevel capacity planning with rational markets

The most intuitive way to model a competitive environment is to assume that the markets have the possibility to select their providers according to their own interest. The rational behavior of the markets can be modeled with a mathematical program that optimizes their objective function. The behavior of the markets is included in the constraints of the capacity planning problem, yielding a bilevel optimization formulation. In this formulation, the upper-level problem is intended to find the optimal capacity expansion plan by selecting the investments that maximize the NPV of the leader. The lower-level represents the response of markets that select production plants as providers with the unique interest of satisfying their demands at lowest cost.

The formulation presented in Section 5.4 is modified to ensure that market demands are completely satisfied. This is achieved by transforming constraint set (5.6) into equality constraints. This change is necessary to avoid the market cost from dropping to zero by leaving all demands unsatisfied. Additionally, the set of potential providers is expanded to include plants from independent producers. We assume that the initial capacity of all production plants is large enough to satisfy all market demands regardless of the expansion plan of the leader. This assumption is also useful to avoid unprofitable investments in capacity expansions driven by the need to maintain feasibility of the problem.

The products offered by the competing producers are considered homogeneous and the markets have no other preference for producers than price. Cases in which the markets have no preference between two or more plants are resolved by the upper level according to the interest of the leader; this modeling framework is known as the *optimistic approach* [149]. In our model, the *optimistic approach* is a key assumption because all plants controlled by the leader offer the same price to each market. Therefore, the optimization problem of the markets is degenerate. However, the markets are only concerned about selecting the producer that offers the lowest price and they are indifferent to the plant from which they are served; consequently, the leader is free to choose the plants it uses to satisfy its demands.

The bilevel optimization formulation for the capacity expansion planning in a competitive environment is presented in Equations (5.9)-(5.19),

$$\max_{v,w,x} \quad \sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k}$$

$$-\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\frac{1}{(1+R)^t}\left(A_{t,i}v_{t,i}+B_{t,i}w_{t,i}\right)$$

$$-\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{k\in K}\frac{1}{(1+R)^t}\left(E_{t,i,k}x_{t,i,k}+F_{t,i,k}\sum_{j\in J}y_{t,i,j,k}\right)$$

$$-\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}G_{t,i,j,k}y_{t,i,j,k} \tag{5.9}$$

$$\text{s.t.} \quad w_{t,i}=V_i^0+\sum_{t'\in T_t'}v_{t',i} \qquad \forall\, t\in T, i\in I^{\mathcal{L}} \tag{5.10}$$

$$x_{t,i,k}\le w_{t,i} \qquad \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \tag{5.11}$$

$$c_{t,i,k}=C_{i,k}^0+\sum_{t'\in T_t'}H_{i,k}x_{t',i,k} \qquad \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \tag{5.12}$$

$$\min_{y} \quad \sum_{t\in T}\sum_{i\in I}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k} \tag{5.13}$$

$$\text{s.t.} \quad \sum_{j\in J}y_{t,i,j,k}\le c_{t,i,k} \qquad \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \tag{5.14}$$

$$\sum_{j\in J}y_{t,i,j,k}\le C_{i,k}^0 \qquad \forall\, t\in T, i\in I^{\mathcal{C}}, k\in K \tag{5.15}$$

$$\sum_{i\in I}y_{t,i,j,k}=D_{t,j,k} \qquad \forall\, t\in T, j\in J, k\in K \tag{5.16}$$

$$y_{t,i,j,k}\in\mathbb{R}^+ \qquad \forall\, t\in T, i\in I, j\in J, k\in K \tag{5.17}$$

$$c_{t,i,k}\in\mathbb{R}^+ \qquad \forall\, t\in T, i\in I^{\mathcal{L}}, j\in J, k\in K \tag{5.18}$$

$$v_{t,i},\ w_{t,i},\ x_{t,i,k}\in\{0,1\} \qquad \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \tag{5.19}$$

where I is the set of all production plants, $I^{\mathcal{L}}\subset I$ is the subset of plants controlled by the leader, and $I^{\mathcal{C}}\subset I$ is the subset of plants controlled by the competitors. It should be noted that Equations (5.9)-(5.12) are identical to Equations (5.1)-(5.4) of the single-level formulation. However, in the bilevel formulation the upper-level decision-maker only controls variables $v_{t,i}$, $w_{t,i}$, $x_{t,i,k}$, and

$c_{t,i,k}$. Demand assignment decisions ($y_{t,i,j,k}$) are controlled by the lower level with the objective of minimizing the cost paid by the markets according to Equation (5.13). Equations (5.14) and (5.15) constrain the production capacity of the plants; Equation (5.16) enforces demand satisfaction in every time period. The domains of the variables are presented in Equations (5.17)-(5.19). It is important to note that upper-level variables only take discrete values and all lower-level variables are continuous. This attribute of the model is crucial for the reformulations that we propose.

5.6 Reformulation as a single-level optimization problem

An optimistic bilevel program with a convex and regular lower-level can be transformed into a single-level optimization problem using its optimality conditions [43]. The key property of convex programs is that their KKT conditions are necessary and sufficient to characterize the corresponding global optimal solutions. In the case of linear programs, KKT optimality conditions are equivalent to the satisfaction of primal feasibility, dual feasibility, and strong duality [74]. Based on this equivalence, we derive two single-level reformulations for the bilevel capacity planning model; both reformulations yield *optimistic* solutions of the problem.

5.6.1 KKT reformulation

The classic reformulation for bilevel programs with a lower-level LP is to replace the lower lower-level problem by its KKT conditions. In the case of the capacity planning with rational markets, the KKT reformulation is obtained by introducing constraints that guarantee the stationarity conditions, primal feasibility, dual feasibility, and complementary slackness for the cost minimization problem modeling markets behavior. The resulting reformulation is presented in Equations (5.20)-(5.33),

$$
\begin{aligned}
\max \quad & \sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k} \\
& -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\frac{1}{(1+R)^t}\left(A_{t,i}v_{t,i}+B_{t,i}w_{t,i}\right) \\
& -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{k\in K}\frac{1}{(1+R)^t}\left(E_{t,i,k}x_{t,i,k}+F_{t,i,k}\sum_{j\in J}y_{t,i,j,k}\right) \\
& -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}G_{t,i,j,k}y_{t,i,j,k} && (5.20) \\
\text{s.t.} \quad & w_{t,i}=V_i^0+\sum_{t'\in T'_t}v_{t',i} && \forall\, t\in T, i\in I^{\mathcal{L}} \quad (5.21) \\
& x_{t,i,k}\le w_{t,i} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \quad (5.22) \\
& c_{t,i,k}=C_{i,k}^0+\sum_{t'\in T'_t}H_{i,k}x_{t',i,k} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \quad (5.23) \\
& \sum_{j\in J}y_{t,i,j,k}\le c_{t,i,k} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \quad (5.24) \\
& \sum_{j\in J}y_{t,i,j,k}\le C_{i,k}^0 && \forall\, t\in T, i\in I^{\mathcal{C}}, k\in K \quad (5.25) \\
& \sum_{i\in I}y_{t,i,j,k}=D_{t,j,k} && \forall\, t\in T, j\in J, k\in K \quad (5.26) \\
& \frac{1}{(1+R)^t}P_{t,i,k}+\lambda_{t,j,k}+\mu_{t,i,k}-\gamma_{t,i,j,k}=0 && \forall\, t\in T, i\in I, j\in J, k\in K \quad (5.27) \\
& \mu_{t,i,k}\left(\sum_{j}y_{t,i,j,k}-c_{t,i,k}\right)=0 && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \quad (5.28) \\
& \mu_{t,i,k}\left(\sum_{j}y_{t,i,j,k}-C_{i,k}^0\right)=0 && \forall\, t\in T, i\in I^{\mathcal{C}}, k\in K \quad (5.29) \\
& \gamma_{t,i,j,k}\; y_{t,i,j,k}=0 && \forall\, t\in T, i\in I, j\in J, k\in K \quad (5.30) \\
& y_{t,i,j,k},\ \mu_{t,i,k},\ \gamma_{t,i,j,k}\in\mathbb{R}^+ && \forall\, t\in T, i\in I, j\in J, k\in K \quad (5.31) \\
& \lambda_{t,j,k}\in\mathbb{R}, && \forall\, t\in T, j\in J, k\in K \quad (5.32) \\
& c_{t,i,k}\in\mathbb{R}^+ && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \quad (5.33) \\
& v_{t,i},\ w_{t,i},\ x_{t,i,k}\in\{0,1\} && \forall\, t\in T, i\in I^{\mathcal{L}}, k\in K \quad (5.34)
\end{aligned}
$$

where $\mu_{t,i,k}$, $\lambda_{t,j,k}$, and $\gamma_{t,i,j,k}$ are the Lagrange multipliers of the lower-level constraints presented in Equations (5.14)-(5.15), (5.16), and (5.17), respectively. The upper-level problem is kept unchanged as shown in Equations (5.20)-(5.23). Constraints (5.24)-(5.26) ensure primal feasibility of the lower level; the constraints presented in Equation (5.27) are the stationary conditions for the lower level; Equations (5.28) and (5.29) represent the complementary conditions corresponding to inequalities (5.14) and (5.15); the constraints (5.30) are the complementary conditions corresponding to the domain of the lower-level variables presented in Equation (5.17). The domains are presented in Equations (5.31)-(5.34).

The main disadvantage associated to this reformulation is the introduction of non-linear complementary constraints. In order to avoid the solution of a nonconvex Mixed-Integer Non-Linear Program (MINLP), the complementary constraints can be formulated as disjunctions that are transformed into mixed-integer constraints [94]. In particular, we rewrite Equations (5.24) and (5.25) as equality constraints by introducing the slack variables $s_{t,i,k}$,

$$\sum_{j \in J} y_{t,i,j,k} + s_{t,i,k} = c_{t,i,k} \qquad \forall \ t \in T, i \in I^{\mathcal{L}}, k \in K \tag{5.35}$$

$$\sum_{j \in J} y_{t,i,j,k} + s_{t,i,k} = C^0_{i,k} \qquad \forall \ t \in T, i \in I^{\mathcal{C}}, k \in K \tag{5.36}$$

and use the Big-M reformulation to express that either constraints (5.35) and (5.36) are active or the corresponding multipliers ($\mu_{t,i,k}$) are zero. The Big-M constraints modeling this disjunction are presented in Equation (5.37) using binary variable $z^1_{t,i,k}$.

$$\left.\begin{aligned} & s_{t,i,k} \leq M z^1_{t,i,k} \\ & \mu_{t,i,k} \leq M \left(1 - z^1_{t,i,k}\right) \\ & z^1_{t,i,k} \in \{0,1\} \end{aligned}\right. \qquad \forall \ t \in T, i \in I, k \in K \tag{5.37}$$

Similarely, the Big-M reformulation of constraint set (5.30) is presented in Equation (5.38).

$$\left.\begin{aligned} & y_{t,i,j,k} \leq M z^2_{t,i,k} \\ & \gamma_{t,i,j,k} \leq M \left(1 - z^2_{t,i,k}\right) \\ & z^2_{t,i,j,k} \in \{0,1\} \end{aligned}\right. \qquad \forall \ t \in T, i \in I, j \in J, k \in K \tag{5.38}$$

We obtain the linearized KKT reformulation of the bilevel model presented in Section 5.5 by replacing constraints (5.24), (5.25), (5.28), (5.29) and (5.30) with constraints (5.35), (5.36), (5.37), and (5.38). The resulting single-level reformulation yields optimal solutions for the bilevel capacity planning model.

5.6.2 Duality-based reformulation

The alternative reformulation for the bilevel capacity planning problem is obtained by introducing constraints that guarantee the satisfaction of strong duality [164, 75]. This is achieved by replacing the lower-level problem described by Equations (5.13)-(5.17) with its primal and dual constraints, and equating their objective functions. The dual formulation corresponding to the lower-level LP is presented by Equations (5.39)-(5.42).

$$\max \quad \sum_{t\in T}\sum_{k\in K}\left[\sum_{j\in J} D_{t,j,k}\lambda_{t,j,k} - \sum_{i\in I^{\mathcal{L}}} c_{t,i,k}\mu_{t,i,k} - \sum_{i\in I^{\mathcal{C}}} C^0_{i,k}\mu_{t,i,k}\right] \tag{5.39}$$

$$\text{s.t.} \quad \lambda_{t,j,k} - \mu_{t,i,k} \leq \frac{1}{(1+R)^t}P_{t,i,j,k} \qquad \forall\, t\in T, i\in I, j\in J, k\in K \tag{5.40}$$

$$\mu_{t,i,k} \in \mathbb{R}^+ \qquad \forall\, t\in T, i\in I, k\in K \tag{5.41}$$

$$\lambda_{t,j,k} \in \mathbb{R}, \qquad \forall\, t\in T, j\in J, k\in K \tag{5.42}$$

The resulting duality-based reformulation is presented in Equations (5.43)-(5.55).

$$\begin{aligned}\max \quad & \sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k} \\ & -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\frac{1}{(1+R)^t}\left(A_{t,i}v_{t,i} + B_{t,i}w_{t,i}\right) \\ & -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{k\in K}\frac{1}{(1+R)^t}\left(E_{t,i,k}x_{t,i,k} + F_{t,i,k}\sum_{j\in J}y_{t,i,j,k}\right) \\ & -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}G_{t,i,j,k}y_{t,i,j,k}\end{aligned} \tag{5.43}$$

$$\text{s.t.} \quad w_{t,i} = V^0_i + \sum_{t'\in T'_t} v_{t',i} \qquad \forall\, t\in T, i\in I^{\mathcal{L}} \tag{5.44}$$

$$x_{t,i,k} \leq w_{t,i} \qquad \forall\, t \in T, i \in I^{\mathcal{L}}, k \in K \tag{5.45}$$

$$c_{t,i,k} = C^0_{i,k} + \sum_{t' \in T'_t} H_{i,k} x_{t',i,k} \qquad \forall\, t \in T, i \in I^{\mathcal{L}}, k \in K \tag{5.46}$$

$$\begin{aligned} &\sum_{t \in T} \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} \frac{1}{(1+R)^t} P_{t,i,j,k} y_{t,i,j,k} \\ &\quad = \sum_{t \in T} \sum_{k \in K} \left[\sum_{j \in J} D_{t,j,k} \lambda_{t,j,k} - \sum_{i \in I^{\mathcal{L}}} c_{t,i,k} \mu_{t,i,k} - \sum_{i \in I^{\mathcal{C}}} C^0_{i,k} \mu_{t,i,k} \right] \end{aligned} \tag{5.47}$$

$$\sum_{j \in J} y_{t,i,j,k} \leq c_{t,i,k} \qquad \forall\, t \in T, i \in I^{\mathcal{L}}, k \in K \tag{5.48}$$

$$\sum_{j \in J} y_{t,i,j,k} \leq C^0_{i,k} \qquad \forall\, t \in T, i \in I^{\mathcal{C}}, k \in K \tag{5.49}$$

$$\sum_{i \in I} y_{t,i,j,k} = D_{t,j,k} \qquad \forall\, t \in T, j \in J, k \in K \tag{5.50}$$

$$\lambda_{t,j,k} - \mu_{t,i,k} \leq \frac{1}{(1+R)^t} P_{t,i,j,k} \qquad \forall\, t \in T, i \in I, j \in J, k \in K \tag{5.51}$$

$$y_{t,i,j,k},\ \mu_{t,i,k} \in \mathbb{R}^+ \qquad \forall\, t \in T, i \in I, j \in J, k \in K \tag{5.52}$$

$$\lambda_{t,j,k} \in \mathbb{R}, \qquad \forall\, t \in T, j \in J, k \in K \tag{5.53}$$

$$c_{t,i,k} \in \mathbb{R}^+ \qquad \forall\, t \in T, i \in I^{\mathcal{L}}, k \in K \tag{5.54}$$

$$v_{t,i},\ w_{t,i},\ x_{t,i,k} \in \{0,1\} \qquad \forall\, t \in T, i \in I^{\mathcal{L}}, k \in K \tag{5.55}$$

The upper-level problem represented by Equations (5.43)-(5.46) remains unchanged in the duality-based reformulation. Strong duality is enforced by equating the primal and dual objective functions as presented by Equation (5.47). Lower-level primal constraints (5.48) and (5.50) are kept in the formulation to guarantee primal feasibility. Dual feasibility of the lower level is ensured with constraints (5.51).

It must be noted that this reformulation yields a Mixed-Integer Non-Linear Program (MINLP). The nonlinearity arises from the dual objective function in the right hand side of Equation (5.47), because of the product of upper-level variable $c_{t,i,k}$ and lower-level dual variable $\mu_{t,i,k}$. Fortunately, the problem can be posed as an MILP because the variable $c_{t,i,k}$ only takes values in discrete increments as indicated by Equation (5.46). The linearization procedure is based on eliminating variable $c_{t,i,k}$ from the formulation by replacing it according to Equation (5.46). The resulting bilinear terms are products of continuous variables ($\mu_{t,i,k}$) and binary variables ($x_{t',i,k}$). Therefore, they can be modeled with a set of mixed-integer constraints by including a continuous variable ($u_{t',t,i,k}$) for each bilinear term.

The resulting linearized MILP formulation is obtained after replacing Equation (5.47) with Equation (5.56),

$$\sum_{t\in T}\sum_{i\in I}\sum_{j\in J}\sum_{k\in K}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k} = \sum_{t\in T}\sum_{k\in K}\left[\sum_{j\in J}D_{t,j,k}\lambda_{t,j,k} - \sum_{i\in I}C^0_{i,k}\mu_{t,i,k} - \sum_{i\in I^{\mathcal{L}}}\sum_{t'\in T'_t}H_{i,k}u_{t,t',i,k}\right] \tag{5.56}$$

and introducing the mixed-integer constraints presented in Equations (5.57)-(5.58).

$$u_{t,t',i,k} \geq \mu_{t,i,k} - M\left(1 - x_{t',i,k}\right) \qquad t\in T, t'\in T'_t, i\in I^{\mathcal{L}}, k\in K \tag{5.57}$$

$$u_{t,t',i,k} \in \mathbb{R}^+ \qquad t\in T, t'\in T'_t, i\in I^{\mathcal{L}}, k\in K \tag{5.58}$$

It is important to note that only the two terms presented in Equations (5.57) and (5.58) are necessary to linearize the bilinear terms because they are sufficient to bound the values of $u_{t,t',i,k}$ in the improving direction of the objective function.

5.7 Illustrative example

Both MILP reformulations of the bilevel capacity planning problem are implemented to solve a small case study from the air separation industry. The illustrative example considers two existing plants of the leader, one candidate location for a new plant of the leader, and a single plant of the competition. Facilities controlled by the leader and the competitor must satisfy the demand of 8 markets for a single commodity. The problem has a time horizon of 3 years divided in 12 time periods (quarters of year). In this time horizon, the leader is allowed to execute investment decisions in time periods 1, 5, and 12. Capacity expansion is achieved by installing additional production lines with capacity of 9,000 ton/period. The complete dataset for this illustrative example is presented in Appendix C. The example uses a discount rate (R) of 3% per time period.

The computational statistics of the single-level capacity planning with captive markets and the reformulations of the capacity planning with rational markets are presented in Table 5.1. All MILP problems were implemented in GAMS 24.4.1 and were solved using GUROBI 6.0.0 on an Intel Core i7 CPU 2.93 Ghz processor with 4 GB of RAM.

Model statistic	Single-level with captive markets	KKT reformulation	Duality-base reformulation
Number of constraints:	225	1,473	682
Number of continuous variables:	420	996	636
Number of binary variables:	48	480	48
LP relaxation at rootnode:	110	110	101
Final incumbent value:	110	97	97
Final optimality gap:	0.00%	0.00%	0.00%
Number of B&B nodes:	1	262	1
Solution time:	0.01 s	0.63 s	0.19 s

Table 5.1: Model statistics of the illustrative example.

Table 5.1 shows the number of constraints and variables for the proposed formulations. It can be observed that the KKT reformulation is significantly larger than the duality-based reformulation; in particular, it requires 10 times more binary variables because of the complementarity constraints. The growth in the number of binary variables does not have much impact for the solution time of this small example, but it is likely to complicate the solution of larger instances.

The solutions obtained from the optimization problems establish the investment plans of the leader. The plan obtained from the formulation with captive markets does not expand any plants in the time horizon. The optimal investment plan obtained from the bilevel formulation (both reformulations) expands plant 1 in the first time period. The bilevel optimal demand assignments in the first time period of this illustrative example are presented in Figure 5.2; it can be observed that some markets have dual sourcing because of the capacity limitations of production plants. Table 5.2 compares the income, investment costs, and operating costs for the single-level and bilevel expansion plans. In order to quantify the potential regret of implementing an expansion plan that ignores the decision criterion of markets, the expansion plan obtained from the single-level formulation is also evaluated in an environment of rational markets.

Table 5.2 shows the benefits of the expansion plan obtained from the bilevel formulation when markets are considered rational. The single-level formulation with captive markets predicts a level of income that is not attainable with rational markets. The bilevel formulation offers the lowest

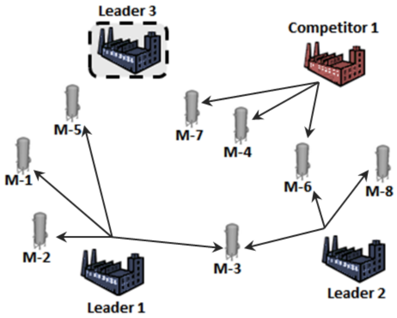


Figure 5.2: Optimal demand assignments obtained in the first time period of the illustrative example using the bilevel formulation.

Term in objective function	Single-level with captive markets	Single-level with rational markets	Bilevel with rational markets
Income from sales (M$):	354	345	398
Investment in new plants (M$):	0	0	0
Expansion cost (M$):	0	0	29
Maintenance cost (M$):	31	31	31
Production cost (M$):	139	139	162
Transportation cost (M$):	74	118	79
NPV (M$):	110	57	97
Market cost (M$):	523	510	508

Table 5.2: Results of the single-level and bilevel expansion plans in the illustrative example.

cost for the markets with a small deterioration of the leader's NPV in comparison to what could be obtained with captive markets. When markets are considered rational, the NPV obtained with the bilevel expansion plan is M$40 higher than the one obtained by the single-level expansion plan; this measure of regret accounts for 41% of the potential NPV.

5.8 Middle-size instances

From the illustrative example presented in Section 5.7, we observe that the KKT and the duality-based reformulations yield exactly the same results. Despite the difference in formulation sizes shown in Table 5.1, both reformulation solve the illustrative example in approximately the same time. In order to predict the performance of the reformulations on large-scale instances, we use a middle-size example of the capacity planning problem.

The example comprises the production and distribution of one product to 15 markets. Initially, the leader has three production plants with capacities equal to 27,000 ton/period, 13,500 ton/period, and 31,500 ton/period. The leader also considers the possibility of opening a new plant at a candidate location. We evaluate the investment decisions in a time horizon of 5 years divided in 20 time periods.

We analyze two instances that share the same data but allow different timing for the investment decisions. In the first instance (Middle-size 1), the leader is allowed to open the new plant and expand capacities in every fourth time period. In the second instance (Middle-size 2), the leader is allowed to execute the investments only every eight time periods. In both cases, capacity must be expanded in discrete increments of 9,000 ton/period. The investment costs associated with opening the new plant and expanding production capacity grow in time according to inflation; the maintenance cost of open plants also increase with time.

Market demands in each time period vary during the time horizon. Figure 5.3 shows the trajectory of the demands for the middle-size example. The selling prices offered by the leader to the markets are presented in Figure 5.4; each market is offered a different price based on their proximity to the production plants of the leader. Unit production costs at the plants controlled by the leader are presented in Figure 5.5; they show the characteristic seasonal variation caused by the electricity cost. Other cost coefficients of the example are available electronically as part of the Supplementary material; they maintain the same magnitudes presented in Appendix C.

The computational statistics for the two middle-size instances of the capacity planning problem with rational markets are presented in Table 5.3. The KKT and the duality-based reformulations were implemented in GAMS 24.4.1 and were solved using GUROBI 6.0.0.

Table 5.3 demonstrates the benefits of the duality-based reformulation in comparison to the KKT reformulation. The time required to solve both instances using the duality-based reformulation is less than 1 second, whereas the KKT reformulation requires a few minutes for each instance. Interestingly, the KKT reformulation takes longer to solve the second middle-size instance that has fewer investment options. The reason behind this counter-intuitive behavior is that the solver takes longer to find a feasible solution to the problem.

The significant difference in solution time for both reformulations is explained by the number of constraints and variables in the problem. The KKT reformulation requires in both instances 2,240

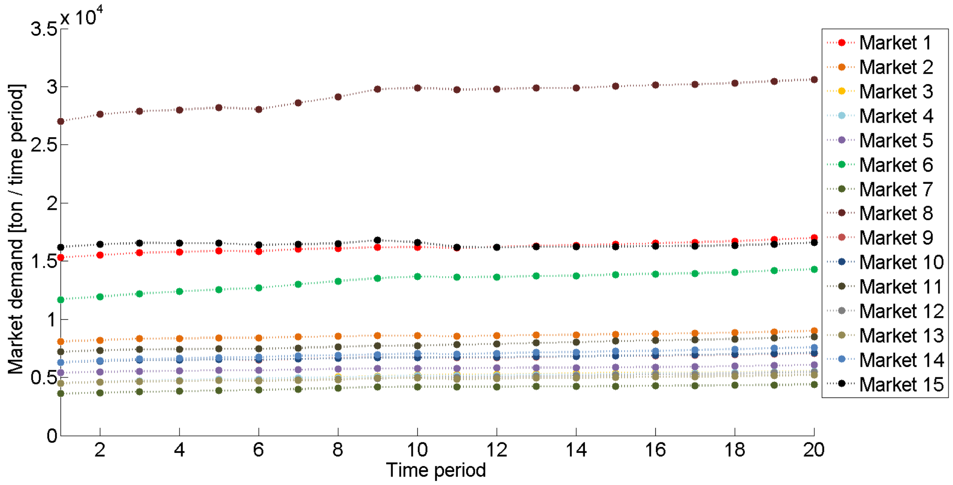


Figure 5.3: Evolution of market demands in the middle-size instances.

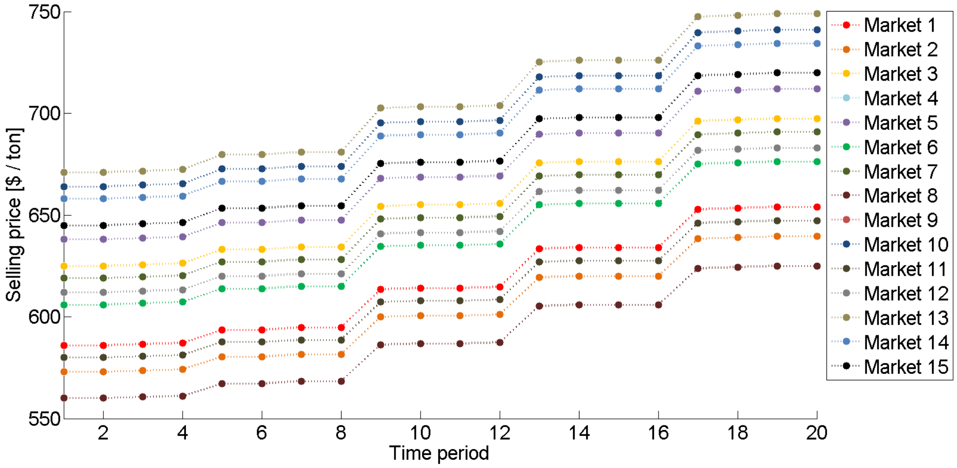


Figure 5.4: Evolution of selling prices in the middle-size instances.

additional binary variables to model complementarity conditions. The growth in the number of binary variables has a severe impact in the solution time of the problem.

Table 5.4 compares the income, investment costs, and operating costs of the proposed expansion plans. It shows that the expansion plan obtained for the first instance produces a slightly higher NPV when compared with the plan obtained for the second instance. This result can be anticipated because the feasible region of the first instance contains the feasible region of the second instance

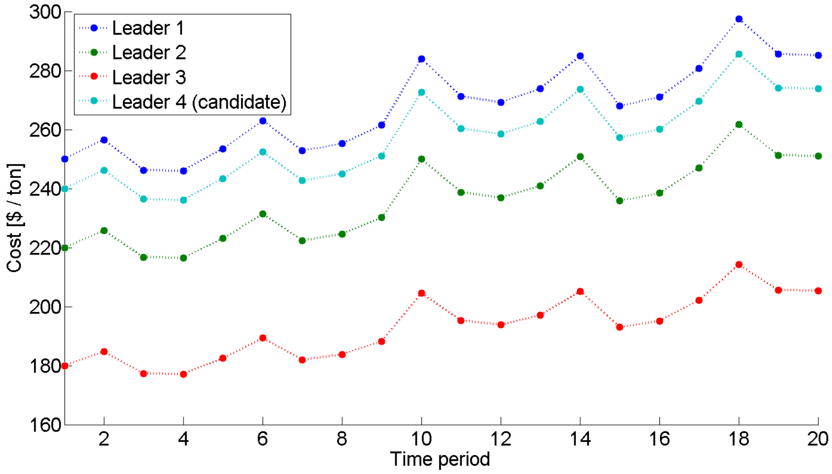


Figure 5.5: Evolution of production costs in the middle-size instances.

Model statistic	**Middle-size 1**		**Middle-size 2**	
	KKT reformulation	Duality-base reformulation	KKT reformulation	Duality-base reformulation
Number of constraints:	7,200	2,961	7,192	2,857
Number of continuous variables:	4,860	2,965	4,860	2,763
Number of binary variables:	2,345	105	2,335	95
LP relaxation at rootnode:	372	346	346	324
Final incumbent value:	316	316	308	308
Final optimality gap:	0.01%	0.00%	0.01%	0.00%
Number of B&B nodes:	11,367	1	16,786	1
Solution time:	157 s	0.83 s	282 s	0.73 s

Table 5.3: Model statistics of middle-size instances.

completely. However, the additional restrictions for the execution of investment decisions in the second instance only produces a decrease of 1.1% in its NPV.

