SUPPLY CHAIN NETWORK DESIGN
UNDER UNCERTAIN AND DYNAMIC DEMAND

A Dissertation

by

AYMAN HASSAN RAGAB

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

December 2010

Major Subject: Industrial Engineering

SUPPLY CHAIN NETWORK DESIGN

UNDER UNCERTAIN AND DYNAMIC DEMAND

A Dissertation

by

AYMAN HASSAN RAGAB

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Brett A. Peters
Committee Members,	Jianer Chen
	Illya V. Hicks
	Lewis Ntaimo
	Powell Robinson
Head of Department,	Brett A. Peters

December 2010

Major Subject: Industrial Engineering

ABSTRACT

Supply Chain Network Design

Under Uncertain and Dynamic Demand. (December 2010)

Ayman Hassan Ragab, B.Sc., Alexandria University, Egypt;

M.Sc., Alexandria University, Egypt

Chair of Advisory Committee: Dr. Brett A. Peters

Supply chain network design (SCND) identifies the production and distribution resources essential to maximizing a network's profit. Once implemented, a SCND impacts a network's performance for the long-term. This dissertation extends the SCND literature both in terms of model scope and solution approach.

The SCND problem can be more realistically modeled to improve design decisions by including: the location, capacity, and technology attributes of a resource; the effect of the economies of scale on the cost structure; multiple products and multiple levels of supply chain hierarchy; stochastic, dynamic, and correlated demand; and the gradually unfolding uncertainty. The resulting multistage stochastic mixed-integer program (MSMIP) has no known general purpose solution methodology. Two decomposition approaches—end-of-horizon (EoH) decomposition and nodal decomposition—are applied.

The developed EoH decomposition exploits the traditional treatment of the *end-of-horizon effect*. It rests on independently optimizing the SCND of every node of the last level of the scenario-tree. Imposing these optimal configurations before optimizing the design decisions of the remaining nodes produces a smaller and thus easier to solve MSMIP. An optimal solution results when the discount rate is 0%. Otherwise, this decomposition deduces a bound on the optimality-gap. This decomposition is neither SCND nor MSMIP specific; it pertains to any application sensitive to the EoH-effect and to special cases of MSMIP. To demonstrate this versatility, additional computational experiments for a two-stage mixed-integer stochastic program (SMIP) are included.

This dissertation also presents the first application of nodal decomposition in both SCND and MSMIP. The developed column generation heuristic optimizes the nodal sub-problems using an iterative procedure that provides a restricted master problem's columns. The heuristic's computational efficiency rests on solving the sub-problems independently and on its novel handling of the master problem. Conceptually, it reformulates the master problem to avoid the duality-gap. Technologically, it provides the first application of Leontief substitution flow problems in MSMIP and thereby shows that hypergraphs lend themselves to loosely coupled MSMIPs. Computational results demonstrate superior performance of the heuristic approach and also show how this heuristic still applies when the SCND problem is modeled as a SMIP where the restricted master problem is a shortest-path problem.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

TABLE Page

TABLE Page

TABLE Page

TABLE Page

LIST OF FIGURES

CHAPTER I

INTRODUCTION

With the cost of US business logistics hitting \$1.3 trillion (9.4% of the GDP) in 2009 (Burnson, 2009), the efficiency of logistics networks remains a priority (Chopra and Meindle, 2007). As a strategic decision, supply chain network design (SCND) controls the long-term efficiency of logistics networks by setting the frame within which decisions of the tactical and operational levels have to take place (Chopra and Meindle, 2007). Tactical policies (such as inventory and transportation) and operational decisions (such as scheduling and routing) reduce logistics cost by optimizing the utilization of existing resources (Shapiro, 2007). Identifying the right resources to acquire is the goal of SCND (Shapiro, 2007) and the focus of this work.

1. Modeling the supply chain network design problem

A realistic SCND model would include, at minimum, the following factors:

- *All the attributes (location, capacity, and technology) of a resource.* Selecting the number and location of facilities involves a trade off between economies of scale in investment cost and the transportation cost among distant supply chain network nodes. Setting the timing and size of the capacity needed to equip spatial resources involves a trade off between the economies of scale in capacity acquisition and the cost of holding excess capacity. Selecting the types of technologies needed to fulfill the capacity plan involves a production trade off between economies of scale provided by highly automated lines and economies of scope provided by flexible manufacturing systems.

This dissertation follows the style of *IIE Transactions.*

- *The effect of the economies of scale on the cost structure.* SCND involves capital intensive facilities with long service lives (Owen and Daskin, 1998). Including fixed-cost components in the cost function better approximates resource acquisition cost.

- *Multiple products and multiple levels of supply chain hierarchy.* Different product families compete for the network's finite resources. Any given product might need to visit multiple types of manufacturing and distribution facilities. These facilities are traditionally located in a multi-level hierarchy. The flow of the different products from suppliers to customer zones links successive supply chain hierarchical levels. Fig. 1 shows an example of a supply chain network.

- *Stochastic, dynamic, and correlated demand.* Demand is the main source of uncertainty in supply chains (Davis, 1993; Mo and Harrison, 2005). Demand uncertainty arises from volatile demand or inaccurate forecasts (Davis, 1993). Usually, this demand is correlated among different locations and various time periods (Tsiakis *et al.*, 2001; Snyder, 2006). Recently, fluctuating demand has

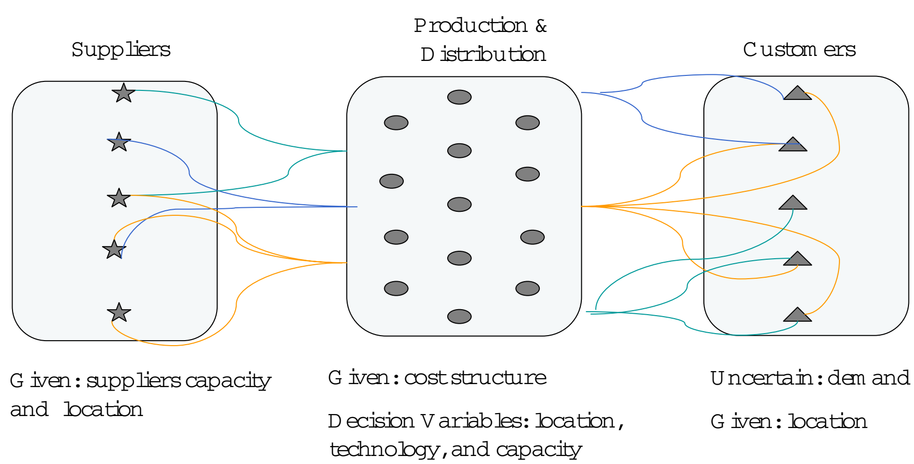

Fig. 1.: Supply chain network design problem.

been more pronounced due to the shrinking product life-cycle (Chopra and Meindle, 2007). Periodic redesign of a supply chain network maintains its efficiency in face of dynamic demand patterns (Chandra and Gragis, 2007). Fig. 2 depicts a scenario representing one possible future. Each arrow represents the reconfiguration actions that result in an evolving SCND.

- *Mimicking the natural unfolding order of events.* Fig. 3 shows the unfolding of uncertainty over time and the decision process that ensues. At each point in time, design decisions occur while future events remain uncertain. Once an uncertain event unfolds, the design can be tweaked using the information gained by the now realized event. Forfeiting the benefit of using unfolding information to improve design decisions detracts from their quality (Bienstock and Shapiro, 1988), especially as the planning horizon grows and as the variability of uncertain parameters increases (Huang and Ahmed, 2009).

Current SCND models do not integrate all these factors in a single model (Melo *et al.*, 2009). Developing and solving such an integrated model is the focus of this work.

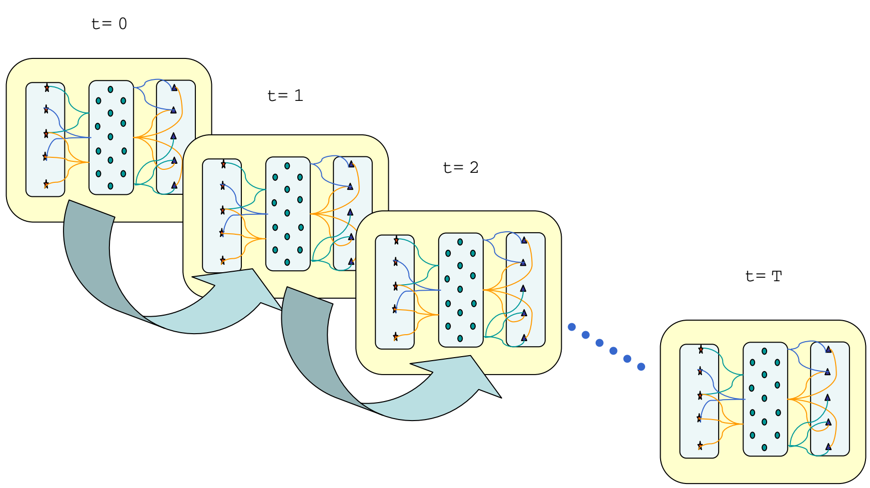

Fig. 2.: Effect of dynamic market condition on SCND problem.

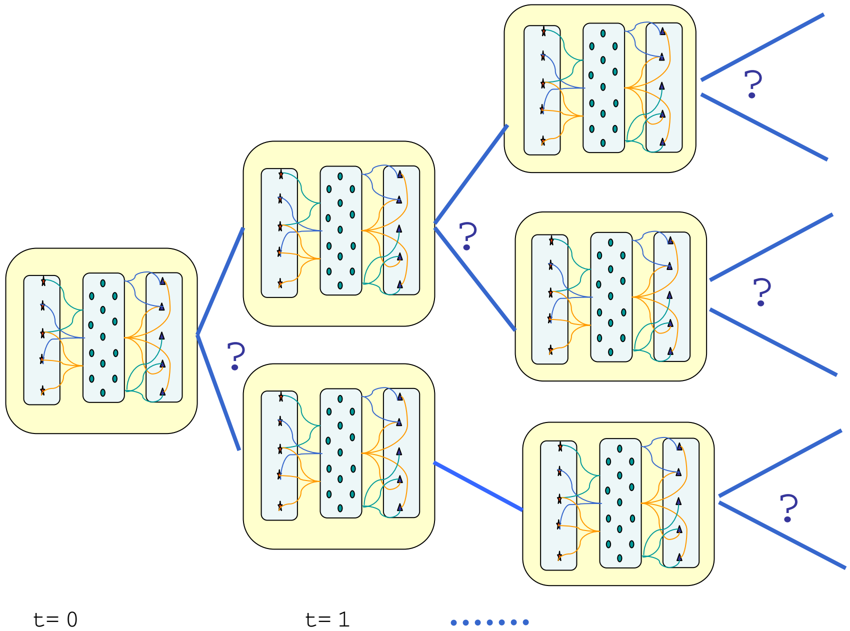

Fig. 3.: Effect of dynamic and stochastic market condition on SCND problem.

2. Solving the supply chain network design problem

Integrating all these factors in one model results in a multistage stochastic mixed-integer program (MSMIP), which has no known general purpose solution methodology (Ahmed and Sahinidis, 2003). This dissertation applies two decomposition approaches to attack this problem: end-of-horizon decomposition and nodal decomposition.

2.1. End-of-horizon decomposition

The developed end-of-horizon (EoH) decomposition exploits the *end-of-horizon effect* to produce a smaller-sized MSMIP. This resulting MSMIP is still NP-hard but its smaller size renders it easier to solve. The EoH decomposition rests on independently optimizing the SCND of every node of the last level of the scenario-tree. These nodal subproblems are NP-hard, but tackling them independently makes them easier to solve. Subsequently, the configurations prescribed by the optimal solutions of these nodal subproblems are imposed before optimizing the design decisions of the remaining nodes of the scenario-tree. The last level of the scenario-tree includes a large portion of the scenario-tree's nodes. By imposing given configurations to these nodes, this decomposition results in a smaller problem with a significantly smaller global feasible search space.

Optimizing the resulting smaller-sized model produces a SCND identical to that prescribed by a global optimal solution when the discount rate is 0%. When the discount rate is greater than 0%, a bound on the gap between the value of the globally optimal solution and that resulting from the EoH decomposition is deduced. The computational results of Chapter V indicate a 91% reduction in solution time and a 6% bound on the optimality-gap is typically less than 4%.

The EoH decomposition is not SCND specific; it applies to any (two-stage or multistage) stochastic program sensitive to the end-of-horizon effect. Its resulting smaller-sized problem can be further decomposed using any of the traditional stochastic programming decompositions (scenario, component, or nodal decompositions). To solve the developed SCND MSMIP, this dissertation applies nodal decomposition on this resulting problem.

2.2. Nodal decomposition

A scenario-tree formulation is used for SCND to allow nodal decomposition. Nodal decomposition has two advantages:

1. It provides smaller subproblems than those currently achieved by the commonly used scenario decomposition (Schultz *et al.*, 2003). SCND subproblems are still NP-hard, but the smaller size allows them to be solved more efficiently.

2. It provides a conveniently structured master problem that is amenable to refor-

mulation into a *Leontief substitution flow problem* (LSFP). This LSFP has the integrality property and thereby it voids the need for MIP solution techniques (Jeroslow *et al.*, 1992).

Together, smaller subproblems and an integral master problem have the potential to radically cut the computational effort, which allows solving models for larger, realistic problem sizes.

A SCND subproblem resulting from nodal decomposition is a two-stage stochastic mixed-integer program (SMIP) for which a multitude of efficient algorithms exist to solve small sized problems. A subproblem seeks the best SCND for a single period of the planning horizon. Design decisions (location, capacity, and technology) must be made before the realization of uncertain demand. Whereas tactical decisions (production, distribution, and subcontracting) react to the unfolding demand.

The master problem resulting from nodal decomposition is a large scale MIP. It tracks the evolution of the network's configuration throughout the planning horizon under all possible demand scenarios and thereby accounts for network reconfiguration costs. The integrality property results from reformulating the master problem as a LSFP. This property serves to prove that a column generation procedure based on the reformulated master problem features a zero duality-gap.

The LSFP reformulation is achieved by graphically representing the master problem as a *hypergraph* consisting of vertices and *hyperarcs*. A vertex represents a possible solution of a SCND subproblem. A hyperarc is a special arc that can join more than two vertices. It represents the necessary reconfiguration actions to transition from a network's configuration at a given time period to a number of potential configurations at the following time period, one configuration per possible realization of uncertain demand. Hypergraphs lend themselves to polynomial-time solution algorithms.

The proposed SCND heuristic is a column generation type-I procedure (Wilhelm, 2001) that consists of two major steps. First, each nodal subproblem generates a set of nodal solutions. These provide columns to a Leontief substitution flow master problem. Second, this restricted master problem constructs the best possible global feasible solution out of the thus far generated columns. This same heuristic still applies if the SCND problem is modeled as an SMIP (which forfeits the benefits of using gradually unfolding information to improve design decisions.) In this case, the restricted master problem is a shortest path problem.

The noteworthy success of column generation type-I procedures in tackling NP-hard problems (Wilhelm, 2001) motivated this approach. The computational results of Chapter VI indicate a 88% reduction in solution time and a 13% bound on the optimality-gap when the SCND problem is modeled as an SMIP. When the SCND is modeled as an MSMIP, results of the computational experiments of Chapter VII indicate a 98% reduction in solution time and a 6% bound on the optimality-gap.

3. Conclusion

Not only will this research provide a practical and realistic methodology to model and solve SCND problems, but will also provide the first application of nodal decomposition and LSFP in MSMIP. This research also provides a faster solution approach for SCND problems modeled as a two-stage stochastic program (which forfeits the benefits of using gradually unfolding information to improve design decisions.) Furthermore, this research capitalizes on the potential to extend this methodology beyond SCND problems by demonstrating that their underlying properties can be induced in other MSMIP applications.

This dissertation is organized as follows. Chapter II reviews relevant SCND literature. Chapter III discusses the research objectives, originality, and plan. Chapter IV models the SCND problem using a scenario-tree formulation. Chapter V develops the EoH decomposition, deduces a bound on the gap between the global optimal solution and the EoH solution, and uses computational experiments to assess the potential gain in computational efficiency and loss in solution value. Chapter VI exploits the special case of two-stage stochastic program formulation developed in Chapter IV with nodal decomposition, reformulates the resulting master problem as a shortest-path problem, and proposes a practical heuristic to design supply chains. Chapter VII extends this heuristic to tackle the multistage stochastic program developed in Chapter IV. This extension involves reformulating the master problem resulting from nodal decomposition as a shortest-hyperpath problem. Chapter VIII outlines a methodology to extend the application of the developed heuristic to problems beyond SCND. Finally, Chapter IX summarizes this research and lists its contributions.

CHAPTER II

REVIEW OF RELEVANT LITERATURE

The literature is rich with supply chain network design (SCND) models; albeit, the scope of this review is limited to stochastic models that include both the effect of economies of scale on the cost structure and the ability to correlate dynamic demand between different locations and various time periods. For reviews covering the breadth of the SCND literature, see Schmidt and Wilhelm (2000), Min and Zhou (2002), Chopra (2003), Mo and Harrison (2005), Snyder and Daskin (2007), Chandra and Gragis (2007), Goetschalckx and Fleischmann (2008), Melo *et al.* (2009), Peidro *et al.* (2009), Farahani *et al.* (2010), and Klibi *et al.* (2010).