The investment plans obtained from the bilevel formulation do not invest to open the new plant in any of the instances. In the first instance, the plan expands plants 2 and 3 in the first time period, and plant 3 in the fifth time period. The optimal capacities and production levels of the plants controlled by the leader in the first instance are presented in Figure 5.6; arrows indicate the time periods in which capacity is expanded. We can observe in Figure 5.6 that all production plants have high utilization. The expanded capacities in plants 2 and 3 are used as soon as they are available;

Term in objective function	**Middle-size 1**	**Middle-size 2**
Income from sales (M$):	895	885
Investment in new plants (M$):	0	0
Expansion cost (M$):	85	82
Maintenance cost (M$):	94	94
Production cost (M$):	315	313
Transportation cost (M$):	85	88
NPV (M$):	316	308
Market cost (M$):	1,319	1,319

Table 5.4: Results of the bilevel expansion plans for the middle-size instances.

plant 1 experiences a temporary decrease in its production because of the capacity increase at plant 3, but it returns to full utilization with demand growth. The bilevel expansion plan obtained for the second instance is very similar to the plan obtained for the first instance; it expands plants 2 and 3 in the first time period, and delays the second expansion of plant 3 until the ninth time period. In both instances, investment and maintenance cost are equal for all plants controlled by the leader; therefore, the expansion trends observed are good indicators of the competitiveness of plants with respect production and transportation cost.

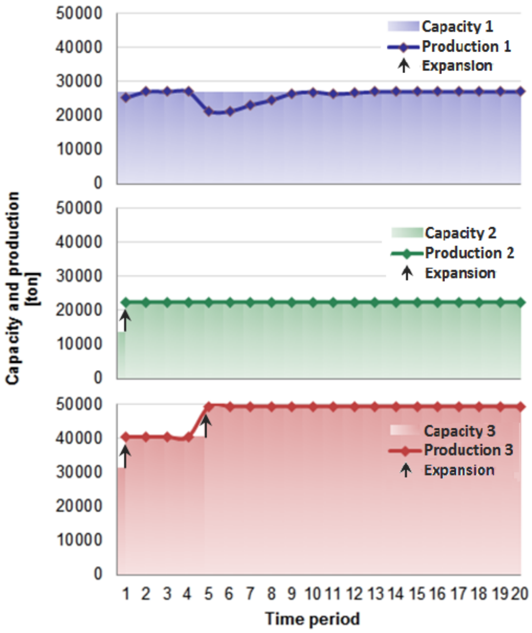


Figure 5.6: Capacity and production of the leader in the first instance of the middle-size example.

5.9 Solution strategies for large-scale problems

The implementation of the bilevel formulation for capacity planning problems in industrial instances requires developing a solution strategy for large-scale problems. The results obtained from the middle-size example suggest that the KKT reformulation is not appropriate to solve large instances. Additionally, we can expect the duality-based reformulation to struggle solving large-scale instances given the relative weakness of its LP relaxation. Therefore, we propose an improved duality-based reformulation and a domain reduction scheme; these solution strategies are evaluated in Section 5.10 with an industrial example.

5.9.1 Strengthened duality-based reformulation

The LP relaxation of the duality-based reformulation can be strengthened by enforcing strong duality independently for each commodity in every time period. The justification for this modification comes from the observation that once the leader has fixed its capacity, the optimization problem of the follower can be decomposed by time period and commodity. Consequently, we can replace Equation (5.56) by its disaggregated version presented in Equation (5.59).

$$\begin{aligned}&\sum_{i\in I}\sum_{j\in J}\frac{1}{(1+R)^t}P_{t,i,j,k}y_{t,i,j,k}\\&\qquad=\sum_{j\in J}D_{t,j,k}\lambda_{t,j,k}-\sum_{i\in I}C^0_{i,k}\mu_{t,i,k}-\sum_{i\in I^{\mathcal{L}}}\sum_{t'=1}^{t}H_{i,k}u_{t,t',i,k}\qquad\forall\ t\in T,k\in K\end{aligned}\tag{5.59}$$

Replacing Equation (5.56) by Equation (5.59) yields a modest improvement in the LP relaxation of the duality-based reformulation. In the first instance of the middle-size example presented in Section 5.8, the value of the LP relaxation is reduced from M$346 to M$343 (9.49% to 8.54% initial optimality gap).

5.9.2 Domain reduction for the duality-based reformulation

A clever strategy to reduce the size of the capacity planning problem with rational markets derives from the analysis of the feasible region of the bilevel optimization problem. In the bilevel optimization literature, the bilevel feasible region is called the inducible region [9]. In essence, the inducible region is the set of upper-level feasible solutions and their corresponding rational reactions in the lower-level problem. In order to describe the inducible region mathematically, we define the set of upper-level feasible solutions as the capacity expansion plans that satisfy upper-level constraints. This set of upper-level feasible solutions is represented in Equation (5.60),

$$(v, w, x, c) \in X \tag{5.60}$$

where X denotes the polyhedron described by upper-level constraints (5.10)-(5.12) and upper-level domains (5.18)-(5.19).

The rational reaction set for the follower is defined by expression (5.61) as a function of the upper-level variables,

$$\Psi(v, w, x, c) = \left\{ y : \underset{y \in Y}{\arg\min} \left[\sum_{t \in T} \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} \frac{1}{(1+R)^t} P_{t,i,j,k} y_{t,i,j,k} \right] \right\} \tag{5.61}$$

where Y denotes the polyhedron described by lower-level constraints (5.14)-(5.17).

According to expressions (5.60) and (5.61), the inducible region of the bilevel capacity expansion problem is defined by expression (5.62).

$$IR = \{(v, w, x, c, y) : \ (v, w, x, c) \in X, \ y \in \Psi(v, w, x, c)\} \tag{5.62}$$

We know from our original assumptions that any expansion plan satisfying Equation (5.60) has a nonempty rational reaction set ($\Psi(v, w, x, c)$). However, not all demand assignments satisfying the lower-level constraints ($y \in Y$) are bilevel feasible because some of them might be suboptimal for all expansion plans of the leader. Hence, it is possible to reduce the dimension of the bilevel formulation by excluding from its domain those demand assignments ($y \in Y$) that are never optimal in the lower level.

The first step to identify demand assignments that are bilevel infeasible is to solve the lower-level problem with the production capacities of the leader fixed to their upper bounds. Once we know the optimal demand assignments in the lower-level problem with maximum capacity, we can infer which demands are never assigned to the leader. The intuition for this inference is that only the demands ($D_{t,j,k}$) that are assigned to the leader when the capacity is at its upper bound, can be assigned to the leader when its capacity is more constrained.

The idea behind the domain reduction is that demand assignments that are nonbasic in the optimal solution of the LP with maximum capacity, must remain nonbasic when capacity is reduced. Proposition 5.1 formalizes this idea. Its proof can be found in Appendix D.

Proposition 5.1. *A demand assignment ($y_{t,i,j,k}$) with positive reduced cost in the optimal solution of the lower-level problem with maximum capacity also has a positive reduced cost when capacities are reduced.*

For the implementation of the domain reduction strategy, it is important to remember that nonbasic variables are associated with positive reduced costs in the minimization problem. In order to identify nonbasic variables, we denote by $\mu^U_{t,i,k}$ and $\lambda^U_{t,j,k}$ the optimal dual solution of the lower-level problem with capacities of the leader are at their upper bound ($C^U_{t,i,k} \ \ \forall\, i \in I^{\mathcal{L}}$). Then, according to Proposition 5.1, Equation (5.63) establishes valid upper bounds for the demand assignments in the bilevel capacity planning problem.

$$y_{t,i,j,k} \leq \begin{cases} 0 & \text{if } \frac{1}{(1+R)^t} P_{t,i,j,k} + \mu^U_{t,i,k} - \lambda^U_{t,j,k} > 0 \\ D_{t,j,k} & \text{otherwise} \end{cases} \qquad \forall\ t \in T, i \in I^{\mathcal{L}}, j \in J, k \in K \tag{5.63}$$

The range reduction proposed in expression (5.63) can have a significant impact in the size of the bilevel formulation because many assignment variables can be fixed if we determine that zero is their only bilevel feasible value. However, it is not the only advantage of the domain reduction strategy when we use the duality-based reformulation. If we analyze the lower-level LP in light of complementary slackness, we can conclude that expression (5.63) also implies that some dual constraints (5.51) are never active. In particular, those dual constraints (5.51) corresponding to the variables $y_{t,i,j,k}$ that can be fixed to zero are irrelevant in the duality-based formulation. Therefore, the domain reduction strategy proposed for the bilevel capacity expansion planning offers the double benefit of reducing the number of continuous variables and the number of constraints in the duality-based reformulation.

5.10 Industrial example

The solution strategies proposed for large-scale instances are tested with a capacity planning problem for an air separation company. This large-scale example includes 3 existing plants of the leader, 2 candidate plants of the leader, and 5 plants of competitors. Demands of 20 markets for 2 different commodities are considered in a time horizon of 20 years divided in 80 time periods. Two instances allowing different timing for the investment decisions are analyzed: the first instance allows investments every fourth time period and the second instance allows investments every eighth time period.

According to the formulation, the leader maximizes the NPV obtained during the 20-year time horizon. Markets select their providers by controlling the demand assignments with the objective of minimizing the discounted cost they pay. A discount rate (R) of 3% per time period is used in both objective functions. Cost coefficients and all other parameters are omitted because of confidentiality reasons.

The computational statistics for the original duality-based reformulation and the large-scale duality-based reformulation are presented in Table 5.5; the large-scale reformulation enforces strong duality for each commodity in every time period and implements the domain reduction strategy to fix variables and eliminate constraints. Table 5.5 shows that both instances of the industrial example have a significant number of constraints, continuous and discrete variables. However, if we compare the original and the large-scale duality-based reformulations, we observe a reduction between 13% and 17% in the number of continuous variables and constraints.

	Industrial 1		**Industrial 2**	
Model statistic	Original duality-based	Large-scale duality-based	Original duality-based	Large-scale duality-based
Number of constraints:	46,601	40,025	42,501	35,925
Number of continuous variables:	46,000	39,905	42,520	35,265
Number of binary variables:	640	640	520	520
LP relaxation at rootnode:	4,289	2,906	4,002	2,851
Final incumbent value:	2,662	2,791	2,689	2,746
Final optimality gap:	33.2%	1.27%	26.4%	0.98%
Solution time:	60 min*	60 min*	60 min*	5 min

* Time limit reached

Table 5.5: Model statistics of industrial instances.

Term in objective function	Industrial 1	Industrial 2
Income from sales (M$):	5,984	5,888
Investment in new plants (M$):	0	0
Expansion cost (M$):	439	411
Maintenance cost (M$):	215	215
Production cost (M$):	2,100	2,072
Transportation cost (M$):	439	444
NPV (M$):	2,791	2,746
Market cost (M$):	10,545	10,546

Table 5.6: Results of the bilevel expansion plans for the industrial instances.

The performance of both reformulations is also presented in Table 5.5; we observe a significant difference in the performance of the original and the large-scale duality-based reformulations. A major advantage of the large-scale reformulation is related to its LP relaxation at the rootnode. This improvement derives partially from disaggregating strong duality, and more importantly from excluding demand assignments that are bilevel infeasible. In the first industrial instance the LP relaxation gap is reduced from 34.9% to 3.9%, whereas in the second industrial instance the reduction is from 34.4% to 3.7%.

Even after implementing the proposed strategies for large-scale problems, the industrial instances are still difficult to solve using GUROBI 6.0.0. For our industrial example, only the second instance was solved to the desired optimality gap of 1% with the large-scale duality-based reformulation. However, if we compare the best solutions obtained for both industrial instances, we observe that allowing more frequent expansions in the first instance produces a NPV that is M$45 higher, which accounts for 1.6% of the potential profit. Table 5.6 presents in detail the terms in the objective function for the best solutions obtained; the table shows that allowing more frequent expansions in the first instance generates a more dynamic expansion plan that can capture a higher market share. However, the optimal number of expansions is the same for both instances and none of them includes investments in new plants.

The optimal capacity and production levels of plants controlled by the leader in the first industrial instance are presented in Figures 5.7 and 5.8 for commodities 1 and 2, respectively; the figures show that utilization of the production capacities is high for all the plants being expanded. The only capacity that is not expanded in the entire time horizon is the production capacity of commodity 1 at plant 1; the utilization of this production capacity fluctuates according to the available capacity at plants 2 and 3. The expansion trends observed preserve a close relation with the competitiveness of plants that is mainly determined by their production and distribution costs.

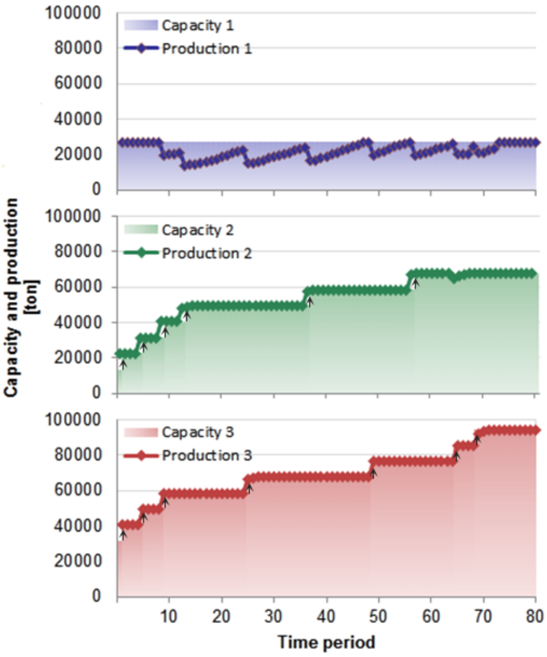


Figure 5.7: Capacity and production of commodity 1 at the plants controlled by the leader in the first instance of the industrial example.

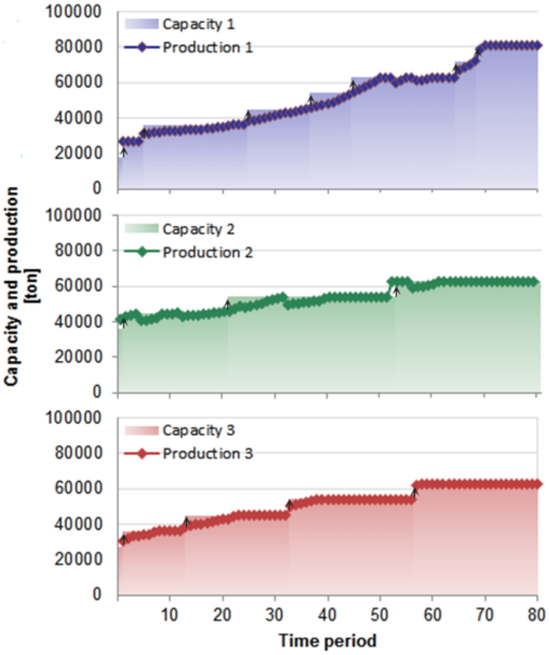


Figure 5.8: Capacity and production of commodity 2 at the plants controlled by the leader in the first instance of the industrial example.

5.11 Summary

We have developed a novel formulation for capacity planning problems that considers markets as rational decision makers. The formulation is based on bilevel optimization, a framework that allows modeling the conflicting interests of producers and consumers. The expansion plans obtained from the bilevel formulation produce greater economic benefits when the producers operate in a competitive environment. In particular, the single-level formulation tends to overestimate the market share that can be obtained and might generate expensive investment plans that are less profitable.

The bilevel formulation for capacity planning is a challenging optimization problem. We have proposed two different approaches to reformulate it as a single-level MILP. The first approach ensures optimality of the lower-level problem through its KKT conditions. The second approach uses strong duality of LPs for the reformulation. In the middle-size instances, we have shown that the duality-based reformulation offers superior performance compared to the KKT reformulation; this result is explained by the large number of binary variables required in the KKT approach to linearize the complementarity constraints. The duality-based reformulation does not require the addition of binary variables, but the strong duality condition gives rise to nonlinearities; for the case in which all upper-level variables are discrete, the nonlinearities can be avoided with the introduction of continuous variables and linear constraints.

Despite the relative advantage of the duality-based reformulation, the solution of large-scale instances of the bilevel capacity planning problem is still computationally demanding. We proposed two strategies to improve the duality-based reformulation. The first strategy leverages separability of the lower-level problem by disaggregating the strong duality constraint. The second strategy uses the topology of the bilevel feasible region to reduce the number of variables and constraints in the duality-based reformulation. The implementation of these strategies yields a significant reduction in the solution time of large-scale problems.

The bilevel formulation for capacity planning has shown to be useful for developing capacity expansion plans that considers markets as rational decision makers. This novel approach is more realistic than the traditional formulation because it models the dynamic nature of industrial markets. Furthermore, we have proposed an effective strategy to solve large-scale instances that allows using the bilevel capacity planning formulation in industrial applications.

Chapter 6

Capacity Planning with Competitive Decision-makers

6.1 Motivation

Industrial gas companies rely on capacity expansion models to plan the investments that allow them to satisfy future demands. In this industry, the proximity of producers to customers is a key competitive advantage that increases supply reliability and reduces transportation costs. This feature makes capacity planning a major strategic decision that impacts the market share that can be obtained in an environment of rational customers. The conflicting interests of several producers and consumers can be modeled using multilevel optimization to capture the rational behavior of all the players involved; such a model allows developing a robust expansion plan that anticipates the decisions of competitors and potential costumers.

Capacity planning has been widely studied in areas requiring the development of long-term infrastructure. In the process systems engineering community, the capacity planning problem has been extended to consider aspects of process design [233] and product development [154]. The model can be applied to problems requiring large capital investments whose feasibility, effectiveness, and profitability can only be assessed in a long time horizon. Therefore, capacity planning is critical for many industries [151] and it is recognized as a key topic in Enterprise-Wide Optimization [93].

Nevertheless, the capacity planning problem in a fully competitive environment has not been formulated before in a mathematical programming framework. The model presented in Chapter 5 addresses the capacity planning problem with rational markets in a bilevel formulation, but ignores potential expansion strategies of the competitors. The proposed bilevel formulation models the Stackelberg competition [239] between a company planning its investments and markets minimizing the cost of satisfying their demands. The addition of rational competitors that are allowed to expand implies a different dynamic for the capacity planning model.

A major extension of the bilevel formulations is needed to model the rational behavior of all players involved in the capacity planning problem with competitive decision-makers. Three hierarchical optimization problems are required because decisions are made sequentially by three players with conflicting interests. Similarly to the model presented in Chapter 5, the company planning its capacity is the first to decide its expansion strategy. Then, the competition optimizes its own capacity expansion plan. After all expansions are fixed, the markets select providers to minimize the cost of satisfying their demands. The investment decisions of the first two levels require discrete variables, whereas the third level can be formulated as a Linear Program (LP).

The growth in the area combining mathematical programming and game theory has been tightly linked to its fruitful range of applications. The potential of optimization models to coordinate decision-making in decentralized systems has been recognized by Anandalingam & Apprey [3]. Interesting multilevel programming models have been developed for traffic planning [157], optimal taxation of biofuels [10], parameter estimation [162], and product introduction [211].

However, there has been little work on multilevel optimization involving more that two players with discrete variables, as is required to model a fully competitive environment. The electrical network defense is the only problem for which a trilevel Mixed-Integer Linear Programming (MILP) model has been proposed [248]; the solution procedures for this formulation are problem specific and there is scarce theoretical study of the general properties of trilevel optimization problems. Our research presents a novel framework for capacity planning, introduces new concepts of degeneracy for multilevel problems, and proposes two solution algorithms for the models. The first step in both algorithms is to reformulate the trilevel problem as a bilevel problem, replacing the third-level by its optimality conditions. The reformulation is based on strong duality of the lower-level LP. The duality-based reformulation offers documented advantages over the standard KKT reformulation because it avoids the addition of discrete variables [80]. In the bilevel reformulation, the second level models the capacity expansion of the competitors and enforces optimality of the third-level problem. The resulting formulation is a Bilevel Mixed-Integer Linear Program (BMILP) with discrete variables in both levels; efficient solution methods for these type of problems is still considered an open question in Operations Research [51].

The numerical solution of BMILPs has been receiving increasing attention during the past years, but the existing literature has only considered academic examples with few discrete variables. The first Branch-&-Bound algorithm was developed by Moore & Bard [163]; it was based exclusively on the solution of LPs. Later, the same authors proposed a binary search tree algorithm that obtains the rational reaction of the lower level by solving an MILP after fixing the decision of the leader [9]; in the worst case, both algorithms conduct an exhaustive exploration of the leader's decision space. DeNegre & Ralphs [55] derived a locally valid cut that can be added to the Branch-&-Bound procedure proposed by Bard & Moore [9]; however, these cuts tend to be weak in problems with parameters of different magnitudes or with non-integer coefficients.

The framework proposed by Gümüş & Floudas [98] is based on convexification of the lower-level MILP to guarantee that its optimal solution lays in a vertex of the lower-level constraint region. This strategy allows using the reformulation techniques developed for lower-level LPs, but it comes at the expense of introducing an exponential number of variables and constraints. Faisca et al. [62] have used multi-parametric programming to characterize the optimal lower-level response for any potential first-level decision. This procedure can be extremely involved, but is interesting from a theoretical point of view because the multi-parametric solution completely describes the inducible region of the bilevel problem.

Recently, there have been two relevant contributions for our research. Xu & Wang [247] proposed a general spatial Branch-&-Bound search that splits the inducible region in polyhedral sets called stability regions; stability regions characterize the decisions of the leader that share the same optimal reaction of the follower. Zeng & An [255] developed a reformulation-decomposition approach that iteratively approximates the rational reaction of the follower based on linear inequalities in the space of the leader decision variables. Both contributions have been important for the development of our algorithms.

We present two algorithms that exploit different properties of the trilevel capacity expansion problem. Both algorithms leverage the relaxation obtained by eliminating the objective function of the lower level, known as the *high-point* problem. The first algorithm is a constraint-directed exploration; it eliminates decisions of the leader that have been explored, as well as all other decisions that induce the same reaction of the other players. The second algorithm is a decomposition method involving a master problem and a subproblem. The main idea is to incorporate in the master problem the reactions of the competition that are iteratively observed.

The rest of this chapter is structured as follows. In Section 6.2, we describe the capacity planning problem in a competitive environment. In Section 6.3, we propose the trilevel optimization formulation. Section 6.5 explores the implications of degeneracy in trilevel optimization problems. In Section 6.6, we elaborate on the properties of the capacity planning model that are useful for the development of our algorithms. The algorithms are described in Sections 6.7 and 6.8. In Section 6.9, we illustrate the implementation of the algorithms on two instances of the capacity planning problem. Finally, Section 6.10 reviews the novelty of this work and the results obtained.

6.2 Problem statement

The capacity planning problem in a competitive environment considers three players with independent decision criteria: the first-level industrial producer (leader) planning its expansion strategy, competitors that are allowed to expand, and costumers that select their providers. The objective of the capacity planning problem is to establish the expansion plan that maximizes the Net Present Value (NPV) obtained by the leader during a finite time horizon. The optimal expansion plan must balance investment and operational costs with the income obtained from sales. Sales are bounded by deterministic demands during the entire time horizon and depend on the actions taken by the other decision-makers. Initially, the leader is given a set of plants with finite production capacity and a set of candidate locations where new plants can be built. Capacity of the plants can be expanded in discrete increments by paying a fixed cost; only discrete capacity expansions are considered to model the installation of new production lines. The attractiveness of expanding plants depends on the possibility of obtaining a higher market share or reducing operational costs.

The competition represents other industrial producers that observe the decisions of the leader and develop expansion plans with the objective of maximizing their own NPV. All production plants that are not controlled by the leader are assumed to behave as a rational decision-maker with a centralized planner. The competition controls a set of open plants with given initial capacity; they are also allowed to open new plants in a set of candidate locations. Expansions at open plants are modeled with discrete capacity increments and fixed costs. We assume that the total initial capacity in the plants controlled by the leader and the competition is enough to satisfy all demands throughout the horizon.

Demand assignments in each time period are decided by customers after observing the available capacities in the plants controlled by the leader and by the competition. The market behaves as a centralized decision-maker that minimizes the total cost paid by all customers; demand assignments are based exclusively on availability and price of the products. The industrial producers can only influence market decisions by changing their production capacity since prices are fixed parameters. The leader offers a single selling price to each potential customer in a given time period, regardless of the plant that is used to satisfy the demand. Under this assumption, the customers are only concerned with selecting their providers, and the leader can choose the plants that satisfy the demands that are assigned to it. We consider two approaches to construct the prices in this industrial environment.

- *The leader offers homogeneous prices:* the price $P_{t,i,j}$ offered to a certain customer (j) is the same regardless of the plant $i \in I^{\mathcal{L}}$.

$$P_{t,i,j} = P_{t,j} \qquad \forall\, t \in T,\ i \in I^{\mathcal{L}},\ j \in J \tag{6.1}$$

- *Competitors offer site-dependent prices:* the price $P_{t,i,j}$ offered to customers depend on a raw price $(P^{raw}_{t,i})$ and the transportation cost $(G_{t,i,j})$ from that plant to the customers (j) .

$$P_{t,i,j} = P^{raw}_{t,i} + G_{t,i,j} \qquad \forall\, t \in T,\ i \in I^{\mathcal{C}},\ j \in J \tag{6.2}$$

The decision process takes place sequentially and perfect information is assumed for all players. The perfect information assumption implies that higher level decision-makers are aware of the decision criteria of the lower levels, and lower-level decision-makers observe the actions of the higher levels before selecting their response. The optimal solution of the problem characterizes the expansion plan that optimizes the objective function of the leader, considering a rational reaction of the competitors and the market. It implies that the leader cannot improve its objective unilaterally, the competitors select their optimal expansion plan given the decisions of the leader, and the market minimizes its total cost according to the available capacity.

6.3 Capacity planning with competitive decision-makers: trilevel formulation

According to the problem statement, we define the objective function of the industrial producers as the maximization of their NPV over a finite time horizon. The objective function presented in Equation (6.3) is the decision criterion of the leader,

$$\begin{aligned} NPV^{\mathcal{L}} = & \sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J} P_{t,i,j}y_{t,i,j} \\ & -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}} \left(A_{t,i}v_{t,i} + B_{t,i}w_{t,i} + E_{t,i}x_{t,i}\right) \\ & -\sum_{t\in T}\sum_{i\in I^{\mathcal{L}}}\sum_{j\in J} \left(F_{t,i} + G_{t,i,j}\right) y_{t,i,j} \end{aligned} \tag{6.3}$$

where T, $I^{\mathcal{L}}$, and J are respectively the index sets for time periods (t), plants controlled by the leader ($i \in I^{\mathcal{L}}$), and customers (j). The first term represents the income obtained from sales. Variables $y_{t,i,j}$ indicate the quantities sold from plant i to customer j at time t; coefficients $P_{t,i,j}$ are the selling prices. In the second term, $v_{t,i}$ is a binary variable indicating if a new plant is built at location i at time period t; the fixed cost to open a new plant is given by coefficients $A_{t,i}$. Maintenance costs are modeled with binary variables $w_{t,i}$ that indicate which plants are open; the maintenance cost of per time period is given by $B_{t,i}$. Capacity expansion decisions are modeled with binary variables $x_{t,i}$; the fixed cost of the expansions is given by $E_{t,i}$. The third term models the operational costs by associating demand assignments ($y_{t,i,j}$) with the unit costs of production ($F_{t,i}$) and transportation ($G_{t,i,j}$).

It is worth noticing that the objective presented in Equation (6.3) is not only a function of the variables that model planning decisions; it also depends on demand assignment variables ($y_{t,i,j}$) that are controlled by the customers. The competing industrial producers are assumed to maximize their own NPV with the same income and cost structure of the leader. In this framework, the trilevel formulation can be modeled with Equations (6.4)-(6.16),

$$\max_{v^{\mathcal{L}},w^{\mathcal{L}},x^{\mathcal{L}},c^{\mathcal{L}}} NPV^{\mathcal{L}}\left(v^{\mathcal{L}},w^{\mathcal{L}},x^{\mathcal{L}},c^{\mathcal{L}},y\right) \tag{6.4}$$

$$\text{s.t.} \quad w_{t,i} = V_{0,i} + \sum_{t' \in T_t^-} v_{t',i} \qquad \forall\ t \in T, i \in I^{\mathcal{L}} \tag{6.5}$$

$$x_{t,i} \leq w_{t,i} \qquad \forall\ t \in T, i \in I^{\mathcal{L}} \tag{6.6}$$

$$c_{t,i} = C_{0,i} + \sum_{t' \in T_t^-} H_i x_{t',i} \qquad \forall\ t \in T, i \in I^{\mathcal{L}} \tag{6.7}$$

$$\max_{v^{\mathcal{C}},w^{\mathcal{C}},x^{\mathcal{C}},c^{\mathcal{C}}} \quad NPV^{\mathcal{C}}\left(v^{\mathcal{C}},w^{\mathcal{C}},x^{\mathcal{C}},c^{\mathcal{C}},y\right) \tag{6.8}$$

$$\text{s.t.} \quad w_{t,i} = V_{0,i} + \sum_{t' \in T_t^-} v_{t',i} \qquad \forall\ t \in T, i \in I^{\mathcal{C}} \tag{6.9}$$

$$x_{t,i} \leq w_{t,i} \qquad \forall\ t \in T, i \in I^{\mathcal{C}} \tag{6.10}$$

$$c_{t,i} = C_{0,i} + \sum_{t' \in T_t^-} H_i x_{t',i} \qquad \forall\ t \in T, i \in I^{\mathcal{C}} \tag{6.11}$$

$$y = \operatorname*{arg\,min}_{y \in Y(c^{\mathcal{L}},c^{\mathcal{C}})} \left\{ \sum_{t \in T} \sum_{i \in I} \sum_{j \in J} P_{t,i,j} y_{t,i,j} \right\} \tag{6.12}$$

$$c_{t,i} \in \mathbb{R}^+ \qquad \forall\ t \in T, i \in I^{\mathcal{C}} \tag{6.13}$$

$$v_{t,i},\ w_{t,i},\ x_{t,i} \in \{0,1\} \qquad \forall\ t \in T, i \in I^{\mathcal{C}} \tag{6.14}$$

$$c_{t,i} \in \mathbb{R}^+ \qquad \forall\ t \in T, i \in I^{\mathcal{L}} \tag{6.15}$$

$$v_{t,i},\ w_{t,i},\ x_{t,i} \in \{0,1\} \qquad \forall\ t \in T, i \in I^{\mathcal{L}} \tag{6.16}$$

where the set of production plants I is divided in two subsets denoting the plants controlled by the leader ($I^{\mathcal{L}}$) and the plants controlled by the competitors ($I^{\mathcal{C}}$). The superscript $\mathcal{L}$ identifies the variables controlled by the leader and the superscript $\mathcal{C}$ the plants controlled by the competitors. The constraints modeling the feasible investment strategies for the leader are presented in Equations (6.5)-(6.7). We define T_t^- as the subset of time periods before time t; formally, $T_t^- = \{t' : t' \in T, t' \leq t\}$. Equation (6.5) enforces maintenance for open plants; the parameter $V_{0,i}$ indicates if plant i is initially open. Equation (6.6) restricts expansions to the open plants; only one expansion per time period is allowed in each plant. Equation (6.7) models capacity ($c_{t,i}$)

of plants according to their initial capacity ($C_{0,i}$) and discrete expansions ($x_{t,i}$) of size H_i. The corresponding feasible expansion plans for the competitors are presented in Equations (6.9)-(6.11). Domains for the decisions of the competitors and the leader are expressed by Equations (6.13)-(6.14) and Equations (6.15)-(6.16), respectively.