This chapter has two primary goals: to identify gaps in the stochastic modeling of SCND problems and to show that nodal decomposition has not been implemented in solving multistage stochastic mixed-integer programs (MSMIPs).

1. Stochastic SCND models and solution approaches

Propelled by advances in computational technology, recent SCND models moved closer than ever towards realism. Nonetheless, room for improvement continues to exist. Table I shows that current stochastic SCND models incorporate either the integrated nature of location, capacity, and technology decisions or exploit the benefit of using unfolding demand over time in making these decisions—none incorporate both features. The following sections explore this gap in current SCND research.

1.1. Integrating location, capacity, and technology decisions

SCND has been segregated into several well-established research areas, which include location analysis, capacity planning, and technology selection (Verter and Dincer, 1992; Paquet *et al.*, 2004; Martel, 2005). Verter and Dincer (1992) and Verter (2002) show that these decisions are interdependent and conclude that it is vital to integrate them in a single modeling framework.

Most current stochastic SCND models do not incorporate location, capacity, and technology in a single modeling framework. For example, Tsiakis *et al.* (2001), Silva and Wood (2006), and Schütz *et al.* (2008) concentrate on facility location. Huang and Ahmed (2009) focus on capacity expansion. Eppen *et al.* (1989), Gupta *et al.* (1992), Ahmed and Sahinidis (2003), and Ahmed *et al.* (2003) integrate capacity planning and technology selection decisions.

To what I found in the literature, the only model that integrates location, capacity, and technology decisions was developed by Lucas *et al.* (1996) and later adopted by MirHassani *et al.* (2000), Lucas *et al.* (2001), and Mitra *et al.* (2006). This two-stage stochastic mixed-integer program (SMIP) models a network consisting of plants, distribution centers, and customer zones. The first stage includes all design decisions (namely opening and closing sites, setting capacity levels, and selecting technology types) for all periods of the planning horizon. The second stage decides production and transportation amounts for the entire planning horizon.

Some stochastic, yet *static*, models, such as Santoso *et al.* (2005), integrate

Table I. Amount of SCND literature classified by integration level of design decisions (location, capacity, and technology) and uncertainty modeling approach

Integration level of design decisions	Model Type		
	Deterministic	SMIP	MSMIP
Single design decision	Abundant cf. Van Roy and Erlenkotter (1982)	Moderate cf. Tsiakis *et al.* (2001)	Little cf. Huang and Ahmed (2009)
Integrating 2 decisions	Abundant cf. Melo *et al.* (2005)	Moderate cf. Eppen *et al.* (1989)	Little cf. Ahmed and Sahinidis (2003)
Integrating 3 decisions	Moderate cf. Wilhelm *et al.* (2005)	Little cf. Lucas *et al.* (2001)	**None**

Little: 1 to 5 publications; Moderate: 6 to 15 publications; Abundant: more than 15 publications.

location, capacity, and technology. Nonetheless, static models inherently preclude capacity expansion and reduction, resource replacement, and resource relocation.

1.2. Reacting to unfolding uncertainty

Stochastic SCND models fall in two groups regarding their ability to react to unfolding uncertainty. The first use two-stage stochastic programs (SMIP), and thereby their design decisions do not interact with unfolding uncertainty. This group includes the overwhelming majority of stochastic SCND models; see for example: Eppen *et al.* (1989), Louveaux and Peeters (1992), Liu and Sahinidis (1996), Lucas *et al.* (1996; 2001), MirHassani *et al.* (2000), Tsiakis *et al.* (2001), Silva and Wood (2006), Mitra *et al.* (2006), and Schütz *et al.* (2008).

The second group consists of the models of Ahmed and Sahinidis (2003) and Ahmed *et al.* (2003), which study capacity expansion under demand uncertainty with fixed-charge expansion cost. Both models adopt MSMIP and thereby include a sequence of capacity expansion (design) decisions that interacts with a sequence of realizations of the uncertain demand.

As the number of stages grows and as the variability of uncertain parameters increases, the quality of a MSMIP solution increases when compared with a SMIP solution (Huang and Ahmed, 2009). Unfortunately, the accuracy comes at a heavy computational price (Ahmed and Garcia, 2004; Sen, 2005).

1.3. Gap in stochastic SCND models

While the models of Lucas *et al.* (1996; 2001), MirHassani *et al.* (2000), and Mitra *et al.* (2006) integrate location, capacity, and technology decisions, these decisions do not benefit from the information gained gradually by the realization of uncertain events. In contrast, the design decisions in Ahmed and Sahinidis (2003) and Ahmed *et al.* (2003) interact with uncertainty but do not capture the interactions among the location, capacity, and technology of a resource.

Current literature lacks a model that extends Lucas *et al.* (1996; 2001), MirHassani *et al.* (2000), and Mitra *et al.* (2006) with the ability to respond to unfolding uncertainty; or, equivalently, a model that extends those of Ahmed and Sahinidis (2003) and Ahmed *et al.* (2003) with integrated location, capacity, and technology decisions.

In essence, this gap in SCND models results from the lack of suitable solution techniques; current SCND models "seem to be guided by the availability of solution methods" (Melo *et al.*, 2009). Consequently, improving solution techniques advances SCND research.

1.4. Solution approaches of MSMIP in the context of SCND

The literature lacks a suitable approach to solve SCND problems modeled as MSMIPs. To date, no practical general purpose solution algorithm exists for MSMIP (Schultz, 2009; Huang and Ahmed, 2009; Sen, 2005; Ahmed *et al.*, 2003). Regardless, notable attempts continue to emerge.

In theory, the pioneering Lagrangian algorithm of Carøe and Schultz (1999) applies to MSMIP. However, the authors acknowledge that “some work still remains to be done since problem sizes increase dramatically” for multistage problems (Carøe and Schultz, 1999). Consequently, this algorithm application to MSMIPs remains elusive (Ahmed *et al.*, 2003).

The innovative branch-and-fix coordination (BFC) algorithm (Alonso-Ayuso *et al.*, 2003) only suits models with *all* binary variables. When a model includes continuous variables, this algorithm becomes problematic (Schultz *et al.*, 2003).

Recently, Escudero *et al.* (2009) extends the BFC of Alonso-Ayuso *et al.* (2003)

to include continuous variables. This results in an exact approach for solving MSMIPs with *complete recourse*. However, Escudero (2009) acknowledges that this method is not suitable for realistic problem sizes due to the ensuing massive-sized BFC tree.

Escudero (2009) presents a heuristic framework aimed at solving realistically-sized problems. This heuristic rests on relaxing the integrality and non-anticipativity constraints of an MSMIP resulting in independent linear programs, each representing one scenario. Gradually fixing the variables of these linear programs to suitable values recovers the integrality and non-anticipativity properties. To date, no computational results are available for this heuristic as its implementation remains in progress (Escudero, 2009). In short, "challenging implementation issues remain" (Schultz, 2009) for this theoretical framework.

Several heuristics have been specifically tailored to solve subsets of the SCND problem. Generalizing these heuristics to tackle an integrated SCND remains elusive.

Extending the SMIP heuristics of MirHassani *et al.* (2000), Lucas *et al.* (2001), and Mitra *et al.* (2006) to MSMIP is possible but not promising. They rest on selecting the best among a set of heuristically generated configurations. These configurations result from a *wait and see* approach, which analyzes each scenario independently.

The three-stage heuristic of MirHassani *et al.* (2000) relies on detecting commonalities among promising solutions. The first step finds an optimal SCND for each individual scenario. The second step narrows down these configurations to those that perform reasonably well under all scenarios. The last step synthesizes one solution by detecting patterns in these configurations. The computational results of Mitra *et al.* (2006) show that a solution that performs well on a subset of scenarios needs not perform well on the entire set. This suggests that the first step might not be always able to choose globally promising configurations.

Lucas *et al.* (2001) and Mitra *et al.* (2006) approximate the solution space of the original problem with a set of heuristically generated configurations. Generating candidate configurations rests on a *wait and see* analysis coupled with the Lagrangian relaxation of capacity constraints. The *complete-recourse* structure of this model allows extending the configuration prescribed by each iteration of the relaxed problem into a feasible solution (with respect to the SMIP). The final step evaluates these configurations and selects the best among them. The computational results show that the first step is computationally efficient while the last step is exceptionally demanding. In MSMIP, the increased number of binary variables, and hence possible configurations, can render this last step computationally prohibitive.

Extending the application-dependent, MSMIP algorithms of Ahmed and Sahinidis (2003) and Ahmed *et al.* (2003) to SCND problems is problematic. The rounding heuristic of Ahmed and Sahinidis (2003) requires nondecreasing demand over time (which precludes capacity reduction). The Branch-and-Bound algorithm of Ahmed *et al.* (2003) relies on the model's tight lower bound. This bound results from the tight linear relaxation achieved by reformulating capacity expansion as a lot sizing problem (which precludes resource replacement). In this clever reformulation, capacity expansion sizes are cast as linear batch sizes (which precludes discrete expansion sizes), and capacity investment cost is cast as the batch setup cost (which precludes fixed and variable operating costs).

Based on the same principles, Huang and Ahmed (2009) tried a different tack. Their approach cannot be extended to SCND problems for the same reasons.

2. Decomposition schemes for multistage stochastic programs

Among the three decomposition schemes that have been proposed for MSMIP (see Römisch and Schultz (2001) for a review of MSMIP decomposition schemes), all current SCND models that I am aware of use *scenario decomposition*. Little incentive exists to adopt *component decomposition* in SCND models. This decomposition is beneficial only when the decision space dominates the *component coupling constraints* (Römisch and Schultz, 2001). In SCND problems, the decision space is typically dominated by constraints expressing logistical details (such as those enforcing the conservation of material flow and capacity limits).

Römisch and Schultz (2001) observe that *nodal decomposition* has never been used in MSMIP. Consistent with their observation, I am unaware of any SCND model that adopts this decomposition. The perceived weakness of nodal decomposition stems from its large duality-gap compared to that of scenario decomposition (Dentcheva and Römisch, 2004). In SCND problems, where the subproblems are NP-hard, closing a large duality-gap can be computationally prohibitive (Wilhelm, 2001).

3. Conclusion

This chapter highlights the need for the following research advances:

1. A SCND MSMIP that integrates location, capacity, and technology decisions

2. An algorithm capable of solving such a model

It also identifies nodal decomposition as an unexplored technique for MSMIP.

This research aims to fill this need with the model and solution algorithm proposed in Chapter III and developed in Chapter IV and Chapter V.

CHAPTER III

RESEARCH QUESTION

This research expands the ongoing research in supply chain network design (SCND) with a model that has more fidelity to the actual problem than current models and to develop a practical solution methodology for this model. The objectives, originality, and plan of this research are summarized in the following sections.

1. Research objectives

This research has three main objectives:

1. Formulate the SCND problem as a multistage stochastic mixed-integer program (MSMIP) with a structure that can be exploited for solution effectiveness while still capturing the essential trade offs encountered in SCND.
2. Develop a practical solution methodology for this model.
3. Characterize other applications beyond SCND that have the potential to benefit from the developed solution methodology, and establish guidelines to formulate these applications as MSMIPs with structures amenable to this methodology.

2. Research originality

This research is unique in many ways. Unlike current models, my model includes all of the following characteristics:

- Integrate the location, capacity, and technology attributes of a resource.
- Capture the effect of the economies of scale on the cost structure.

- Allow for multiple products and multiple levels of supply chain hierarchy.
- Model the stochastic and dynamic natures of demand.
- Mimic the natural unfolding of events.

The mathematical formulation of this model is unique. It differs from those of Lucas *et al.* (1996), MirHassani *et al.* (2000), Lucas *et al.* (2001), and Mitra *et al.* (2006) in how resources are modeled. In my model, a set of resources that perform a specific function are modeled as a single unit. This provides the formulation with the following advantages:

1. Inclusion of miscellaneous costs associated with retooling a site (such as those related to layout, lighting, and wiring modifications) and savings instigated by the effect of economies of scale in capacity procurement.
2. A mathematical structure amenable to exploitation for solution efficiency.

My representation of resources relates mostly to that of Eppen *et al.* (1989). However, my model provides more details in SCND representation and less focus on the financial concerns associated with capacity expansion.

The decomposition of this model is unique. This research presents the first application of nodal decomposition in both SCND and MSMIP.

The proposed solution approach is novel. Conceptually, it reformulates the master problem to avoid the duality-gap, which is a departure from the stochastic programming tradition of efficiently closing the duality-gap. Technologically, it develops the first application of Leontief substitution flow problems (LSFPs) in MSMIP.

Like Lucas *et al.* (2001) and Mitra *et al.* (2006), my solution approach exploits the successive solutions generated by an iterative procedure to construct feasible configurations (which serve as columns in the master problem). In contrast to their Lagrangian-based algorithm, my approach adopts the L-shaped method (Van Slyke and Wets, 1969). Unlike Lucas *et al.* (2001) and Mitra *et al.* (2006), my master problem is computationally efficient. This results from two ideas:

1. Applying an end-of-horizon decomposition that significantly reduces the size of the master problem. This decomposition optimizes subsets of the master problem and then inserts these optimal solutions as parameters into the master problem.

2. Using the generated columns to populate a Leontief substitution flow master problem (or a shortest path problem in the case of SMIP), which exhibits the integrality property. This property renders the master problem easy to solve.

3. Approach

The major steps of this dissertation are as follows:

1. Formulate the SCND problem as a MSMIP with a structure amenable to solution effectiveness while still capturing the essential trade-offs of SCND.

2. Derive the SMIP formulation of the SCND problem as a special case of the MSMIP formulation.

3. Apply nodal decomposition on both the SMIP and MSMIP to decompose each of them into a conveniently structured master problem and (relatively) small-sized nodal subproblems.

4. Develop end-of-horizon decomposition and apply it to reduce the sizes of these master problems (without altering their structures.)

5. As a stepping stone towards a solution approach for the MSMIP formulation, develop a practical solution approach for the SMIP formulation. This results from following these steps: first, exploit the structure of the reduced master problem with a shortest path reformulation, which provides the integrality property; second, use the L-shaped method to optimize the subproblems and thereby provide columns for the reformulated master problem; fourth, develop a heuristic inspired from column generation type I approach; and finally, conduct computational experiences to evaluate the effectiveness of this heuristic procedure.

6. Adapt the developed solution procedure to suit the MSMIP formulation. This results from following the same steps except for replacing the shortest path reformulation by a LSFP reformulation, which also provides the integrality property. Afterwards, tailor existing LSFP's polynomial-time algorithms to suit the special structure of the reformulated master problem. Finally, conduct computational experiences to evaluate the effectiveness of this heuristic procedure.

7. Identify the characteristics that render MSMIPs amenable to nodal decomposition and LSFP reformulation, and characterize other applications likely to benefit from the developed methodology.

CHAPTER IV

SUPPLY CHAIN NETWORK DESIGN MODEL

This chapter models the supply chain network design (SCND) problem with a multi-stage stochastic mixed-integer program (MSMIP). This model aims to plan resource acquisition by picking those resources with location, capacity, and technology that best complement the existing supply chain network structure. The model does so by implicitly exploring the common features of the different forecasted demand scenarios to develop a design that works reasonably well under various scenarios or a design prone to tweaking at a future point in time where the future may be clearer.

This chapter is organized as follows. First, the SCND problem and major modeling assumptions are stated. Second, the general structure of the model is outlined. Third, the decision variables are described. Fourth, the constraints and objective function are developed. Fifth, a compact representation of the model is presented. Finally, how to reduce this formulation to model special cases of the SCND problem is illustrated.

1. Model's foundation and general structure

The developed model adopts a scenario-tree formulation (Römisch and Schultz, 2001) for the SCND problem. The following sections discuss the characteristics of the SCND problem, explain how the supply chain network and scenario-tree are expressed in the model, and present an overview of the model's general structure.

1.1. Supply chain network design problem

The SCND problem aims to determine the number, location, capacity, and technology of the supply chain's facilities that minimize the expected long run cost of the network. The supply chain under consideration has the following characteristics:

- A single entity owns and controls all manufacturing and distribution resources.
- Design decisions adapt a known initial structure of the supply chain network (which could be null) to achieve better long run efficiency.
- The length of the planning horizon is predetermined, and, at the beginning of each of its periods, the design of the supply chain network can be adjusted. Both the length of the horizon and the duration of each period are application-dependent. Typically the planning horizon ranges between five to ten years, and the duration of each period ranges between one to three years.
- The candidate supply chain partners (i.e., suppliers and target customer segments) are predefined, and their locations are known. The suppliers' capacity and customers' demand forecasts are provided. These forecasts define a probability density function of possible scenarios. Every scenario defines the location and amount of demand for each product family throughout the planning horizon.
- Products are shipped from the suppliers to a series of manufacturing and distribution facilities, then to the customers. Each shipping channel and product family combination is associated with a per unit shipping cost.
- Manufacturing and distribution facilities are located in a multi-level hierarchy. Any given product might need to visit multiple facilities if the operations it needs are fragmented into different facilities.
- The sets of promising locations, technologies, and capacities are provided.

- A change of network configuration (by establishing a new facility, closing an existing one, or retooling an existing facility with different resources) is associated with a one-time fixed cost. This cost depends on the nature and timing of the event and on the location, technology, and capacity of that facility.

- An open facility is associated with a recurrent fixed cost that is time, location, capacity, and technology dependent. Moreover, processing a product involves a per unit cost that is time, location, capacity and technology dependent.

- Fulfilling customers' demand of each product family generates a per unit revenue that is time and market dependent.

- No inventory is held from one time period to the next; i.e., customer demand at a given period must be satisfied by products processed during that same period.

These characteristics have the following implications:

- The problem is a hierarchical, multi-commodity, dynamic, and stochastic supply chain network design.

- The hierarchical and multi-commodity aspects of the problem are intertwined. The different product families compete for the finite capacity installed in the different hierarchical levels. Thus, the flow of the different product families links successive levels. This linkage becomes especially strong as the number of product families grows. Likewise, the finite capacity of individual resources links the flow of the different product families. This linkage becomes especially strong as the number of hierarchical levels (and thereafter the number of candidate resources) grows.

- The dynamic and stochastic aspects of the problem are intertwined. At each time period, several scenarios exist. To adequately describe a point in the future, one needs to specify both its corresponding time period and its associated anticipated sequence of uncertain demand outcomes.

- Periodically tweaking a SCND links successive time periods since each time period inherits the SCND of its predecessor. Absent this consideration, each time period can be dealt with in isolation of all other periods.

1.2. Modeling the supply chain network

The ownership of and control over a supply chain network is divided among three contributors collaborating to fulfill customers' demand: preselected supplier zones ($s \in \mathcal{S}$) that provide the inputs; predefined customer zones ($k \in \mathcal{K}$) that consume the outputs; and plants and distribution centers that process the inputs received from the suppliers into the outputs provided to the customer zones.

Both plants and distribution centers are modeled the same way: material arrives, value is added, and product exits. As such, they will be collectively referred to as *facilities*. A facility has three attributes: location, technology, and capacity level.

A facility's location is selected from a predefined set of candidate sites ($j \in \mathcal{J}$). The set of candidate sites for plants ($j \in \mathcal{J}_r$) and that for distribution centers ($j \in \mathcal{J}_w$) can intersect. This allows for co-locating a plant and a distribution center.

Each open facility is fitted with exactly one technology. In this context, a technology ($q \in \mathcal{Q}_j$) is a group of resources that enables a facility to perform its particular function. Examples for technologies include assembly lines, packaging lines, storage/retrieval systems, etc. A specific technology can process, at different rates, a subset of the product families.

Each technology comes in discrete capacity levels ($\ell \in \mathcal{L}_q$). That is, this model does not explicitly mix-and-match equipment to achieve suitable capacity; it rather picks one whole package among available capacity options. This representation allows incorporating the effect of economies of scale on resource acquisition cost and accounting for miscellaneous costs associated with retrofitting sites (such as those related to layout, lighting, and wiring modifications).

The existing supply chain network structure serves as the initial configuration (which can be null). Adapting this configuration to an evolving business environment is the focus of this model. This entails opening new facilities, retooling existing facilities, adjusting the capacities of installed technologies, and closing existing facilities. Opening a new facility involves selecting its location and fitting it with a technology at some suitable capacity level. Retooling an existing facility involves replacing its technology. Capacity upgrade and downgrade of a given technology result from acquiring and shedding units of the same technology group. Closing a facility involves removing the technology within.

1.3. Modeling the planning horizon

The planning horizon is approximated by a finite set of discrete periods $\mathcal{T} = \{0, 1, \cdots, T\}$. These periods can be of equal or different length. $t = 0$ indicates the initial configuration of the supply chain network (which could be null).

The end of the planning horizon warrants special treatment to circumvent the *end-of-horizon effect*. The end-of-horizon effect refers to a model's bias against acquiring new resources as the remaining portion of the planning horizon becomes insufficient to recoup investment expenditures. Chapter V addresses this phenomenon.

1.4. Modeling the scenario-tree

This model approximates customer demand by scenarios. A scenario is the "joint realization of uncertain parameters over all time periods" (Ahmed and Sahinidis, 2003). Scenarios are defined on a finite probability space (Ω, Ξ, ϕ), where Ω is a finite sample space, Ξ is a power set on Ω, and ϕ is a probability measure. $\omega = \{\omega^1, \ldots, \omega^T\}$ is a multi-dimensional sample point of Ω. The probability that a scenario ω will realize, $\phi(\{\omega\})$, is provided for all forecasted scenarios $\omega \in \Omega$. Scenario generation is an active research field by itself. For example, Dupačová *et al.* (2000) and Hoyland and Wallace (2001) developed techniques to generate scenarios for multistage stochastic programs.

In scenario-trees, a node represents a decision point and an arc represents a specific realization of the uncertain event. A scenario ω is represented by the unique path leading from the root node to a leaf node $n_\omega \in \mathcal{N}_T$, where $\mathcal{N}_T \subset \mathcal{N}$ is the set of leaf nodes. Accordingly, each leaf node represents a scenario, while the root node $(n = 0)$ represents the inherited design of the network. The paths of different scenarios are not necessarily exclusive throughout the planning horizon—scenarios can share a common history before they branch into diverging paths.

Each scenario-tree node is associated with a bundle. A bundle $\mathcal{B}_n$ is the set of scenarios passing through node $n \in \mathcal{N}$. For example, at the root node $\mathcal{B}_0 = \Omega$, and at a leaf node $n_\omega \in \mathcal{N}_T$ corresponding to scenario ω, $\mathcal{B}_{n_\omega} = \{\omega\}$.

Each node of the scenario-tree is associated with a nodal probability and an arc probability. The *nodal probability* of node n, ϕ_n, is the probability of reaching scenario-tree node n starting from the root node. It is expressed in terms of scenarios' probability as follows: $\phi_n = \phi(\{\omega | \omega \in \mathcal{B}_n\}) = \sum_{\omega \in \mathcal{B}_n} \phi(\{\omega\})$. The *arc probability* of node n, $\phi_{n|a(n)}$, is the probability to reach $n \in \mathcal{N}$ given that its parent,

$a(n) \in \mathcal{N}$, has been reached (i.e., the probability to select the arc leading to n). The arc probability of a node n can be computed as follows:

$$\phi_{n|a(n)} = \frac{\phi(\{\omega|\omega \in \mathcal{B}_n \cap \mathcal{B}_{a(n)}\})}{\phi(\{\omega|\omega \in \mathcal{B}_{a(n)}\})} = \frac{\phi(\{\omega|\omega \in \mathcal{B}_n\})}{\phi(\{\omega|\omega \in \mathcal{B}_{a(n)}\})} = \frac{\phi_n}{\phi_{a(n)}}.$$

For notational convenience, let $\mathcal{N} = \{1, 2, \ldots, N\}$; i.e., $\{0\} \nsubseteq \mathcal{N}$. Also, let $\mathcal{N}_B = \mathcal{N} \setminus \mathcal{N}_T$. Furthermore, let $\mathcal{D}(n) \subset \mathcal{N}$ be the set of immediate descendants of $n \notin \mathcal{N}_T$ (i.e., $n \in \{0\} \cup \mathcal{N}_B$), and $\mathcal{D}(n) = \emptyset$ for $n \in \mathcal{N}_T$.

1.5. Model's general structure

The model aims to find values of the strategic decision variables (x, y) that minimize the expected cost (4.1) over all possible scenarios $\omega \in \Omega$. The optimal values of the strategic variables are related to those of the tactical variables (z) by the *component-coupling constraints* (4.3), which characterize a decision's feasibility for any given period $t \in \mathcal{T}$. The relationship between decisions at different periods of the planning horizon are restricted by the *stage-coupling constraints* (4.2).

$$\kappa = \min\ E_{\omega \in \Omega}\left[\kappa_1(x) + \kappa_2(y) + \kappa_3(z)\right], \tag{4.1}$$

subject to

$$\varphi_1(x, y, \omega) = 0, \quad x \in X, \tag{4.2}$$

$$\varphi_2(y, z, \omega) = 0, \quad y \in Y. \tag{4.3}$$

2. Decision variables

The goal of the model is to determine the *present* design decisions that minimize the *long-term* expected cost of the network. Future uncertainty and future decisions (strategic and tactical) take part in the model to assess the impact of present design decisions on the long-term performance of the network.

2.1. Strategic variables

Strategic decisions describe the evolution of the supply chain network over the planning horizon. At each node of the scenario-tree, strategic decisions are resolved before the realization of the uncertain parameters. Strategic decisions fall into two categories: configuration selection and reconfiguration planning.

Configuration selection answers the following question: which technology ($q \in \mathcal{Q}_j$) operates at which capacity level ($\ell \in \mathcal{L}_q$) in which site ($j \in \mathcal{J}$)? The model selects a configuration for each scenario-tree node except for the root node (for which a given, inherited configuration is imposed) and the leaf nodes (which are terminal and don't emanate further scenarios). Thus, configuration selection decisions are indexed over $\mathcal{N}_B$.

Configuration selection involves location, technology, and capacity aspects:

- Location selection ($y_j^n, j \in \mathcal{J}, n \in \mathcal{N}_B$) indicates whether site j houses an operational facility.

- Technology selection ($y_{0,q,j}^n, q \in \mathcal{Q}_j, j \in \mathcal{J}, n \in \mathcal{N}_B$) indicates whether the operational facility at site j uses technology q.

- Capacity level selection ($y_{\ell,q,j}^n, \ell \in \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J}, n \in \mathcal{N}_B$) indicates whether level ℓ is the capacity level of technology q that operates in site j.

For each node $n \in \mathcal{N}_B$, variables pertaining to configuration selection are defined as follows:

$$y_j^n = \begin{cases} 1, & \text{if site } j \text{ houses an open facility,} \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{0,q,j}^n = \begin{cases} 1, & \text{if technology } q \text{ is \textit{not used} in site } j, \\ 0, & \text{otherwise.} \end{cases}$$

$$y_{\ell,q,j}^n = \begin{cases} 1, & \text{if technology } q \text{ operates at capacity level } \ell \text{ in site } j, \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{y_n}$ groups these variables into one vector:

$$\mathbf{y_n} = \left(\{y_j^n\}_{j\in\mathcal{J}}, \{y_{\ell,q,j}^n\}_{\ell\in\{0\}\cup\mathcal{L}_q, q\in\mathcal{Q}_j, j\in\mathcal{J}}\right).$$

Reconfiguration planning answers the following question: what actions are needed to update configuration $\mathbf{y_{a(n)}}$ into $\mathbf{y_n}$? Therefore, reconfiguration planning is also indexed over $n \in \mathcal{N}_B$ and involves location, technology, and capacity aspects:

- Relocation planning indicates whether to establish a new facility in site j ($x_{\text{open } j}^n$) and whether to dismantle the existing facility in site j ($x_{\text{close } j}^n$).
- Technology planning indicates whether to install technology q in the facility at site j ($x_{0,\ell,q,j}^n$) and whether to remove technology q from the facility at site j ($x_{\ell,0,q,j}^n$).
- Capacity planning ($x_{\ell_1,\ell_2,q,j}^n$) indicates whether to upgrade/downgrade the capacity of technology q that operates in site j from level $\ell_1 \in \mathcal{L}_j$ to level $\ell_2 \in \mathcal{L}_j$, where $\ell_1 \neq \ell_2$.

For each node $n \in \mathcal{N}_B$, variables pertaining to configuration selection are defined as follows.

$$x^n_{\text{open } j} = \begin{cases} 1, & \text{if site } j \text{ is opened,} \\ 0, & \text{otherwise.} \end{cases}$$

$$x^n_{\text{close } j} = \begin{cases} 1, & \text{if site } j \text{ is closed,} \\ 0, & \text{otherwise.} \end{cases}$$

$$x^n_{0,\ell,q,j} = \begin{cases} 1, & \text{if technology } q \text{ is installed at level } \ell \text{ in site } j, \\ 0, & \text{otherwise.} \end{cases}$$

$$x^n_{\ell,0,q,j} = \begin{cases} 1, & \text{if technology } q \text{ operating at level } \ell \text{ is removed from site } j, \\ 0, & \text{otherwise.} \end{cases}$$

$$x^n_{\ell_1,\ell_2,q,j} = \begin{cases} 1, & \text{if capacity level of technology } q \text{ in site } j \text{ is adjusted from } \ell_1 \text{ to } \ell_2, \\ 0, & \text{otherwise.} \end{cases}$$

$\mathbf{x_n}$ groups these variables into one vector:

$$\mathbf{x_n} = \left(\{x^n_{\text{open } j}\}_{j\in\mathcal{J}}, \{x^n_{\text{close } j}\}_{j\in\mathcal{J}}, \{x^n_{\ell_1,\ell_2,q,j}\}_{\ell_1\neq\ell_2\in\{0\}\cup\mathcal{L}_j, q\in\mathcal{Q}_j, j\in\mathcal{J}}\right).$$

2.2. Tactical variables

Tactical decisions describe the production and distribution of products ($p \in \mathcal{P}$) at a given period of the planning horizon. At each node of the scenario-tree, tactical decisions are resolved after the realization of the uncertain parameters. Tactical decisions are continuous, and they fall into two categories: transportation decisions and processing decisions.

Transportation decisions represent the amount of each product shipped between entities of the supply chain. z^n_{p,s,j_r} is the amount of product p shipped from supplier s to manufacturing location j_r. z^n_{p,j_r,j_w} is the amount of product p shipped from manufacturing location j_r to distribution location j_w. $z^n_{p,j_w,k}$ is the amount of product p shipped from distribution location j_w to market k.

Processing decisions represent the amount of each product processed in a given facility. z^n_{p,ℓ_r,q_r,j_r} is the amount of product p processed at manufacturing location j_r by manufacturing technology q_r that operates at capacity level ℓ_r. z^n_{p,ℓ_w,q_w,j_w} is the amount of product p processed at distribution location j_w by distribution technology q_w that operates at capacity level ℓ_w.

$\mathbf{z_n}$ groups these variables in one vector:

$$\mathbf{z_n} = \left(\{z^n_{p,s,j_r}\}_{p,s,j_r}, \{z^n_{p,j_r,j_w}\}_{p,j_r,j_w}, \{z^n_{p,j_w,k}\}_{p,j_w,k}, \{z^n_{p,\ell,q,j}\}_{p,\ell,q,j}\right),$$

where p, s, j_r stands for $p \in \mathcal{P}, s \in \mathcal{S}, j_r \in \mathcal{J}_r$; p, j_r, j_w stands for $p \in \mathcal{P}, j_r \in \mathcal{J}_r, j_w \in \mathcal{J}_w$; p, j_w, k stands for $p \in \mathcal{P}, j_w \in \mathcal{J}_w, k \in \mathcal{K}$; and p, ℓ, q, j stands for $p \in \mathcal{P}, \ell \in \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J} = \mathcal{J}_r \cup \mathcal{J}_w$.

3. Constraints

The constraints fall into two groups: stage-coupling constraints and component-coupling constraints.

3.1. Stage-coupling constraints

Stage-coupling constraints ensure that a supply chain acquires all the resources that it uses. They do so by enforcing reconfiguration actions to update the configuration at node $a(n)$ into node n whenever these configurations are not identical. These reconfiguration actions involve either opening and closing sites as constraint sets

(4.4)–(4.11) dictate or retooling facilities as constraint sets (4.12)–(4.15) dictate.