The rational response of the customers is modeled with Equation (6.12). The market minimizes its total discounted cost by controlling the demand assignment variables $y_{t,i,j}$ on the polyhedral set $Y(c^{\mathcal{L}}, c^{\mathcal{C}})$; the decision space of the assignments depends on the capacity planning strategies chosen by the leader and the competitors. The complete third-level optimization problem that decides demand assignments is presented in Equations (6.17)-(6.20),

$$\min_{y} \quad \sum_{t \in T} \sum_{i \in I} \sum_{j \in J} P_{t,i,j} y_{t,i,j} \tag{6.17}$$

$$\text{s.t.} \quad \sum_{j \in J} y_{t,i,j} \leq c_{t,i} \qquad \forall\, t \in T, i \in I \tag{6.18}$$

$$\sum_{i \in I} y_{t,i,j} = D_{t,j} \qquad \forall\, t \in T, j \in J \tag{6.19}$$

$$y_{t,i,j} \geq 0 \qquad \forall\, t \in T, i \in I, j \in J \tag{6.20}$$

where $D_{t,j}$ is the demand of customer j in time period t.

6.4 Capacity planning with competitive decision-makers: bilevel reformulation

The trilevel formulation for the capacity planning in a competitive environment is a difficult mathematical problem. There are no standard techniques to solve this kind of problems and most of the available literature in multilevel programming focuses on bilevel problems. Therefore, the first step to address this challenge is to reformulate the two lower levels as a single-level optimization problem. Equations (6.8)-(6.14) is a bilevel formulation modeling the problem of the competitors and the market; the upper level only has discrete variables and the lower level is an LP. Hence, the problem can be transformed into a single-level formulation by replacing the lower level with its optimality conditions.

The most common approach to reformulate a bilevel optimization leverages convexity of the lower level to characterize the set of optimal lower-level solutions using the Karush-Kuhn-Tucker (KKT) optimality conditions. However, in lower-level problems with inequality constraints the KKT approach might be ineffective because it requires the addition of many complementarity constraints. The duality-based approach described in Chapter 5 is better suited to reformulate the two lower-level problems because it does not require the addition of binary variables. We replace the lower-level LP by constraints guaranteeing primal feasibility, dual feasibility, and strong duality. The constraints presented in Equations (6.21)-(6.25) characterize the set of optimal solutions for the cost minimization problem controlling demand assignments in the model with rational markets [80],

$$\sum_{t\in T}\sum_{i\in I}\sum_{j\in J} P_{t,i,j} y_{t,i,j} = \sum_{t\in T}\left[\sum_{j\in J} D_{t,j}\lambda_{t,j} - \sum_{i\in I} c_{t,i}\mu_{t,i}\right] \tag{6.21}$$

$$\sum_{j\in J} y_{t,i,j} \leq c_{t,i} \qquad \forall\, t\in T, i\in I \tag{6.22}$$

$$\sum_{i\in I} y_{t,i,j} = D_{t,j} \qquad \forall\, t\in T, j\in J \tag{6.23}$$

$$\lambda_{t,j} - \mu_{t,i} \leq P_{t,i,j} \qquad \forall\, t\in T, i\in I, j\in J \tag{6.24}$$

$$y_{t,i,j};\quad \mu_{t,i}\in\mathbb{R}^+;\quad \lambda_{t,j}\in\mathbb{R} \qquad \forall\, t\in T, i\in I, j\in J \tag{6.25}$$

where Equation (6.21) enforces strong duality and Equation (6.24) are the dual constraints corresponding to primal variables $y_{t,i,j}$. Dual variables associated to Equation (6.18) are denoted by $\mu_{t,i}$ and dual variables associated to Equation (6.19) are denoted by $\lambda_{t,j,k}$.

It is important to note that Equation (6.21) contains bilinear terms in the product of upper-level variables $c_{t,i}$ and dual variables $\mu_{t,i}$. Bilinear terms are nonconvex; however, in this case we can apply an exact linearization procedure [87] because variables $c_{t,i}$ only take discrete values. The non-linearity is avoided by describing capacities ($c_{t,i}$) in terms of the expansion decisions, according to Equation (6.7) and Equation (6.11). Additionally, new variables ($u_{t,t',i}$) defined for the product of dual variables $\mu_{t,i}$ and expansion variables $x_{t',i}$ are introduced in the formulation. Then, Equation (6.21) can be replaced by the Equations (6.26)-(6.28).

$$\sum_{t\in T}\sum_{i\in I}\sum_{j\in J} P_{t,i,j}y_{t,i,j} = \sum_{t\in T}\left(\sum_{j\in J} D_{t,j}\lambda_{t,j} - \sum_{i\in I} C_{0,i}\mu_{t,i} - \sum_{i\in I}\sum_{t'\in T_t^-} H_i u_{t,t',i}\right) \tag{6.26}$$

$$u_{t,t',i} \geq \mu_{t,i} - M(1 - x_{t',i}) \qquad \forall\, t \in T, t' \in T_t^-, i \in I \tag{6.27}$$

$$u_{t,t',i} \in \mathbb{R}^+ \qquad \forall\, t \in T, t' \in T_t^-, i \in I \tag{6.28}$$

We achieve the exact linearization of the bilinear terms in Equation (6.21) with only two linear inequalities, Equations (6.27) and (6.28), because they are sufficient to bound variables $u_{t,t',i}$ in the improving direction of the objective function.

The bilevel reformulation of the capacity planning problem in a competitive environment is obtained by replacing Equation (6.12) in the trilevel formulation with the constraints modeling the rational behavior of the market. The optimal response of the market is characterized by the primal feasibility constraints presented in Equations (6.22)-(6.23), the dual feasiblity constraints presented in Equation (6.24), the linearized version of the strong duality constraint presented in Equations (6.26)-(6.28), and the domains presented in Equation (6.25).

6.5 Multilevel programming and degeneracy

The optimal solution of a multilevel program might not be strictly defined if a lower-level problem has several optimal responses to the decisions of the higher levels. Degeneracy gives rise to ambiguity in the lower-level decision criterion because the same optimal objective values can be obtained from a set of responses producing different effects in the higher levels. The interpretations of degeneracy have been studied for bilevel programs [52], but it has not been addressed before in multilevel programming. We first offer some background on degeneracy in bilevel optimization in order to present the definitions needed for our algorithms.

6.5.1 Degeneracy in bilevel programming

One complication of bilevel optimization problems is the characterization of optimal solutions when the lower-level problem has multiple optima. Definition 1 describes the most common modeling approach for degenerate bilevel problems.

Definition 1. The solution of a bilevel program is considered *optimistic* if any degeneracy in the lower level is resolved in favor of the leader. The rational reaction of the lower-level problem in the *optimistic approach* is formally defined by Equation (6.29).

$$\Psi^{\mathcal{U}}(x) = \arg\max_{y \in \Psi(x)} \{F(x, y)\} \tag{6.29}$$

where x is the set of variables controlled by the upper level, y is the set of variables controlled by the lower level, $F(x, y)$ is the objective function being maximized in the upper level, and $\Psi(x)$ is the set of lower-level optimal reactions described in Equation (1.32).

The *optimistic approach* is a common assumption to resolve degeneracy in bilevel programming, mainly because *optimistic solutions* are easy to find using reformulation techniques. However, there is an increasing interest on extending the treatment of degeneracies to study alternative resolution models. The *pessimistic approach* can be defined as the model in which the lower level selects the response that is most detrimental to the leader in case of degeneracy [53]. These alternative models are considered harder to solve.

6.5.2 Degeneracy in trilevel programming

Hierarchical optimization problem with three levels might exhibit new types of solutions. In order to comply with the perfect information assumption, the decision criteria must be completely specified in the case of degeneracy, such that decision-makers that are hierarchically higher can calculate the response of the lower levels. In the following, we propose definitions to clear out ambiguity in our trilevel formulation when the second and third levels have multiple optima.

Definitions of the *Constraint Region* (Ω) presented in Equation (1.29) and the *High-Point* (HP) problem presented in Equations (1.34)-(1.37) can be extended directly to trilevel optimization problems. Similarly, the *Inducible Region* (IR) follows the same intuition presented for bilevel models in Equation (1.33), but its interpretation depends on a new definition of the *Rational Reaction sets*. For notational convenience, we denote by $x^{\mathcal{L}}$ and $x^{\mathcal{C}}$ the first- and second-level decisions, respectively; all other first- and second-level variables can be easily related to them in the capacity planning problem.

Definition 2. The following *Rational Reaction* sets can be identified in a trilevel program.

- The rational reaction set of the third level:

$$\Psi_y(x^{\mathcal{L}}, x^{\mathcal{C}}) = \underset{y \in Y(x^{\mathcal{L}}, x^{\mathcal{C}})}{\arg\min} \left\{ \sum_{t \in T} \sum_{i \in I} \sum_{j \in J} P_{t,i,j} y_{t,i,j} \right\} \tag{6.30}$$

- The first-level optimistic reaction set of the third level:

$$\Psi_y^{\mathcal{L}}(x^{\mathcal{L}}, x^{\mathcal{C}}) = \underset{y \in \Psi_y(x^{\mathcal{L}}, x^{\mathcal{C}})}{\arg\max} \left\{ NPV^{\mathcal{L}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) \right\} \tag{6.31}$$

- The second-level optimistic reaction set of the third level:

$$\Psi_y^{\mathcal{C}}(x^{\mathcal{L}}, x^{\mathcal{C}}) = \underset{y \in \Psi_y(x^{\mathcal{L}}, x^{\mathcal{C}})}{\arg\max} \left\{ NPV^{\mathcal{C}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) \right\} \tag{6.32}$$

- The rational reaction set of the second level:

$$\Psi_{x^{\mathcal{C}}}(x^{\mathcal{L}}) = \underset{x^{\mathcal{C}} \in X^{\mathcal{C}}(x^{\mathcal{L}})}{\arg\max} \left\{ NPV^{\mathcal{C}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) : y \in \Psi_y(x^{\mathcal{L}}, x^{\mathcal{C}}) \right\} \tag{6.33}$$

- The first-level optimistic reaction set of the second level:

$$\Psi_{x^{\mathcal{C}}}^{\mathcal{L}}(x^{\mathcal{L}}) = \underset{x^{\mathcal{C}} \in \Psi_{x^{\mathcal{C}}}(x^{\mathcal{L}})}{\arg\max} \left\{ NPV^{\mathcal{L}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) : y \in \Psi_y(x^{\mathcal{L}}, x^{\mathcal{C}}) \right\} \tag{6.34}$$

The *Rational Reaction* sets presented in Definition 2 suggest a variety of interpretations for degenerate solutions in trilevel programs. We classify the approaches to resolve degeneracy in trilevel programming according to the order in which the upper-level objective functions are favored.

Definition 3. The optimal solution to a trilevel program is considered *Sequentially Optimistic* if degeneracy in the third level is resolved in favor of the second level, and degeneracy in the second level is resolved in favor of the first level. A *Sequentially Optimistic* optimal solution is characterized according to Equation (6.35).

$$\left(\hat{x}^{\mathcal{L}}, \hat{x}^{\mathcal{C}}, \hat{y}\right) = \arg\max \left\{ NPV^{\mathcal{L}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) :\ x^{\mathcal{L}} \in X,\ x^{\mathcal{C}} \in \Psi^{\mathcal{L}}_{x^{\mathcal{C}}}(x^{\mathcal{L}}),\ y \in \Psi^{\mathcal{C}}_{y}(x^{\mathcal{L}}, x^{\mathcal{C}}) \right\} \tag{6.35}$$

Definition 4. The optimal solution to a trilevel program is considered *Hierarchically Optimistic* if degeneracy in the third level is resolved in favor of the first level, and degeneracy in the second level is also resolved in favor of the first level. A *Hierarchically Optimistic* optimal solution is characterized according to Equation (6.36).

$$\left(\hat{x}^{\mathcal{L}}, \hat{x}^{\mathcal{C}}, \hat{y}\right) = \arg\max \left\{ NPV^{\mathcal{L}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) :\ x^{\mathcal{L}} \in X,\ x^{\mathcal{C}} \in \Psi^{\mathcal{L}}_{x^{\mathcal{C}}}(x^{\mathcal{L}}),\ y \in \Psi^{\mathcal{L}}_{y}(x^{\mathcal{L}}, x^{\mathcal{C}}) \right\} \tag{6.36}$$

Surprisingly, the *Hierarchically Optimistic* model for resolving degeneracy does not guarantee the best possible objective for the first-level decision-maker. Therefore, we present a third optimistic approach to degeneracy.

Definition 5. The optimal solution to a trilevel program is considered *Strategically Optimistic* if degeneracy in the second level is resolved in favor of the first level, and degeneracy in the third level is resolved such that the best first-level solution is obtained. In order to define the *Strategically Optimistic* optimal solution, we characterize the second-level pessimistic reaction set of the third level ($\Upsilon^{\mathcal{C}}_{y}$) according to Equation (6.37).

$$\Upsilon^{\mathcal{C}}_{y}(x^{\mathcal{L}}, x^{\mathcal{C}}) = \underset{y \in \Psi_y(x^{\mathcal{L}}, x^{\mathcal{C}})}{\arg\min} \left\{ NPV^{\mathcal{C}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) \right\} \tag{6.37}$$

The idea behind the *Strategically Optimistic* model is that the second-level decision-maker accepts any resolution of degeneracy yielding a better objective value than the second-level pessimistic model. First, let us define in Equation (6.38) the rational reaction set for the second level in the pessimistic framework.

$$\Psi^{\Upsilon}_{x^{\mathcal{C}}}(x^{\mathcal{L}}) = \underset{x^{\mathcal{C}} \in X^{\mathcal{C}}(x^{\mathcal{L}})}{\arg\max} \left\{ NPV^{\mathcal{C}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) : y \in \Upsilon^{\mathcal{C}}_{y}(x^{\mathcal{L}}, x^{\mathcal{C}}) \right\} \tag{6.38}$$

Now, the strategic reaction set is defined as the tuple of second- and third-level decisions that belong to the rational reaction set of the third level and yield a better solution to the second level than the second-level pessimistic model. The strategic rational reaction set is defined in Equation (6.39).

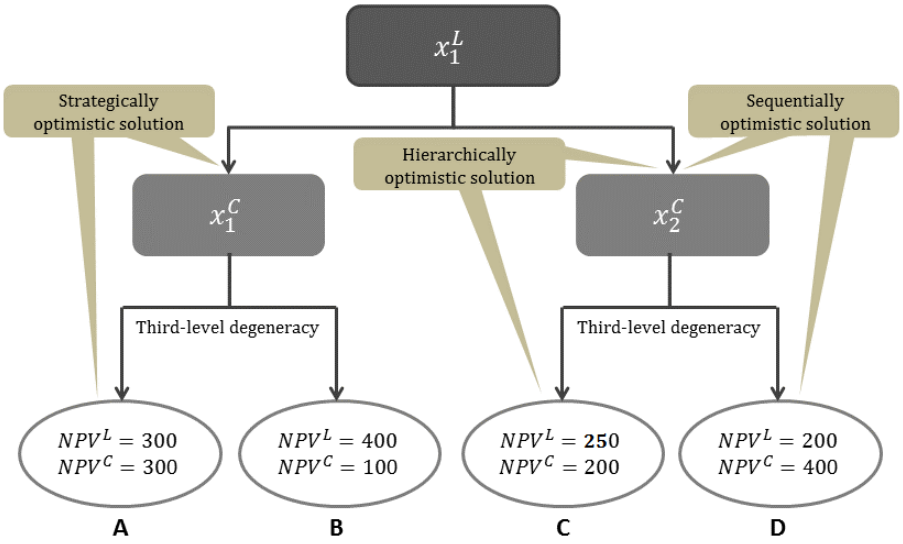
$$\Psi^{S}_{(x^{\mathcal{C}},y)}(x^{\mathcal{L}}) = \big\{(x^{\mathcal{C}},y) : NPV^{\mathcal{C}}(x^{\mathcal{L}},x^{\mathcal{C}},y) \geq NPV^{\mathcal{C}}(x^{\mathcal{L}},\tilde{x}^{\mathcal{C}},\tilde{y}), \\ \tilde{x}^{\mathcal{C}} \in \Psi^{\Upsilon}_{x^{\mathcal{C}}}(x^{\mathcal{L}}),\ \tilde{y} \in \Upsilon^{\mathcal{C}}_{y}(x^{\mathcal{L}},\tilde{x}^{\mathcal{C}}),\ y \in \Psi_{y}\big\} \tag{6.39}$$

Finally, a *Strategically Optimistic* optimal solution is characterized according to Equation (6.40).

$$(\hat{x}^{\mathcal{L}},\hat{x}^{\mathcal{C}},\hat{y}) = \arg\max\left\{NPV^{\mathcal{L}}(x^{\mathcal{L}},x^{\mathcal{C}},y) :\ x^{\mathcal{L}} \in X,\ (x^{\mathcal{C}},y) \in \Psi^{S}_{(x^{\mathcal{C}},y)}(x^{\mathcal{L}})\right\} \tag{6.40}$$

The difference between the three degeneracy resolution models is illustrated in Example 1.

Example 1. Figure 6.1 describes a case in which different approaches to degeneracy produce different solutions for a fixed first-level decision ($x^{\mathcal{L}}$). Here, the second level has two rational reactions $x^{\mathcal{C}}_1$ and $x^{\mathcal{C}}_2$, corresponding to different interpretations of third-level degeneracy. For each second-level solution, the third level has two alternative optimal reactions.



x_1^L
Strategically optimistic solution
Hierarchically optimistic solution
Sequentially optimistic solution
x_1^C
x_2^C
Third-level degeneracy
Third-level degeneracy
$NPV^L = 300$ $NPV^C = 300$
$NPV^L = 400$ $NPV^C = 100$
$NPV^L = 250$ $NPV^C = 200$
$NPV^L = 200$ $NPV^C = 400$
A
B
C
D

Figure 6.1: Example of different degeneracy resolution models.

Outcome D is the optimal solution under the *Sequentially Optimistic* model because degeneracy in the third level favors the objective of the competition ($NPV^{\mathcal{L}} = 200$, $NPV^{\mathcal{C}} = 400$). Under the *Hierarchically Optimistic* model, the optimal solution of the problem is given by outcome C ($NPV^{\mathcal{L}} = 250$, $NPV^{\mathcal{C}} = 200$). In this case, the third level tries to benefit the leader locally, pushing the competitor to select $x_2^{\mathcal{C}}$, which is detrimental for the first-level objective function. The *Strategically Optimistic* solution is given by outcome A ($NPV^{\mathcal{L}} = 300$, $NPV^{\mathcal{C}} = 300$), which is the best solution for the first level from all degeneracy resolution models. It is interesting to note that outcome B is not trilevel feasible because it does not belong to the inducible region under any degeneracy resolution model.

We have only presented degeneracy resolution models that characterize *optimistic* approaches. However, models for *pessimistic* resolution or mixed resolution (e.g. *optimistic-pessimistic*) can be easily extended from our definitions.

6.6 Properties of the trilevel capacity planning formulation

Solution algorithms for the capacity planning with competitive decision-makers rely on particular properties of the trilevel formulation. In this section we describe the most relevant properties for the algorithms we propose, and indicate the how to exploit them.

6.6.1 Stability regions

We study regions $R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y})$ of the first-level decision space that aggregate points $x^{\mathcal{L}}$ producing the same rational reaction of the lower levels $(\hat{x}^{\mathcal{C}}, \hat{y})$. We can expect the trilevel capacity planning formulation to have large stability regions because the second and third levels are indifferent to the distribution of capacities in the plants controlled by the first level. From the point of view of the market, only the total capacity of the leader in a given time period ($\mathscr{C}_t$) is relevant since all its plants offer the same price.

Definition 6. A *Stability Region* $R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y})$ in the trilevel capacity planning formulation is the set of first-level decisions that produce exactly the same rational reactions in the second and third levels. Formally, the stability region for second- and third-level reactions $(\hat{x}^{\mathcal{C}}, \hat{y})$ is characterized according to Equation (6.41).

$$R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y}) = \left\{ x^{\mathcal{L}} : \; x^{\mathcal{L}} \in X^{\mathcal{L}}, \; \hat{x}^{\mathcal{C}} \in \Psi^0_{x^{\mathcal{C}}}(x^{\mathcal{L}}), \; \hat{y} \in \Psi^0_y(x^{\mathcal{L}}, \hat{x}^{\mathcal{C}}) \right\} \tag{6.41}$$

where $\Psi^0_{x^{\mathcal{C}}}(x^{\mathcal{L}})$ and $\Psi^0_y(x^{\mathcal{L}}, \hat{x}^{\mathcal{C}})$ refer to one of the degeneracy resolution models described in Section 6.5.2.

Another property of the trilevel planning formulation that implies large stability regions can be derived from the intuition that expanding plants with slack capacity does not change the rational response of the second and third levels. Proposition 6.1 gives the mathematical description of this property.

Proposition 6.1. *Let $(\hat{Q})$ be the bilevel problem obtained after fixing the first-level decisions to $\hat{x}^{\mathcal{L}}$ in the second- and third-level problems presented in Equations (6.8)-(6.14). We denote by $(\hat{x}^{\mathcal{C}}, \hat{y})$ the corresponding optimal bilevel solution and by $\hat{\mu}$ the optimal multipliers associated with capacity constraints (6.18). Then,*

$$\bar{x}^{\mathcal{L}} \in \Bigg\{ (c_{1,1}, .., c_{|T|,|I^{\mathcal{L}}|}) : \qquad \left[\sum_{i \in I^{\mathcal{L}}} c_{t,i} = \sum_{i \in I^{\mathcal{L}}} \hat{c}_{t,i} \right] \vee \left[\begin{array}{c} \sum_{i \in I^{\mathcal{L}}} c_{t,i} \geq \sum_{i \in I^{\mathcal{L}}} \hat{c}_{t,i} \\ \hat{\mu}_{t',i} = 0 \quad \forall\, t' \in T^+_t, \; i \in I^{\mathcal{L}} \end{array} \right] \; \forall\, t \in T \Bigg\} \tag{6.42}$$

$$\implies \bar{x}^{\mathcal{L}} \in R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y}) \tag{6.43}$$

where T^+_t is the subset of periods after time t: $T^+_t = \{t' : t' \in T, t' \geq t\}$.

Proof. We want to prove that a first-level decision $\bar{x}^{\mathcal{L}}$ satisfying conditions (6.42) produces the same rational reaction $(\hat{x}^{\mathcal{C}}, \hat{y})$ as $\hat{x}^{\mathcal{L}}$. We divide the proof of Proposition 6.1 in three steps.

Step 1. *In the third-level problem, increasing the capacity of one plant cannot increase the demand assigned to any other plant.*

Let us denote by $(\hat{P})$ the third-level problem with capacities equal to $c_{t,i}$, and by $(\tilde{P})$ the problem in which plant i' increases its capacity by $\Delta C_{i'}$. We want to show that the optimal demand assignments $(y_{t,i,j})$ corresponding to problems $(\hat{P})$ and $(\tilde{P})$ satisfy the conditions presented in Equation (6.44).

$$\sum_{j\in J} \tilde{y}_{t,i,j} \leq \sum_{j\in J} \hat{y}_{t,i,j} \qquad \forall\, t \in T,\ i \in I\backslash \{i'\} \tag{6.44}$$

First, we notice that fully utilized plants ($\sum_{j\in J} \hat{y}_{t,i,j} = c_{t,i}$) in problem ($\hat{P}$) cannot increase the demand assigned to them. For all other plants with slack capacity ($\sum_{j\in J} \hat{y}_{t,i,j} + \hat{\epsilon}_{t,i} = c_{t,i}$), the Lagrange multiplier ($\hat{\mu}_{t,i}$) associated with the capacity constraint (6.18) must be zero according to complementary slackness of the third-level LP.

We prove in Appendix D that increasing the capacity of one plant cannot produce an increase in the optimal Lagrange multipliers associated with capacity constraints (6.18). In this case, the Lagrange multipliers ($\hat{\mu}_{t,i}$) of plants with slack capacity in problem ($\hat{P}$) remain at zero. Therefore, the condition presented in Equation (6.45) must be satisfied for the optimal Lagrange multipliers corresponding to problems ($\tilde{P}$) and ($\hat{P}$).

$$\tilde{\mu}_{t,i} \leq \hat{\mu}_{t,i} = 0 \qquad \forall\,(t,i) \in \{(t,i):\ t \in T, i \in I\backslash \{i'\}, \hat{\epsilon}_{t,i} > 0\} \tag{6.45}$$

Since the slack ($\hat{\epsilon}_{t,i}$) in plants that are not fully utilized in problem ($\hat{P}$) can be arbitrarily small, we conclude that the condition in Equation (6.44) must be satisfied. Otherwise, an increase in the demand assignments would produce a positive value in the Lagrange multipliers ($\tilde{\mu}_{t,i} > 0$) associated to capacity constraints.

Step 2. *The optimal objective value of the second level cannot improve if the total capacity of the leader increases* $\left(\sum_{i\in I^{\mathcal{L}}} \tilde{x}_{t,i} \geq \sum_{i\in I^{\mathcal{L}}} \hat{x}_{t,i}\ \forall\ t \in T\right)$ *and capacities of the competitors remain constant.*

Recall that the prices offered by competitors are given by Equation (6.2). Rewriting the objective function of the competition as in Equation (6.46), it is easy to note that the margin obtained from every unit sold only depends on the production cost ($F_{t,i}$) and the raw price ($P^{raw}_{t,i}$) of each plant.

$$NPV^{\mathcal{C}} = \sum_{t\in T}\sum_{i\in I^{\mathcal{C}}}\sum_{j\in J}\left(P^{raw}_{t,i} - F_{t,i}\right) y_{t,i,j} - \sum_{t\in T}\sum_{i\in I^{\mathcal{C}}}\left(A_{t,i}v_{t,i} + B_{t,i}w_{t,i} + E_{t,i}x_{t,i}\right) \tag{6.46}$$

Therefore, the condition presented in Equation (6.44) also implies that the objective function of the second level cannot improve from problem ($\hat{P}$) to ($\tilde{P}$); this is formalized in Equation (6.47).

$$NPV^{\mathcal{C}}(\tilde{x}^{\mathcal{L}}, x^{\mathcal{C}}, \tilde{y}) \leq NPV^{\mathcal{C}}(\hat{x}^{\mathcal{L}}, x^{\mathcal{C}}, \hat{y}) \qquad \forall\, x^{\mathcal{C}} \in X^{\mathcal{C}},\ (\tilde{y},\hat{y}) \in \Psi^0_y(\hat{x}^{\mathcal{L}}, x^{\mathcal{C}}) \tag{6.47}$$

Step 3. *If the expansion strategy of the leader ($\bar{x}_{t,i}$) satisfies the conditions presented in Equation (6.42), the bilevel problems ($\hat{Q}$) and ($\bar{Q}$) resulting from fixing the variables of the leader have the same rational reactions.*

First, we use the duality-based reformulation presented in Equations (6.21)-(6.25) to verify that optimal solutions ($\hat{y}$) to problem ($\hat{P}$) are feasible in ($\tilde{P}$). This is the case because capacity constraints (6.18) are relaxed with the additional expansions of the leader, and the dual objective function (right-hand side of Equation (6.21)) only changes in coefficients ($c_{t,i}$) for which the optimal Lagrange multipliers ($\hat{\mu}_{t,i}$) are equal to zero. Then, the optimal solution ($\hat{x}^{\mathcal{C}}, \hat{y}$) of the bilevel problem ($\hat{Q}$) resulting from fixing the first-level decisions to $\hat{x}^{\mathcal{L}}$, is feasible in the bilevel problem ($\bar{Q}$) since second-level constraints (6.9)-(6.11) are not affected by first-level decisions. Therefore, the optimal solution to problem ($\bar{Q}$) must be at least as good as the optimal solution to problem ($\hat{Q}$); this condition is formalized in Equation (6.48).

$$NPV^{\mathcal{C}}(\hat{x}^{\mathcal{L}}, \hat{x}^{\mathcal{C}}, \hat{y}) \leq NPV^{\mathcal{C}}(\bar{x}^{\mathcal{L}}, \bar{x}^{\mathcal{C}}, \bar{y}) \tag{6.48}$$

Furthermore, we can establish the inequalities given in Equation (6.49),

$$NPV^{\mathcal{C}}(\bar{x}^{\mathcal{L}}, \bar{x}^{\mathcal{C}}, \bar{y}) \leq NPV^{\mathcal{C}}(\hat{x}^{\mathcal{L}}, \bar{x}^{\mathcal{C}}, \bar{y}) \leq NPV^{\mathcal{C}}(\hat{x}^{\mathcal{L}}, \hat{x}^{\mathcal{C}}, \hat{y}) \tag{6.49}$$

where the inequality on the left is derived from Equation (6.47), and the inequality on the right follows from optimality of ($\hat{x}^{\mathcal{L}}, \hat{x}^{\mathcal{C}}, \hat{y}$) in problem ($\hat{Q}$). Equations (6.48) and (6.49) together demonstrate that problems ($\hat{Q}$) and ($\bar{Q}$) have the same optimal objective value. Since the optimal solutions of ($\hat{Q}$) are always feasible in ($\bar{Q}$), we conclude that they belong to the same stability region:

$\hat{x}^{\mathcal{L}} \in R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y})$

$\bar{x}^{\mathcal{L}} \in R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y})$ □

6.6.2 A cut to eliminate solutions in $R_{x^{\mathcal{L}}}(\hat{x}^{\mathcal{C}}, \hat{y})$

The capacity planning problem with competitive decision-makers is likely to have large stability regions as a consequence of Proposition 6.1. These stability regions can be partially characterized from the optimal solution of the bilevel problem ($\hat{Q}$). In order to describe the stability regions, we introduce the following parameters and sets.