Constraint sets (4.4)–(4.11) regulate opening sites to house new facilities. Constraint set (4.4) acquires the site that houses a newly established facility. Constraint sets (4.5) and (4.6) prevent unnecessary opening of sites. Constraint set (4.7) enforces the binary nature of $x^n_{\text{open } j}$.

$$x^n_{\text{open } j} \geq y^n_j - y^{a(n)}_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \tag{4.4}$$

$$x^n_{\text{open } j} \leq y^n_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \tag{4.5}$$

$$x^n_{\text{open } j} \leq 1 - y^{a(n)}_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \tag{4.6}$$

$$x^n_{\text{open } j} \in \{0, 1\} \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B. \tag{4.7}$$

Constraint sets (4.8)–(4.11) regulate closing sites that no longer house operating facilities. Constraint set (4.8) closes a site whenever its facility ceases being operational. Constraint sets (4.9) and (4.10) prevent unnecessary closing of sites. Constraint set (4.11) enforces the binary nature of $x^n_{\text{close } j}$.

$$x^n_{\text{close } j} \geq y^{a(n)}_j - y^n_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \tag{4.8}$$

$$x^n_{\text{close } j} \leq 1 - y^n_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \tag{4.9}$$

$$x^n_{\text{close } j} \leq y^{a(n)}_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \tag{4.10}$$

$$x^n_{\text{close } j} \in \{0, 1\} \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B. \tag{4.11}$$

Constraint sets (4.12)–(4.15) regulate retooling facilities. When $\ell_1 = 0$, constraint set (4.12) ensures that a facility uses a technology in production/distribution operations only after this technology has been installed. Likewise, it ensures that non-operational technologies are removed from a facility when $\ell_2 = 0$. When $\ell_1 \neq 0$ and $\ell_2 \neq 0$, constraint sets (4.12) enforce upgrading/downgrading the capacity level of a technology. Constraint sets (4.13) and (4.14) prevent unnecessary retooling actions. Constraint set (4.15) enforces the binary nature of $x^n_{\ell_1,\ell_2,q,j}$.

$$x^n_{\ell_1,\ell_2,q,j} \geq y^n_{\ell_2,q,j} + y^{a(n)}_{\ell_1,q,j} - 1 \qquad \forall\, \ell_1 \neq \ell_2 \in \{0\} \cup \mathcal{L}_q,\, q \in \mathcal{Q}_j,\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \quad (4.12)$$

$$x^n_{\ell_1,\ell_2,q,j} \leq y^n_{\ell_2,q,j} \qquad \forall\, \ell_1 \neq \ell_2 \in \{0\} \cup \mathcal{L}_q,\, q \in \mathcal{Q}_j,\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \quad (4.13)$$

$$x^n_{\ell_1,\ell_2,q,j} \leq y^{a(n)}_{\ell_1,q,j} \qquad \forall\, \ell_1 \neq \ell_2 \in \{0\} \cup \mathcal{L}_q,\, q \in \mathcal{Q}_j,\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \quad (4.14)$$

$$x^n_{\ell_1,\ell_2,q,j} \in \{0,1\} \qquad \forall\, \ell_1 \neq \ell_2 \in \{0\} \cup \mathcal{L}_q,\, q \in \mathcal{Q}_j,\, j \in \mathcal{J},\, n \in \mathcal{N}_B. \quad (4.15)$$

3.2. Component-coupling constraints

Component coupling constraints confine the capacity consumed in producing and distributing products within the capacity limits of available resources. Constraint set (4.16) dictates that an open facility has exactly one technology and a closed one has none. Constraint set (4.17) keeps track of non-operational technologies. Constraint set (4.18) enforces the binary nature of $y^n_{\ell,q,j}$.

$$\sum_{q \in \mathcal{Q}_j} \sum_{\ell \in \mathcal{L}_q} y^n_{\ell,q,j} = y^n_j \qquad \forall\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \quad (4.16)$$

$$\sum_{\ell \in \{0\} \cup \mathcal{L}_q} y^n_{\ell,q,j} = 1 \qquad \forall\, q \in \mathcal{Q}_j,\, j \in \mathcal{J},\, n \in \mathcal{N}_B, \quad (4.17)$$

$$y^n_{\ell,q,j} \in \{0,1\} \qquad \forall\, \ell \in \{0\} \cup \mathcal{L}_q,\, q \in \mathcal{Q}_j,\, j \in \mathcal{J},\, n \in \mathcal{N}_B. \quad (4.18)$$

Constraint sets (4.19)–(4.29) involve the tactical variables, $\mathbf{z_m}, m \in \mathcal{D}(n), n \in \mathcal{N}_B$. These constraints help assess the impact of the strategic decisions on the performance of the network. $\mathbf{z_1}$ depends only on the inherited initial configuration (which is not a decision variable) and thereby is not impacted by the strategic decisions. Consequently, constraint sets (4.19)–(4.29) are indexed over $n \in \mathcal{N} \setminus \{1\}$.

Constraint sets (4.19) and (4.20) enforces the capacities of production and distribution technologies; where $c_{p,\ell,q}$ is the portion of the capacity ℓ of technology q required to process one product of family $p \in \mathcal{P}$. Constraint sets (4.21) and (4.22) relate the processing decisions to the transportation decisions.

$$\sum_{p} c_{p,\ell,q} z^{n}_{p,\ell,q,j_r} \leq y^{a(n)}_{\ell,q,j} \qquad \forall\, \ell \in \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J} = \mathcal{J}_r \cup \mathcal{J}_w, n \in \mathcal{N} \setminus \{1\}, \quad (4.19)$$

$$z^{n}_{p,\ell,q,j} \geq 0 \qquad \forall\, p \in \mathcal{P}, \ell \in \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J} = \mathcal{J}_r \cup \mathcal{J}_w, n \in \mathcal{N} \setminus \{1\}, \qquad (4.20)$$

$$\sum_{s \in \mathcal{S}} z^{n}_{p,s,j_r} = \sum_{\ell \in \mathcal{L}_q} \sum_{q \in \mathcal{Q}_{j_r}} z^{n}_{p,\ell,q,j_r} \qquad \forall\, p \in \mathcal{P}, j_r \in \mathcal{J}_r, n \in \mathcal{N} \setminus \{1\}, \qquad (4.21)$$

$$\sum_{k \in \mathcal{K}} z^{n}_{p,j_w,k} = \sum_{\ell \in \mathcal{L}_q} \sum_{q \in \mathcal{Q}_{j_w}} z^{n}_{p,\ell,q,j_w} \qquad \forall\, p \in \mathcal{P}, j_w \in \mathcal{J}_w, n \in \mathcal{N} \setminus \{1\}. \qquad (4.22)$$

Constraint sets (4.23) and (4.24) enforce the balance of flow at production and distribution locations, respectively.

$$\sum_{s} z^{n}_{p,s,j_r} = \sum_{j_w} z^{n}_{p,j_r,j_w} \qquad \forall\, p \in \mathcal{P}, j_r \in \mathcal{J}_r, n\mathcal{N} \setminus \{1\}, \qquad (4.23)$$

$$\sum_{j_r} z^{n}_{p,j_r,j_w} = \sum_{k} z^{n}_{p,j_w,k} \qquad \forall\, p \in \mathcal{P}, j_w \in \mathcal{J}_w, n\mathcal{N} \setminus \{1\}. \qquad (4.24)$$

Constraint set (4.25) enforces the capacity limits of suppliers, $d^{n}_{p,s}$. Constraint set (4.26) restricts the flow of products to markets such that supply doesn't exceed demand.

$$\sum_{j_r} z^{n}_{p,s,j_r} \leq d^{n}_{p,s} \qquad \forall\, p \in \mathcal{P}, s \in \mathcal{S}, n\mathcal{N} \setminus \{1\}, \qquad (4.25)$$

$$\sum_{j_w} z^{n}_{p,j_w,k} \leq d^{n}_{p,k} \qquad \forall\, p \in \mathcal{P}, k \in \mathcal{K}, n\mathcal{N} \setminus \{1\}. \qquad (4.26)$$

Constraints sets (4.27) to (4.29) impose the non-negativity of processing and transportation variables.

$$z^{n}_{p,s,j_r} \geq 0 \qquad \forall\, p \in \mathcal{P}, s \in \mathcal{S}, j_r \in \mathcal{J}_r, n\mathcal{N} \setminus \{1\}, \qquad (4.27)$$

$$z^{n}_{p,j_r,j_w} \geq 0 \qquad \forall\, p \in \mathcal{P}, j_r \in \mathcal{J}_r, j_w \in \mathcal{J}_w, n\mathcal{N} \setminus \{1\}, \qquad (4.28)$$

$$z^n_{p,j_w,k} \geq 0 \qquad \forall\, p \in \mathcal{P}, j_w \in \mathcal{J}_w, k \in \mathcal{K}, n\mathcal{N} \setminus \{1\}. \tag{4.29}$$

4. Objective function

The objective function minimizes the total expected cost. The total cost function consists of three terms: supply chain network restructuring cost, $\kappa_1(x)$; fixed operating cost, $\kappa_2(y)$; and variable production/distribution cost, $\kappa_3(z)$.

4.1. Supply chain network restructuring cost

Restructuring cost involves location, technology, and capacity aspects. Establishing a new facility in site j involves a one-time cost ($f^n_{\text{open } j}$). Closing the facility in site j involves a one-time cost ($f^n_{\text{close } j}$). Likewise, a one-time cost ($f^n_{0,\ell,q,j}$) is incurred to install a new technology q in facility j. This cost accounts for all fixed costs associated with the acquisition and installation of technology q and capacity-dependent costs associated with acquiring capacity level ℓ. Similarly, removing an existing technology q from the facility in site j results in a one-time cost ($f^n_{\ell,0,q,j}$), which depends on technology q, capacity level ℓ, and location j. Finally, upgrading/downgrading the capacity level of technology q from level ℓ_1 to level ℓ_2 involves a one-time cost ($f^n_{\ell_1,\ell_2,q,j}$) that depends on location, technology and capacity.

Relation (4.30) combines these costs, where $\mathbf{f_n x_n}$ is the cost to update the supply chain network from its configuration at node $a(n)$ to that of node n, and $\mathbf{f_n}$ groups all restructuring parameters for node n in one vector. Equation (4.30) aggregates this cost over relevant scenario-tree nodes.

$$\begin{aligned}\kappa_1(x) &= \sum_{n\in\mathcal{N}_B}\sum_{j\in\mathcal{J}} f^n_{\text{open } j} x^n_{\text{open } j} + \sum_{j\in\mathcal{J}} f^n_{\text{close } j} x^n_{\text{close } j} + \sum_{j\in\mathcal{J}}\sum_{q\in\mathcal{Q}_j}\sum_{\ell\in\{0\}\cup\mathcal{L}_q} f^n_{\ell_1,\ell_2,q,j} x^n_{\ell_1,\ell_2,q,j} \\ &= \sum_{n\in\mathcal{N}_B} \mathbf{f_n x_n}. \end{aligned} \tag{4.30}$$

4.2. Fixed operating cost

An operational facility results in a fixed, periodically-recurrent cost such as insurance, maintenance, and labor wages. This fixed operating cost ($g^n_{\ell,q,j}$) differs from one period to another and depends on a facility's technology, capacity, and location. Once a facility is closed, this cost ceases. Equation (4.31) neglects the fixed operating cost at the beginning of the planning horizon since the imposed initial configuration fixes this cost to a constant.

$$\kappa_2(y) = \sum_{n \in \mathcal{N}_B} \sum_{j \in \mathcal{J}} \sum_{q \in \mathcal{Q}_j} \sum_{\ell \in \mathcal{L}_q} g^n_{\ell,q,j} y^n_{\ell,q,j} = \sum_{n \in \mathcal{N}_B} \mathbf{g_n y_n}. \tag{4.31}$$

4.3. Variable production/distribution cost

The production/distribution cost parameters include the processing costs per unit at a manufacturing facility (h_{p,ℓ,q,j_r}) and a distribution facility (h_{p,ℓ,q,j_w}), the transportation cost per unit between entities of the supply chain, and the revenue generated by selling products $p \in \mathcal{P}$ at markets $k \in \mathcal{K}$.

On top of shipping related expenses, the transportation cost from a supplier s to a manufacturing location j_r (h_{p,s,j_r}) includes the procurement cost of the supplies and the pipeline inventory cost. Likewise, the transportation cost from a distribution location j_w to a market k ($h_{p,j_w,k}$) is combined with the revenue per unit at that market and the pipeline inventory cost. Thus, $h_{p,j_w,k}$ are likely to assume negative values. The transportation cost from a manufacturing location j_r to a distribution location j_w (h_{p,j_r,j_w}) includes shipping related expenses plus the pipeline inventory cost. Equation (4.32) combines these costs.

$$\begin{aligned}\kappa_3(z) = \sum_{n\in\mathcal{N}\setminus\{1\}} \Bigg(&\sum_{p\in\mathcal{P}}\sum_{j\in\mathcal{J}}\sum_{q\in\mathcal{Q}_j}\sum_{\ell\in\mathcal{L}_q} h_{p,\ell,q,j} z_{p,\ell,q,j} + \sum_{p\in\mathcal{P}}\sum_{s\in\mathcal{S}}\sum_{j_r\in\mathcal{J}_r} h_{p,s,j_r} z_{p,s,j_r} \\ &+ \sum_{p\in\mathcal{P}}\sum_{j_r\in\mathcal{J}_r}\sum_{j_w\in\mathcal{J}_w} h_{p,j_r,j_w} z_{p,j_r,j_w} + \sum_{p\in\mathcal{P}}\sum_{j_w\in\mathcal{J}_w}\sum_{k\in\mathcal{K}} h_{p,j_w,k} z_{p,j_w,k} \Bigg) \\ = \sum_{n\in\mathcal{N}\setminus\{1\}} &\mathbf{h_n z_n}. \end{aligned} \tag{4.32}$$

4.4. Total expected cost

The model aims to minimize the total expected cost (κ), which sums the nodal-probability weighted costs over all scenario-tree nodes. The objective function is expressed by (4.33). In the last term, $\kappa_3(z)$, every node m is expressed in terms of its parent $n = a(m)$. This rests on substituting $\phi_m = \phi_{m|n}\phi_n, m \in \mathcal{D}(n)$. To simplify the notation, let $g^n_{0,q,j} = h^n_{p,0,q,j} = 0$ for $p \in \mathcal{P}, q \in \mathcal{Q}_j, j \in \mathcal{J}, n \in \mathcal{N}$.

$$\begin{aligned}\kappa &= \min\ E_{\omega\in\Omega}\left[\kappa_1(x) + \kappa_2(y) + \kappa_3(z)\right] \qquad (4.33)\\ &= \min \sum_{n\in\mathcal{N}_B} \phi_n \left(\mathbf{f_n x_n} + \mathbf{g_n y_n} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n} \mathbf{h_m z_m} \right).\end{aligned}$$

5. Model's compact representation

To simplify the mathematical exposition, the model's compact representation (P_c) integrates the components discussed in §3 and §4 as shown by (4.34)–(4.37).

$$\kappa = \min \sum_{n\in\mathcal{N}_B} \phi_n \left(\mathbf{f_n x_n} + \mathbf{g_n y_n} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n} \mathbf{h_m z_m} \right), \tag{4.34}$$

subject to

$$\mathbf{A x_n} + \mathbf{B_1 y_n} + \mathbf{B_2 y_{a(n)}} = \mathbf{b} \qquad \forall\, n \in \mathcal{N}_B, \tag{4.35}$$

$$\mathbf{y_n} \in Y_n \qquad \forall n \in \mathcal{N}_B, \tag{4.36}$$

$$\mathbf{D}\mathbf{z_m} + \mathbf{B_3}\mathbf{y_n} = \mathbf{d_m} \qquad \forall\, m \in \mathcal{D}(n), n \in \mathcal{N}_B. \tag{4.37}$$

Constraint (4.35) compactly represents the stage coupling constraints (4.4)–(4.15). All other constraints represent the component coupling constraints. Constraint (4.36) compactly represents constraints (4.16)–(4.18). Constraint (4.37) compactly represents constraints (4.19)–(4.29). The compact representation of constraint (4.35) results from the followings:

- Extending $\mathbf{x_n}$ to include the needed slack and surplus variables to turn inequalities (4.4)–(4.6), (4.8)–(4.10), and (4.12)–(4.14) into equations.
- Extending $\mathbf{f_n}$ to match the new size of $\mathbf{x_n}$ and assigning all new parameters equal to zero.
- Appropriate choice of $\mathbf{A}, \mathbf{B_1}, \mathbf{B_2}$, and $\mathbf{b}$ to reflect the proper coefficients.

The compact representation of constraint (4.37) follows from similar measures.

5.1. Special cases

Manipulating the sets over which the decision variables are indexed reduces P_c into the following special cases. Therefore, the solution approach described in the following chapters also applies to all these special cases:

- The single product case results from omitting the index p from the model (which is equivalent to using $\mathcal{P} = \{1\}$).
- A location selection model (i.e., neglecting the capacity and technology attributes of a resource) results from removing the indices ℓ and q from the model (i.e., both $\mathcal{L}$ and $\mathcal{Q}$ are empty sets). In this case, $y_j^n = 0$ or 1 when location j is closed or open, respectively, g_j^n represents the fixed operating cost

of the facility in location j at node n, and $c_{p,j}$ represents the portion of facility j's capacity consumed to process one unit of product p.

- A capacity planning model in which capacity and technology decisions for existing locations evolve through a given planning horizon results from replacing $\ell \in \{0\} \cup \mathcal{L}_q$ by $\ell \in \mathcal{L}_q$ for all strategic decision variables.

- The deterministic SCND problem results from replacing node indices by their time indices counterparts in all decision variables.

- The static SCND problem results from omitting the time index (i.e., using $\mathcal{T} = \{1\}$) from the formulation of the deterministic SCND problem.

- A dynamic and stochastic SCND problem in which design decisions must be taken only at the beginning of the planning horizon (and remain unchanged thereafter) results from removing time and node indices from all strategic decision variables. This also effectively reduces the model into a two-stage stochastic program.

- A two-stage stochastic program in which strategic decisions can evolve through the planning horizon results from replacing node indices by their time indices counterparts in all strategic decision variables of P_c. This two-stage stochastic program is addressed in Chapter VI.

5.2. Solving the SCND model

For practical purposes, P_c is a massive MIP that is beyond current computational capacity (Schultz, 2009; Sen, 2005). To solve this model, the following chapter develops an End-of-Horizon (EoH) decomposition that rests on exploiting the end-of-horizon effect. This EoH decomposition pre-processes a stochastic program to reduce its size. The resulting smaller stochastic program can be further exploited by any stochastic programming decomposition; c.f. Römisch and Schultz (2001).

By construction, P_c exhibits a block angular structure amenable to decomposition by scenario-tree node. Chapter VI and Chapter VII exploit this structure by a nodal decomposition and a Leontief substitution flow problem reformulation that significantly reduces solution time. Chapter VI and Chapter VII show that combining the EoH decomposition and nodal decomposition provide the potential to radically cut computational effort, which allows larger, realistic problem sizes to be solved.

CHAPTER V

END-OF-HORIZON DECOMPOSITION

This chapter exploits the end-of-horizon effect to decompose the developed supply chain network design (SCND) multistage stochastic mixed-integer program (MSMIP) into a sequence of smaller—and thus easier to solve—subproblems. This chapter shows that when the discount rate is 0%, the SCND prescribed by this decomposition is identical to that of the global optimal solution. Moreover, this chapter derives a bound on the (suboptimal) solution resulting from this decomposition when the discount rate is greater than 0%.

The end-of-horizon (EoH) decomposition is neither SCND nor MSMIP specific. While this chapter frames the EoH decomposition in the context of the SCND problem, this decomposition applies to any application sensitive to the EoH effect. Likewise, the EoH decomposition applies to special cases of MSMIP such as two-stage mixed-integer stochastic programs (SMIP) and dynamic (yet deterministic) models.

This chapter is organized as follows. First, the EoH effect is discussed. Second, the EoH decomposition is developed. Third, extensions driven by practical considerations are presented. Finally, the computational experiments and their results are discussed.

1. End-of-horizon effect

For strategic analyses, multi-period models provide insight into how to adapt to an evolving business environment. Ideally, long planning horizons are preferred to enable the proper evaluation of the long lasting impact of SCND decisions. Practically, the lack of reliable forecasts for distant time periods prevents adopting such long horizons. In most practical circumstances, reliable forecasts pertinent to SCND are not available beyond 10 years (Snyder and Daskin, 2007), which gives rise to the EoH effect.

The EoH effect refers to a model's bias against acquiring new resources as the remaining portion of the planning horizon becomes insufficient to recoup investment expenditures. This bias affects a model's ability to properly evaluate investment options (Hübner, 2007). The SCND literature (and its location selection ancestor) resolves the EoH effect in different ways; for a survey of these methods, see Hübner (2007). Among these methods, Eppen *et al.*'s (1989) approach is the most prominent.

Eppen *et al.*'s (1989) approach rests on optimizing over an infinite planning horizon in which all parameters remain stationary after the last period for which a reliable forecast exists. This infinite horizon can be modeled by a discrete finite horizon, $t \in \{0, ..., T\}$, in which each t represents one period except for the final $t = T$, which stretches into infinity. Fixed and variable production cost parameters for this final period result from discounting infinite cost series to the beginning of this period. To render this final period similar to all other periods—albeit with larger fixed and variable production costs—no network reconfiguration decisions are allowed to take place during this final period except at its beginning.

2. End-of-horizon decomposition

The EoH decomposition sequentially optimizes subsets of a stochastic program's deterministic equivalent model (DEM). It starts by independently optimizing the SCND of every node of the last level of the scenario-tree for which strategic variables apply; i.e., $n \in \mathcal{N}_{T-1}$. Subsequently, the configurations prescribed by the optimal solutions of these nodal subproblems are imposed before optimizing the design decisions of the remaining nodes of the scenario-tree. The resulting MSMIP is smaller in size and thus easier to optimize. Sections 2.1, 2.2, and 2.3 elaborate on the steps of this decomposition.

2.1. Find EoH target configurations

The decomposition imposes EoH target configurations on *boundary nodes.* Boundary nodes, $n \in \mathcal{N}_{T-1}$, belong to the last level of the scenario-tree for which strategic variables apply. Descendants of boundary nodes, leaf nodes, involve only tactical decisions since these nodes don't emanate further scenarios.

An EoH target configuration for a given boundary node results from independently optimizing the nodal subproblem, P_n, corresponding to this boundary node, $n \in \mathcal{N}_{T-1}$. P_n, represented by (5.1), is the deterministic equivalent model (DEM) of an SMIP in which the first stage selects the optimal configuration $(\mathbf{y}_\mathbf{n}^{\circledast})$[1] for a single boundary node ($n \in \mathcal{N}_{T-1}$) and the recourse stage selects the optimal production/distribution decisions ($\mathbf{z}_\mathbf{m}^{\circledast}$) at nodes $m \in \mathcal{D}(n)$. Consequently, $\kappa_n^{\circledast} = \mathbf{g_n}\mathbf{y}_\mathbf{n}^{\circledast} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m}\mathbf{z}_\mathbf{m}^{\circledast}$, where $\kappa_n^{\circledast}$ is the optimal solution value of P_n.

$$\kappa_n^{\circledast} = \min_{\mathbf{y_n},\mathbf{z_m}} \{\mathbf{g_n}\mathbf{y_n} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m}\mathbf{z_m} \mid \mathbf{y_n} \in Y_n, \mathbf{D}\mathbf{z_m} + \mathbf{B_3}\mathbf{y_n} = \mathbf{d_m}\}. \tag{5.1}$$

$P_n, n \in \mathcal{N}_{T-1}$, is an NP-hard SMIP. However, tackling nodal subproblems independently makes them easier to solve. Moreover, the following properties of P_n makes it amenable to solution by the L-shaped method (Van Slyke and Wets, 1969):

- Fixed recourse-matrix $\mathbf{D}$, as previously defined in section 3.2 of Chapter IV.

[1]Throughout this document, the superscript $\circledast$ refers to nodal optima while superscript $*$ distinguishes elements of the (global) optimal solution for P_C

- Relatively complete-recourse; i.e., for each $\mathbf{y_n} \in Y_n$, there exists a vector $\mathbf{z_m}$ for each $m \in \mathcal{D}(n)$ that satisfies $\mathbf{Dz_m} + \mathbf{B_3y_n} = \mathbf{d_m}$.
- Binary first-stage variables ($\mathbf{y_n}$) and continuous recourse variables ($\mathbf{z_m}$).

The L-shaped method, which is based on Benders Decomposition, has been effective in solving similar SMIPs; see, for example, MirHassani *et al.* (2000) and Santoso *et al.* (2005).

2.2. Find configurations for all other scenario-tree nodes

Imposing the EoH target configurations results from pegging $\mathbf{y_n} = \mathbf{y_n^{\circledast}}$ in P_c, (4.34)–(4.37), for all boundary nodes, $n \in \mathcal{N}_{T-1}$. This also eliminates all variables and constraints relating to leaf nodes $m \in \mathcal{N}_T$. The last two levels of the scenario-tree include a large portion of the scenario-tree's nodes. By imposing given configurations to the boundary nodes (and thereby eliminating their corresponding binary variables) and eliminating leaf nodes all together, this decomposition results in a significantly smaller MSMIP. This resulting MSMIP, P_B, can be expressed by (5.2)–(5.5).

$$\kappa_B = \min \sum_{n \in \mathcal{N}_B} \phi_n \mathbf{f_n x_n} + \sum_{n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}} \phi_n \left(\mathbf{g_n y_n} + \sum_{m \in \mathcal{D}(n)} \phi_{m|n} \mathbf{h_m z_m} \right), \tag{5.2}$$

subject to

$$\mathbf{Ax_n} + \mathbf{B_1 y_n} + \mathbf{B_2 y_{a(n)}} = \mathbf{b} \qquad \forall\, n \in \mathcal{N}_B, \tag{5.3}$$

$$\mathbf{y_n} \in Y_n \qquad \forall n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}, \tag{5.4}$$

$$\mathbf{Dz_m} + \mathbf{B_3 y_n} = \mathbf{d_m} \qquad \forall\, m \in \mathcal{D}(n), n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}. \tag{5.5}$$

2.3. Evaluate the resulting SCND

Equation (5.6) defines the objective function's value of the solution resulting from the EoH decomposition, κ_{EoH}. κ_{EoH} is the sum of the optimal solution value of P_B, κ_B, and the probability-weighted sum of the optimal nodal solution values for all boundary nodes.

$$\kappa_{EoH} = \kappa_B + \sum_{n \in \mathcal{N}_{T-1}} \phi_n \kappa_n^{\circledast}. \tag{5.6}$$

3. Approximating infinite horizons by imposing EoH target configurations

The EoH decomposition captures the essence of Eppen *et al.*'s (1989) infinite planning horizon approach. In their approach, stretching the last period into infinity increases its decisions' relative influence over the supply chain's cost. Thus, a global optimal solution must prescribe nearly optimal configurations for every boundary node.

3.1. Case (1): discount rate=0%

The following proposition shows that the optimal solution is identical to the solution resulting from the EoH decomposition when the discount rate is 0% provided that each nodal subproblem has a unique optimal solution.

Proposition 1 *When the discount rate is* 0%, *the solution resulting from the EoH decomposition is identical to the optimal solution for P_c (expressed by* (4.34)–(4.37)*), provided that the optimal solution for each nodal subproblem, $P_n, n \in \mathcal{N}_{T-1}$, is unique.*

Proof: Let $(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*)$ be an optimal solution for P_c (expressed by (4.34)–(4.37)), where $\mathbf{X}^* = \left(\mathbf{x}_1^*, \ldots, \mathbf{x}_{\mathbf{N_B}}^*\right)$, $\mathbf{Y}^* = \left(\mathbf{y}_1^*, \ldots, \mathbf{y}_{\mathbf{N_B}}^*\right)$, $\mathbf{Z}^* = \left(\mathbf{z}_2^*, \ldots, \mathbf{z}_{\mathbf{N}}^*\right)$, $N_B = |\mathcal{N}_B|$, and $N = |\mathcal{N}|$. Furthermore, let $(\mathbf{X}^{\circledast}, \mathbf{Y}^{\circledast}, \mathbf{Z}^{\circledast})$ be a feasible solution for P_c in which decisions for all nodes $n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$ are identical to those of $(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*), \mathbf{y_n} = \mathbf{y_n^{\circledast}}$ for all $n \in \mathcal{N}_{T-1}, \mathbf{z_m} = \mathbf{z_m^{\circledast}}$ for all $m \in \mathcal{D}(n), n \in \mathcal{N}_{T-1}$, and $\mathbf{x_n} = \mathbf{x_n^{\circledast}}$ for all $n \in \mathcal{N}_{T-1}$, where $\mathbf{x_n^{\circledast}}$ is defined by (5.7);

$$\mathbf{A}\mathbf{x_n^{\circledast}} + \mathbf{B_2}\mathbf{y_n^{\circledast}} + \mathbf{B_3}\mathbf{y^*_{a(n)}} = \mathbf{b} \qquad \forall\, n \in \mathcal{N}_{T-1}. \tag{5.7}$$

Accordingly, the optimality gap, δ, can be defined by (5.8), where κ^* is the objective function's value for $(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*)$ and $\kappa^{\circledast}$ is the objective function's value for $(\mathbf{X}^{\circledast}, \mathbf{Y}^{\circledast}, \mathbf{Z}^{\circledast})$, $\kappa_n^* = \mathbf{g_n}\mathbf{y_n^*} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m}\mathbf{z_m^*}$, and $\kappa_n^{\circledast}$ is defined by (5.1).

$$\delta = \kappa^{\circledast} - \kappa^* = \sum_{n\in\mathcal{N}_{T-1}} \phi_n \left(\mathbf{f_n}\mathbf{x_n^{\circledast}} - \mathbf{f_n}\mathbf{x_n^*} + \left(\kappa_n^{\circledast} - \kappa_n^*\right)\right) \geqslant 0. \tag{5.8}$$

Consider a nodal subproblem $P_n, n \in \mathcal{N}_{T-1}$, for which a unique nodal optimal solution $\mathbf{y_n^{\circledast}}$ exists. To get a contradiction, suppose that $\mathbf{y_n^{\circledast}} \neq \mathbf{y_n^*}$ (and thereby $\mathbf{z_m^{\circledast}} \neq \mathbf{z_m^*}$ for some $m \in \mathcal{D}(n), n \in \mathcal{N}_T$). Suppose that the last period of the planning horizon begins at epoch $t = T-1$ and stretches up to the end of the planning horizon, $t = \tau$. Because the discount rate remains 0% throughout this last time period, $\kappa_n^{\circledast} = \sum_{t=T-1}^{\tau} \kappa_{n,t}^{\circledast}$ and $\kappa_n^* = \sum_{t=T-1}^{\tau} \kappa_{n,t}^*$. The uniqueness of optimal solution of P_n implies that $\kappa_n^{\circledast} - \kappa_n^* < 0$. Define $\epsilon_{n,t} = \kappa_{n,t}^{\circledast} - \kappa_{n,t}^*$. $\epsilon_{n,t} < 0$ because all parameters remain constant and the configuration of the supply chain remains unchanged throughout this last time period. Finally, because this last time period stretches into infinity, $\kappa_n^{\circledast} - \kappa_n^* = \lim_{\tau\to\infty} \sum_{t=T-1}^{\tau} \epsilon_{n,t} = -\infty$, which contradicts (5.8). ∎

3.2. Case (2): discount rate>0%

For practical purposes, discount rates usually exceed 0%. In this case, the solution resulting from the EoH decomposition might not be optimal (with respect to P_c).

Proposition 2 provides a bound on the optimality gap, δ, resulting from the (sub-optimal) solution of the EoH decomposition. (5.9) defines the value of this bound, where δ_e is the optimal value of the SMIP described by (5.10)–(5.15).

$$\delta \leq \sum_{e \in \mathcal{N}_{T-2}} \left(\sum_{n \in \mathcal{D}(e)} \phi_n \kappa_n^{\circledast} - \delta_e \right), \tag{5.9}$$

$$\delta_e = \min \left\{ \sum_{n \in \mathcal{D}(e)} \phi_n \left(\mathbf{f_n x_n} - \mathbf{f_n x_n^{\bullet}} + \mathbf{g_n y_n} + \sum_{m \in \mathcal{D}(n)} \phi_{m|n} \mathbf{h_m z_m} \right) \right\}, \tag{5.10}$$

subject to

$$\mathbf{A x_n} + \mathbf{B_2 y_n} + \mathbf{B_3 y_e} = \mathbf{b} \qquad \forall\, n \in \mathcal{D}(e), \tag{5.11}$$

$$\mathbf{A x_n^{\bullet}} + \mathbf{B_2 y_n^{\circledast}} + \mathbf{B_3 y_e} = \mathbf{b} \qquad \forall\, n \in \mathcal{D}(e), \tag{5.12}$$

$$\mathbf{y_n} \in Y_n \qquad \forall n \in \mathcal{D}(e), \tag{5.13}$$

$$\mathbf{y_e} \in Y_e, \tag{5.14}$$

$$\mathbf{D z_m} + \mathbf{B_3 y_n} = \mathbf{d_m} \qquad \forall\, m \in \mathcal{D}(n), n \in \mathcal{D}(e). \tag{5.15}$$

Proposition 2 *When the discount rate exceeds* 0%, *the optimality gap resulting from the solution of the EoH decomposition is bounded from above as defined by* (5.9).

Proof: Let $(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*)$ be an optimal solution for P_c (expressed by (4.34)–(4.37)), where $\mathbf{X}^* = \left(\mathbf{x_1^*}, \ldots, \mathbf{x_{N_B}^*}\right)$, $\mathbf{Y}^* = \left(\mathbf{y_1^*}, \ldots, \mathbf{y_{N_B}^*}\right)$, $\mathbf{Z}^* = \left(\mathbf{z_2^*}, \ldots, \mathbf{z_N^*}\right)$, $N_B = |\mathcal{N}_B|$, and $N = |\mathcal{N}|$. Furthermore, let $(\mathbf{X}^{\circledast}, \mathbf{Y}^{\circledast}, \mathbf{Z}^{\circledast})$ be a feasible solution for P_c in which decisions for all nodes $n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$ are identical to those of $(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*), \mathbf{y_n} = \mathbf{y_n^{\circledast}}$ for all $n \in \mathcal{N}_{T-1}, \mathbf{z_m} = \mathbf{z_m^{\circledast}}$ for all $m \in \mathcal{D}(n), n \in \mathcal{N}_{T-1}$, and $\mathbf{x_n} = \mathbf{x_n^{\circledast}}$ for all $n \in \mathcal{N}_{T-1}$, where $\mathbf{x_n^{\circledast}}$ is defined by (5.16).

$$\mathbf{A x_n^{\circledast}} + \mathbf{B_1 y_n^{\circledast}} + \mathbf{B_2 y_e^*} = \mathbf{b} \qquad \forall\, n \in \mathcal{D}(e). \tag{5.16}$$

Accordingly, the optimality gap, δ, can be expressed by (5.17), where $\kappa_n^{\circledast} = \mathbf{g_n y_n^{\circledast}} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m z_m^{\circledast}}$, and $\kappa_n^* = \mathbf{g_n y_n^*} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m z_m^*}$.

$$\delta \leqslant \delta^{\circledast} = \sum_{e\in\mathcal{N}_{T-2}} \left[\sum_{n\in\mathcal{D}(e)} \phi_n \left(\mathbf{f_n x_n^{\circledast}} + \kappa_n^{\circledast}\right) - \sum_{n\in\mathcal{D}(e)} \phi_n \left(\mathbf{f_n x_n^*} + \kappa_n^*\right) \right]. \tag{5.17}$$

Consider the following mathematical program:

$$q_e^* = \min\{ \sum_{n\in\mathcal{D}(e)} \phi_n \left(\mathbf{f_n x_n} + \mathbf{g_n y_n} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m z_m} \right) \mid (\mathbf{x_n}, \mathbf{y_n}, \mathbf{z_m}) \in Q_e\},$$

where the set Q_e that is given by $(\mathbf{x_n}, \mathbf{y_n}, \mathbf{z_m})$ values that satisfy (5.18)–(5.21).