Definition 7. Given the optimal bilevel solution $(x^{\mathcal{C},k}, y^k, \mu^k, \lambda^k)$ corresponding to problem (Q^k) with first-level decisions fixed to $(x^{\mathcal{L},k})$, we define:

- The total capacity of the leader in time period t:

$$\mathscr{C}_t^k = \sum_{i \in I^{\mathcal{L}}} C_{0,i} + \sum_{t' \in T_t^-} \sum_{i \in I^{\mathcal{L}}} H_i x_{t,i}^k \tag{6.50}$$

- The subset of time periods in which all plants controlled by the leader do not expand but expansions could change demand assignments:

$$\Gamma_{x0}^k = \left\{ t \in T : \sum_{i \in I^{\mathcal{L}}} x_{t,i}^k = 0, \quad \sum_{t' \in T_t^+} \mu_{t',i}^k > 0 \right\} \tag{6.51}$$

- The subset of time periods in which the leader expands and further expansions could change demand assignments:

$$\Gamma_{\mu+}^k = \left\{ t \in T : \sum_{i \in I^{\mathcal{L}}} x_{t,i}^k > 0, \quad \sum_{t' \in T_t^+} \mu_{t',i}^k > 0 \right\} \tag{6.52}$$

- The subset of time periods in which the leader expands but further expansions would not change demand assignments:

$$\Gamma_{\mu0}^k = \left\{ t \in T : \sum_{i \in I^{\mathcal{L}}} x_{t,i}^k > 0, \quad \sum_{t' \in T_t^+} \mu_{t',i}^k = 0 \right\} \tag{6.53}$$

We describe the *Stable Region* (R^k) corresponding to first-level decisions $(v^{\mathcal{L},k}, w^{\mathcal{L},k}, x^{\mathcal{L},k}, c^{\mathcal{L},k})$ by identifying if an alternative first-level solution satisfies the conditions presented in Proposition 6.1. In Equation (6.54), we introduce binary variables $z_{0,t}^k$ and $z_{1,t}^k$ to indicate if alternative solutions offer more, less, or the same capacity of the leader with respect to $\mathscr{C}_t^k$.

$$\begin{bmatrix} z_{0,t}^k = 1 \\ \sum\limits_{i \in I^{\mathcal{L}}} c_{t,i} = \mathscr{C}_t^k \end{bmatrix} \vee \begin{bmatrix} z_{1,t}^k = 1 \\ \sum\limits_{i \in I^{\mathcal{L}}} c_{t,i} < \mathscr{C}_t^k \end{bmatrix} \vee \begin{bmatrix} z_{0,t}^k + z_{1,t}^k = 0 \\ \sum\limits_{i \in I^{\mathcal{L}}} c_{t,i} > \mathscr{C}_t^k \end{bmatrix} \qquad \forall\, t \in T \tag{6.54}$$

Based on the variables that compare capacities in alternative solutions to capacities in the solution of problem (Q^k), we can characterized the *Stable Region* (R^k) of solution $(v^{\mathcal{L},k}, w^{\mathcal{L},k}, x^{\mathcal{L},k}, c^{\mathcal{L},k})$ with Equation (6.55).

$$\sum_{i \in I^{\mathcal{L}}} \sum_{t \in \Gamma^k_{x0}} x_{t,i} + \sum_{t \in \Gamma^k_{\mu 0}} z^k_{1,t} + \sum_{t \in \Gamma^k_{\mu +}} (1 - z^k_{0,t}) = 0 \tag{6.55}$$

A *no-good* cut to exclude all solutions that belong to this region $(x^{\mathcal{L}} \in R^k)$ is obtained by forcing the left-hand side of Equation (6.55) to be greater or equal than one (≥ 1).

6.6.3 Equations to tighten the HP relaxation

The *High-Point* (HP) relaxation of the bilevel reformulation presented in Section 6.4 can be obtained by removing from the model the second-level objective function. The HP problem usually yields a weak upper bound to the bilevel optimization problem because it gives control of all variables to the first level. The column-and-constraint generation method developed by Zeng & An [255] proposes a strategy to tighten the HP relaxation based on second-level reactions for which the second-level objective value is known. The idea is to generate cuts that constrain the second-level objective function to be at least as good as it would be with any of the second-level solutions that have been observed. These constraints, presented in Equation (6.56), are included in the HP problem to model the reactions of the second level.

$$NPV^{\mathcal{C}}(x^{\mathcal{L}}, x^{\mathcal{C}}, y) \geq NPV^{\mathcal{C}}(x^{\mathcal{L}}, \hat{x}^{\mathcal{C},k}, y^k) \tag{6.56}$$

where $\hat{x}^{\mathcal{C},k}$ are parameters modeling a fixed second-level response, and y^k are duplicate variables that model optimal demand assignments for any first-level decision $(x^{\mathcal{L}})$ when the second-level response $(\hat{x}^{\mathcal{C},k})$ is fixed. In order to enforce the third-level optimality of demand assignments, a full set of duplicate variables $(y^k_{t,i,j}, \mu^k_{t,i}, u^k_{t,t',i,} \lambda^k_{t,k})$ and constraints must be appended to the HP problem for each solution that has been observed. The constraints correspond to the duality-based reformulation of the third-level problem; they are presented in Equations (6.57)-(6.63).

$$\sum_{t\in T}\sum_{i\in I}\sum_{j\in J} P_{t,i,j} y^k_{t,i,j} = \sum_{t\in T}\left(\sum_{j\in J} D_{t,j}\lambda^k_{t,j} - \sum_{i\in I} C_{0,i}\mu^k_{t,i} + \sum_{i\in I}\sum_{t'\in T^-_t} H_i u^k_{t,t',i}\right) \tag{6.57}$$

$$\sum_{j\in J} y^k_{t,i,j} \le c_{t,i} \qquad \forall\ t\in T,\ i\in I^{\mathcal{L}} \tag{6.58}$$

$$\sum_{j\in J} y^k_{t,i,j} \le \hat{c}^k_{t,i} \qquad \forall\ t\in T,\ i\in I^{\mathcal{C}} \tag{6.59}$$

$$\sum_{i\in I} y^k_{t,i,j} = D_{t,j} \qquad \forall\ t\in T,\ j\in J \tag{6.60}$$

$$\lambda^k_{t,j} - \mu^k_{t,i} \le P_{t,i,j} \qquad \forall\ t\in T,\ i\in I,\ j\in J \tag{6.61}$$

$$u^k_{t,t',i} \ge \mu^k_{t,i} - M(1-x_{t',i}) \qquad \forall\ t\in T,\ t'\in T^-_t,\ i\in I^{\mathcal{L}} \tag{6.62}$$

$$y^k_{t,i,j};\quad \mu^k_{t,i},\quad u^k_{t,t',i}\in\mathbb{R}^+;\quad \lambda^k_{t,j}\in\mathbb{R} \qquad \forall\ t\in T,\ i\in I,\ j\in J \tag{6.63}$$

We observe that the cuts modeled by Equations (6.56)-(6.63) do not exclude any solution that is trilevel feasible. All first-level solutions remain feasible after the cuts are appended to the HP problem because we assume that there is always enough capacity in the third level to satisfy all demands. Therefore, the duality-based reformulation of the third level always have a feasible solution. Additionally, we can guarantee that no point in the *Inducible Region* of the trilevel problem can be excluded from the tightened HP problem because Equation (6.56) provides lower bounds on $NPV^{\mathcal{C}}$ based on solutions that are feasible in the second- and third-level problems; solutions in the *Inducible Region* must be optimal in the second and third levels, which implies that their corresponding $NPV^{\mathcal{C}}$ must be greater or equal than any bound imposed by inequality (6.56). Furthermore, if inequality (6.56) is active, then $(y_{t,i,j}, \mu_{t,i}, u_{t,t',i}, \lambda_{t,k}) = (y^k_{t,i,j}, \mu^k_{t,i}, u^k_{t,t',i}, \lambda^k_{t,k})$ satisfies all other constraints in the HP problem.

6.7 Algorithm 1: Constraint-directed exploration

We use the stability regions of the capacity planning problem and the equations describing them to design a constraint-directed exploration of the leader's decision space. Algorithm 1 performs an accelerated search on the inducible region containing the optimal trilevel solution. The details of the algorithm are presented below.

6.7.1 Reaching the sequentially optimistic solution

Algorithm 1 uses *no-good* cuts derived from Equation (6.55) to iteratively characterize the first-level decision space in terms of stability regions. The algorithm finds the best trilevel feasible solution inside each region (R^k) by solving the single-level reformulation (Q^k) of the second- and third-level problems. The search is directed towards unexplored first-level decisions by adding to the high-point (HP^k) of the bilevel reformulation, cuts that exclude the regions previously analyzed ($R^k \; \forall \; k \in K \backslash \{|K|\}$). Convergence of the algorithm is guaranteed because the problem has a discrete number of first-level decisions, which implies a finite number of regions. The operations performed by the algorithm are divided in six steps.

Step 1: Solve HP^k over the unexplored first-level decision space. Identify the first-level solution ($x^{\mathcal{L}}_{HP^k}$). If HP^k is infeasible, terminate and return the incumbent.

Step 2: Update the upper bound (UB^k). If UB^k is less than the best lower bound (LB^*), terminate and return the incumbent.

Step 3: Solve Q^k with first-level variables fixed to $x^{\mathcal{L}}_{HP^k}$. Identify the second-level solution ($x^{\mathcal{C}}_{Q^k}$).

Step 4: Identify the sets Γ^k_{x0}, $\Gamma^k_{\mu+}$, and $\Gamma^k_{\mu 0}$ describing the region R^k that contains $x^{\mathcal{L}}_{HP^k}$ and all other first-level solutions satisfying the condition given by Equation (6.42).

Step 5: Update LB^* if solution ($x^{\mathcal{L}}_{HP^k}, x^{\mathcal{C}}_{Q^k}, y_{Q^k}$) is better than the incumbent. Terminate if UB^k is equal to LB^*.

Step 6: Generate *no-good* cuts to exclude R^k from HP^{k+1}. Go back to Step 1.

Algorithm 1 has two possible stopping criteria:

C1: If $UB^k < LB^*$ in Step 2 or Step 5, return incumbent. In this case, no solution contained in the unexplored region of the first-level decision space can be better than the incumbent.

C2: If HP^k is infeasible in Step 1, return incumbent. In this case, the first-level decision space has been exhaustively analyzed.

It is worth noticing that Step 1 produces an improving UB because the feasible region of problem HP^k is successively reduced. On the other hand, Step 3 finds a trilevel feasible solution that corresponds to the *sequentially optimistic* model of degeneracy because problem Q^k resolves degeneracy in favor of the second level. A *sequentially optimistic* solution might be very detrimental for the first level since demands assigned to the leader are degenerate according to the pricing model presented in Equation (6.1). Furthermore, instances with a degenerate third level might not close the gap between UB^k and LB^* because problems HP^k and Q^k use different degeneracy resolution models. In this case, an exhaustive search could be necessary to meet stopping criterion C2.

6.7.2 Reaching the hierarchically optimistic solution

Several additional operations are needed to instruct the algorithm to obtain the *hierarchically optimistic* solution. The idea is modify Step 4 to find among the degenerate solutions the one that favors the leader according to the *hierarchically optimistic* model. Two additional optimization problems must be defined: the high-point problem $(HP_R^K(x_{Q^k}^{\mathcal{C}}))$ constraint to region R^k with second-level variables fixed to $x_{Q^k}^{\mathcal{C}}$, and the high-point problem $(HP_R^K(NPV^{\mathcal{C}}))$ constraint to region R^k with second-level objective value fixed to $NPV^{\mathcal{C}}(x_{HP^k}^{\mathcal{L}}, x_{Q^k}^{\mathcal{C}}, y_{Q^k})$. If the solutions obtained from these problems are found not to resolve degeneracy in favor of the first level, we add penalties to the objective functions of the second and third levels and go back to Step 3. A detailed description of the steps required to reach the hierarchically optimistic solution are presented below.

Step 4a: Identify the sets Γ_{x0}^k, $\Gamma_{\mu+}^k$, and $\Gamma_{\mu 0}^k$ describing the region R^k that contains $x_{HP^k}^{\mathcal{L}}$ and all other first-level solutions satisfying the condition given by Equation (6.1).

Step 4b: Solve $HP_R^K(x_{Q^k}^{\mathcal{C}})$ and identify the third-level response $(y_{HP_R^k})$. If the third-level solution is different from the one obtained in Step 3 $(\sum_{i\in I^{\mathcal{C}}} y_{Q^k} \neq \sum_{i\in I^{\mathcal{C}}} y_{HP_R^k} \; \forall \; t \in T)$, add a penalty to the third-level objective to resolve degeneracy in favor of the first level. Go back to Step 3.

Step 4c: Solve $HP_R^K(NPV^{\mathcal{C}})$. If the first-level objective is different from the one obtained in Step 3 $(NPV^{\mathcal{L}}(x_{HP^k}^{\mathcal{L}}, x_{Q^k}^{\mathcal{C}}, y_{Q^k}) \neq NPV^{\mathcal{L}}(x_{HP_R^k}^{\mathcal{L}}, x_{Q^k}^{\mathcal{C}}, y_{HP_R^k}))$, add a penalty to the second-level objective to resolve degeneracy in favor of the first level. Go back to Step 3.

Problem $HP_R^K(x_{Q^k}^{\mathcal{C}})$ has two purposes: to find the best first-level solution in R^k after the second-level response $(x_{Q^k}^{\mathcal{C}})$ has been observed, and to resolve third-level degeneracy in favor of the first level $(y_{HP_R^k} \in \Psi_y^{\mathcal{L}}(x_{HP_R^k}^{\mathcal{L}}, x_{Q^k}^{\mathcal{C}}))$. Problem $HP_R^K(NPV^{\mathcal{C}})$ is intended to check if second-level degeneracy is being resolved in favor of the first level $(x_{HP_R^k}^{\mathcal{C}} \in \Psi_{x^{\mathcal{C}}}(x_{HP_R^k}^{\mathcal{L}}))$. The steps of the algorithm are presented schematically in Figure 6.2; diamonds control the flow of the algorithm, light gray boxes are simple operations and dark gray boxes involve optimization problems.

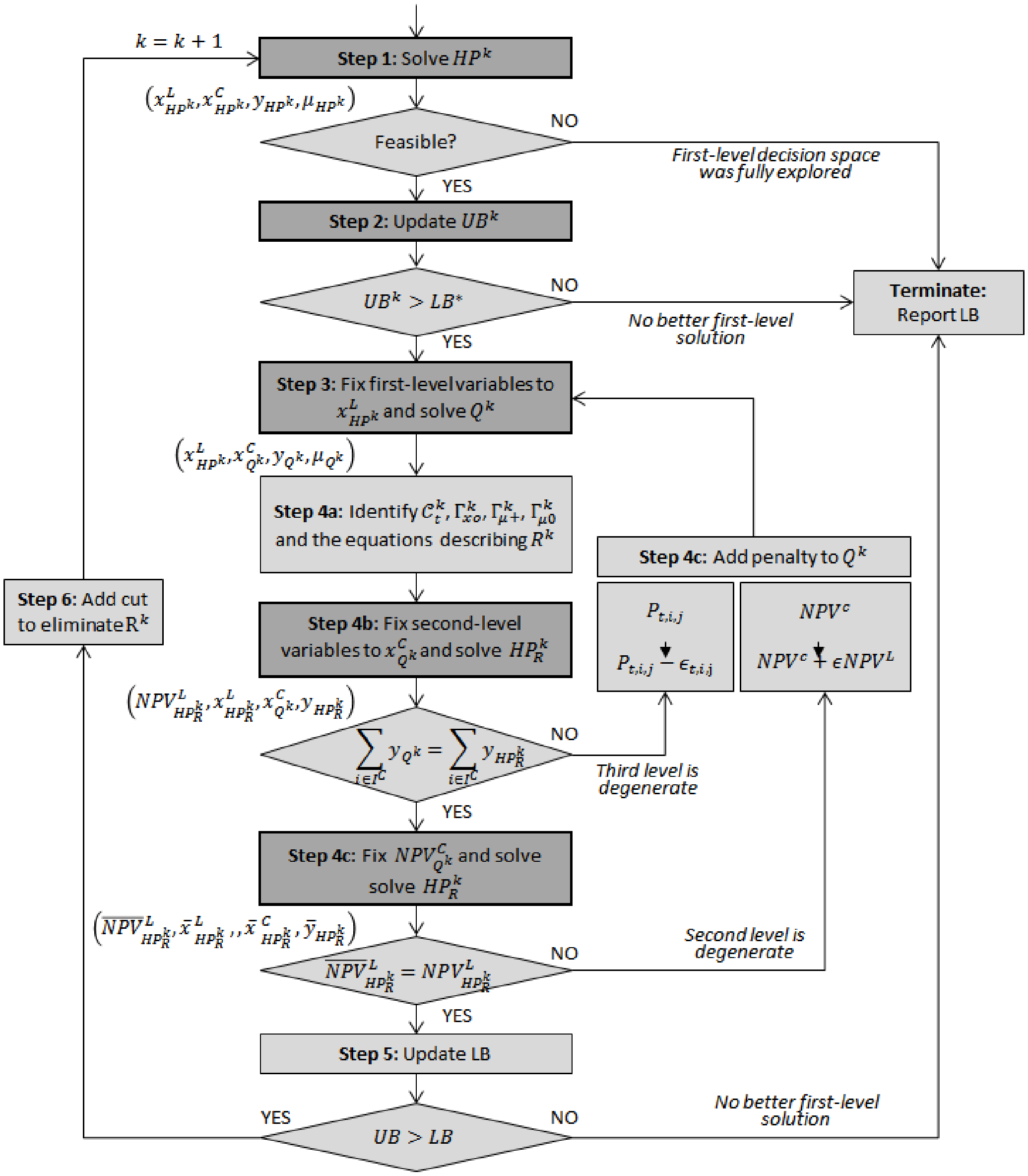

$k = k + 1$
Step 1: Solve HP^k
$(x^L_{HP^k}, x^C_{HP^k}, y_{HP^k}, \mu_{HP^k})$
Feasible?
NO
First-level decision space was fully explored
YES
Step 2: Update UB^k
$UB^k > LB^*$
NO
No better first-level solution
Terminate: Report LB
YES
Step 3: Fix first-level variables to $x^L_{HP^k}$ and solve Q^k
$(x^L_{HP^k}, x^C_{Q^k}, y_{Q^k}, \mu_{Q^k})$
Step 4a: Identify $C^k_t, \Gamma^k_{xo}, \Gamma^k_{\mu+}, \Gamma^k_{\mu 0}$ and the equations describing R^k
Step 4c: Add penalty to Q^k
$P_{t,i,j}$
$P_{t,i,j} - \epsilon_{t,i,j}$
NPV^c
$NPV^c + \epsilon NPV^L$
Step 6: Add cut to eliminate R^k
Step 4b: Fix second-level variables to $x^C_{Q^k}$ and solve HP^k_R
$(NPV^L_{HP^k_R}, x^L_{HP^k_R}, x^C_{Q^k}, y_{HP^k_R})$
$\sum_{i \in I^C} y_{Q^k} = \sum_{i \in I^C} y_{HP^k_R}$
NO
Third level is degenerate
YES
Step 4c: Fix $NPV^C_{Q^k}$ and solve solve HP^k_R
$(\overline{NPV}^L_{HP^k_R}, \bar{x}^L_{HP^k_R},, \bar{x}^C_{HP^k_R}, \bar{y}_{HP^k_R})$
$\overline{NPV}^L_{HP^k_R} = NPV^L_{HP^k_R}$
NO
Second level is degenerate
YES
Step 5: Update LB
YES
NO
No better first-level solution
$UB > LB$

Figure 6.2: Algorithm 1 towards the hierarchically optimistic solution.

6.8 Algorithm 2: Column-and-constraint generation algorithm

As opposed to Algorithm 1, Algorithm 2 finds optimal trilevel solutions by exploring the decision space of the second-level problem. Algorithm 2 is inspired by the column-and-constraint generation algorithm developed by Zeng & An [255] for linear bilevel problems with mixed-integer variables in both levels. However, our algorithm operates over the bilevel reformulation of the capacity planning problem presented in Section 6.4, which already enforces optimality of the variables control by the markets; therefore, no additional reformulation is needed for the continuous variables. The details of the algorithm are presented below.

6.8.1 Reaching the strategically optimistic solution

Algorithm 2 uses the cuts presented in Section 6.6.3 to sequentially tighten the *high-point* relaxation of the trilevel capacity planning problem. The algorithm iterates between a master problem (MP^k) that provides upper bounds (UP^k) and the single-level reformulation of the second- and third-level problems (Q^k_R) that explores the decision space of the second level. Problem MP^k is the high-point relaxation of the bilevel reformulation with the cuts modeled by Equations (6.56)-(6.63). The search is directed towards unexplored second-level decisions by adding no-good cuts to problem Q^k_R, such that second-level decisions that were already observed are not considered in future iterations. The no-good cuts used to diversify the search in the second-level decision space are presented in Equation (6.64).

$$\sum_{(t,i)\in\left\{(t,i):\ [x^{\mathcal{C}}_{i,j}]_{Q^k_R}=1\right\}} (1 - x^{\mathcal{C}}_{t,i}) \quad + \sum_{(t,i)\in\left\{(t,i):\ [x^{\mathcal{C}}_{i,j}]_{Q^k_R}=0\right\}} x^{\mathcal{C}}_{t,i} \quad \geq \quad 1 \tag{6.64}$$

where $[x^{\mathcal{C}}_{i,j}]_{Q^k_R}$ denotes the second-level optimal solution for problem Q^k_R.

The algorithm is identified as a column-and-constraint generation approach because at every iteration, a new second-level candidate solution ($\hat{c}_{t,i}\ \forall\, t \in T, i \in I^{\mathcal{C}}$) is appended to MP^k, together with the constraints and variables modeling third-level optimal responses. Convergence of Algorithm 2 is guaranteed because the problem has a discrete number of second-level decisions; therefore, a finite number of different columns and constraints can be added to MP^k. The operations performed by Algorithm 2 are divided in five steps; they are presented schematically in Figure 6.3.

Step 1: Solve MP^k. Identify the first-level solution ($x^{\mathcal{L}}_{MP^k}$) and the second-level objective value $NPV^{\mathcal{C}}(x^{\mathcal{L}}_{MP^k},x^{\mathcal{C}}_{MP^k},y_{MP^k})$.

Step 2: Update the upper bound (UB^k). If UB^k is less or equal to the best lower bound (LB^*), terminate and return the incumbent.

Step 3: Fix first-level variables to $x^{\mathcal{L}}_{MP^k}$ and solve Q^k_R including the no-good cuts presented in Equation (6.64). If Q^k_R is infeasible, terminate and return the solution obtained from MP^k ($x^{\mathcal{L}}_{MP^k}, x^{\mathcal{C}}_{MP^k}, y_{MP^k}$). Otherwise, identify the second-level solution ($x^{\mathcal{C}}_{Q^k}$) and the second-level objective value. If $NPV^{\mathcal{C}}(x^{\mathcal{L}}_{MP^k},x^{\mathcal{C}}_{Q^k_R},y_{Q^k_R}) < NPV^{\mathcal{C}}(x^{\mathcal{L}}_{MP^k},x^{\mathcal{C}}_{MP^k},y_{MP^k})$, terminate and return the solution obtained from MP^k ($x^{\mathcal{L}}_{MP^k}, x^{\mathcal{C}}_{MP^k}, y_{MP^k}$).

Step 4: Update the best LB^*. If UB^k is less or equal to the best lower bound (LB^*), terminate and return the incumbent.

Step 5: Generate the columns and constraints to tighten MP^{k+1}, and the cuts to exclude $x^{\mathcal{C}}_{Q^k_R}$ from Q^k_R. Go back to Step 1.

Algorithm 2 has three possible stopping criteria:

C1: If $UB \leq LB^*$ in Step 2 or in Step 4, both problems MP^k and Q^*_R yield the same optimal value $(NPV^{\mathcal{L}}(x^{\mathcal{L}}_{MP^k},x^{\mathcal{C}}_{MP^k},y_{MP^k}))$ because there is no third-level degeneracy favoring the second-level objective in problem Q^*_R.

C2: If Q^k_R is infeasible in Step 3, return the solution obtained from MP^k ($x^{\mathcal{L}}_{MP^k}, x^{\mathcal{C}}_{MP^k}, y_{MP^k}$). In this case, the second-level decision space has been exhaustively analyzed.

C3: If $NPV^{\mathcal{C}}(x^{\mathcal{L}}_{MP^k},x^{\mathcal{C}}_{MP^k},y_{MP^k}) \geq NPV^{\mathcal{C}}(x^{\mathcal{L}}_{MP^k},x^{\mathcal{C}}_{Q^k_R},y_{Q^k_R})$, return the solution obtained from MP^k ($x^{\mathcal{L}}_{MP^k}, x^{\mathcal{C}}_{MP^k}, y_{MP^k}$). In this case, no other solution contained in the unexplored region can be better for the second level than ($x^{\mathcal{L}}_{MP^k}, x^{\mathcal{C}}_{MP^k}, y_{MP^k}$).

It is worth noticing that Step 1 produces an improving UB^k because the feasible region of problem HP^k is successively reduced. The solutions obtained from HP^k correspond to the *strategically optimistic* model of degeneracy since the control of all variables are granted to the first level and only a constraint on the second-level objective value is imposed. Step 3 directs the second-level decisions towards the *strategically optimistic* solution but resolves third-level degeneracy in favor of the second level. Consequently, the gap between UB^k and LB^* might not close; in that case, either criterion C2 or C3 must be met.

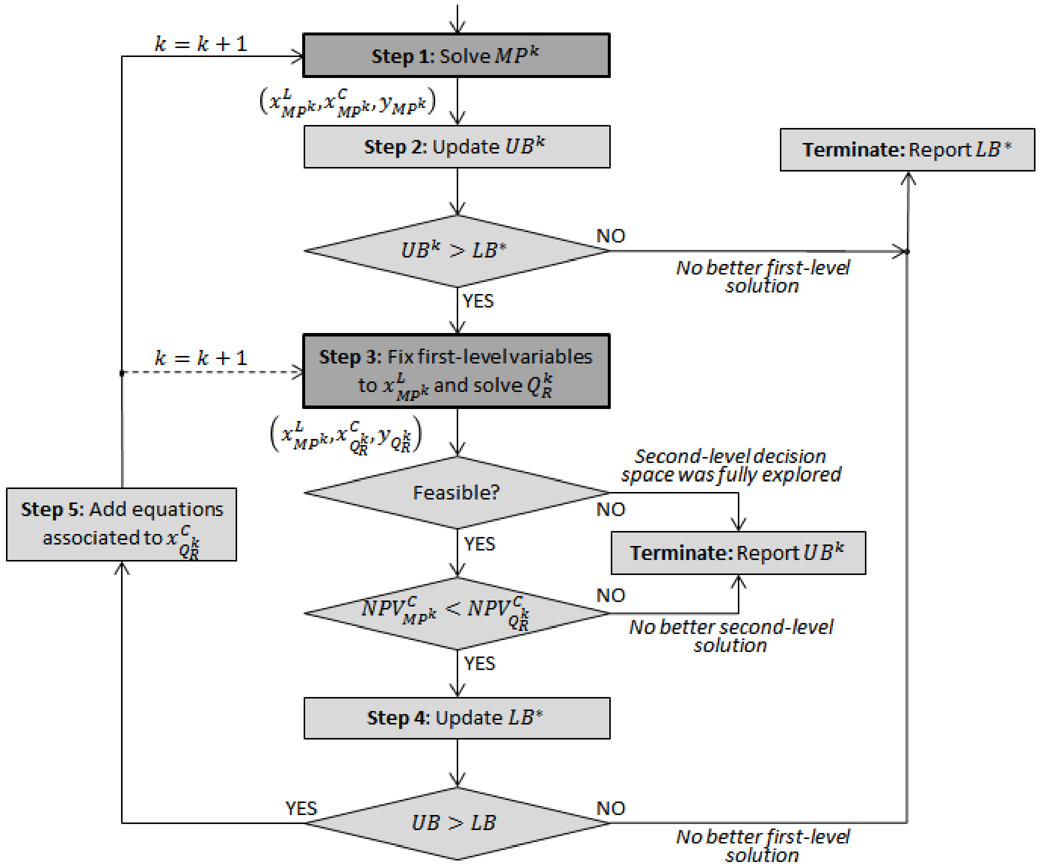

$k = k + 1$
Step 1: Solve MP^k
$(x^L_{MP^k}, x^C_{MP^k}, y_{MP^k})$
Step 2: Update UB^k
Terminate: Report LB^*
$UB^k > LB^*$
NO
No better first-level solution
YES
$k = k + 1$
Step 3: Fix first-level variables to $x^L_{MP^k}$ and solve Q^k_R
$(x^L_{MP^k}, x^C_{Q^k_R}, y_{Q^k_R})$
Second-level decision space was fully explored
Feasible?
NO
Step 5: Add equations associated to $x^C_{Q^k_R}$
YES
Terminate: Report UB^k
$NPV^C_{MP^k} < NPV^C_{Q^k_R}$
NO
No better second-level solution
YES
Step 4: Update LB^*
YES
$UB > LB$
NO
No better first-level solution

Figure 6.3: Algorithm 2 towards the strategically optimistic solution.