$$\mathbf{Ax_n} + \mathbf{B_1 y_n} + \mathbf{B_2 y_e} = \mathbf{b} \qquad \forall\, n \in \mathcal{D}(e), \tag{5.18}$$

$$\mathbf{y_n} \in Y_n \qquad \forall n \in \mathcal{D}(e), \tag{5.19}$$

$$\mathbf{y_e} \in Y_e, \tag{5.20}$$

$$\mathbf{Dz_m} + \mathbf{B_3 y_n} = \mathbf{d_m} \qquad \forall\, m \in \mathcal{D}(n), n \in \mathcal{D}(e). \tag{5.21}$$

Clearly, $(\mathbf{x_n^*}, \mathbf{y_n^*}, \mathbf{z_m^*}) \in Q_e$ (because $(\mathbf{X}^*, \mathbf{Y}^*, \mathbf{Z}^*)$ satisfies all constraints of P_c, which enforce (5.18)–(5.21)). This implies (5.22):

$$\sum_{n\in\mathcal{D}(e)} \phi_n \left(\mathbf{f_n x_n^*} + \kappa_n^*\right) \geqslant q_e^*, \tag{5.22}$$

A new bound, (5.23), results from (5.17) and (5.22):

$$\delta \leqslant \delta^{\circledast} \leqslant \sum_{e\in\mathcal{N}_{T-2}} \left[\sum_{n\in\mathcal{D}(e)} \phi_n \kappa_n^{\circledast} - \left(q_e^* - \sum_{n\in\mathcal{D}(e)} \phi_n \left(\mathbf{f_n x_n^{\circledast}}\right) \right) \right]. \tag{5.23}$$

$q_e^* - \sum_{n\in\mathcal{D}(e)} \phi_n\left(\mathbf{f_n}\mathbf{x_n^{\circledast}}\right)$ is the value of the SMIP expressed by (5.24):

$$q_e^* - \sum_{n\in\mathcal{D}(e)} \phi_n \mathbf{f_n}\mathbf{x_n^{\circledast}} = \min\left\{ \sum_{n\in\mathcal{D}(e)} \phi_n(\mathbf{f_n}\mathbf{x_n} - \mathbf{f_n}\mathbf{x_n^{\circledast}} \right. \\ \left. + \mathbf{g_n}\mathbf{y_n} + \sum_{m\in\mathcal{D}(n)} \phi_{m|n}\mathbf{h_m}\mathbf{z_m}) \mid (\mathbf{x_n}, \mathbf{y_n}, \mathbf{z_m}) \in Q_e \text{ and } (5.16) \right\}. \tag{5.24}$$

The SMIP expressed by (5.10) and (5.15) is a relaxation of that expressed by (5.24), which implies (5.25):

$$q_e^* - \sum_{n\in\mathcal{D}(e)} \phi_n \mathbf{f_n}\mathbf{x_n^{\circledast}} \geqslant \delta_e. \tag{5.25}$$

Finally, (5.9) results from (5.23) and (5.25).

■

4. Practical considerations

Stretching the planning horizon into infinity (or, equivalently, imposing EoH target configurations for nodes $n \in \mathcal{N}_{T-1}$) might overly bias decisions from earlier periods. Rolling the planning horizon and adopting a finite approximation of the infinite-horizon are strategies that can reduce such a bias.

4.1. Rolling horizon

A rolling horizon approach re-applies the model periodically. In every such application, only decisions related to the first period of the planning horizon are implemented. When updated forecasts for subsequent periods become available, the planning horizon rolls by dropping the now elapsed first period and appending an extra period at the end of the horizon. Before re-applying the model, EoH target configurations for the newly added period are imposed. The recently implemented

SCND serves as the initial configuration for the updated horizon, and the configurations of all other periods become decision variables.

4.2. Finite approximation of the infinite-horizon

Instead of extending the last period into infinity, this last period can include only the minimum number of years sufficient to cover the payback periods of all candidate resources. The size of the last period of a planning horizon is therefore instance-dependent. To assess the EoH decomposition's sensitivity to this approximation of a planning horizon, the computational experiment of the following section assumes different lengths for this last period.

5. Computational experiment

This section uses two sets of computational experiments to demonstrate the computational efficiency of the EoH decomposition. The first set tests the performance of the EoH decomposition when the SCND problem is modeled as an MSMIP. The second set tests the performance of the EoH decomposition when the SCND problem is modeled as an SMIP.

Each set of experiments applies the EoH decomposition over three different problem sizes. Table II summarizes the sizes of these problems, which are referred to hereafter as size A, size B and size C. Depending on the problem size, either 3 or 5 instances are tested per problem size. Each instance is solved thirty different times, each using a different combination of discount rates and finite approximations for the infinite last period combinations. For each of these combinations, an instance is solved twice: using CPLEX 11.0 to optimize the DEM and using the EoH decomposition, where CPLEX 11.0 optimizes the DEMs for the subproblems.

Table II. Dimensions of test problems

Problem Size	A	B	C
Products	5	5	5
Suppliers	5	5	5
Customers	10	10	15
Manufacturing locations	5	5	10
Manufacturing technologies	2	2	3
Capacity levels per manufacturing technology	2	2	3
Distribution locations	5	10	15
Distribution technologies	1	1	1
Capacity levels per distribution technology	1	3	3
Time periods	5	5	5
Scenarios	81	81	81

All computational experiments were conducted on a quad-core Intel Xeon X5355 processor running at 2.66 GHz with 12 GB RAM. The parameters for all instances were generated as described by Appendix G.

5.1. Computational experiments for MSMIPs

The results of one instance of each problem size are presented in this section. Appendices A–C present the results of the remaining instances, an appendix per problem size.

Tables III and IV summarize the results for an instance of problem size A. Tables V and VI summarize the results for an instance of problem size B. Tables VII and VIII summarize the results for an instance of problem size C.

Problems size A were allowed unlimited run time under both the DEM and the EoH decomposition. Because it takes a considerable amount of time to optimize a DEM of the second or third problem size, their optimization was halted after one and two hours, respectively. The solution values listed in their corresponding tables indicate the best solutions achieved within these allotted times. The EoH decomposition always completed within these allotted times.

Table III. Solution values for an example MSMIP instance of size A under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	25,255.4	25,236.5	0.07%	0.26%
2%	100	21,926.9	21,898.8	0.13%	0.62%
2%	50	16,276.4	16,239.8	0.22%	0.86%
2%	20	9,061.3	9,031.1	0.33%	0.95%
2%	10	5,472.3	5,444.3	0.51%	1.42%
2%	5	3,442.5	3,413.8	0.84%	1.46%
5%	infinity	9,345.4	9,328.1	0.19%	0.87%
5%	100	9,276.7	9,258.0	0.20%	0.95%
5%	50	8,649.2	8,631.3	0.21%	1.14%
5%	20	6,254.3	6,232.8	0.34%	1.16%
5%	10	4,246.1	4,222.0	0.57%	1.16%
5%	5	2,846.1	2,828.0	0.64%	1.18%
10%	infinity	4,085.9	4,075.3	0.26%	1.02%
10%	100	4,083.2	4,072.0	0.27%	1.08%
10%	50	4,099.7	4,078.9	0.51%	1.50%
10%	20	3,633.3	3,602.8	0.84%	1.79%
10%	10	2,857.7	2,832.7	0.88%	1.85%
10%	5	2,086.4	2,067.9	0.89%	1.88%
20%	infinity	2,685.4	2,661.1	0.90%	2.12%
20%	100	2,652.9	2,627.4	0.96%	2.17%
20%	50	2,562.2	2,537.3	0.97%	2.21%
20%	20	2,513.8	2,484.0	1.19%	2.70%
20%	10	1,481.0	1,462.9	1.22%	2.74%
20%	5	1,457.2	1,438.7	1.27%	2.78%
50%	infinity	273.4	267.8	2.05%	4.04%
50%	100	270.0	264.4	2.07%	4.15%
50%	50	266.5	260.9	2.11%	4.18%
50%	20	265.9	260.1	2.18%	4.70%
50%	10	265.2	259.3	2.20%	4.91%
50%	5	246.8	241.3	2.22%	4.99%

Table IV. Solution times for an example MSMIP instance of size A under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	64.10	0.49	0.77%	1.07
2%	100	58.91	0.48	0.82%	1.34
2%	50	58.03	0.51	0.88%	1.04
2%	20	24.32	0.44	1.81%	1.38
2%	10	21.07	0.46	2.19%	1.55
2%	5	9.51	0.50	5.23%	1.50
5%	infinity	31.33	0.59	1.87%	2.11
5%	100	54.01	0.58	1.07%	2.56
5%	50	56.75	0.54	0.95%	2.00
5%	20	21.87	0.56	2.57%	2.54
5%	10	9.34	0.53	5.68%	2.42
5%	5	8.72	0.54	6.24%	2.47
10%	infinity	10.67	0.66	6.14%	2.22
10%	100	9.35	0.62	6.63%	1.96
10%	50	17.08	0.66	3.84%	1.98
10%	20	46.00	0.67	1.46%	2.32
10%	10	34.34	0.64	1.87%	1.96
10%	5	8.38	0.63	7.48%	1.93
20%	infinity	64.80	0.61	0.95%	1.77
20%	100	59.02	0.66	1.11%	1.28
20%	50	42.70	0.68	1.58%	1.10
20%	20	8.85	0.66	7.43%	1.57
20%	10	8.05	0.65	8.10%	1.74
20%	5	12.76	0.64	5.02%	1.35
50%	infinity	39.82	0.76	1.91%	1.21
50%	100	36.04	0.76	2.12%	1.32
50%	50	43.30	0.76	1.76%	0.93
50%	20	38.89	0.76	1.95%	0.96
50%	10	41.59	0.74	1.78%	0.98
50%	5	13.36	0.77	5.80%	0.93

Table V. Solution values for an example MSMIP instance of size B under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at one hour runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM	Bound on optimality gap
2%	infinity	34,776.9	36,437.5	4.6%	0.8%
2%	100	30,343.2	31,635.8	4.1%	0.9%
2%	50	22,456.9	23,537.3	4.6%	1.1%
2%	20	12,448.9	13,013.2	4.3%	1.2%
2%	10	7,557.0	7,920.9	4.6%	1.3%
2%	5	4,748.2	4,951.5	4.1%	1.3%
5%	infinity	12,835.0	13,527.5	5.1%	1.0%
5%	100	12,726.4	13,394.0	5.0%	1.0%
5%	50	11,947.5	12,489.0	4.3%	1.1%
5%	20	8,630.8	9,005.7	4.2%	1.7%
5%	10	5,858.5	6,161.5	4.9%	1.7%
5%	5	3,901.6	4,114.7	5.2%	1.8%
10%	infinity	5,832.0	5,969.5	2.3%	1.5%
10%	100	5,811.4	5,967.5	2.6%	1.9%
10%	50	5,695.5	5,861.2	2.8%	1.9%
10%	20	5,125.3	5,232.4	2.0%	2.1%
10%	10	4,032.4	4,122.1	2.2%	2.2%
10%	5	2,890.4	2,976.7	2.9%	2.2%
20%	infinity	3,713.7	3,902.2	4.8%	2.5%
20%	100	3,629.8	3,795.5	4.4%	2.6%
20%	50	3,564.2	3,738.8	4.7%	2.7%
20%	20	3,403.8	3,579.4	4.9%	2.7%
20%	10	2,050.4	2,136.7	4.0%	2.8%
20%	5	1,927.7	2,020.1	4.6%	2.8%
50%	infinity	488.6	494.2	1.1%	3.0%
50%	100	479.1	487.7	1.8%	3.1%
50%	50	471.6	477.8	1.3%	3.2%
50%	20	458.9	467.3	1.8%	3.2%
50%	10	464.0	464.5	0.1%	3.3%
50%	5	455.1	460.2	1.1%	3.6%

Table VI. Solution times for an example MSMIP instance of size B under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM	Solution time for bound on gap (minutes)
2%	infinity	60.0	11.2	19%	50.1
2%	100	60.1	13.9	23%	59.4
2%	50	60.1	11.1	18%	50.1
2%	20	60.0	10.5	18%	55.0
2%	10	60.0	13.3	22%	56.8
2%	5	60.1	11.0	18%	55.8
5%	infinity	60.1	12.9	21%	69.6
5%	100	60.0	9.7	16%	75.8
5%	50	60.0	11.3	19%	67.6
5%	20	60.0	11.4	19%	67.9
5%	10	60.0	13.2	22%	71.2
5%	5	60.0	10.4	17%	69.8
10%	infinity	60.0	15.4	26%	50.5
10%	100	60.1	15.9	26%	58.7
10%	50	60.1	15.8	26%	56.9
10%	20	60.0	14.9	25%	55.7
10%	10	60.1	16.1	27%	58.2
10%	5	60.1	14.1	23%	52.1
20%	infinity	60.1	19.9	33%	76.5
20%	100	60.1	22.0	37%	74.7
20%	50	60.0	21.0	35%	78.3
20%	20	60.0	19.6	33%	73.2
20%	10	60.0	19.2	32%	73.8
20%	5	60.0	18.7	31%	74.0
50%	infinity	60.0	11.4	19%	42.9
50%	100	60.0	10.2	17%	43.3
50%	50	60.1	11.4	19%	43.2
50%	20	60.1	13.0	22%	42.2
50%	10	60.0	12.9	21%	47.4
50%	5	60.1	11.7	20%	47.1

Table VII. Solution values for an example MSMIP instance of size C under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at two hours runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM
2%	infinity	271,662.0	289,398.7	6.1%
2%	100	225,135.4	251,049.6	10.3%
2%	50	161,838.8	186,441.3	13.2%
2%	20	94,493.9	103,322.4	8.5%
2%	10	54,146.3	62,643.1	13.6%
2%	5	35,144.4	39,104.8	10.1%
5%	infinity	92,875.0	107,150.2	13.3%
5%	100	97,106.2	106,337.0	8.7%
5%	50	91,759.7	98,965.2	7.3%
5%	20	65,429.3	71,320.5	8.3%
5%	10	42,203.3	48,459.4	12.9%
5%	5	28,894.0	32,019.0	9.8%
10%	infinity	42,569.4	47,031.5	9.5%
10%	100	41,378.4	46,911.5	11.8%
10%	50	41,501.8	46,633.2	11.0%
10%	20	36,740.3	41,422.0	11.3%
10%	10	27,627.2	32,220.7	14.3%
10%	5	20,552.3	22,901.6	10.3%
20%	infinity	27,572.0	30,694.3	10.2%
20%	100	26,645.0	30,648.8	13.1%
20%	50	25,965.2	29,010.4	10.5%
20%	20	24,687.9	28,158.8	12.3%
20%	10	14,384.6	16,417.9	12.4%
20%	5	13,736.4	15,500.5	11.4%
50%	infinity	2,924.0	3,163.5	7.6%
50%	100	2,915.9	3,157.1	7.6%
50%	50	2,816.2	3,052.9	7.8%
50%	20	2,767.2	2,978.1	7.1%
50%	10	2,657.6	2,879.3	7.7%
50%	5	2,483.3	2,672.7	7.1%

Table VIII. Solution times for an example MSMIP instance of size C under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
2%	infinity	120.1	59.2	49%
2%	100	120.1	53.8	45%
2%	50	120.0	50.9	42%
2%	20	120.1	50.1	42%
2%	10	120.1	53.6	45%
2%	5	120.0	58.9	49%
5%	infinity	120.1	46.7	39%
5%	100	120.1	50.1	42%
5%	50	120.0	52.5	44%
5%	20	120.1	54.8	46%
5%	10	120.1	51.3	43%
5%	5	120.1	55.3	46%
10%	infinity	120.1	61.3	51%
10%	100	120.0	59.2	49%
10%	50	120.1	55.9	47%
10%	20	120.0	55.3	46%
10%	10	120.1	56.4	47%
10%	5	120.0	60.8	51%
20%	infinity	120.1	73.6	61%
20%	100	120.0	68.1	57%
20%	50	120.0	65.1	54%
20%	20	120.0	63.5	53%
20%	10	120.0	68.5	57%
20%	5	120.0	73.7	61%
50%	infinity	120.1	58.3	49%
50%	100	120.0	51.5	43%
50%	50	120.0	57.9	48%
50%	20	120.0	59.4	50%
50%	10	120.0	61.0	51%
50%	5	120.1	55.0	46%

The results of the experiments exhibit the following features:

- For problem size A, the optimality gap never exceeds 4% and the bound on this gap never exceeds 6%. For problems sizes B and C, the solution of the DEM was halted after an hour or two hours, respectively, and the optimal solution was not achieved. Regardless, the EoH solution was compared to the best solution achieved by the EoH for the instances of problem sizes B and C. The solution for the EoH decomposition outperformed that of the DEM model in all these instances. The improvement over the DEM solution varied between 0.1% and 5.7% for problem size B and between 6.1% and 18.5% for problem size C.

- Both the optimality gap and the bound on this gap are more sensitive to the discount rate than they are to the approximation of the infinite planning horizon. Consistent with the spirit of proposition 1 (section 3), the lower the discount rate, the smaller the optimality gap and its bound become. Also, the longer the approximation of the infinite planning horizon, the smaller the optimality gap and its bound become.