Remark. Algorithms 1 and 2 are guaranteed to find the same trilevel optimal solution only in instances with no degeneracy in any level. If degeneracy is present, no result can be established about the relative performance of the algorithms because they look for different solutions, and these two problems can be arbitrarily difficult to solve. For non-degenerate instances we can establish that Algorithm 1 requires at least the same number iterations as Algorithm 2. This is the case because Algorithm 2 explores at most one point in each stability region, which is not true for Algorithm 1. However, it does not imply that Algorithm 2 outperforms Algorithm 1 in execution time because Algorithm 2 adds many variables and constraints to MP^k at every iteration, which increases the complexity of the iterations.

6.9 Capacity planning instances

We test Algorithms 1 and 2 using two instances of the capacity planning problem with competitive decision-makers. The algorithms are implemented to find the *hierarchically* and *strategically optimistic* solutions, respectively. The first instance is an illustrative example that we use to provide insight about the performance of the algorithms; the second instance is an industrial example of practical interest for the air separation industry.

Instance 1. Illustrative instance of trilevel capacity planning

This example considers one existing plant ($\mathcal{L}_1$) and one potential plant ($\mathcal{L}_2$) controlled by the leader, as well as one existing plant ($\mathcal{C}_1$) and one potential plant ($\mathcal{C}_2$) controlled by the competition. The market comprises four customers (M_j) with demands for a single commodity. The planning problem has a horizon of 20 time periods, in which the plants are allowed to expand in periods 1, 5, 9, 13, and 17. The trilevel model has 100 discrete variables and 120 constraints in the first level, 100 discrete variables and 120 constraints in the second level, and 320 continuous variables and 160 constraints in the third level. A scheme representing the location of plants and customers is presented in Figure 6.4; the parameters of the instance are given in Tables 6.1, 6.2, and 6.3.

The optimal expansion strategy for the leader comprises expanding plant $\mathcal{L}_2$ at time 9 to capture the demand from M_2. The rational reaction of the competition is to expand plant $\mathcal{C}_2$ at time 9 to maintain M_4 by offering a lower price than the leader. The optimal assignments in the first and last time periods are presented Figure 6.5. The elements of the objective functions for the trilevel optimal solution are presented in Table 6.4.

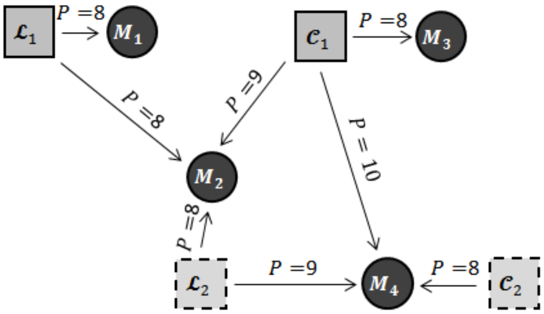


Figure 6.4: Network of plants and markets in Instance 1.

Time (t)	**Customer** (j)			
	$D_{t,1}$	$D_{t,2}$	$D_{t,3}$	$D_{t,4}$
1-4	3.75	0	3	10
5-8	3.75	0	3	10
9-12	3.75	8	3	10
13-16	3.75	10	3	10
17-20	3.75	10	3	10

Table 6.1: Market demands in Instance 1.

Parameter	**Plant** (i)			
	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{C}_1$	$\mathcal{C}_2$
$A_{t,i}$	-	0	-	0
$B_{t,i}$	15	15	15	15
$E_{t,i}$	110	110	110	110
$F_{t,i}$	3	3	2	4
$G_{t,i,1}$	1	10	10	10
$G_{t,i,2}$	10	1	2	10
$G_{t,i,3}$	10	10	1	10
$G_{t,i,4}$	10	2	3	1
$C_{0,i}$	3.75	0	30	0
$H_{t,i}$	30	30	30	30

Table 6.3: Cost parameters and initial capacities in Instance 1.

Costumer	**Plant** (i)			
(j)	$P_{t,\mathcal{L}_1,j}$	$P_{t,\mathcal{L}_2,j}$	$P_{t,\mathcal{C}_1,j}$	$P_{t,\mathcal{C}_2,j}$
M_1	8	8	17	17
M_2	8	8	9	17
M_3	17	17	8	17
M_4	9	9	10	8

Table 6.2: Selling prices in Instance 1.

Instance 1 has been designed such that Algorithms 1 and 2 find exactly the same solution at every iteration. This is possible because the *hierarchically* and *strategically optimistic* solutions coincide. The convergence of the upper and lower bounds for both algorithms can be observed in Figure 6.6.

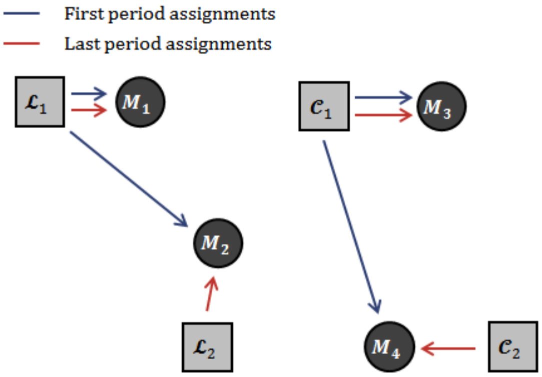


Figure 6.5: Optimal demand assignments for Instance 1.

Element of objective function	Leader	Competition
Income from sales [M$]:	1,496	2,240
Expansion cost [M$]:	110	110
Maintenance cost [M$]:	480	480
Production cost [M$]:	561	760
Transportation cost [M$]:	187	420
Total NPV [M$]:	158	470

Table 6.4: Optimal objective values for Instance 1.

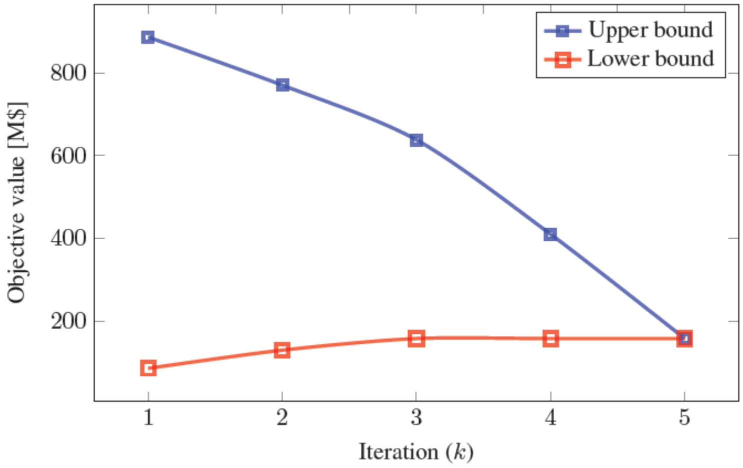


Figure 6.6: Convergence of Algorithms 1 and 2 in Instance 1.

Both algorithms were implemented in GAMS 24.4.1 and the optimization problems were solved using GUROBI 6.0.0 on an Intel Core i7 CPU 2.93 Ghz processor with 4 GB of RAM. Table 6.5 presents the computational statistics for problems HP^k of Algorithm 1 and MP^k of Algorithm 2 in the first and last iterations. We observe that both problems have the same number of continuous variables and constraints in the first iteration, but they grow much faster in Algorithm 2 than in Algortihm 1; on the other hand, Algorithm 1 has a modest increase in the number of binary variables. Our analysis indicates that instances for which both algorithms explore the solution space in the same order can be solved faster with Algorithm 1 because the complexity of iterations increases at a lower rate.

Problem	**First iteration**	**Final iteration**	
	HP^k & MP^k	HP^k	MP^k
Constraints:	1,015	1,035	3,596
Continuous variables:	835	835	3,331
Binary variables:	120	128	120
CPU time [s]:	2	5	9

Table 6.5: Computational statistics of Algorithms 1 and 2 in Instance 1.

Instance 2. Industrial example

This example is based on the instance Middle-size 1 presented in Chapter 5; we extend the problem by considering expansions in the plants controlled by the competition. The problem comprises the production and distribution of one product to 15 customers. Initially, the leader has three plants with initial capacities equal to 27,000 ton/period, 13,500 ton/period, and 31,500 ton/period. Additionally, the leader considers the possibility of opening a new plant at a candidate location. As for the competition, it controls three plants with initial capacities of 22,500 ton/period, 45,000 ton/period and 49,500 ton/period; the competition also has a candidate location for a new plant. The investment decisions are evaluated over a time horizon of 5 years divided in 20 time periods; all producers are allowed to expand only every fourth time period.

Selling prices and market demands follow an increasing trend during the time horizon. Investment and maintenance costs grow in time to adjust for inflation. The costs of production also have an increasing trend but exhibit a seasonal variation related to electricity prices. The exact data for this industrial example is provided as part of the Supplementary material.

The trilevel model has 200 discrete variables and 240 constraints in the first level, 200 discrete variables and 240 constraints in the second level, and 1,200 continuous variables and 460 constraints in the third level. The algorithms were implemented in GAMS 24.4.1 and were solved using GUROBI 6.0.0 on an Intel Core i7 CPU 2.93 Ghz processor with 4 GB of RAM. In this industrial instance, Algorithm 2 is very efficient; it only needs two iterations to find the trilevel optimal solution, while Algorithm 1 requires 7 iterations. Both algorithms find the same solution because the *hierarchically* and *strategically optimistic* solutions coincide. The convergence of the upper and lower bounds to the optimal solution (M$302) can be observed in Figure 6.7.

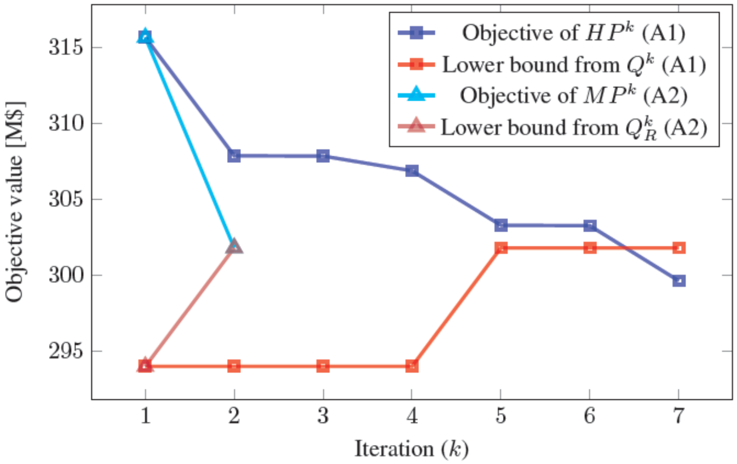


Figure 6.7: Convergence of Algorithm 1 (A1) and Algorithm 2 (A2) in Instance 2.

Table 6.6 presents the computational statistics for problems HP^k of Algorithm 1 and MP^k of Algorithm 2 in the first and last iterations. We observe that the number of continuous variables and constraints grows very quickly for problem MP^k in just one iteration, even though the number of binary variables stay constant. The total time required by Algorithm 1 to solve the instance is 46 s, in contrast to Algorithm 2 that only takes 8 s. This instance shows the advantage of Algorithm 2 for problems that are solved in few iterations.

Problem	**First iteration**	**Final iteration**	
	HP^k & MP^k	HP^k	MP^k
Constraints:	4,174	4,229	7,620
Continuous variables:	3,835	3,835	7,259
Binary variables:	240	264	240
Solution time [s]:	2	12	6

Table 6.6: Computational statistics of Algorithms 1 and 2 in Instance 2.

The optimal investment plan for the leader in this industrial example is to expand plant $\mathcal{L}_3$ at time 1 and 5. The rational reaction of the competition is not to expand at all. The optimal capacities and production levels of the plants controlled by the leader are presented in Figure 6.8; we can observe in Figure 6.8 that all production plants have high utilization. The elements of the objective functions for the trilevel optimal solution are presented in Table 6.7.

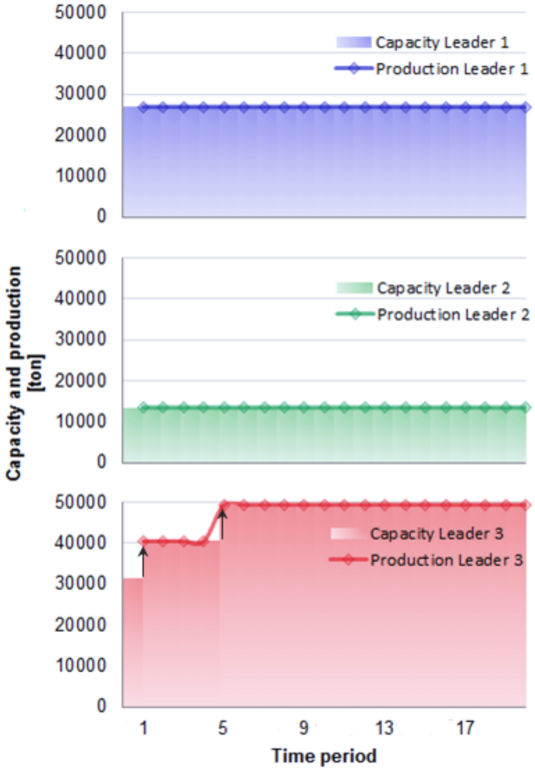


Figure 6.8: Capacity and production of the leader in Instance 2.

Element of objective function	**Leader**	**Competition**
Income from sales [M$]:	816	504
Investment in new plants [M$]:	0	0
Expansion cost [M$]:	56	0
Maintenance cost [M$]:	94	97
Production cost [M$]:	288	171
Transportation cost [M$]:	76	41
Total NPV [M$]:	302	195

Table 6.7: Optimal objective values for Instance 2.

The optimal capacity expansion plan for the trilevel formulation differs from the results reported in Chapter 5 for the bilevel formulation in which the competitions are not allowed to expand. Even though the optimal expansion strategy for the competition is not to expand, considering the competition as a rational decision-maker changes the optimal plan of the leader. This result exposes some of the counter-intuitive mechanisms present in multilevel optimization problems. In this particular instance, if the leader implements the bilevel optimal plan prescribing three expansions instead of two, the rational reaction of the competition is to expand plant $\mathcal{C}_1$ at time 1. This expansion plans would produce a NPV for the leader equal to M\$294, which is 2.5% lower than the trilevel optimal solution (M\$302). This measure of regret illustrates the value of obtaining the trilevel optimal solution in comparison to a bilevel formulation that assumes static competitors.

6.10 Summary

For the first time, a fully competitive model for the capacity planning problem has been formulated as a trilevel optimization. It allows simultaneously considering the conflicting interests of three rational decision-makers within a mathematical programming framework. We have also addressed for the first time the topic of degeneracy in multilevel decision problems. Our research found a void in definitions and models that induce ambiguity in the characterization of trilevel optimal solutions. We have introduced several extensions of the *optimistic* models from bilevel programming and we have provided algorithms that allow finding different optimal solutions.

The proposed model belongs to a challenging class of mathematical problems: multilevel programming with integer variables in more than one level. The few general methods available to solve this type of problems are in an early stage. We have developed two problem specific solution methods that rely on different properties of the formulation. The examples show that none of the two algorithms strictly dominates the other in terms of performance, indicating that both are interesting approaches to solve this problem. The solutions obtained from the new formulation unveil complex interactions that are very difficult to predict. A significant improvement over previously proposed models is quantified in monetary terms for the industrial instance.

The type of problems that we have addressed are of interest in applications where discrete decisions are taken by different players. As the range of applications is expected to increase, we consider the generalization of the algorithms as an important direction for future research; additionally, efficiency and numerical stability of the algorithms can still improve. For the industrial application of the capacity expansion model, we believe that it is important to extend the model to include stochastic parameters.

Chapter 7

Conclusions

7.1 Review of the thesis

This thesis has proposed optimization models and solution methods for supply chain planning with uncertainty and hierarchical decision-makers. Uncertainty and external decision-makers are the main factors affecting supply chain performance because they define conditions that are not under control of supply chain planners. The examples presented throughout the chapters have provided evidence of the value added by considering both uncontrollable factors in logistic and production planning models. In the following sections, we offer a critical review of the proposed optimization models, the solution methods, and the results obtained.

7.1.1 Design of resilient supply chains with risk of disruptions

In Chapter 2, we developed an MILP two-stage stochastic programming model for the design of resilient supply chains. We focused on the uncertain availability of distribution centers (DCs) as a consequence of potential disruptions. The proposed model includes, in the first stage, the design decisions that determine the location and capacity of DCs; the model considers in the second stage demand assignments in scenarios describing disruptions at the candidate locations for DCs. The main challenge for the implementation of our supply chain design approach arises from the rapid growth in the number of scenarios as a function of the number of DC candidate locations.

We proposed several strategies that allow solving large-scale MILP models for the design of resilient supply chains. The initial step to develop an efficient solution method was to strengthen the original MILP model with a set of redundant constraints. The addition of tightening constraints implied an increase in the size of the model, but also a significant improvement of its LP relaxation. In the illustrative example presented in Section 2.5, the optimality gap of the relaxed model was reduced from 30.0% to 1.9%.

The main benefit of the tightened model for the design of large-scale supply chains is that it allows an effective implementation of Benders decomposition. The additional constraints expand the set of cuts that can be generated from the Benders subproblems in comparison to the original model. Given this cut-richer formulation, it becomes important to select solutions producing good Benders cuts from the dual degenerate solutions of the subproblems. We achieved a judicious selection of the cuts by solving additional LPs that provided non-dominated pareto-optimal cuts. The effort required to solve additional LPs is compensated by a significant reduction in the number of iterations in Benders decomposition.

We implemented the multi-cut version of Benders decomposition and strengthened the Benders master problem by including the main-scenario assignments. The Benders cuts disaggregated per scenarios and commodities were effective to achieve convergence in few iterations, but they implied a rapid increase in the complexity of the master problem with the number of iterations. The addition of the main-scenario assignments to the master problem further reduced the total number of iterations in the Benders algorithm; the strengthened Benders algorithm solved the large-scale example in 8 iterations, whereas the implementation without main-scenario assignments required 15 iterations. For larger instances that might require more iterations, a partial disaggregation of cuts per scenario can be implemented to reduce the complexity of the master problem; the idea would be to select the scenarios with disaggregated cuts according to the impact that we expect them to have in the objective function.

The potential impact of scenarios on the optimal supply chain design was also considered to reduce the number of scenarios in large-scale problems. For scenarios describing disruptions at candidate DC locations, we selected relevant subsets of scenarios according to their probability and the number of simultaneous disruptions. In addition, we proposed a procedure to obtain deterministic bounds on the expected cost of scenarios that were excluded from the optimization model. The bounds were obtained based on a policy characterized by the main-scenario assignments. The upper bound is calculated based on the probability of having feasible main-scenario assignments in the excluded scenarios. The lower bound is calculated assuming that the main-scenario assignments are feasible in all excluded scenarios. The implementation of the bounding procedure allowed assessing the quality of the designs obtained for supply chains with a large number of DC candidate locations and scenarios. However, the proposed bounding procedure was implemented in the final step of the algorithm and does not have any influence in the supply chain design. Including the upper bounding policy in the optimization model might lead to tighter deterministic bounds and supply chain designs with better performance in adverse scenarios.

The results obtained in the examples demonstrated the importance of building supply chain resilience from the design stage. Neglecting disruptions at DCs lead to centralized supply chains that are vulnerable to adverse scenarios. The model proposed for the design of resilient supply chains has shown to be effective at balancing the investment costs associated to the selection and capacity of DCs with the expected transportation and penalty costs in the scenarios with disruptions. An interesting addition to the model would be to include risk measures in the objective function with the purpose of representing different risk preferences of the supply chain planner.

7.1.2 Implementation of a novel cross-decomposition algorithm for two-stage stochastic programming

In Chapter 3, we presented a cross-decomposition scheme for two-stage stochastic programs with mixed-integer variables in the first stage and continuous variables in the second stage. The proposed method integrates ideas from Benders and Lagrangean decomposition to leverage their complementary strengths. The algorithm is significantly different from previously proposed cross-decomposition schemes [237, 111] that try to avoid Benders and Lagrangean master problems. Our implementation of cross-decomposition not only solves the Benders and Lagrangean master problems at every iteration, but it also strengthens them with cuts generated from both Benders and Lagrangean subproblems. Strengthening the master problems addresses the limitations responsible for slow convergence in the original decomposition methods: weak bounds from the Benders master problem, erratic multipliers update in Lagrangean decomposition, and the need for a heuristic to find feasible solutions in Lagrangean decomposition.

The Benders master problem in our cross-decomposition scheme provides lower bounds for the minimization problems and candidate first-stage solutions. We include two sets of cuts that are disaggregated by scenarios in the Benders master problem. The first set of cuts corresponds to the standard Benders cuts. The second set of cuts is derived from the Lagrangean subproblems; they guarantee that the bounds obtain in the Benders master problem are at least as good as the best known Lagrangean bound. The Lagrangean master problem is responsible for the update of the Lagrange multipliers. It also includes disaggregated cuts generated from the Benders and Lagrangean subproblems. The Lagrangean cuts correspond to the standard cuts used in the cutting planes method for multipliers update [35, 126]; the Benders cuts ensure that the Lagrangean master problem is bounded and its objective value is at least as tight as the incumbent solution.

The convergence of our cross-decomposition algorithm relies on the properties of Benders decomposition. However, the computational experiments presented in Section 3.4 have shown a significant improvement in the solution time over multi-cut Benders decomposition; the improvement is noticeable in models with both strong and weak LP relaxations, as demonstrated by the two formulations of the resilient supply chain design (RSCD) model. The multi-cut implementation of Benders decomposition was only capable of solving the smallest instances with 639 scenarios of the tightened RSCD formulation within the specified time limit, whereas the cross-decomposition algorithm solved all instances with 639 and 1,025 scenarios. The comparative performance of cross-decomposition is especially good in large-scale problems. Those instances that cannot be solved directly with commercial solvers because of their size seem to be the most appropriate for cross-decomposition; in our experiments, RSCD instances with 1,587 scenarios could only be solved through cross-decomposition.

The cross-decomposition algorithm has the additional advantage of being well suited for parallelization. We have reduced the CPU time required for the Benders and Lagrangean subproblems by solving them simultaneously in multiple threads. This framework offers interesting opportunities given the expansion of grid computer infrastructure. Therefore, we expect growing attention to solution methods like cross-decomposition that exploit the decomposable structure of optimization problems and leverage parallelization.

7.1.3 Optimizing inventory policies in process networks under uncertainty

In Chapter 4, we developed a framework for inventory planning in process networks based on logic-based optimization models. The proposed stochastic inventory planning model includes decision rules that are established by inventory policies; the model is used to find the optimal parameters of these policies. Our planning model is inspired by the simplicity and effectiveness of popular inventory management strategies, but we have implemented significant adjustments to address the challenges posed by the complexity of chemical process networks. The proposed inventory planning framework explicitly addresses the inventory management of intermediate products, which has the potential to increase capacity utilization and avoid the formation of bottlenecks.

The formulations proposed for inventory planning in process networks are based on stochastic programming models. The scenario-based approach allows representing any type of uncertainty for which discrete-time forecasts can be generated. This flexibility has remarkable value for industrial applications given the large availability of historical data and the complex probabilistic descriptions of uncertain events.

In problems with a large number of potential scenarios, the standard approach to integrate forecasts of uncertainty in a stochastic programming model usually leads to a two-stage approximation of the multiperiod problem, since forecasts are unlikely to share indistinguishable trajectories. The two-stage approximation of a multiperiod problem often yields suboptimal decisions because it does not consider the sequence in which uncertainty is revealed after the first time period. In contrast, our logic-based model preserves the non-anticipativity condition even on sets of indistinguishable scenarios because the decisions are based on rules that apply across all scenarios.

We proposed basestock policies for inventory planning in process networks with arrangements of inventories in parallel and in series. The policy for inventories in parallel is based on establishing replenishment priorities among the inventories competing for shared upstream resources. The optimization model yields the optimal order of priorities and the optimal basestock levels for the inventories in the parallel arrangement. The policy for inventories in series is based on the theory of multi-echelon basestock policies. According to multi-echelon basestock policies, the replenishment decision for an inventory depends not only on its own level, but also on the inventory available in the downstream storage units. The optimization model yields the optimal basestock levels for each echelon in the arrangements of inventories in series.

We developed an LP model for stochastic inventory planning in general process networks, and included the logic establishing the inventory management strategies through Generalized Disjunctive Programming (GDP). The resulting formulation can represent process networks of arbitrary topologies. We also proposed MILP reformulations for the logic that models basestock policies in arrangements of inventories in parallel and in series.

The inventory planning decisions obtained from the logic-based model were compared with the decisions obtained from the equivalent two-stage stochastic programming model. We proposed two methodologies to assess the quality of the solutions: the *Residual Expected Value* (REV) and closed-loop Monte Carlo simulations. In the illustrative example with randomly sampled scenarios presented in Section 4.7, the logic-based model yielded solutions with lower expected cost than the two-stage model for the whole range of sample sizes; the reductions obtained in REV by implementing the logic-based decisions were around 2.5%. Interestingly, the logic-based model yielded solutions with a slightly lower expected cost than the multistage model for small sample sizes. In the examples with arrangements of inventories in parallel and in series, the logic-based model also yielded solutions with lower expected costs. These examples were evaluated through closed-loop Monte Carlo simulations to mimic the implementation of the decision-making strategies in an industrial environment. In comparison to the two-stage model, the logic-based model yielded reductions in the expected cost of 2.7% for the example with inventories in parallel and 7.0% for the example with inventories in series. More importantly, the decisions obtained from the logic-based model outperform those obtained from the two-stage model in the majority of simulations.

One of the topics that we have not addressed for the implementation of inventory policies in process networks is their stability in an infinite horizon. The stability of the policies in our examples can be demonstrated from the stability of a zero basestock policy; process networks that require no inventories to be stable are also stable under any basestock policy. However, instability appears in systems that accumulate increasing backorders because the expected throughput of the process network is less than the expected demands. It would be interesting to conduct further analysis on the specific conditions required for stability of inventory policies in process networks.

The policies developed for inventory planning in process networks with arrangements of inventories in parallel and in series can be extended to networks with arbitrary topologies. The proposed methodology has been designed to coordinate inventory management in complex process networks and to deal with diverse sources of uncertainty. The GDP model offers alternative reformulations and allows efficient solution methods that should be explored for inventory planning in large-scale process networks.

7.1.4 Bilevel optimization for capacity planning with rational markets

In Chapter 5, we developed a mixed-integer linear bilevel optimization model for capacity planning that considers markets as rational decision-makers. The bilevel formulation models the behavior of two independent decision-makers: a company planning the expansion of its production capacity and consumers that are allowed to select their providers from a pool of different producers. Discrete investment decisions are controlled by the upper level with the purpose of maximizing its Net

Present Value (NPV) during a finite time horizon. The lower level is an LP that minimizes the cost paid by the markets. The bilevel formulation models a Stackelberg competition in which the upper level decides its capacity expansion plan, and then the lower level decides the demand assignments.

The bilevel model is a challenging mathematical program with a nonconvex feasible region. The standard approach to address these models is to reformulate them as a single-level optimization problem. We implemented two alternative reformulations that characterize the same bilevel feasible region. The first reformulation replaces the lower-level LP with the KKT conditions guaranteeing lower-level optimality; the reformulation is linear except for the complementarity conditions that require the addition of new binary variables and constraints to obtain an MILP model. The second reformulation leverages the strong-duality property of the lower-level LP to replace it with constraints enforcing optimality of the lower-level solutions. The duality-based reformulation does not require the addition of binary variables, but it includes bilinear terms in the constraint enforcing strong duality. However, the bilinear terms are the product of upper-level discrete variables and lower-level dual (continuous) variables; therefore, they can be linearized exactly with the addition of linear constraints.

The middle-size example of the bilevel capacity planning problem showed a decisive advantage of the duality-based reformulation over the KKT reformulation; the duality-based reformulation required less than one second to solve each of the middle-size instances, whereas the KKT reformulation took over 150 s to solve each of them. The number of binary variables needed to reformulate the complementary constraints is responsible for the increase in computational complexity. However, we cannot determine from our results which reformulation would offer better performance in problems with continuous upper-level variables, since the KKT reformulation would remain unchanged and the duality-based reformulation would include bilinear terms that cannot be linearized.

Despite the good performance of the duality-based reformulation in the middle-size example, the solution of large-scale instances was still challenging. Therefore, we proposed two strategies that improved the solution times of the duality-based reformulation: disaggregated strong-duality constraints and domain reduction for the demand assignment variables. The major impact from these strategies derives from the number of continuous variables and constraints that can be excluded from the reformulation after identifying assignments that always have positive reduced cost in the lower-level problem. Not only does it allow reducing the size of the model, but it also produces a significant improvement in the LP relaxation of the reformulation by excluding many assignments that would be beneficial for the upper level but are not bilevel feasible; in the industrial instances studied in Section 5.10, the optimality gap of the LP relaxations improved from around 35% to less than 4%.

The comparison of the expansion plans obtained from the bilevel formulation with the plans obtained from the traditional single-level formulation shows the importance of considering potential customers as rational decision-makers. The formulations modeling captive markets tend to yield aggressive investment plans because they overestimate the market share that can be obtained. These investments can be very expensive and ultimately unnecessary; in the illustrative example presented in Section 5.7, the captive market model only yielded 41% of the NPV obtained with the bilevel expansion plan. The proposed bilevel formulation represents an important contribution for the development of investment plans that can be applied beyond the capacity planning problem. The model could be enhanced by relaxing the perfect information assumption, allowing capacity expansions in the competing producers, and including uncertainty.

7.1.5 Capacity planning with competitive decision-makers

In Chapter 6, we proposed an important enhancement to the model presented in Chapter 5. We extended the bilevel formulation for capacity planning to model expansions of the competing producers in a trilevel formulation. The trilevel optimization problem models the behavior of three rational decision-makers: the main company planning the expansion of its production capacity, the competition also planning the expansion of its capacity, and the consumers. The hierarchical optimization model establishes the sequence in which decisions are made. The upper-level decision-maker decides the expansion plan of the main company, the second-level controls the expansion plan of the competitors, and the third level decides demand assignments. The first and second level only control discrete variables, whereas the third level is an LP.

The formal definition of a trilevel optimal solution requires characterizing the behavior of the decision-makers in the case of degenerate solutions. This topic had been addressed in bilevel optimization but not for problems with more than two levels; we extended the definition of *optimistic solutions* for trilevel optimization problems. The behavior of the second- and third-level decision-makers with respect to degeneracy might give rise to several types of solutions in the trilevel problem. We identified and characterized mathematically three of them: *hierarchically optimistic* solutions, *sequentially optimistic* solutions, and *strategically optimistic* solutions. Other models to resolve degenerancy in trilevel optimization models can be easily extended from our definitions.

The first step for the solution of the trilevel capacity planning problem was to reformulate it as a bilevel optimization model. We obtained the bilevel reformulation using the duality-based approach. The resulting model is a linear bilevel program with discrete variables in the upper level and mixed-integer variables in the lower level. Few methods have been developed to solve this type of problems; we proposed two different algorithms to solve the bilevel reformulation. The first algorithm is a constraint-directed exploration of the *stability regions* of the bilevel reformulation; it is based on the observation that many different first-level solutions produce the same optimal response of the lower levels. We showed that different variants of the constraint-directed exploration algorithm can be used to obtain *sequentially optimistic* and *hierarchically optimistic* solutions. The second algorithm is a decomposition algorithm based on column-and-constraint generation; it iteratively improves the *high-point* relaxation of the bilevel reformulation by including the lower-level solutions that have been already observed. We showed that the column-and-constraint generation algorithm yields *strategically optimistic* solutions of the trilevel model.

We compared the performance of the algorithms in two instances of the trilevel capacity planning problem. For direct comparison of the algorithms, the instances presented in Section 6.9 were designed such that the *hierarchically* and *strategically optimistic* solutions coincide. In the first instance, both algorithms require the same number of iterations but the constraint-directed exploration takes 15 s, whereas the column-and-constraint generation algorithm takes 26 s. In the second instance, the constraint-directed exploration requires 7 iterations and the column-and-constraint generation algorithm only 2 iterations; the total solution times were 8 s and 46 s, respectively. Our analysis showed that instances requiring a similar number of iterations are likely to be solved faster with the constraint-directed exploration algorithm. The column-and-constraint generation algorithm requires fewer iterations to solve the instances, but it might take a longer time because the complexity of its iterations increases at a higher rate.

The second instance of the capacity planning problem with competitive decision-makers illustrates the value of considering the competitors as rational decision-makers. The same instance was solved in Chapter 5 with competitors that were not allowed to expand. The results obtained from the bilevel and the trilevel models are different and their interpretation is rather involved. The expansion strategy of the competitors obtained from both models implies no expansion; however, the bilevel model prescribes three expansions for the main company instead of the two expansions prescribed by the trilevel model. Our analysis showed that the trilevel optimal strategy only includes two expansions because a third expansion would trigger an expansion in the competitors that would be detrimental; the bilevel optimal expansion plan yields an NPV that is 2.5% lower than the trilevel expansion plan when the competitors are allowed to expand. We have found this type of counter-intuitive results to be common in trilevel optimization problems. Therefore, hierarchical optimization models are a very valuable tool to develop investment plans that involve several decision-makers.

7.2 Contributions of this thesis

The main contributions of the thesis to the research community are summarized below.

1. We proposed a novel two-stage stochastic programming model for the design of resilient supply chains with risk of disruptions at DCs. The model included capacity of the distribution centers as design decisions, which allowed considering inventory management as a key element to build supply chain resilience from the design stage.
2. We developed a tailor-made solution method for the design of large-scale resilient supply chains. The method includes tightening the formulation, a multi-cut implementation of Benders decomposition, pareto-optimal cuts, a strategy to select subsets of relevant scenarios, and a procedure to obtain deterministic bounds on the expected cost in problems with a large number of scenarios.
3. We presented a novel cross-decomposition algorithm for two-stage stochastic programming investment planning problems and implemented it for the design of resilient supply chains. We provided evidence of the superior performance of cross-decomposition for the solution large-scale problems, in comparison to the direct solution with commercial solvers and Benders decomposition.
4. We presented a new model for inventory planning in process networks under uncertainty based on the implementation of inventory policies. We proposed policies for arrangements of inventories in parallel and in series; we developed the logic that models inventory management rules; and we presented MILP reformulations. The proposed logic-based stochastic programing models offer better planning decisions when compared to two-stage stochastic programming models because they use policies as an alternative to avoid anticipativity in

multiperiod problems.

5. We proposed a new metric to assess the quality of decision-making strategies in multiperiod problems with uncertainty that we designated as the *Residual Expected Value* (REV). We also developed a method based on closed-loop Monte Carlo simulations to assess the quality of decision-making strategies in instances with a large number of scenarios.
6. We presented a novel model for capacity planning that considers potential customers as rational decision-makers. We demonstrated the value of considering customers to be rational for capacity planning in comparison to the traditional models that assume captive markets.
7. We presented two reformulation techniques for bilevel problems with lower-level LPs. Our implementations demonstrated the advantages of the duality-based reformulation over the KKT reformulation in problems with purely discrete upper-level variables.
8. We proposed a domain reduction strategy for the bilevel capacity planning formulation that allowed the solution of large-scale problems. This domain reduction strategy has the potential to lead the development of techniques that allow an improved description of the inducible region in bilevel programs.
9. We developed a trilevel capacity planning model that considers all producers and consumers as rational decision-makers. The analysis of the implementation revealed the complex interactions that take place in optimization problems with multiple decision-makers.
10. We addressed the issue of degeneracy in trilevel optimization problems for the first time. We presented definitions characterizing different types of optimal solutions that can be obtained in the presence of degeneracy in trilevel models.
11. We developed and implemented two algorithms for the trilevel capacity planning problem. The algorithms allowed solving a medium-size instance of the capacity planning problem with competitive decision-makers.

7.3 Publications related to this thesis

7.3.1 Full-length articles

- P. Garcia-Herreros, J.M. Wassick, & I.E. Grossmann. Design of resilient supply chains with risk of facility disruptions. *Industrial & Engineering Chemistry Research*, 53:17240–17251, 2014.
- S. Mitra, P. Garcia-Herreros, & I.E. Grossmann. An Enhanced Cross-Decomposition Scheme with Primal-Dual Multi-cuts for Two-Stage Stochastic Programming Investment Planning Problems. *Submitted to Mathematical Programming*, 2015.
- P. Garcia-Herreros, A. Agarwal, J.M. Wassick, & I.E. Grossmann. Optimizing inventory policies in process networks under uncertainty. *To be submitted to Computers & Chemical Engineering*, 2016.
- P. Garcia-Herreros, L. Zhang, P. Misra, E. Arslan, & I.E. Grossmann. Mixed-integer bilevel optimization for capacity planning with rational markets. *Submitted for publication in Computers & Chemical Engineering*, 2015.
- C. Florensa Campo, P. Garcia-Herreros, P. Misra, E. Arslan, S. Mehta, & I.E. Grossmann. Capacity planning with competitive decision-makers: Trilevel MILP formulation and solution approaches. *To be submitted to European Journal of Operations Research*, 2016.

7.3.2 Review articles

- I.E. Grossmann, B.A. Calfa, & P. Garcia-Herreros. Evolution of concepts and models for quantifying resiliency and flexibility of chemical processes. *Computers & Chemical Engineering*, 70:22–34, 2014.
- I.E. Grossmann, R.M. Apap, B.A. Calfa, P. Garcia-Herreros, & Q. Zhang. Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty. *Computer Aided Chemical Engineering*, 37:1–14, 2015.

7.3.3 Conference articles

- P. Garcia-Herreros, I.E. Grossmann, & J.M. Wassick. Design of supply chains under the risk of facility disruptions. *Computer Aided Chemical Engineering*, 32:577–582, 2013.
- S. Mitra, P. Garcia-Herreros, & I.E. Grossmann. A novel cross-decomposition multi-cut scheme for two-stage stochastic programming. *Computer Aided Chemical Engineering*, 33: 241–246, 2014.
- P. Garcia-Herreros & I.E. Grossmann. Stochastic Programming for Supply Chains Resilience [in Spanish]. *XXVII Congreso Interamericano y Colombiano de Ingenieria Quimica*, pages 1309–1314, 2014.
- P. Garcia-Herreros, I.E. Grossmann, B. Sharda, A. Agarwal, & J.M. Wassick. Empirical study of the behavior of capacitated production-inventory systems. In *Proceedings of the 2014 Winter Simulation Conference*, WSC '14, pages 2251–2260. IEEE Press, 2014.
- P. Garcia-Herreros, P. Misra, E. Arslan, S. Mehta, & I.E. Grossmann. A duality-based approach for bilevel optimization of capacity expansion. *Computer Aided Chemical Engineering*, 37:2021–2026, 2015.
- A. Kandiraju, P. Garcia-Herreros, P. Misra, E. Arslan, S. Mehta, & I.E. Grossmann. Capacity planning for the air separation industry with rational markets and demand uncertainty. *Submitted to Computer Aided Chemical Engineering*, 2016.

7.4 Directions for future research

In the following subsections we describe the directions in which our research can be further developed.

7.4.1 Include other sources of uncertainty in the resilient supply chain design

The model proposed for the design of resilient supply chains in Chapter 2 considers disruptions at candidate locations for DCs as the only source of uncertainty. The model can be extended to include other sources of uncertainty. It would be interesting to analyze the impact of disruptions in arcs connecting plants to DCs and DCs to customers; disruptions in the arcs play an important role in supply chains with limited transportation modes, such as petrochemical networks connected by pipelines. A more comprehensive model for the design of resilient supply chains should also include uncertainty in the amount of supply available at the plants and uncertainty in customer demands. Given the large number of scenarios that these design problems are likely to have, we believe that Sample Average Approximation (SAA) [206, 130] would be an appropriate framework to address the resulting supply chain design problem.

7.4.2 Automate the implementation of cross-decomposition for two-stage stochastic programming investment planning problems

The cross-decomposition scheme presented in Chapter 3 can be implemented for a wide range of investment planning problems that can be formulated as two-stage stochastic programming models. The only restriction for cross-decomposition is that the model must be an MILP with only continuous variables in the second stage. Given the performance observed in our computational experiments, it would be interesting to develop an algorithm that automatically implements the cross-decomposition scheme from the full-space model of a two-stage stochastic programming investment planning problem. Such an implementation could leverage the recent advances in modeling languages such as PYOMO [103] or Julia [150] that offer high performance computing and a diverse set of programming tools.

7.4.3 Develop efficient solution methods for the optimization of inventory policies

The logic-based models developed in Chapter 4 for inventory planning in process networks are very promising because of the quality of their solutions and the intuitive appeal of inventory policies. However, the reformulation of the logic-based models as MILPs produce optimization problems with significant computational complexity. Fortunately, the original GDP formulations offer alternative MILP reformulations and the possibility of using logic-based solution methods [112, 231]. Additionally, the multiperiod inventory planning model might be a good candidate for the implementation of a solution method based on dynamic programming [54, 180]. More efficient solution methods will allow addressing instances representing larger process networks, increase the number of scenarios in the models, and increase the planning horizon. We believe that addressing more complex instances will reveal to a larger extent the benefits of using inventory polices in process networks.

7.4.4 Include uncertainty in the bilevel capacity planning model

The bilevel capacity planning model presented in Chapter 5 considers the markets as rational decision-makers, but assumes that all the information is deterministic. The model can be extended to include uncertainty in several ways. The deterministic parameters that have the largest influence in the optimal capacity plan are the capacity of competitors and market demands. A bilevel model that includes scenarios characterizing uncertain capacities at the comptetitor's plants and uncertain demands would be more realistic. The problem can be formulated as a bilevel-stochastic capacity planning model [50, 37, 2]; we expect the solution methods developed in Chapter 5 to be applicable for such problems. Another interesting direction would be to relax the perfect information assumption. This would imply that the company planning its capacity does not know exactly the decision criterion of the markets. Instead, the decision criterion of the markets could be modeled with a probabilistic description; such a model could also be formulated as a bilevel-stochastic program.

7.4.5 Include contracts, mergers, and acquisitions in the capacity planning models

The capacity planning models presented in Chapters 5 and 6 assume that the markets select their providers according to their cost minimization criterion, and they are free to switch providers any time. However, industrial producers and consumers often establish long term partnerships that allow them to share their interests and increase coordination; these interactions are regulated by contracts [5]. Contracts might play an important role for capacity expansion plans because they reduce the variability of future business conditions. Different types of contracts between producers and consumers can be included in the capacity planning models through Generalized Disjunctive Programming [174]. It would be interesting to analyze the benefits of considering contracts in capacity planning models and study their impact in the resulting expansion strategies. Furthermore, mergers and acquisitions among industrial producers can significantly change the market landscape. This type of decisions could also be modeled through Generalized Disjunctive Programming. Mathematical programming models representing potential mergers can be very useful for producers and for industry regulators to analyze the response of markets to different competitive environments.

7.4.6 Develop general domain reduction strategies for hierarchical optimization

The domain reduction strategy implemented in Chapter 5 was very successful to reduce the solution time of the reformulated bilevel capacity planning model. The integration of Constraint Programming and Mathematical Programming techniques has also shown to be effective to reduce the solution time of a wide range of problems [113]. Domain reduction strategies and other Constraint Programming techniques have great potential to improve the formulation of any hierarchical optimization model because the feasible (inducible) region in these problems is usually much smaller than the constraint region. Hierarchal optimization models with a reduced domain should require less time to be solved regardless of the solution method.

Appendix A

Data for the large-scale example of the resilient supply chain design problem

The independent disruption probabilities for candidate DCs are presented in Table A.1.

Candidate DC	Probability of disruption
1	0.026
2	0.100
3	0.030
4	0.018
5	0.063
6	0.090
7	0.072
8	0.031
9	0.046

Table A.1: Probability of disruption at DC candidate locations (P_i^0).

Table A.2 presents the cost coefficients for the objective function of the optimization model.

The unit transportation costs for commodity 1 from plant to DCs are given in Table A.3; the unit transportation costs for commodity 2 from plant to DCs (i) are given by Equation (A.1) as a func-

Parameter	**Value**	**Units**
N	365	periods
F_i	200,000	\$/DC
V_i	100	\$/ton
H_k	0.01	\$/(ton period)
$penalty$	25	\$/ton

Table A.2: Cost coefficients.

Parameter	**Value [\$/ton]**
$A_{1,1}$	0.298
$A_{1,2}$	0.340
$A_{1,3}$	0.264
$A_{1,4}$	0.109
$A_{1,5}$	0.312
$A_{1,6}$	0.333
$A_{1,7}$	0.270
$A_{1,8}$	0.289
$A_{1,9}$	0.286

Table A.3: Transportation cost for commodity 1 from plant to DCs.

tion of the transportation costs for commodity 1.

$$A_{i,2} = 1.15 A_{i,1} \tag{A.1}$$

Customer demands for both commodities are presented in Table A.4. The unit transportation costs for commodity 1 from DCs to customers are given in Table A.5; the unit transportation costs for commodity 2 from the DCs to customers can be calculated from Equation (A.2).

$$B_{i,j,2} = 1.15 B_{i,j,1} \tag{A.2}$$

Customer (j)	**Demand for commodity 1**	**Demand for commodity 2**
1	243	133
2	200	181
3	194	176
4	112	108
5	236	136
6	108	53
7	114	247
8	204	83
9	119	71
10	264	124
11	264	90
12	244	148
13	130	118
14	232	240
15	204	234
16	295	61
17	230	198
18	260	104
19	191	135
20	186	160
21	265	239
22	117	134
23	127	247
24	135	110
25	178	190
26	266	183
27	261	158
28	112	190
29	180	183
30	205	86

Table A.4: Customer demands ($D_{j,1}$ and $D_{j,2}$) for commodities.

Customer (j)					**DC** (i)				
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
1	1.481	2.139	0.928	2.265	0.580	1.192	0.615	0.743	2.559
2	2.237	1.293	2.876	0.586	1.597	1.454	2.414	2.488	0.967
3	1.724	1.614	2.116	2.273	2.387	1.190	2.199	2.138	0.907
4	0.797	1.746	2.899	1.351	1.963	1.060	2.378	1.138	1.765
5	2.248	2.727	2.898	1.868	0.847	0.873	1.144	2.602	1.136
6	2.536	1.109	2.823	1.375	0.991	1.128	2.040	1.683	1.379
7	2.577	1.963	1.874	2.793	1.215	2.393	2.384	1.451	1.920
8	0.690	0.635	1.827	2.448	2.835	0.825	1.922	1.673	0.530
9	1.343	0.905	2.486	1.278	1.821	0.914	2.005	1.157	2.135
10	2.223	2.370	1.626	0.710	1.072	2.783	0.881	2.565	1.846
11	2.990	0.695	1.607	0.767	2.905	0.512	2.437	2.543	2.672
12	0.711	1.499	1.150	2.500	1.579	2.777	0.955	1.160	0.864
13	0.840	2.673	1.949	1.875	0.862	2.633	2.055	1.377	1.783
14	1.505	0.690	1.100	0.808	0.960	1.100	1.543	0.624	2.757
15	2.862	1.727	1.723	1.344	2.750	1.423	0.778	2.451	1.474
16	1.104	1.510	0.741	0.830	2.855	2.890	1.938	0.649	1.087
17	1.383	2.553	0.539	0.608	0.922	2.123	2.329	2.119	1.627
18	1.868	1.241	2.362	0.972	2.217	0.959	1.421	2.064	2.451
19	0.703	2.823	2.439	1.717	1.590	1.617	1.266	1.771	1.777
20	2.544	2.487	2.111	1.447	2.529	1.832	1.377	2.848	2.690
21	1.875	2.056	1.968	1.019	1.253	1.677	1.076	2.611	0.987
22	1.065	0.927	1.069	1.589	1.278	2.808	1.576	0.962	2.762
23	2.949	1.597	0.778	1.145	1.522	1.987	1.156	2.007	2.278
24	1.054	0.794	1.242	1.297	1.560	1.770	0.714	1.156	2.503
25	0.573	2.822	2.326	1.722	1.946	1.093	1.647	2.908	1.867
26	1.803	1.079	1.722	2.060	2.198	1.489	1.419	2.970	0.594
27	2.713	2.783	2.490	0.747	1.155	1.338	2.199	0.841	2.303
28	0.767	2.134	1.735	2.448	2.288	2.759	2.727	1.335	2.247
29	0.995	0.576	2.360	1.750	1.700	2.762	2.025	2.044	2.649
30	2.514	1.942	0.957	1.100	2.716	0.572	1.725	0.920	2.947

Table A.5: Transportation cost for commodity 1 from DCs to customers ($B_{i,j,1}$).

Appendix B

Data for the random instances of the resilient supply chain design problem

The data for the instances is taken from Daskin [48]. The original problem considers 49 US cities that simultaneously serve as demand sites and potential distribution centers (DCs). The formulation includes uncapacitated DCs with investment costs estimated from the real-state market. Variable costs associated with DC capacities have been added at a rate of $\$0.0001$ per unit of product. Originally, demands for a single commodity are assumed to be proportional to the state populations in 1990, and transportation costs are proportional to the great-circle distance between locations. The original demands and transportation cost are used to generate the first instance; all other instances are generated randomly by sampling demands and transportation costs from uniform distributions bounded between 80% and 120% of the original values.

Given the very large number of possible scenarios (2^{49}), we have selected subsets of 10, 11, and 12 candidate locations for DCs in the different instances of the problem. The locations included in the smallest instance are: Sacramento (CA), Albany (NY), Austin (TX), Tallahassee (FL), Harrisburg (PA), Springfield (IL), Columbus (OH), Montgomery (AL), Salem (OR), and Des Moines (IA). The additional location included in the instance with 11 candidate DCs is Lansing (MI). The largest instance with 12 candidate DCs also includes Trenton (NJ).

Given the very small probability of scenarios with more than 5 simultaneous disruptions, they have been grouped into a single scenario in which all demands are penalized. The effect of this approximation is limited by the magnitude of the corresponding probabilities ($< 1.2 * 10^{-5}$ in all cases). Furthermore, the approximation improves the numerical stability of the algorithm. Despite the reduction in size, the problem still implies minimizing the cost over large sets of scenarios. The failure probability of each DC has been left to the value originally used by Daskin [48]: $q = 0.05$.

Appendix C

Data for the illustrative example of the bilevel capacity planning problem

Table C.1 shows the cardinality of the datasets used in the three examples presented in Chapter 5.

	Illustrative example	**Middle-size instance**	**Industrial instance**
Existing plants of the leader:	2	3	3
Candidate plants of the leader:	1	1	2
Facilities of the competitors:	1	3	5
Markets:	8	15	20
Commodities:	1	1	2
Time periods:	12	20	80

Table C.1: Summary of datasets used in capacity planning examples.

The complete dataset for the illustrative example is presented in Tables C.2 - C.10.

The initial production capacity of the plants is presented in Table C.2.

Facility	**Commodity 1**
	[ton/period]
Leader 1	22,500
Leader 2	36,000
Leader 3	0
Competitor 1	36,000

Table C.2: Initial capacity of plants.

Market demands for all time periods are presented in Table C.3.

Time	**Market demand** [ton/period]							
period	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
1	15,300	8,100	4,500	4,500	5,400	11,700	3,600	27,000
2	15,500	8,200	4,600	4,600	5,500	11,900	3,700	27,600
3	15,700	8,300	4,600	4,700	5,500	12,200	3,800	27,900
4	15,800	8,400	4,700	4,700	5,600	12,400	3,800	28,000
5	15,900	8,400	4,800	4,800	5,600	12,600	3,900	28,200
6	15,900	8,400	4,800	4,900	5,600	12,700	3,900	28,100
7	16,000	8,500	4,900	5,000	5,700	13,000	4,000	28,600
8	16,100	8,500	5,000	5,000	5,700	13,300	4,100	29,100
9	16,200	8,600	5,100	5,100	5,800	13,500	4,200	29,800
10	16,200	8,600	5,200	5,100	5,800	13,700	4,200	29,900
11	16,100	8,500	5,300	5,200	5,800	13,600	4,200	29,700
12	16,200	8,600	5,300	5,200	5,800	13,600	4,200	29,800

Table C.3: Market demands ($D_{t,j,k}$).

Tables C.4-C.10 present the cost coefficients for the objective function of the illustrative example. Table C.4 shows the cost ($A_{t,3}$) of opening the candidate production plant in different time periods. In the illustrative example, it is allowed to open the new plant only in time periods 1, 5, and 9.

Table C.5 presents the maintenance cost per time period ($B_{t,i}$) incurred by open plants.

Table C.6 presents the investment cost ($E_{t,i,1}$) associated to the expansion of production capacity by 9,000 ton/period. In the illustrative example, all plants are assumed to have the same expansion cost and expansions are allowed only in time periods 1, 5, and 9.

Time period	**Investment cost** [MM$]
1	20.00
5	20.40
9	20.86

Table C.4: Investment cost ($A_{t,3}$) of the leader to open plant 3.

Time	**Maintenance cost** [MM$/period]		
period	Leader 1	Leader 2	Leader 3
1	1.000	2.000	3.000
2	1.005	2.010	3.015
3	1.010	2.020	3.030
4	1.013	2.026	3.039
5	1.020	2.040	3.060
6	1.029	2.058	3.087
7	1.032	2.064	3.096
8	1.035	2.070	3.105
9	1.043	2.086	3.129
10	1.049	2.098	3.147
11	1.054	2.108	3.162
12	1.058	2.116	3.174

Table C.5: Maintenance cost ($B_{t,i}$).

Time	**Expansion cost** [MM$/9,000 ton]		
period	Leader 1	Leader 2	Leader 3
1	30.00	30.00	30.00
5	30.60	30.60	30.60
9	31.29	31.29	31.29

Table C.6: Expansion costs ($E_{t,i,1}$).

The production cost of plants ($F_{t,i,1}$) in the illustrative example are presented in Table C.7.

The transportation cost from plants to markets in each time period are calculated from the transportation costs at the initial time period and their growth rate, according to Equation (C.1). Initial transportation costs ($G^0_{i,j,k}$) are presented in Table C.8; their growth rate (G^{Rt}_t) are presented in Table C.10.

$$G_{t,i,j,k} = G^0_{i,j,k} G^{Rt}_t \tag{C.1}$$

Time	**Production cost** [$/ton]		
period	Leader 1	Leader 2	Leader 3
1	250	220	180
2	257	226	185
3	246	217	177
4	246	216	177
5	254	223	183
6	263	231	189
7	253	222	182
8	255	225	184
9	262	230	188
10	284	250	204
11	271	239	195
12	269	237	194

Table C.7: Production costs ($F_{t,i,1}$).

Market	**Transportation cost** [$/ton]		
	Leader 1	Leader 2	Leader 3
1	26	325	234
2	13	299	260
3	65	195	325
4	104	130	156
5	78	260	221
6	208	195	46
7	195	169	59
8	234	169	0.4

Table C.8: Initial transportation cost ($G_{t,i,j,1}$).

Selling prices offered by plants to markets are calculated from the selling prices at the initial time period and their growth rate according to Equation (C.2). Initial selling prices ($P^0_{i,j,k}$) are presented in Table C.9; their growth rates (P^{Rt}_t) are presented in Table C.10.

$$P_{t,i,j,k} = P^0_{i,j,k} P^{Rt}_t \tag{C.2}$$

Market	Leader 1, 2 & 3	Competitor 1
	[$/ton]	[$/ton]
1	586	615
2	573	726
3	625	785
4	664	633
5	638	794
6	606	619
7	619	606
8	560	580

Table C.9: Initial selling prices ($P^0_{i,j,k}$) from plants to markets.

Time period	Growth rate	Grow rate
	for transportation	for selling prices
1	1.00	1
2	1.00	1
3	1.03	1.001
4	1.05	1.002
5	1.09	1.013
6	1.09	1.013
7	1.12	1.015
8	1.12	1.015
9	1.12	1.047
10	1.14	1.048
11	1.14	1.048
12	1.16	1.049

Table C.10: Growth rates for transportation costs (G^{Rt}_t) and selling prices (P^{Rt}_t).

Appendix D

Capacity planning with rational markets: proof of Proposition 5.1

Proposition 5.1 A demand assignment ($y_{t,i,j,k}$) with positive reduced cost in the optimal solution of the lower-level problem with maximum capacity also has a positive reduced cost when capacities are reduced.

Proof.

We want to prove that the optimal reduced cost of the leader's assignment variables cannot decrease when capacities are reduced from their maximum feasible value ($C^U_{t,i,k}$). For this analysis, we decompose the lower-level problems by time periods ($t \in T$) and by commodities ($k \in K$); the problem minimizing the cost paid by markets is decomposable since all terms in the objective function and constraints are indexed by (t, k). Intuitively, this means that we can solve independent problems to minimize the cost paid at time period t for commodity k. The lower-level problem resulting from this decomposition is presented in Equations (D.1)-(D.5).

$$\min \quad \frac{1}{(1+R)^t} \sum_{i \in I} \sum_{j \in J} P_{i,j} y_{i,j} \tag{D.1}$$

$$\text{s.t.} \quad \sum_{j \in J} y_{i,j} \leq C_i \qquad [\mu_i] \qquad \forall\, i \in I^{\mathcal{L}} \tag{D.2}$$

$$\sum_{j \in J} y_{i,j} \leq C_i^0 \qquad [\mu_i] \qquad \forall\, i \in I^{\mathcal{C}} \tag{D.3}$$

$$\sum_{i \in I} y_{i,j} = D_j \qquad [\lambda_j] \qquad \forall\, j \in J \tag{D.4}$$

$$y_{i,j} \in \mathbb{R}^+ \qquad \forall\, i \in I, j \in J \tag{D.5}$$

Similarly, the dual lower-level problem disaggregated by time periods and commodities is presented in Equations (D.6)-(D.10).

$$\max \quad \sum_{j \in J} D_j \lambda_j - \sum_{i \in I^{\mathcal{L}}} C_i \mu_i - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i \tag{D.6}$$

$$\text{s.t.} \quad \lambda_j - \mu_i \leq \frac{1}{(1+R)^t} P_{i,j} \qquad \forall\, i \in I^{\mathcal{L}}, j \in J \tag{D.7}$$

$$\lambda_j - \mu_i \leq \frac{1}{(1+R)^t} P_{i,j} \qquad \forall\, i \in I^{\mathcal{C}}, j \in J \tag{D.8}$$

$$\mu_i \in \mathbb{R}^+ \qquad \forall\, i \in I \tag{D.9}$$

$$\lambda_j \in \mathbb{R} \qquad \forall\, j \in J \tag{D.10}$$

We assume that the dual lower-level problem is bounded (and the primal lower-level problem is feasible). The condition that guarantees a finite solution for the dual of the lower-level problem is presented in Equation (D.11).

$$\sum_{j \in J} D_j \leq \sum_{i \in I^{\mathcal{L}}} C_i + \sum_{i \in I^{\mathcal{L}}} C_i^0 \tag{D.11}$$

An important observation regarding dual variables μ_i $(i \in I^{\mathcal{L}})$ is that they all have the same optimal value. It is the case because constraints (D.7) are identical for all plants of the leader (plants of the leader offer the same price to each market) and the coefficients of all μ_i have the same sign in the objective function. We also note that the condition presented in Equation (D.12) must be satisfied by the optimal solution of the dual lower-level problem in order to obtain the largest values of λ_j allowed by dual constraints (D.7)-(D.8).

$$\lambda_j = \min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i \right) \qquad \forall\, j \in J \tag{D.12}$$

Using Equation (D.12), we can rewrite the dual lower-level problem (D.6)-(D.10) as in Equation

(D.13).

$$\max_{\mu_i \geq 0} \left\{ \sum_{j \in J} D_j \left[\min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i \right) \right] - \sum_{i \in I^{\mathcal{L}}} C_i \mu_i - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i \right\} \tag{D.13}$$

In order to prove that the optimal reduced costs of the leader's assignment variables cannot decrease when capacities are reduced, we divide the proof in four steps.