- Solution time for the EoH decomposition is much shorter than that of the DEM. Table IV shows for an example instance of problem size A for which the solution time for the EoH decomposition is at most 8.01% of that of the DEM. The EoH decomposition also outperformed the DEM in solution time for problem sizes B and C (where the optimization of the DEM was halted after an hour or two hours, respectively). Table VI shows that the maximum EoH runtime was 37% of that of the DEM for an instance of size B. Similarly, for an example instance of problem size C, table VIII shows that the maximum EoH runtime was 61% of that of the DEM (the improvement over the DEM solution value was 10%).

- Solution time for the EoH decomposition seems sensitive to the discount rate but insensitive to the approximation of the infinite horizon. In contrast, the solution time for the DEM seems sensitive to both factors and exhibits considerable variability. For example, depending on the combination of discount rate and the approximation for the infinite planning horizon, table IV shows that solution times for the DEM vary between 8.05 to 64.10 minutes. The solution times exhibit variability for the SMIP that provides a bound on the optimality gap. Depending on the combination of discount rate and approximation of the infinite last period, table IV shows that solution times vary between 0.93 to 2.56 minutes.

These observations lead to the following conclusions:

- The optimality gap vanishes as the discount rate converges to 0% and the last periods' length approaches infinity.

- The size of the optimality gap is acceptable for practical applications even when the discount rate is as high as 50% and the approximation for the infinite last period is as short as 5 years.

- The EoH decomposition significantly cuts the solution time, and this computational efficiency increases along with the problem size.

- The EoH decomposition's induced solution efficiency allows solving problems of larger sizes, which enables modeling SCND problems more realistically than was possible with previous models.

5.2. Computational experiments for SMIPs

The results of one instance of each problem size are presented in this section. Appendices D–F present the results of the remaining instances, an appendix per problem size.

Tables IX and X summarize the results for an instance of problem size A. Tables XI and XII summarize the results for an instance of problem size B. Tables XIII and XIV summarize the results for an instance of problem size C. All problem sizes were allowed unlimited run time under both the DEM and the EoH decomposition.

The results of the experiments exhibit the following features:

- The optimality gap never exceeds 7% and the bound on that gap never exceeds 13%. Both the optimality gap and the bound on this gap are more sensitive to the discount rate than they are to the approximation of the infinite planning horizon. Consistent with the spirit of proposition 1 (section 3), the lower the discount rate, the smaller the optimality gap and its bound become. Also, the longer the approximation of the infinite planning horizon, the smaller the optimality gap and its bound become.

- Solution time for the EoH decomposition is shorter than that of the DEM. Table X shows for an example instance of problem size A for which the solution time for the EoH decomposition is at most 12% of that of the DEM. Table XII shows that the maximum EoH runtime was 39% of that of the DEM for an instance of size B. Similarly, for an example instance of problem size C, table XIV shows that the maximum EoH runtime was 38% of that of the DEM.

These observations lead to the following conclusions:

- The optimality gap vanishes as the discount rate converges to 0% and the last

Table IX. Solution values for an example SMIP instance of size A under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	23,254.2	23,254.2	0.0%	0.0%
2%	100	20,164.3	20,164.3	0.0%	0.0%
2%	50	15,119.7	15,119.7	0.0%	0.0%
2%	20	8,331.5	8,331.5	0.0%	0.0%
2%	10	4,919.1	4,919.1	0.0%	0.0%
2%	5	3,124.9	3,103.8	0.7%	0.7%
5%	infinity	8,388.7	8,388.7	0.0%	0.0%
5%	100	8,349.1	8,349.1	0.0%	0.0%
5%	50	7,989.3	7,989.3	0.0%	0.0%
5%	20	5,677.6	5,677.6	0.0%	0.0%
5%	10	3,816.3	3,795.8	0.5%	0.5%
5%	5	2,644.3	2,626.7	0.7%	1.8%
10%	infinity	3,670.0	3,670.0	0.0%	0.0%
10%	100	3,669.4	3,669.4	0.0%	0.0%
10%	50	3,389.7	3,389.7	0.0%	0.0%
10%	20	3,348.7	3,348.7	0.0%	0.0%
10%	10	2,579.5	2,546.9	1.3%	1.3%
10%	5	1,864.4	1,836.0	1.5%	2.6%
20%	infinity	2,457.2	2,457.2	0.0%	0.0%
20%	100	2,394.6	2,394.6	0.0%	0.0%
20%	50	2,339.0	2,339.0	0.0%	0.0%
20%	20	2,284.5	2,235.0	2.2%	2.2%
20%	10	1,349.5	1,313.1	2.7%	3.7%
20%	5	1,338.2	1,297.1	3.1%	4.8%
50%	infinity	252.6	252.6	0.0%	0.0%
50%	100	248.4	244.5	1.6%	2.1%
50%	50	244.9	239.6	2.2%	3.0%
50%	20	241.9	234.6	3.0%	3.6%
50%	10	239.7	231.2	3.6%	4.7%
50%	5	224.6	215.7	4.0%	6.3%

Table X. Solution times for an example SMIP instance of size A under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	9.35	0.41	4.4%	2.45
2%	100	9.24	0.49	5.3%	1.00
2%	50	10.19	0.42	4.1%	2.36
2%	20	9.22	0.42	4.6%	2.85
2%	10	9.11	0.49	5.4%	3.26
2%	5	8.02	0.50	6.3%	2.73
5%	infinity	6.98	0.49	7.0%	2.52
5%	100	10.20	0.43	4.2%	1.46
5%	50	9.65	0.46	4.8%	1.82
5%	20	7.02	0.41	5.8%	1.42
5%	10	7.10	0.49	6.9%	1.93
5%	5	7.21	0.32	4.4%	1.10
10%	infinity	7.04	0.82	11.6%	1.11
10%	100	7.09	0.80	11.3%	1.50
10%	50	10.27	0.98	9.5%	2.00
10%	20	9.45	0.79	8.4%	1.58
10%	10	7.49	0.79	10.5%	2.52
10%	5	6.91	0.72	10.4%	1.45
20%	infinity	8.77	0.66	7.6%	0.99
20%	100	7.50	0.68	9.1%	2.13
20%	50	8.55	0.67	7.8%	1.67
20%	20	8.12	0.70	8.6%	2.92
20%	10	8.37	0.65	7.8%	2.10
20%	5	9.10	0.63	6.9%	2.63
50%	infinity	9.69	0.95	9.8%	1.48
50%	100	9.21	0.81	8.8%	3.22
50%	50	9.15	0.83	9.0%	1.49
50%	20	8.59	0.76	8.8%	3.04
50%	10	7.65	0.60	7.9%	2.81
50%	5	7.62	0.59	7.7%	0.61

Table XI. Solution values for an example SMIP instance of size B under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	32,337.1	32,337.1	0.00%	0.00%
2%	100	28,010.9	28,010.9	0.00%	0.00%
2%	50	20,369.3	20,369.3	0.00%	0.06%
2%	20	11,090.5	11,090.5	0.00%	0.37%
2%	10	7,003.7	6,998.7	0.07%	0.61%
2%	5	4,345.3	4,286.7	1.35%	0.76%
5%	infinity	11,655.2	11,655.2	0.00%	0.06%
5%	100	11,619.3	11,619.3	0.00%	0.22%
5%	50	10,779.9	10,779.9	0.00%	0.30%
5%	20	7,768.4	7,768.4	0.00%	0.60%
5%	10	5,251.2	5,240.7	0.20%	1.50%
5%	5	3,607.7	3,539.2	1.90%	1.70%
10%	infinity	5,372.6	5,372.6	0.00%	0.33%
10%	100	5,294.0	5,294.0	0.00%	0.50%
10%	50	5,185.1	5,185.1	0.00%	0.90%
10%	20	4,503.0	4,503.0	0.00%	1.10%
10%	10	3,665.9	3,634.2	0.87%	2.10%
10%	5	2,633.0	2,580.7	1.99%	2.20%
20%	infinity	3,327.4	3,327.4	0.00%	0.51%
20%	100	3,221.8	3,221.8	0.00%	0.50%
20%	50	3,190.0	3,190.0	0.00%	1.03%
20%	20	3,148.7	3,099.3	1.57%	1.40%
20%	10	1,827.5	1,790.7	2.02%	2.10%
20%	5	1,702.6	1,633.8	4.04%	2.46%
50%	infinity	436.5	436.5	0.00%	0.80%
50%	100	419.9	419.9	0.00%	0.90%
50%	50	417.8	417.7	0.03%	1.30%
50%	20	415.9	407.4	2.04%	2.01%
50%	10	412.9	399.6	3.22%	4.39%
50%	5	395.2	371.0	6.12%	10.01%

Table XII. Solution times for an example SMIP instance of size B under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	26.4	1.80	6.8%	11.0
2%	100	31.4	2.79	26.7%	12.3
2%	50	36.8	2.61	21.3%	11.5
2%	20	34.9	1.53	13.2%	9.2
2%	10	34.7	3.32	28.8%	13.7
2%	5	31.4	1.93	18.4%	10.7
5%	infinity	20.7	2.68	38.8%	8.6
5%	100	30.5	1.62	15.9%	7.0
5%	50	27.1	1.77	19.6%	8.3
5%	20	29.5	2.47	25.1%	9.7
5%	10	40.1	2.32	17.3%	10.5
5%	5	23.4	2.34	30.0%	9.4
10%	infinity	30.4	3.18	31.4%	12.1
10%	100	35.9	1.71	14.3%	11.1
10%	50	37.0	2.11	17.1%	9.3
10%	20	22.3	2.89	38.9%	9.3
10%	10	23.4	3.30	42.3%	10.5
10%	5	40.1	1.64	12.2%	11.0
20%	infinity	37.7	2.72	21.7%	11.5
20%	100	31.1	2.81	27.1%	10.5
20%	50	37.2	3.12	25.2%	12.0
20%	20	41.0	1.84	13.5%	11.0
20%	10	29.2	2.44	25.0%	11.7
20%	5	39.0	1.51	11.6%	8.9
50%	infinity	26.1	1.69	19.4%	9.0
50%	100	38.9	1.55	12.0%	12.0
50%	50	26.1	2.06	23.7%	10.7
50%	20	28.0	2.88	30.8%	9.7
50%	10	27.2	2.26	25.0%	10.6
50%	5	30.1	1.88	18.7%	10.5

Table XIII. Solution values for an example SMIP instance of size C under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	223,465.4	223,465.4	0.0%	0.0%
2%	100	185,440.5	185,255.1	0.1%	0.1%
2%	50	134,097.8	133,887.2	0.2%	0.2%
2%	20	86,575.9	86,304.9	0.3%	1.9%
2%	10	45,008.0	43,812.1	2.7%	3.2%
2%	5	30,558.3	29,651.3	3.0%	3.6%
5%	infinity	94,840.2	94,870.1	0.0%	0.0%
5%	100	88,509.8	88,216.0	0.3%	0.6%
5%	50	87,882.3	87,038.0	1.0%	1.4%
5%	20	62,643.4	61,352.7	2.1%	2.7%
5%	10	35,163.3	34,147.4	2.9%	3.0%
5%	5	25,123.6	24,327.8	3.2%	3.7%
10%	infinity	41,039.6	40,987.7	0.1%	0.1%
10%	100	40,770.3	40,546.6	0.5%	0.6%
10%	50	38,289.8	37,573.9	1.9%	2.2%
10%	20	31,874.4	31,055.0	2.6%	2.8%
10%	10	26,791.4	25,955.4	3.1%	3.1%
10%	5	19,520.5	18,786.4	3.8%	3.8%
20%	infinity	24,664.4	24,467.4	0.8%	1.6%
20%	100	22,714.2	22,295.2	1.8%	2.4%
20%	50	23,079.5	22,494.8	2.5%	2.5%
20%	20	21,595.3	20,746.8	3.9%	4.6%
20%	10	12,649.9	12,139.7	4.0%	4.8%
20%	5	11,399.7	10,937.5	4.1%	4.9%
50%	infinity	2,761.4	2,714.9	1.7%	2.2%
50%	100	2,641.6	2,587.1	2.1%	2.8%
50%	50	2,533.1	2,458.9	2.9%	3.1%
50%	20	2,408.8	2,312.4	4.0%	5.3%
50%	10	2,228.1	2,136.7	4.1%	5.5%
50%	5	2,198.8	2,093.3	4.8%	6.9%

Table XIV. Solution times for an example SMIP instance of size C under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	62.2	16.3	26%	36.0
2%	100	87.7	11.2	13%	87.4
2%	50	62.7	16.1	26%	40.3
2%	20	77.3	12.7	16%	47.7
2%	10	73.9	18.0	24%	70.3
2%	5	63.8	20.7	32%	47.0
5%	infinity	82.5	18.4	22%	39.9
5%	100	60.9	18.9	31%	45.0
5%	50	72.3	13.2	18%	47.7
5%	20	64.0	11.5	18%	54.2
5%	10	75.1	12.1	16%	29.8
5%	5	51.0	19.2	38%	50.0
10%	infinity	88.8	20.0	23%	44.3
10%	100	98.8	12.3	12%	86.0
10%	50	56.2	11.8	21%	42.0
10%	20	60.5	17.0	28%	40.4
10%	10	66.7	14.6	22%	51.3
10%	5	66.4	10.4	16%	55.2
20%	infinity	62.5	10.4	17%	54.3
20%	100	56.1	18.1	32%	40.4
20%	50	65.2	14.7	23%	62.9
20%	20	53.3	11.7	22%	31.7
20%	10	69.6	18.6	27%	60.8
20%	5	75.3	16.4	22%	70.5
50%	infinity	74.5	19.9	27%	61.2
50%	100	77.6	13.1	17%	37.0
50%	50	81.7	11.7	14%	76.8
50%	20	72.4	18.5	26%	56.9
50%	10	68.2	16.9	25%	51.3
50%	5	77.7	19.2	25%	74.0

periods' length approaches infinity.

- The size of the optimality gap is acceptable for practical applications even when the discount rate is as high as 20% and the approximation for the infinite last period is as short as 5 years.
- The EoH decomposition significantly reduces the solution time.
- The EoH decomposition's induced solution efficiency allows solving problems of larger sizes, which enables modeling SCND problems more realistically than was possible with previous models.

CHAPTER VI

TWO-STAGE SCND DECOMPOSITION AND HEURISTIC

This chapter presents a practical solution approach for the supply chain network design (SCND) problem when modeled as a two-stage stochastic mixed-integer program (SMIP). The solution approach for this SMIP is developed as a stepping stone towards solving the multistage stochastic mixed-integer program P_c, (4.35)–(4.37).

The developed solution approach is a type-I column generation procedure, which has proved successful in solving NP-hard problems (Wilhelm, 2001). It rests on applying nodal decomposition on a SMIP, as section 2 of this chapter explains. This decomposition results in a conveniently structured master problem that links $T-2$ otherwise independent subproblems.

These subproblems are still NP-hard SMIPs, but their relatively smaller sizes makes them easier to solve. The L-shaped method particularly suits solving these subproblems, as section 2.1 elaborates. Furthermore, its resulting iterative solutions provide the master problem's columns, as shown in section 2.2.

Section 3 reformulates the master problem into a shortest path problem (SPP), which enjoys the integrality property. The reformulated master problem is restricted in the sense that it includes only a subset of all feasible columns. Thus, the heuristic developed in section 4 does not necessarily produce an optimal solution. However, as section 5 shows, the results of the computational experiments indicate that this heuristic's optimality gap is below 6%. The heuristic's solution time is always less than 93% of that of the deterministic equivalent model (DEM), which results in an optimal solution. The computational efficiency of this heuristic enables solving larger problems than current technology allows, which benefits practical business application.

1. Two-stage stochastic program

Chapter IV developed P_c, a DEM of an MSMIP that models the SCND problem. As section 5.1 explained, the SMIP special case of this MSMIP model results from replacing node indices by their time indices counterparts for all strategic decision variables. As such, the resulting SCND decisions still evolve throughout the planning horizon but uncertainty unfolds only after the design decisions of all time periods are implemented. Thereby, SMIP forfeits the benefits of applying the information gained by the gradual unfolding of uncertainty in making design decisions.

Chapter V developed an end-of-horizon (EoH) decomposition that pegs the configuration of the last period for which design decisions exist to the optimal configuration for that period. This reduces the size of a SMIP and thus renders it easier to solve.

The following SMIP, (6.1)–(6.4), results from applying the EoH decomposition to the SMIP special case of P_c. The objective function (6.1) sums the costs associated with design decisions (x_t, y_t) and production and distribution decisions (z_m). The costs of production and distribution decisions are weighted by combining the probability of all scenarios to which they apply (which is equivalent to their respective nodal probability in the MSMIP scenario-tree, ϕ_m), as explained in section 1.4 of Chapter IV. Because uncertainty unfolds after all design decisions are implemented, design decisions are associated with a probability of 1.0.

$$\kappa_{SMIP} = \min \sum_{t=1}^{T-1} \left(\mathbf{f_t x_t} + \mathbf{g_t y_t} + \sum_{m \in \mathcal{N}_{t+1}} \phi_m \mathbf{h_m z_m} \right), \tag{6.1}$$

subject to

$$\mathbf{A x_t} + \mathbf{B_1 y_t} + \mathbf{B_2 y_{t-1}} = \mathbf{b} \qquad \forall\, t = 1, \ldots, T-1, \tag{6.2}$$

$$\mathbf{y_t} \in Y_t \qquad \forall t = 1, \ldots, T-2, \tag{6.3}$$

$$\mathbf{Dz_m} + \mathbf{B_3 y_{t_m - 1}} = \mathbf{d_m} \qquad \forall\, m = 2, \ldots, N_B. \tag{6.4}$$

Constraint set (6.2) compactly represents the stage coupling constraints (4.4)–(4.15). Constraints belonging to this set establish the interdependency among successive time periods. When $t = 1$, $\mathbf{y_{t-1}} = \mathbf{y_0}$, which is the inherited initial configuration that could be null. When $t = T-1$, $\mathbf{y_t} = \mathbf{y_{T-1}}$, which is the target configuration imposed by the EoH decomposition; this model assigns $\mathbf{g_{T-1}} = (\mathbf{0}, \ldots, \mathbf{0})$ to simplify the expression of the objective function.