Step 1: optimal values of μ_i $(i \in I^{\mathcal{L}})$ cannot be less than their optimal values obtained with maximum capacity.

We assume that C_i^U is the upper bound of the coefficient of dual variable μ_i in Equation (D.6), and we denote by (μ_i^U, λ_j^U) the corresponding optimal solution of the dual lower-level problem. Now, let us assume that the coefficients of μ_i are reduced by ΔC_i, and let us denote by $(\mu_i^\Delta, \lambda_j^\Delta)$ the optimal dual solution corresponding to capacities $C_i^\Delta = C_i^U - \Delta C_i$. If we consider that Equation (D.13) is a maximization problem, we can establish the sequence of inequalities (D.14)-(D.17).

$$\sum_{j \in J} D_j \left[\min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^\Delta \right) \right] - \sum_{i \in I^{\mathcal{L}}} C_i^U \mu_i^\Delta - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i^\Delta \tag{D.14}$$

$$\leq \sum_{j \in J} D_j \left[\min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U \right) \right] - \sum_{i \in I^{\mathcal{L}}} C_i^U \mu_i^U - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i^U \tag{D.15}$$

$$\leq \sum_{j \in J} D_j \left[\min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U \right) \right] - \sum_{i \in I^{\mathcal{L}}} \left(C_i^U - \Delta C_i \right) \mu_i^U - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i^U \tag{D.16}$$

$$\leq \sum_{j \in J} D_j \left[\min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^\Delta \right) \right] - \sum_{i \in I^{\mathcal{L}}} \left(C_i^U - \Delta C_i \right) \mu_i^\Delta - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i^\Delta \tag{D.17}$$

where (D.14) is less than (D.15) because μ_i^U is the optimal solution in the maximization problem with capacity C_i^U; (D.15) is less than (D.16) because of its additional term $\sum_{i \in I^{\mathcal{L}}} \Delta C_i \mu_i^U$; and (D.16) is less than (D.17) because μ_i^Δ is the optimal solution in the maximization problem with capacity $C_i^U - \Delta C_i$.

We note that $\sum_{i \in I^{\mathcal{L}}} \Delta C_i \mu_i^\Delta$ is the difference between expressions (D.17) and (D.14). Similarly, the difference between expressions (D.16) and (D.15) is $\sum_{i\in I^{\mathcal{L}}} \Delta C_i \mu_i^U$. Hence, we can infer inequality (D.18).

$$\sum_{i\in I^{\mathcal{L}}} \Delta C_i \mu_i^{\Delta} \geq \sum_{i\in I^{\mathcal{L}}} \Delta C_i \mu_i^U \tag{D.18}$$

Since dual variables μ_i have the same optimal value for all $i \in I^{\mathcal{L}}$, then $\mu_i^{\Delta} \geq \mu_i^U$ for all $i \in I^{\mathcal{L}}$.

Step 2: optimal values of μ_i $(i \in I^{\mathcal{C}})$ cannot be less than their optimal values obtained with maximum capacity.

In order to continue with the argument, let us define ϵ_i according to Equation (D.19).

$$\epsilon_i = \mu_i^{\Delta} - \mu_i^U \tag{D.19}$$

Beforehand, we know that $\epsilon_i \geq -\mu_i^U$ because any feasible μ_i must be nonnegative. By optimality of Equation (D.15), we also know that any deviation of μ_i^U from their optimal values yields a lower bound as presented in Equations (D.20)-(D.21).

$$\sum_{j\in J} D_j \left[\min_{i\in I}\left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U + \min_{i'}[\epsilon_{i'}]\right)\right] - \sum_{i\in I^{\mathcal{L}}} C_i^U(\mu_i^U + \min_{i'}[\epsilon_{i'}]) - \sum_{i\in I^{\mathcal{C}}} C_i^0(\mu_i^U + \min_{i'}[\epsilon_{i'}]) \tag{D.20}$$

$$\leq \sum_{j\in J} D_j \left[\min_{i\in I}\left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U\right)\right] - \sum_{i\in I^{\mathcal{L}}} C_i^U \mu_i^U - \sum_{i\in I^{\mathcal{C}}} C_i^0 \mu_i^U \tag{D.21}$$

subtracting (D.21) from (D.20), we obtain inequality (D.22),

$$-\sum_{j\in J} D_j \min_{i'}[\epsilon_{i'}] + \sum_{i\in I^{\mathcal{L}}} C_i^U \min_{i'}[\epsilon_{i'}] + \sum_{i\in I^{\mathcal{C}}} C_i^0 \min_{i'}[\epsilon_{i'}] \geq 0 \tag{D.22}$$

which implies $\min_{i\in I}[\epsilon_i] \geq 0$ according to inequality (D.11).

Step 3: if capacities of the leader are reduced, optimal values of μ_i $(i \in I^{\mathcal{C}})$ cannot increase faster than the values of μ_i $(i \in I^{\mathcal{L}})$.

We want to show that $\max_{i \in I}[\epsilon_i] = \max_{i \in I^{\mathcal{L}}}[\epsilon_i]$. Since all dual variables μ_i have the same optimal value for all $i \in I^{\mathcal{L}}$, we denote by μ_1^U their optimal value in the problem with maximum capacity and by ϵ_1 their optimal deviation when capacities of the leader are reduced by ΔC_i.

By optimality of Equation (D.15), we can deduce inequality (D.23).

$$\begin{aligned}
&\sum_{j \in J} D_j \left\{ \min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U + \epsilon_i - \epsilon_1 \right) \right\} \\
&\qquad - \sum_{i \in I^{\mathcal{L}}} C_i^U \left(\mu_i^U + \epsilon_i - \epsilon_1 \right) - \sum_{i \in I^{\mathcal{C}}} C_i^0 \left(\mu_i^U + \epsilon_i - \epsilon_1 \right) \\
\leq &\sum_{j \in J} D_j \left\{ \min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U \right) \right\} \\
&\qquad - \sum_{i \in I^{\mathcal{L}}} C_i^U \mu_i^U - \sum_{i \in I^{\mathcal{C}}} C_i^0 \mu_i^U
\end{aligned} \tag{D.23}$$

which implies inequality (D.24),

$$\begin{aligned}
&\sum_{j \in J} D_j \left\{ \min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U + \epsilon_i \right) \right\} \\
&\qquad - \sum_{i \in I^{\mathcal{L}}} C_i^U \left(\mu_i^U + \epsilon_i \right) - \sum_{i \in I^{\mathcal{C}}} C_i^0 \left(\mu_i^U + \epsilon_i \right) \\
\leq &\sum_{j \in J} D_j \left\{ \min_{i \in I} \left(\frac{1}{(1+R)^t} P_{i,j} + \mu_i^U + \epsilon_1 \right) \right\} \\
&\qquad - \sum_{i \in I^{\mathcal{L}}} C_i^U \left(\mu_i^U + \epsilon_1 \right) - \sum_{i \in I^{\mathcal{C}}} C_i^0 \left(\mu_i^U + \epsilon_1 \right)
\end{aligned} \tag{D.24}$$

By optimality, we also know that inequality (D.25) must be satisfied.

$$\begin{aligned}
&\sum_{j\in J} D_j \left\{\min_{i\in I}\left(\frac{1}{(1+R)^t}P_{i,j} + \mu_i^U + \epsilon_1\right)\right\} \\
&\qquad\qquad - \sum_{i\in I^{\mathcal{L}}}\left(C_i^U - \Delta C_i\right)\left(\mu_i^U + \epsilon_1\right) - \sum_{i\in I^{\mathcal{C}}} C_i^0\left(\mu_i^U + \epsilon_1\right) \\
&\leq \sum_{j\in J} D_j \left\{\min_{i\in I}\left(\frac{1}{(1+R)^t}P_{i,j} + \mu_i^U + \epsilon_i\right)\right\} \\
&\qquad\qquad - \sum_{i\in I^{\mathcal{L}}}\left(C_i^U - \Delta C_i\right)\left(\mu_i^U + \epsilon_i\right) - \sum_{i\in I^{\mathcal{L}}} C_i^0\left(\mu_i^U + \epsilon_i\right)
\end{aligned} \tag{D.25}$$

Furthermore, an upper bound on the right-hand side of inequality (D.25) is given by Equation (D.26).

$$\begin{aligned}
&\sum_{j\in J} D_j \left\{\min_{i\in I}\left(\frac{1}{(1+R)^t}P_{i,j} + \mu_i^U + \epsilon_i\right)\right\} \\
&\qquad\qquad - \sum_{i\in I^{\mathcal{L}}}\left(C_i^U - \Delta C_i\right)\left(\mu_i^U + \epsilon_i\right) - \sum_{i\in I^{\mathcal{C}}} C_i^0\left(\mu_i^U + \epsilon_i\right) \\
&\leq \sum_{j\in J} D_j \left\{\min_{i\in I}\left(\frac{1}{(1+R)^t}P_{i,j} + \mu_i^U\right) + \max_i[\epsilon_i]\right\} \\
&\qquad\qquad - \sum_{i\in I^{\mathcal{L}}}\left(C_i^U - \Delta C_i\right)\left(\mu_i^U + \epsilon_i\right) - \sum_{i\in I^{\mathcal{L}}} C_i^0\left(\mu_i^U + \epsilon_i\right)
\end{aligned} \tag{D.26}$$

If we subtract the left-hand side of (D.25) from the right-hand side of (D.26), we can infer inequality (D.27),

$$\sum_{j\in J} D_j \left\{\max_i[\epsilon_i] - \epsilon_1\right\} - \sum_{i\in I^{\mathcal{C}}} C_i^0(\epsilon_i - \epsilon_1) \geq 0 \tag{D.27}$$

Now, let us assume that $\max_i[\epsilon_i] > \epsilon_1$. Then, for $i' = \arg\max[\epsilon_i]$, inequality (D.28) must be satisfied.

$$C_{i'}^0 \leq \frac{\sum_{j\in J} D_j \left\{\max_i[\epsilon_i] - \epsilon_1\right\} - \sum_{i\in I^{\mathcal{C}}\setminus\{i'\}} C_i^0\left(\epsilon_i - \epsilon_1\right)}{\left(\epsilon_{i'} - \epsilon_1\right)} \tag{D.28}$$

But we have not imposed any restrictions on the capacity of the competitors. Therefore,

$\epsilon_1 = \max_{i \in I}[\epsilon_i]$

Step 4: reduced costs of assignment variables for the leader cannot decrease when its capacities are reduced.

A necessary condition for the optimality of a minimization linear program is that the reduced cost of the nonbasic variables must be nonnegative. Therefore, optimal demand assignments to the leader that are nonbasic ($y^U_{i,j} = 0 \quad i \in I^{\mathcal{L}}$) in the problem with maximum capacity must have nonnegative reduced costs as indicated in inequality (D.29).

$$r^U_{i,j} = \frac{1}{(1+R)^t}P_{i,j} + \mu^U_i - \lambda^U_j \geq 0 \qquad \forall\,(i,j) \in \left\{(i,j) :\, i \in I^{\mathcal{L}},\, j \in J,\, y^U_{i,j} = 0\right\} \tag{D.29}$$

Using Equation (D.12), we can rewrite the reduced cost ($r_{i,j}$) for nonbasic variables $y_{i,j}$ only in terms of dual variables μ_i,

$$r^U_{i,j} = \frac{1}{(1+R)^t}P_{i,j} + \mu^U_i - \min_{i' \in I}\left(\frac{1}{(1+R)^t}P_{i',j} + \mu^U_{i'}\right) \geq 0$$
$$\forall\,(i,j) \in \left\{(i,j) :\, i \in I^{\mathcal{L}},\, j \in J,\, y^U_{i,j} = 0\right\} \tag{D.30}$$

Recall that the lower-level problem is degenerate because the leader offers a single price to each market from all plants. This degeneracy implies that some assignment variables are nonbasic but their reduced costs are strictly equal to zero. In order to keep in the bilevel problem the degenerate assignments, we restrict the domain reduction to variables with strictly positive reduced costs in the lower-level problem with maximum capacity.

In Step 3, we established that dual variables μ_i ($i \in I^{\mathcal{C}}$) cannot increase more than dual variables μ_i ($i \in I^{\mathcal{L}}$) when production capacities of the leader are reduced from C^U_i to $C^U_i - \Delta C_i$. Then, according to inequality (D.31), the reduced cost of the variables of the leader cannot decrease when capacities are reduced.

$$\frac{1}{(1+R)^t}P_{i,j} + \mu^U_i - \min_{i' \in I}\left(\frac{1}{(1+R)^t}P_{i',j} + \mu^U_{i'}\right)$$
$$\leq \frac{1}{(1+R)^t}P_{i,j} + \mu^U_i + \epsilon_i - \min_{i' \in I}\left(\frac{1}{(1+R)^t}P_{i',j} + \mu^U_{i'} + \epsilon_{i'}\right)$$
$$\forall\,(i,j) \in \left\{(i,j) :\, i \in I^{\mathcal{L}},\, j \in J\right\} \tag{D.31}$$

Inequality (D.31) implies that variables $y_{i,j}$ $(i \in I^{\mathcal{L}})$ with positive reduced cost in the lower-level problem with maximum capacity have positive reduced costs regardless of the leader's expansion strategy. Therefore, variables $y_{i,j}$ $(i \in I^{\mathcal{L}})$ with positive reduced cost in the lower-level problem with maximum capacity remain nonbasic when capacities of the leader are reduced.

□

Bibliography

[1] I.J.B.F. Adan & J. van der Wal. Combining make to order and make to stock. *Operations Research Spektrum*, 20:73–81, 1998.

[2] S.M. Alizadeh, P. Marcotte, & G. Savard. Two-stage stochastic bilevel programming over a transportation network. *Transportation Research Part B: Methodological*, 58:92–105, 2013.

[3] G. Anandalingam & V. Apprey. Multi-level programming and conflict resolution. *European Journal of Operational Research*, 51(2):233–247, 1991.

[4] G. Anandalingam & T.L. Friesz. Hierarchical optimization: An introduction. *Annals of Operations Research*, 34:1–11, 1992.

[5] R. Anupindi & Y. Bassok. Supply contracts with quantity commitments and stochastic demand. In S. Tayur, R. Ganeshan, & M. Magazine, editors, *Quantitative Models for Supply Chain Management*, pages 197–232. Springer: US, 1999.

[6] E. Balas. Disjunctive programming. *Annals of Discrete Mathematics*, 5:3–12, 1979.

[7] F. Barahona & R. Anbil. The Volume Algorithm: Producing Primal Solutions with a Subgradient Method. *Mathematical Programming*, 87:385–399, 2000.

[8] J.F. Bard & J.E. Falk. An explicit solution to the multi-level programming problem. *Computers & Operations Research*, 9:77–100, 1982.

[9] J.F. Bard & J.T. Moore. An algorithm for the discrete bilevel programming problem. *Naval Research Logistics (NRL)*, 39(3):419–435, 1992.

[10] J.F. Bard, J. Plummer, & J.C. Sourie. A bilevel programming approach to determining tax credits for biofuel production. *European Journal of Operational Research*, 120:30–46, 2000.

[11] M. Ben-Daya & M. Hariga. Integrated single vendor single buyer model with stochastic demand and variable lead time. *International Journal of Production Economics*, 92:75–80, 2004.

[12] A. Ben-Tal, L. El Ghaoui, & A. Nemirovski. *Robust Optimization*. Princeton University Press: US, 2009.

[13] J.F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4:238–252, 1962.

[14] E. Berk & A. Arreolarisa. Note on future-supply uncertainty in EOQ models. *Naval Research Logistics*, 41:129–132, 1994.

[15] G. Bhatia, C. Lane, & A. Wain. Building resilience in supply chains. In *World Economic Forum*, 2013.

[16] W.F. Bialas & M.H. Karwan. On two-level optimization. *Automatic Control, IEEE Transactions on*, 27:211–214, 1982.

[17] D. Bienstock & J.F. Shapiro. Optimizing resource acquisition decisions by stochastic programming. *Management Science*, 34:215–229, 1988.

[18] J.R. Birge & F. Louveaux. A Multicut Algorithm for Two-stage Stochastic Linear Programs. *European Journal of Operational Research*, 34:384–392, 1988.

[19] J.R. Birge & F. Louveaux. *Introduction to Stochastic Programming*. Springer Series in Operations Research. Springer, 2011.

[20] J.A. Bondy & U.S.R. Murty. *Graph Theory with Applications*, chapter Networks, pages 191–211. Elsevier, New York, 1976.

[21] S. Bose & J.F. Pekny. A model predictive framework for planning and scheduling problems: a case study of consumer goods supply chain. *Computers & Chemical Engineering*, 24: 329–335, 2000.

[22] M. Bouakiz & M.J. Sobel. Inventory control with an exponential utility criterion. *Operations Research*, 40:603–608, 1992.

[23] J. Bracken & J.T. McGill. Mathematical programs with optimization problems in the constraints. *Operations Research*, 21:p37–44, 1973.

[24] A. Brooke, D. Kendrick, & A. Meeraus. GAMS: A Users Guide, Release 24.2.1. *The Scientific Press, South San Francisco*, 2013.

[25] A.P. Burgard & C.D. Maranas. Optimization-based framework for inferring and testing hypothesized metabolic objective functions. *Biotechnology and Bioengineering*, 82:670–677, 2003.

[26] M.R. Bussieck, M. C. Ferris, & A. Meeraus. Grid-Enabled Optimization with GAMS. *INFORMS Journal on Computing*, 21:349–362, 2009.

[27] G. Cachon & C. Terwiesch. *Matching supply with demand: An introduction to operations management*. McGraw-Hill: US, 3rd edition, 2013.

[28] W. Candler & R. Townsley. A linear two-level programming problem. *Computers & Operations Research*, 9(1):59–76, 1982.

[29] C.C. Carøe & R. Schültz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24:37–45, 1999.

[30] S. Cerisola, A. Baillo, J.M. Fernandez-Lopez, A. Ramos, & R. Gollmer. Stochastic Power Generation Unit Commitment in Electricity Markets: A Novel Formulation and a Comparison of Solution Methods. *Operations Research*, 57:32–46, 2009.

[31] S-G. Chang & B. Gavish. Telecommunications network topological design and capacity expansion: Formulations and algorithms. *Telecommunication Systems*, 1(1):99–131, 1993.

[32] F. Chen. Optimal policies for multi-echelon inventory problems with batch ordering. *Operations Research*, 48:376–389, 2000.

[33] Q. Chen, X. Li, & Y. Ouyang. Joint inventory-location problem under the risk of probabilistic facility disruptions. *Transportation Research Part B: Methodological*, 45:991–1003, 2011.

[34] X. Chen, M. Sim, D. Simchi-Levi, & P. Sun. Risk aversion in inventory management. *Operations Research*, 55:828–842, 2007.

[35] E.W. Cheney & A.A. Goldstein. Newton's Method for Convex Programming and Tchebycheff Approximation. *Numerische Mathematik*, 1:253–268, 1959.

[36] S. Chopra, G. Reinhardt, & M. Dada. The effect of lead time uncertainty on safety stocks. *Decision Sciences*, 35:1–24, 2004.

[37] S. Christiansen, M. Patriksson, & L. Wynter. Stochastic bilevel programming in structural optimization. *Structural and Multidisciplinary Optimization*, 21:361–371, 2001.

[38] Y. Chu & F. You. Integrated Scheduling and Dynamic Optimization by Stackelberg Game: Bilevel Model Formulation and Efficient Solution Algorithm. *Industrial & Engineering Chemistry Research*, 53(13):5564–5581, 2014.

[39] Y. Chu, F. You, J.M. Wassick, & A. Agarwal. Integrated planning and scheduling under production uncertainties: Bi-level model formulation and hybrid solution method. *Computers & Chemical Engineering*, 72:255 – 272, 2015.

[40] Y. Chu, F. You, J.M. Wassick, & A. Agarwal. Simulation-based optimization framework for multi-echelon inventory systems under uncertainty. *Computers & Chemical Engineering*, 73:1–16, 2015.

[41] A.J. Clark & H. Scarf. Optimal policies for a multi-echelon inventory problem. *Management Science*, 50:1782–1790, 2004.

[42] P.A. Clark & A.W. Westerberg. Bilevel programming for steady-state chemical process design: I. fundamentals and algorithms. *Computers & Chemical Engineering*, 14:87–97, 1990.

[43] B. Colson, P. Marcotte, & G. Savard. An overview of bilevel optimization. *Annals of Operations Research*, 153(1):235–256, 2007.

[44] A.J. Conejo, E. Castillo, R. Minguez, & R. Garcia-Bertrand. *Decomposition Techniques in Mathematical Programming*. Springer: US, 2006.

[45] C.W. Craighead, J. Blackhurst, M.J. Rungtusanatham, & R.B. Handfield. The severity of supply chain disruptions: design characteristics and mitigation capabilities. *Decision Sciences*, 38:131–156, 2007.

[46] T. Cui, Y. Ouyang, & Z-J.M. Shen. Reliable facility location under the risk of disruptions. *Operations Research*, 58:998–1011, 2010.

[47] G.B. Dantzig. *Activity Ananlysis of Production and Allocation*, chapter Maximization of a linear function of variables subject to linear inequalities, pages 339–347. John Wiley: New York, 1951.

[48] M.S. Daskin. *Network and Discrete Location: Models, Algorithms, and Applications*, chap-

ter Appendix H. Longitudes, Latitudes, Demands, and Fixed Cost for SORTCAP.GRT: A 49-Node Problem Defined on the Continental United States. John Wiley & Sons, Inc, 1995.

[49] M.S. Daskin, C.R. Coullard, & Z-J.M. Shen. An inventory-location model: Formulation, solution algorithm and computational results. *Annals of Operations Research*, 110:83–106, 2002.

[50] D. De Wolf & Y. Smeers. A stochastic version of a stackelberg-nash-cournot equilibrium model. *Management Science*, 43:190–197, 1997.

[51] A. Delgadillo, J.M. Arroyo, & N. Alguacil. Analysis of electric grid interdiction with line switching. *Power Systems, IEEE Transactions on*, 25:633–641, 2010.

[52] S. Dempe, J. Dutta, & B.S. Mordukhovich. New necessary optimality conditions in optimistic bilevel programming. *Optimization*, 56:577–604, 2007.

[53] S. Dempe, B.S. Mordukhovich, & A.B. Zemkoho. Necessary optimality conditions in pessimistic bilevel programming. *Optimization*, 63(4):505–533, 2014.

[54] E.V. Denardo. *Dynamic programming: Models and applications.* Prentice-Hall: US, 1982.

[55] S.T. DeNegre & T.K. Ralphs. A branch-and-cut algorithm for integer bilevel linear programs. In J. W. Chinneck, B. Kristjansson, & M. J. Saltzman, editors, *Operations Research and Cyber-Infrastructure*, volume 47 of *Operations Research/Computer Science Interfaces*, pages 65–78. Springer US, 2009.

[56] D. Dentcheva & W. Römisch. Optimal power generation under uncertainty via stochastic programming. In K Marti & P Kall, editors, *Stochastic Programming Methods and Technical Applications*, volume 458, pages 22–56. Springer Berlin Heidelberg, 1998.

[57] E.B. Diks & A.G. de Kok. Optimal control of a divergent multi-echelon inventory system. *European Journal of Operational Research*, 111:75–97, 1998.

[58] G.D. Eppen. Note–effects of centralization on expected costs in a multi-location newsboy problem. *Management Science*, 25:498–501, 1979.

[59] G.D. Eppen & L. Schrage. *Multi-Level Production/Inventory Control Systems: Theory and Practice*, chapter Centralized ordering policies in a multi-warehouse system with lead times and random demand, page 5167. North-Holland, Amsterdam, The Netherlands, 1981.

[60] L.F. Escudero, A. Garín, M. Merino, & G. Pérez. The value of the stochastic solution in multistage problems. *TOP*, 15:48–64, 2007.

[61] M. Ettl, G.E. Feigin, G.Y. Lin, & D.D. Yao. A supply network model with base-stock control and service requirements. *Operations Research*, 48:216–232, 2000.

[62] N.P. Faisca, V. Dua, B. Rustem, P.M. Saraiva, & E.N. Pistikopoulos. Parametric global optimisation for bilevel programming. *Journal of Global Optimization*, 38(4):609–623, 2007.

[63] F. Fantauzzi. Decomposition methods for network optimization problems in the presence of uncertainty. *Lecture notes in economics and mathematical systems*, 450:234–248, 1997.

[64] A. Federgruen & Zipkin P.H. A combined vehicle routing and inventory allocation problem. *Operations research*, 32:1019–1037, 1984.

[65] A. Federgruen & P.H. Zipkin. Allocation policies and cost approximations for multilocation inventory systems. *Naval Research Logistics Quarterly*, 31:97–129, 1984.

[66] A. Federgruen & P.H. Zipkin. Computational issues in an infinite-horizon, multiechelon inventory model. *Operations Research*, 32:818–836, 1984.

[67] A. Federgruen & P.H. Zipkin. An Inventory Model with Limited Production Capacity and Uncertain Demands I. The Average-Cost Criterion. *Mathematics of Operations Research*, 11:193–207, 1986.

[68] A. Federgruen & P.H. Zipkin. An Inventory Model with Limited Production Capacity and Uncertain Demands II. The Discounted-Cost Criterion. *Mathematics of Operations Research*, 11:208–215, 1986.

[69] K. Fischer. Sequential discrete p-facility models for competitive location planning. *Annals of Operations Research*, 111:253–270, 2002.

[70] M.L. Fisher. The Lagrangian Relaxation Method for Solving Integer Programming Problems. *Management Science*, 27:1–18, 1981.

[71] C. Florensa Campo, P. Garcia-Herreros, P. Misra, E. Arslan, S. Mehta, & I.E. Grossmann. Capacity planning with competitive decision-makers: Trilevel MILP formulation and solution approaches. *To be submitted to European Journal of Operations Research*, 2016.

[72] J. Fortuny-Amat & B. McCarl. A representation and economic interpretation of a two-level programming problem. *The Journal of the Operational Research Society*, 32:783–792, 1981.

[73] A. Frangioni. Generalized Bundle Methods. *SIAM Journal on Optimization*, 13:117–156, 2002.

[74] D. Gale, H.W. Kuhn, & A.W. Tucker. *Activity Analysis of Production and Allocation*, chapter Linear programming and the theory of games, page 317329. New York: Wiley, 1951.

[75] L.P. Garces, A.J. Conejo, R. Garcia-Bertrand, & R. Romero. A bilevel approach to transmission expansion planning within a market environment. *Power Systems, IEEE Transactions on*, 24:1513–1522, 2009.

[76] P. Garcia-Herreros & I.E. Grossmann. Stochastic Programming for Supply Chains Resilience [in Spanish]. *XXVII Congreso Interamericano y Colombiano de Ingenieria Quimica*, pages 1309–1314, 2014.

[77] P. Garcia-Herreros, I.E. Grossmann, & J.M. Wassick. Design of supply chains under the risk of facility disruptions. *Computer Aided Chemical Engineering*, 32:577–582, 2013.

[78] P. Garcia-Herreros, I.E. Grossmann, B. Sharda, A. Agarwal, & J.M. Wassick. Empirical study of the behavior of capacitated production-inventory systems. In *Proceedings of the 2014 Winter Simulation Conference*, WSC '14, pages 2251–2260. IEEE Press, 2014.

[79] P. Garcia-Herreros, J.M. Wassick, & I.E. Grossmann. Design of resilient supply chains with risk of facility disruptions. *Industrial & Engineering Chemistry Research*, 53:17240–17251, 2014.

[80] P. Garcia-Herreros, P. Misra, E. Arslan, S. Mehta, & I.E. Grossmann. A duality-based approach for bilevel optimization of capacity expansion. *Computer Aided Chemical Engineering*, 37:2021–2026, 2015.

[81] P. Garcia-Herreros, L. Zhang, P. Misra, E. Arslan, & I.E. Grossmann. Mixed-integer bilevel optimization for capacity planning with rational markets. *Submitted for publication in Computers & Chemical Engineering*, 2015.

[82] P. Garcia-Herreros, A. Agarwal, J.M. Wassick, & I.E. Grossmann. Optimizing inventory policies in process networks under uncertainty. *To be submitted to Computers & Chemical Engineering*, 2016.

[83] A.M. Geoffrion. Generalized Benders Decomposition. *Journal of Optimization Theory and Applications*, 10:237–260, 1972.

[84] A.M. Geoffrion & G.W. Graves. Multicommodity distribution system design by benders decomposition. *Management Science*, 20:822–844, 1974.

[85] A.M. Geoffrion & R. McBride. Lagrangean relaxation applied to capacitated facility location problems. *AIIE Transactions*, 10:40–47, 1978.

[86] P. Glasserman & S. Tayur. Sensitivity analysis for base-stock levels in multiechelon production-inventory systems. *Management Science*, 41:263–281, 1995.

[87] F. Glover. Improved linear integer programming formulations of nonlinear integer problems. *Management Science*, 22:455–460, 1975.

[88] V. Goel & I.E. Grossmann. A class of stochastic programs with decision dependent uncertainty. *Mathematical Programming*, 108:355–394, 2006.

[89] J.L. Goffin, A. Haurie, & J.P. Vial. Decomposition and nondifferentiable optimization with the projective algorithm. *Management Science*, 38:284–302, 1992.

[90] S.C. Graves. A multiechelon inventory model with fixed replenishment intervals. *Management Science*, 42:1–18, 1996.

[91] S.C. Graves & S.P. Willems. Optimizing strategic safety stock placement in supply chains. *Manufacturing & Service Operations Management*, 2:68–83, 2000.

[92] I.E. Grossmann. Enterprise-wide optimization: A new frontier in process systems engineering. *Aiche Journal*, 51:1846–1857, 2005.

[93] I.E. Grossmann. Advances in mathematical programming models for enterprise-wide optimization. *Computers & Chemical Engineering*, 47:2–18, 2012.

[94] I.E. Grossmann & C.A. Floudas. Active constraint strategy for flexibility analysis in chemical processes. *Computers & Chemical Engineering*, 11:675–693, 1987.

[95] I.E. Grossmann, B.A. Calfa, & P. Garcia-Herreros. Evolution of concepts and models for quantifying resiliency and flexibility of chemical processes. *Computers & Chemical Engineering*, 70:22–34, 2014.

[96] I.E. Grossmann, R.M. Apap, B.A. Calfa, P. Garcia-Herreros, & Q. Zhang. Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty. *Computer Aided Chemical Engineering*, 37:1–14, 2015.

[97] M. Guignard & S. Kim. Lagrangean decomposition: A model yielding stronger lagrangean bounds. *Mathematical Programming*, 2(333-353), 1987.

[98] Z.H. Gümüş & C.A. Floudas. Global optimization of mixed-integer bilevel programming problems. *Computational Management Science*, 2(3):181–212, 2005.

[99] V. Gupta & I.E. Grossmann. Solution strategies for multistage stochastic programming with endogenous uncertainties. *Computers & Chemical Engineering*, 35:2235–2247, 2011.

[100] V. Gupta & I.E. Grossmann. A new decomposition algorithm for multistage stochastic programs with endogenous uncertainties. *Computers & Chemical Engineering*, 62:62–79, 2014.

[101] F.W. Harris. How many parts to make at once. *Operations Research*, 38:947–950, 1990.

[102] J.M. Harrison. *Brownian Motion and Stochastic Systems*. Wiley: New York, 1985.

[103] W.E. Hart, J-P. Watson, & D.L. Woodruff. Pyomo: modeling and solving mathematical programs in python. *Mathematical Programming Computation*, 3:219–260, 2011.

[104] X.J. He, J.G. Kim, & J.C. Hayya. The cost of lead-time variability: The case of the exponential distribution. *International Journal of Production Economics*, 97:130–142, 2005.

[105] H. Heitsch & W. Romisch. Scenario Reduction Algorithms in Stochastic Programming. *Computational Optimization and Applications*, 24:187–206, 2003.

[106] M. Held & R.M. Karp. The Traveling-Salesman Problem and Minimum Spanning Trees: Part ii. *Mathematical Programming*, 1:6–25, 1971.

[107] M. Held, P. Wolfe, & H.P. Crowder. Validation of Subgradient Optimization. *Mathematical Programming*, 6:62–88, 1974.

[108] O.K. Helferich & R.L. Cook. *Securing the Supply Chain*. Council of Logistics Management, 2002.

[109] K. Holmberg. On the Convergence of Cross Decomposition. *Mathematical Programming*, 47:269–296, 1990.

[110] K. Holmberg. Linear Mean Value Cross Decomposition: A Generalization of the Kornai-Liptak Method. *European Journal of Operational Research*, 62:55–73, 1997.

[111] K. Holmberg. Mean Value Cross Decomposition Applied to Integer Programming Problems. *European Journal of Operational Research*, 97:124–138, 1997.

[112] J.N. Hooker. Logic-based methods for optimization. In Alan Borning, editor, *Principles and Practice of Constraint Programming*, pages 336–349. Springer: Berlin Heidelberg, 1994.

[113] J.N. Hooker. *Integrated methods for optimization*. Springer: US, 2nd edition, 2012.

[114] D. Huppmann & J. Egerer. National-Strategic Investment in European Power Transmission Capacity. *DIW Berlin Discussion Paper No. 1379*, pages 1–23, 2014.

[115] M.G. Ierapetritou & E.N. Pistikopoulos. Batch plant design and operations under uncertainty. *Industrial & Engineering Chemistry Research*, 35:772–787, 1996.

[116] K. Inderfurth. Safety stock optimization in multi-stage inventory systems. *International Journal of Production Economics*, 24:103–113, 1991.

[117] K. Inderfurth & S. Minner. Safety stocks in multi-stage inventory systems under different service measures. *European Journal of Operational Research*, 106:57–73, 1998.

[118] J.R. Jackson. Networks of waiting lines. *Operations Research*, 5:518–521, 1957.

[119] H-M. Jeon. *Location-inventory models with supply disruptions*. ProQuest, UMI Dissertation Publishing, 2011.

[120] T.W. Jonsbraten, R.J-B. Wets, & D.L. Woodruff. A class of stochastic programs with decision dependent random elements. *Annals of Operations Research*, 82:83–106, 1998.

[121] W.C. Jordan & S.C. Graves. Principles on the benefits of manufacturing process flexibility. *Management Science*, 41:577–594.

[122] J.Y. Jung, G. Blau, J.F. Pekny, Reklaitis G.V., & D. Eversdyk. A simulation based optimization approach to supply chain management under demand uncertainty. *Computers and Chemical Engineering*, 28:20872106, 2004.

[123] J.Y. Jung, G. Blau, J.F. Pekny, G.V. Reklaitis, & D. Eversdyk. Integrated safety stock management for multi-stage supply chains under production capacity constraints. *Computers & Chemical Engineering*, 32:2570–2581, 2008.

[124] A. Kandiraju, P. Garcia-Herreros, P. Misra, E. Arslan, S. Mehta, & I.E. Grossmann. Capacity planning for the air separation industry with rational markets and demand uncertainty. *Submitted to Computer Aided Chemical Engineering*, 2016.

[125] I.A. Karimi & G.V. Reklaitis. Optimal selection of intermediate storage tank capacity in a periodic batch/semicontinuous process. *AIChE Journal*, 29:588–596, 1983.

[126] J.E. Kelley. The Cutting-Plane Method for Solving Convex Programs. *Journal of the Society for Industrial and Applied Mathematics*, 8:703–712, 1960.

[127] G.E. Kimball. General principles of inventory control. *Journal of Manufacturing and Operations Management*, 1:119–130, 1988. Original from 1955.

[128] K.C. Kiwiel. Proximity Control in Bundle Methods for Convex Nondifferentiable Minimization. *Mathematical Programming*, 46:105–122, 1990.

[129] P-M. Kleniati & C.S. Adjiman. A generalization of the branch-and-sandwich algorithm: From continuous to mixed-integer nonlinear bilevel problems. *Computers & Chemical Engineering*, 72:373–386, 2015.

[130] A.J. Kleywegt, A. Shapiro, & T. Homem-de Mello. The sample average approximation method for stochastic discrete optimization. *SIAM J. on Optimization*, 12:479–502, 2002.

[131] W. Klibi & A. Martel. Modeling approaches for the design of resilient supply networks under disruptions. *International Journal of Production Economics*, 135:882–898, 2012.

[132] W. Klibi & A. Martel. Scenario-based supply chain network risk modeling. *European Journal of Operational Research*, 223:644–658, 2012.

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

[134] E. Krausmann & A.M. Cruz. Impact of the 11 march 2011, Great East Japan earthquake and tsunami on the chemical industry. *Natural Hazards*, pages 811–828, 2013.

[135] D. Kuhn, W. Wiesemann, & A. Georghiou. Primal and dual linear decision rules in stochastic and robust optimization. *Mathematical Programming*, 130:177–209, 2011.

[136] J.M. Laínez & L. Puigjaner. Prospective and perspective review in integrated supply chain modelling for the chemical process industry. *Current Opinion in Chemical Engineering*, pages 430–445, 2012.

[137] A. Latour. Trial by fire: A blaze in albuquerque sets off major crisis for cell-phone giants. *Wall Street Journal*, January 29th, 2001.

[138] H.L. Lee & C. Billington. Material management in decentralized supply chains. *Operations Research*, 41(5):835–847, 1993.

[139] Y-J. Lee & P.H. Zipkin. Tandem queues with planned inventories. *Operations Research*, 40: 936–947, 1992.

[140] Y.H. Lee, M.K. Cho, & Y.B. Kim. A discrete-continuous combined modeling approach for supply chain simulation. *Simulation*, 78:321–329, 2002.

[141] C. Lemarechal. An Algorithm for Minimizing Convex Functions. *In: Proceedings IFIP'74 Congress*, pages 552–556, 1974.

[142] P. Li, H. Arellano-Garcia, & G. Wozny. Chance constrained programming approach to process optimization under uncertainty. *Computers & Chemical Engineering*, 32:25–45, 2008.

[143] Q. Li, B. Zeng, & A. Savachkin. Reliable facility location design under disruptions. *Computers & Operations Research*, 40:901–909, 2013.

[144] X. Li & Y. Ouyang. A continuum approximation approach to reliable facility location design under correlated probabilistic disruptions. *Transportation Research Part B: Methodological*, 44:535–548, 2010.

[145] M.J. Liberatore. The eoq model under stochastic lead time. *Operations Research*, 27:391–396, 1979.

[146] M. Lim, M. Daskin, S. Chopra, & A. Bassamboo. A facility reliability problem: Formulation, properties, and algorithm. *Naval Research Logistics*, 57:58–70, 2010.

[147] X. Lin, S.L. Janak, & C.A. Floudas. A new robust optimization approach for scheduling under uncertaint. *Computers & Chemical Engineering*, 28:10691085, 2004.

[148] J. Linderoth, A. Shapiro, & S. Wright. The Empirical Behavior of Sampling Methods for Stochastic Programming. *Annals of Operations Research*, 142:215–241, 2006.

[149] P. Loridan & J. Morgan. Weak via strong Stackelberg problem: New results. *Journal of Global Optimization*, 8:263–287, 1996. ISSN 0925-5001.

[150] M. Lubin & I. Dunning. Computing in operations research using julia. *INFORMS Journal on Computing*, 27:238–248, 2015.

[151] H. Luss. Operations research and capacity expansion problems: A survey. *Operations Research*, 30:907–947, 1982.

[152] T.L. Magnanti & R.T. Wong. Accelerating Benders Decomposition: Algorithmic Enhancement and Model Selection Criteria. *Operations Research*, 29:464–484, 1981.

[153] C.T. Maravelias & I.E. Grossmann. Simultaneous planning for new product development and batch manufacturing facilities. *Industrial & Engineering Chemistry Research*, 40(26): 6147–6164, 2001.

[154] C.T. Maravelias & I.E. Grossmann. Simultaneous planning for new product development and batch manufacturing facilities. *Industrial & engineering chemistry research*, 40(26): 6147–6164, 2001.

[155] M.J. Meixell & V.B. Gargeya. Global supply chain design : A literature review and critique. *Transportation Research*, 41:531–550, 2005.

[156] Q. Meng, T. Wang, & S. Wang. Short-term liner ship fleet planning with container transshipment and uncertain container shipment demand. *European Journal of Operational Research*, 223:96–105, 2012.

[157] A. Migdalas. Bilevel programming in traffic planning: models, methods and challenge. *Journal of Global Optimization*, 7(4):381–405, 1995.

[158] S.A. MirHassani, C. Lucas, G. Mitra, E. Messina, & C.A. Poojari. Computational solution of capacity planning models under uncertainty. *Parallel Computing*, 26:511–538, 2000.

[159] D. Mitra. Stochastic theory of a fluid model of producers and consumers coupled by a buffer. *Advances in Applied Probability*, 20:646–676, 1988.

[160] S. Mitra, P. Garcia-Herreros, & I.E. Grossmann. A novel cross-decomposition multi-cut scheme for two-stage stochastic programming. *Computer Aided Chemical Engineering*, 33: 241–246, 2014.

[161] S. Mitra, P. Garcia-Herreros, & I.E. Grossmann. An Enhanced Cross-Decomposition Scheme with Primal-Dual Multi-cuts for Two-Stage Stochastic Programming Investment Planning Problems. *Submitted to Mathematical Programming*, 2015.

[162] A. Mitsos, G.M. Bollas, & P.I. Barton. Bilevel optimization formulation for parameter esti-

mation in liquid–liquid phase equilibrium problems. *Chemical Engineering Science*, 64(3): 548–559, 2009.

[163] J.T. Moore & J.F. Bard. The mixed integer linear bilevel programming problem. *Operations Research*, 38(5):911–921, 1990.

[164] A.L. Motto, J.M. Arroyo, & F.D. Galiana. A Mixed-Integer LP Procedure for the Analysis of Electric Grid Security Under Disruptive Threat. *Power Systems, IEEE Transactions on*, 20:1357–1365, 2005.

[165] S. Mouret, I.E. Grossmann, & P. Pestiaux. A new Lagrangian Decomposition Approach Applied to the Integration of Refinery Planning and Crude-oil Scheduling. *Computers and Chemical Engineering*, 35:2750–2766, 2011.

[166] F.H. Murphy & Y. Smeers. Generation capacity expansion in imperfectly competitive restructured electricity markets. *Operations Research*, 53(4):646–661, 2005.

[167] F.H. Murphy, S. Sen, & A.L. Soyster. Electric utility capacity expansion planning with uncertain load forecasts. *A I I E Transactions*, 14(1):52–59, 1982.

[168] W.S. Nainis & Y.Y. Haimes. A multilevel approach to planning for capacity expansion in water resource systems. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-5(1): 53–63, 1975.

[169] K. Nwazota. Hurricane katrina underscores tenuous state of u.s. oil refining industry. http://www.pbs.org/newshour/bb/weather/july-dec05/katrina/oil_background.html, September 9th, 2005. Accessed on 01/23/2014.

[170] F. Oliveira, V. Gupta, S. Hamacher, & I.E. Grossmann. A Lagrangean Decomposition Approach for Oil Supply Chain Investment Planning under Uncertainty with Risk Considerations. *Computers & Chemical Engineering*, 50:184–195, 2013.

[171] M.J. Osborne & A. Rubenstein. *A course in game theory*, chapter 6. Extensive games with perfect information, pages 89–115. Cambridge (MA): The MIT Press., 1994.

[172] S.H. Owen & M.S. Daskin. Strategic facility location: A review. *European Journal of Operational Research*, 111:423–447, 1998.

[173] J.C-H. Pan, M-C. Lo, & Y-C. Hsiao. Optimal reorder point inventory models with vari-

able lead time and backorder discount considerations. *European Journal of Operational Research*, 158:488–505, 2004.

[174] M. Park, , S. Park, F.D. Mele, & I.E. Grossmann. Modeling of purchase and sales contracts in supply chain optimization. *Industrial & Engineering Chemistry Research*, 45:5013–5026, 2006.

[175] M. Parlar & D. Berkin. Future-supply uncertainty in EOQ models. *Naval Research Logistics*, 38:107–121, 1991.

[176] P. Peng, L.V. Snyder, Z. Liu, & A. Lim. Reliable logistics networks design with facility disruptions. *Transportation Research Part B: Methodological*, 45:1190–1211, 2011.

[177] E. Perea-López, I.E. Grossmann, B.E. Ydstie, & T. Tahmassebi. Dynamic modeling and decentralized control of supply chains. *Industrial & Engineering Chemistry Research*, 40: 3369–3383, 2001.

[178] E. Perea-López, B.E. Ydstie, & I.E. Grossmann. A model predictive control strategy for supply chain optimization. *Computers & Chemical Engineering*, 27:1201–1218, 2003.

[179] S.B. Petkov & C.D. Maranas. Design of single-product campaign batch plants under demand uncertainty. *AIChE Journal*, 44:896–911, 1998.

[180] W.B. Powell. *Approximate dynamic programming: Solving the curses of dimensionality*. Wiley: US, 2nd edition, 2011.

[181] L. Qi, Z-J.M. Shen, & L.V. Snyder. A continuous-review inventory model with disruptions at both supplier and retailer. *Production and Operations Management*, 18:516–532, 2009.

[182] L. Qi, Z-J.M. Shen, & L.V. Snyder. The effect of supply disruptions on supply chain design decisions. *Transportation Science*, 44:274–289, 2010.

[183] R. Raman & I.E. Grossmann. Modelling and computational techniques for logic based integer programming. *Computers and Chemical Engineering*, 18:563–578, 1994.

[184] J.B. Rawlings, D. Angeli, & C.N. Bates. Fundamentals of economic model predictive control. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 3851–3861, 2012.

[185] J. Rice & F. Caniato. Building a secure and resilient supply network. *Supply Chain Management Review*, 7:22–33, 2003.

[186] S.M. Robinson. Analysis of sample-path optimization. *Mathematics of Operations Research*, 21:513–528, 1996.

[187] K. Rosling. Optimal inventory policies for assembly systems under random demands. *Operations Research*, 37:565–579, 1989.

[188] C. Ruiz, A.J. Conejo, & Y. Smeers. Equilibria in an oligopolistic electricity pool with stepwise offer curves. *IEEE Transactions on Power Systems*, 27:752–761, 2012.

[189] A. Ruszczynski. Decomposition methods in stochastic programming. *Mathematical Programming*, 79:215–228, 1997.

[190] J-H. Ryu, V. Dua, & E.N. Pistikopoulos. A bilevel programming framework for enterprise-wide process networks under uncertainty. *Computers & Chemical Engineering*, 28:1121–1129, 2004.

[191] G.K.D. Saharidis, M. Minoux, & M.G. Ierapetritou. Accelerating benders method using covering cut bundle generation,. *International Transactions in Operational Research*, 17: 221–237, 2010.

[192] G.K.D. Saharidis, A.J. Conejo, & G. Kozanidis. Exact solution methodologies for linear and (mixed) integer bilevel programming. In El-Ghazali Talbi, editor, *Metaheuristics for Bi-level Optimization*, volume 482 of *Studies in Computational Intelligence*, pages 221–245. Springer Berlin Heidelberg, 2013.

[193] N.V. Sahinidis. Optimization under uncertainty: state-of-the-art and opportunities. *Computers & Chemical Engineering*, 28:971–983, 2004.

[194] N.V. Sahinidis & I.E. Grossmann. Convergence Properties of Generalized Benders Decomposition. *Computers & Chemical Engineering*, 15:481–491, 1991.

[195] N.V. Sahinidis & I.E. Grossmann. Reformulation of the Multiperiod MILP Model for Capacity Expansion of Chemical Processes. *Operations Research*, 40:S127–S144, 1992.

[196] N.V. Sahinidis, I.E. Grossmann, R.E. Fornari, & M. Chathrathi. Optimization model for long range planning in the chemical industry. *Computers & Chemical Engineering*, 13: 1049–1063, 1989.

[197] M.I.G. Salema, A.M. Barbosa-Povoa, & A.Q. Novais. An optimization model for the design

of a capacitated multi-product reverse logistics network with uncertainty. *European Journal of Operational Research*, pages 1063–1077, 2007.

[198] J. Salmeron, K. Wood, & R. Baldick. Analysis of electric grid security under terrorist threat. *Power Systems, IEEE Transactions on*, 19:905–912, 2004.

[199] T. Santoso, S. Ahmed, M. Goetschalckx, & A. Shapiro. A stochastic programming approach for supply chain network design under uncertainty. *European Journal of Operational Research*, 167:96–115, 2005.

[200] P. Schültz & A. Tomasgard. The impact of flexibility on operational supply chain planning. *International Journal of Production Economics*, 134:300–311, 2011.

[201] P. Schültz, A. Tomasgard, & S. Ahmed. Supply chain design under uncertainty using sample average approximation and dual decomposition. *European Journal of Operational Research*, 199:409–419, 2009.

[202] P. Schütz, A. Tomasgard, & S. Ahmed. Supply chain design under uncertainty using sample average approximation and dual decomposition. *European Journal of Operational Research*, 199:409–419, 2009.

[203] N. Shah. Process industry supply chains: Advances and challenges. *Computers & Chemical Engineering*, 29:1225–1236, 2005.

[204] N. Shah & C.C. Pantelides. Design of multipurpose batch plants with uncertain production requirements. *Industrial & Engineering Chemistry Research*, 31:1325–1337, 1992.

[205] K.H. Shang & J-S. Song. Newsvendor bounds and heuristic for optimal policies in serial supply chains. *Management Science*, 49:618–638, 2003.

[206] A. Shapiro & T. Homem-de Mello. A simulation-based approach to two-stage stochastic programming with recourse. *Mathematical Programming*, 81:301–325, 1998.

[207] Z-J.M. Shen. Integrated supply chain design models: A survey and future research direction. *Journal of Industrial and Management Optimization*, 3:1–27, 2007.

[208] Z-J.M. Shen, C.R. Coullard, & M.S. Daskin. A joint location - inventory model. *Transportation Science*, 37:40–55, 2003.

[209] Z-J.M. Shen, R. Zhan, & J. Zhang. The reliable facility location problem: Formulations,

heuristics, and approximation algorithms. *INFORMS Journal on Computing*, 23:470–482, 2011.

[210] K.F.Jr. Simpson. In-process inventories. *Operations Research*, 6(6):863–873, 1958.

[211] J.C. Smith, C. Lim, & A. Alptekinoglu. Optimal mixed-integer programming and heuristic methods for a bilevel stackelberg product introduction game. *Naval Research Logistics*, 56: 714–729, 2009.

[212] L. Snyder & Z-J.M. Shen. *Fundamentals of Supply Chain Theory*. Wiley: Hoboken (NJ), 2011.

[213] L.V. Snyder. Facility location under uncertainty: A review. *IIE Transactions*, 38:547–564, 2006.

[214] L.V. Snyder & M.S. Daskin. Reliability models for facility location: The expected failure cost case. *Transportation Science*, 39:400–416, 2005.

[215] L.V. Snyder & Z-J.M. Shen. Supply and demand uncertainty in multi-echelon supply chains. College of Engineering and Applied Sciences, Lehigh University., 2006.

[216] L.V. Snyder & Z-J.M. Shen. *Fundamentals of Supply Chain Theory*, chapter Deterministic Inventory Models, pages 29–62. Wiley, 2011.

[217] L.V. Snyder & Z-J.M. Shen. *Fundamentals of Supply Chain Theory*, chapter Stochastic Inventory Models, pages 63–116. Wiley, 2011.

[218] H.S. Sohn, D.L. Bricker, & T.L. Tseng. Mean Value Cross Decomposition for Two-Stage Stochastic Linear Programming with Recourse. *The Open Operational Research Journal*, 5:30–38, 2011.

[219] A.L. Soyster & F.H. Murphy. *Economic Behaviour of Electric Utilities*. Prentice-Hall, 1989.

[220] D.A. Straub & I.E. Grossmann. Integrated stochastic metric of flexibility for systems with discrete state and continuous parameter uncertainties. *Computers & Chemical Engineering*, 14:967 – 985, 1990.

[221] S. Subrahmanyam, J.F. Pekny, & G.V. Reklaitis. Design of batch chemical plants under market uncertainty. *Industrial & Engineering Chemistry Research*, 33:2688–2701, 1994.

[222] K. Subramanian, C.T. Maravelias, & J.B. Rawlings. A state-space model for chemical production scheduling. *Computers & Chemical Engineering*, 47:97 – 110, 2012.

[223] K. Subramanian, J.B. Rawlings, C.T. Maravelias, J. Flores-Cerrillo, & L. Megan. Integration of control theory and scheduling methods for supply chain management. *Computers & Chemical Engineering*, 51:4–20, 2013.

[224] A. Sundaramoorthy, J.M.B. Evans, & P.I. Barton. Capacity planning under clinical trials uncertainty in continuous pharmaceutical manufacturing, 1: Mathematical framework. *Industrial & Engineering Chemistry Research*, 51(42):13692–13702, 2012.

[225] B. Tarhan, V. Gupta, & I.E. Grossmann. Improving Dual Bound for Stochastic MILP Models Using Sensitivity Analysis. *submitted to European Journal of Operational Research*, 2013.

[226] S.R. Tayur. Computing the optimal policy for capacitated inventory models. *Communications in Statistics. Stochastic Models*, 9:585–598, 1993.

[227] S. Terrazas-Moreno, I.E. Grossmann, J.M. Wassick, & S.J. Bury. Optimal design of reliable integrated chemical production sites. *Computers & Chemical Engineering*, 34:1919–1936, 2010.

[228] S. Terrazas-Moreno, I.E. Grossmann, J.M. Wassick, S.J. Bury, & N. Akiya. An efficient method for optimal design of large-scale integrated chemical production sites with endogenous uncertainty. *Computers & Chemical Engineering*, 37:89–103, 2012.

[229] I. Thomas, W. Abriatis, & C. Savage. *Manufacturing and Trade: Inventories and Sales - December 2013*. U.S. Department of Commerce, 2013.

[230] B. Tomlin. On the value of mitigation and contingency strategies for managing supply chain disruption risks. *Management Science*, 52:639–657, 2006.

[231] F. Trespalacios & I.E. Grossmann. Lagrangean relaxation of the hull-reformulation of linear generalized disjunctive programs and its use in disjunctive branch and bound. *Submitted to European Journal of Operational Research*, 2015.

[232] P. Tsiakis, N. Shah, & C.C. Pantelides. Design of multi-echelon supply chain networks under demand uncertainty. *Industrial & Engineering Chemistry Research*, 40:3585–3604, 2001.

[233] S.A. Van den Heever & I.E. Grossmann. Disjunctive multiperiod optimization methods for

design and planning of chemical process systems. *Computers & Chemical Engineering*, 23 (8):1075–1095, 1999.

[234] G-J. van Houtum & W.H.M. Zijm. Computational procedures for stochastic multi-echelon production systems. *International Journal of Production Economics*, 23:223 – 237, 1991.

[235] G-J. van Houtum & W.H.M. Zijm. On the relationship between cost and service models for general inventory systems. *Statistica Neerlandica*, 54:127–147, 2000.

[236] G-J. van Houtum, A. Scheller-Wolf, & J. Yi. Optimal control of serial inventory systems with fixed replenishment intervals. *Operations Research*, 55(4):674–687, 2007.

[237] T.J. Van Roy. Cross Decomposition for Mixed Integer Programming. *Mathematical Programming*, 25:46–63, 1983.

[238] R. Van Slyke & R.J.B. Wets. L-shaped Linear Programs with Applications to Optimal Control and Stochastic Programming. *SIAM Journal on Applied Mathematics*, 17:638–663, 1969.

[239] H. von Stackelberg. *Market Structure and Equilibrium*. Springer: US, 2011.

[240] H.M. Wagner & T.M. Whitin. Dynamic version of the economic lot size model. *Management Science*, 5:89–96, 1958.

[241] P. Wang & J.A. Hill. Recursive behavior of safety stock reduction: The effect of lead-time uncertainty. *Decision Sciences*, 37:285–290, 2006.

[242] A. Weber & C.J. Friedrich. *Alfred Weber's theory of the location of industries*. The University of Chicago Press, 1929.

[243] H.S. Wellons & G.V. Reklaitis. The design of multiproduct batch plants under uncertainty with staged expansion. *Computers & Chemical Engineering*, 13:115–126, 1989.

[244] P. Wentges. Accelerating benders' decomposition for the capacitated facility location problem. *Mathematical Methods of Operations Research*, 44:267–290, 1996.

[245] B.D. Williams & T. Tokar. A review of inventory management research in major logistics journals: Themes and future directions. *The International Journal of Logistics Management*, 19:212–232, 2008.

[246] R. Wilson. *24th Annual State of Logistic Report*. Counsil of Supply Chain Management Professionals, 2013.

[247] P. Xu & L. Wang. An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. *Computers & operations research*, 41:309–318, 2014.

[248] Y. Yao, T. Edmunds, D. Papageorgiou, & R. Alvarez. Trilevel optimization in power network defense. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(4):712–718, 2007.

[249] F. You & I.E. Grossmann. Mixed-integer nonlinear programming models and algorithms for large-scale supply chain design with stochastic inventory management. *Industrial & Engineering Chemistry Research*, 47:7802–7817, 2008.

[250] F. You & I.E. Grossmann. Integrated multi-echelon supply chain design with inventories under uncertainty: Minlp models, computational strategies. *AIChE Journal*, 56:419–440, 2010.

[251] F. You & I.E. Grossmann. Stochastic inventory management for tactical process planning under uncertainties: Minlp models and algorithms. *AIChE Journal*, 57:1250–1277, 2011.

[252] F. You, J.M. Wassick, & I.E. Grossmann. Risk management for a global supply chain planning under uncertainty: Models and algorithms. *AIChE Journal*, 55:931–946, 2009.

[253] D. Yue & F. You. Planning and scheduling of flexible process networks under uncertainty with stochastic inventory: MINLP models and algorithm. *AIChE Journal*, 59:1511–1532, 2013.

[254] M.A. Zamarripa, A.M. Aguirre, C.A. Mendez, & A. Espuna. Improving supply chain planning in a competitive environment. *Computers & Chemical Engineering*, 42:178–188, 2012.

[255] B. Zeng & Y. An. Solving bilevel mixed integer program by reformulations and decomposition. *Optimization On-line*, 2014.

[256] P.H. Zipkin. *Foundations of inventory management*. McGraw-Hill, 2000.

[257] J. Zowe. Nondifferentiable Optimization. *In K. Schittkowski, Computational Mathematical Program, NATO ASI Series F: Computer and Systems Science*, 15:323–356, 1985.