Constraint sets (6.3) and (6.4) represent the component coupling constraints. Constraint set (6.3) compactly represents constraints (4.16)–(4.18). Each constraint of set (6.3) describes the configuration of the supply chain at a given time period. Constraint set (6.4) compactly represents constraints (4.19)–(4.29). Each constraint of this set describes the production and distribution decisions, $\mathbf{z_m}$, under a unique combination of a time period, $t_m = 2, \ldots, T-1$, and a realization of the demand scenarios. The capacity of the existing supply chain configuration, $\mathbf{y_{t_m-1}}$, and the customers' demand $\mathbf{d_m}$, restrict these production and distribution decisions. $\mathbf{z_1}$ and $\mathbf{z_T}$ are not included in (6.4) since they depend only on the inherited initial configuration and the fixed target configuration, respectively.

2. Model decomposition

The block-angular structure of the SMIP (6.1)–(6.4) allows for decomposing the model by time period. Relaxing the stage coupling constraint set (6.2) results into $T-2$ independent subproblems, each consisting of a single time period. A master problem re-establishes the interdependence among these subproblems.

2.1. Single-period subproblems

The subproblems result from relaxing constraint (6.2) and omitting $\sum_{t=1}^{T-1} \mathbf{f_t x_t}$ from the objective function (6.1). Each subproblem seeks the best SCND for a single time period. An arbitrary nodal subproblem, P_t, is defined by (6.5).

$$\kappa_t = \min \left\{ \mathbf{g_t y_t} + \sum_{m \in \mathcal{N}_{t+1}} \phi_m \mathbf{h_m z_m} \;\middle|\; \mathbf{y_t} \in Y_t, \mathbf{D z_m} + \mathbf{B_3 y_t} = \mathbf{d_m} \right\}. \tag{6.5}$$

P_t is a SMIP in which the first stage, (6.6), selects the operational resources ($\mathbf{y_t}$) at time t. Given $\mathbf{y_t}$ and $\omega \in \Omega$, the recourse stage, (6.7), selects the production/distribution decisions ($\mathbf{z_m}$) at time $t+1$.

$$\kappa_t = \min \left\{ \mathbf{g_t y_t} + Q(\mathbf{y_t}) \mid \mathbf{y_t} \in Y_t \right\}, \tag{6.6}$$

$$Q(\mathbf{y_t}) = E_\omega Q(\mathbf{y_t}, \omega) = \min \{ \sum_{m \in \mathcal{N}_{t+1}} \phi_m \left(\mathbf{h_m z_m} \right) \mid \mathbf{D z_m} + \mathbf{B_3 y_t} = \mathbf{d_m} \}. \tag{6.7}$$

P_t is amenable to solution by the L-shaped method (Van Slyke and Wets, 1969) because it exhibits the following properties:

- Fixed recourse-matrix; i.e., $\mathbf{D}$ is scenario independent, as previously defined in section 3.2 of Chapter IV.
- Relatively complete-recourse; i.e., for each $\mathbf{y_t} \in Y_t$, there exists a vector $\mathbf{z_m}$ for each $m \in \mathcal{N}_{t+1}$ that satisfies (6.4).
- Binary first-stage variables ($\mathbf{y_t}$) and continuous recourse variables ($\mathbf{z_m}$).

MirHassani *et al.* (2000) and Santoso *et al.* (2005) use the L-shaped method to solve similar SMIPs of sizes comparable to those of the single-period subproblems.

2.2. Master problem

The master problem tracks the evolution of the network's configuration throughout the planning horizon and thereby accounts for network reconfiguration costs. It results from expressing $\mathbf{y_t}$ in terms of the members of its feasible solution set as described by (6.8), where $\{\mathbf{y_t^v} | v \in \mathcal{V}_t\}$ is the set of these solution points, and $\mathcal{V}_t$ is its index set.

$$\mathbf{y_t} = \sum_{v \in \mathcal{V}_t} \lambda_t^v \mathbf{y_t^v}, \qquad \sum_{v \in \mathcal{V}_t} \lambda_t^v = 1, \qquad \lambda_t^v \in \{0,1\} \quad \forall v \in \mathcal{V}_t. \tag{6.8}$$

$\sum_{v \in \mathcal{V}_t} \kappa_t^v \lambda_t^v$ substitutes for the contribution of $\mathbf{y_t}$ and $\mathbf{z_m}, m \in \mathcal{N}_{t+1}$, in the objective function. Relation (6.9) defines the value of $\kappa_t^v, t = 1, \ldots, T-2$. κ_{T-1}^v is the value of the fixed target configuration, $\mathbf{y_{T-1}^v}$; both κ_{T-1}^v and $\mathbf{y_{T-1}^v}$ are given by the EoH decomposition.

$$\kappa_t^v = \mathbf{g_t y_t^v} + \sum_{m \in \mathcal{N}_{t+1}} \phi_m \mathbf{h_m z_m^v} \qquad \forall\, t = 1, \ldots, T-2. \tag{6.9}$$

The resulting master problem, P_M, is expressed as follows:

$$\kappa = \min \sum_{t=1}^{T-1} \mathbf{f_t x_t} + \sum_{t=1}^{T-2} \sum_{v \in \mathcal{V}_t} \kappa_t^v \lambda_t^v, \tag{6.10}$$

subject to

$$\mathbf{Ax_1} + \sum_{v \in \mathcal{V}_1} \mathbf{B_1 y_1^v} \lambda_1^v = -\mathbf{B_2 y_0}, \tag{6.11}$$

$$\mathbf{Ax_t} + \sum_{v \in \mathcal{V}_t} \mathbf{B_1 y_t^v} \lambda_t^v + \sum_{u \in \mathcal{V}_{t-1}} \mathbf{B_2 y_{t-1}^u} \lambda_{t-1}^u = \mathbf{b} \qquad \forall\, t = 2, \ldots, T-2, \tag{6.12}$$

$$\mathbf{Ax_{T-1}} + \sum_{u \in \mathcal{V}_{T-2}} \mathbf{B_2 y_{T-2}^u} \lambda_{T-2}^u = -\mathbf{B_1 y_{T-1}}, \tag{6.13}$$

$$\sum_{v \in \mathcal{V}_t} \lambda_t^v = 1 \qquad \forall\, t = 1, \ldots, T-2, \tag{6.14}$$

$$\lambda_t^v \in \{0,1\} \qquad \forall\, v \in \mathcal{V}_t,\ t = 1, \ldots, T-2. \tag{6.15}$$

In this mathematical program, constraints (6.14) and (6.15) dictate selecting exactly one configuration per time period. Constraint sets (6.11)–(6.13) enforce the reconfiguration actions necessary to transition between the configurations of two successive time periods. The objective function accounts for the reconfiguration costs and the costs associated with the selected configurations. The mathematical structure of P_M is amenable to reformulation into a shortest path problem, which is the focus of the following section.

3. Shortest path reformulation

The shortest path problem (SPP) exhibits the integrality property and is amenable to efficient solution algorithms. To take advantage of these properties, the master problem is reformulated into a SPP. Section 3.1 explains the logic behind this reformulation and section 3.2 discusses the ensuing mathematical model.

3.1. Correspondence between the master problem and the SPP

P_M is mainly concerned with selecting a configuration for each time period. In contrast, its SPP reformulation focuses on the reconfiguration actions required to transition between the successive configurations that P_M selects. The following paragraphs use the graph depicted by Fig. 4 to elaborate on the relationship between P_M and its SPP reformulation.

Fig. 4 depicts an example problem involving three time periods. Each such period is depicted by an ellipse. A vertex $v \in \mathcal{V}_t$ drawn inside an ellipse represents a possible feasible solution $\{\mathbf{y_t^v} \mid t = 1, \ldots, T-1\}$ for this time period. Each such feasible solution, $\mathbf{y_t^v}$, describes a possible configuration, v, for node n. Due to the strong association between a feasible solution, the supply chain network configura-

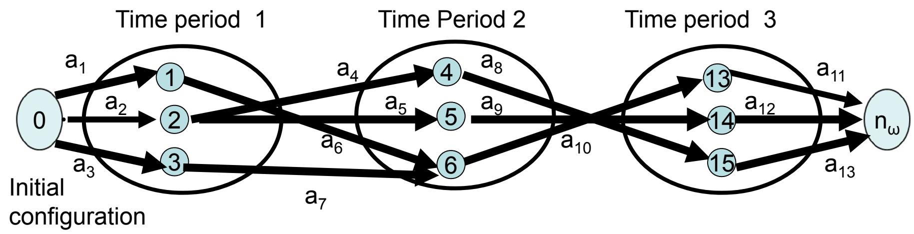

Fig. 4.: Partial graphical representation of the shortest path reformulation, P_{SPP}.

tion it prescribes, and the vertex that represents this solution in the hypergraph, the three terms are used interchangeably hereafter. The initial and target configurations of the supply chain are represented by vertices 0 and $T-1$, respectively. Their distinct shape helps to distinguish them from the other vertices that represent decision variables in P_M.

The reconfiguration action required to transition between every two successive configurations is represented by an arc joining their corresponding vertices. Therefore, the total number of arcs connecting any two successive time periods, $t-1$ and t, is equal to $|\mathcal{V}_{t-1}| \times |\mathcal{V}_t|$. Fig. 4 depicts only a subset of these arcs since depicting all of them would make it cumbersome.

In P_M, constraint (6.14) indicates that a feasible solution must include exactly one configuration per time period. This is equivalent to selecting exactly one vertex per time period in Fig. 4. This constraint is satisfied by selecting a set of arcs that satisfy the following conditions:

- Exactly one arc emanates from each time period $t = 0, \ldots, T-2$.
- If an arc leading to a vertex $v, v \notin V_{T-1}$ is selected, an arc emanating from this vertex must be selected.

Combined, these two conditions are equivalent to selecting exactly one path connecting the root vertex 0 to the target vertex.

The cost of a path is the sum of the weights of its constituent arcs. To achieve equivalence between the weights of these arcs and the cost of the reconfiguration action they represent, variables $x^{v_{t-1}\to v_t}_{\text{open } j}$, $x^{v_{t-1}\to v_t}_{\text{close } j}$, and $x^{v_{t-1}\to v_t}_{\ell_1,\ell_2,q,j}$, are defined by (6.16)–(6.18) for each $v_{t-1} \in \mathcal{V}_{t-1}$ and $v_t \in \mathcal{V}_t$.

$$x^{v_{t-1}\to v_t}_{\text{open } j} = \max\{0, y^{v_t}_{\text{open } j} - y^{v_{t-1}}_{\text{open } j}\} \qquad \forall\, j \in \mathcal{J}, \tag{6.16}$$

$$x^{v_{t-1}\to v_t}_{\text{close } j} = \max\{0, -y^{v_t}_{\text{close } j} + y^{v_{t-1}}_{\text{close } j}\} \qquad \forall\, j \in \mathcal{J}, \tag{6.17}$$

$$x^{v_{t-1}\to v_t}_{\ell_1,\ell_2,q,j} = \max\{0, y^{v_t}_{\ell_2,q,j} + y^{v_{t-1}}_{\ell_1,q,j} - 1\} \qquad \forall\, \ell_1 \neq \ell_2 \in \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J}. \tag{6.18}$$

$\mathbf{x^i_t}$ groups $x^{v_{t-1}\to v_t}_{\text{open } j}$, $x^{v_{t-1}\to v_t}_{\text{close } j}$, and $x^{v_{t-1}\to v_t}_{\ell_1,\ell_2,q,j}$ in a single vector as shown by (6.19).

$$\mathbf{x^i_t} = \left(\{x^{v_{t-1}\to v_t}_{\text{open } j}\}_{j\in\mathcal{J}}, \{x^{v_{t-1}\to v_t}_{\text{close } j}\}_{j\in\mathcal{J}}, \{x^{v_{t-1}\to v_t}_{\ell_1,\ell_2,q,j}\}_{\ell_1\neq\ell_2\in\{0\}\cup\mathcal{L}_j, q\in\mathcal{Q}_j, j\in\mathcal{J}}\right). \tag{6.19}$$

A weight c_i is associated with every arc a_i, where $i \in \mathcal{A}$ and $\mathcal{A}$ is the index set of arcs. Equation (6.20) defines this weight, where $\kappa^{v_t}_t$ is the cost associated with configuration $\mathbf{y^{v_t}_t}$ and is defined by (6.9), $\mathbf{f_t x^i_t}$ is the cost to update the supply chain design from configurations $\mathbf{y^{v_{t-1}}_{t-1}}$ to configuration $\mathbf{y^{v_t}_t}$.

$$c_i = \mathbf{f_t x^i_t} + \kappa^{v_t}_t. \tag{6.20}$$

The following equation defines the value of this reconfiguration cost, where $f^t_{\text{open } j}$ is the cost to open a new facility in location j, $f^t_{\text{close } j}$ is the cost to close the facility at site j, and $f^t_{\ell_1,\ell_2,q,j}$ is the cost to adjust the capacity level of technology q from level ℓ_1 to level ℓ_2.

$$\mathbf{f_t x^i_t} = \sum_{j\in\mathcal{J}} f^t_{\text{open } j} x^{v_{t-1}\to v_t}_{\text{open } j} + \sum_{j\in\mathcal{J}} f^t_{\text{close } j} x^{v_{t-1}\to v_t}_{\text{close } j} + \sum_{j\in\mathcal{J}} \sum_{q\in\mathcal{Q}_j} \sum_{\ell\in\{0\}\cup\mathcal{L}_q} f^t_{\ell_1,\ell_2,q,j} x^{v_{t-1}\to v_t}_{\ell_1,\ell_2,q,j}.$$

3.2. Reformulated master problem

The master problem is reformulated into a single-source shortest hyperpath problem (SPP). Problem P_{SPP} seeks the shortest path connecting the vertex representing the initial configuration ($t = 0$) and that representing the target configuration ($t = T-1$) of the supply chain network. A path, $\boldsymbol{\beta}$, consists of $|T-1|$ arcs; $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_H)$ where β_i is a binary variable indicating whether arc $a_i, i \in \mathcal{A}$, takes part of the path ($\beta_i = 1$) or not ($\beta_i = 0$), and $A = |\mathcal{A}|$. A shortest path, $\boldsymbol{\beta^*}$, is a path with minimum weight; i.e., $\mathbf{c}\boldsymbol{\beta^*} \leq \mathbf{c}\boldsymbol{\beta^j}$ for all possible hyperpaths $\boldsymbol{\beta^j}$, where $\mathbf{c} = (c_1, \ldots, c_A)$.

To ease the mathematical exposition, let $\mathcal{A}_v^+$ and $\mathcal{A}_v^-$ be the index sets of arcs leaving and entering vertex $v \in \mathcal{V}$, respectively, where $\mathcal{V} = \cup_{t=1}^{T-1}\mathcal{V}_t$. Further, $\mathcal{A}_0^- = \mathcal{A}_v^+ = \emptyset$ for $v \in \mathcal{V}_{T-1}$, where $\emptyset$ is the empty set.

The shortest path problem, P_{SPP}, can be expressed by the linear program (6.21)–(6.24)

$$\kappa_{SPP} = \min \sum_{i \in \mathcal{A}} c_i \beta_i, \tag{6.21}$$

subject to

$$\sum_{i \in \mathcal{A}_{v_0}^+} \beta_i = 1, \tag{6.22}$$

$$\sum_{i \in \mathcal{A}_v^+} \beta_i - \sum_{i \in \mathcal{A}_v^-} \beta_i = 0 \qquad \forall\, v \in \mathcal{V}, \tag{6.23}$$

$$\beta_i \geq 0 \qquad \forall\, i \in \mathcal{A}. \tag{6.24}$$

The objective function (6.21) minimizes the cost of the selected arcs, κ_{SPP}. The first constraint (6.22) ensures that exactly one unit of flow leaves the root vertex. Constraint set (6.23) forces a flow leaving the source vertex to eventually reach the target vertex by ensuring conservation of flow at each time period $t = 1, \ldots, T-2$. The nonnegativity constraint set (6.24) suffices to achieve binary variables due to the integrality property of the SPP.

4. SMIP heuristic

The developed solution procedure is a type-I column generation procedure, and thus it consists of two steps:

1. Approximate the solution space of the original problem P_c with a set of feasible configurations.

2. Use a restricted master problem to evaluate the interdependencies among different time periods and select the best combination of generated configurations, one configuration per time period.

As such, this procedure does not guarantee an optimal solution. However, the results of the computational experiments of section 5 reveal satisfactory performance, which is consistent with the results of type-I column generation procedures for other NP-hard problems (Wilhelm, 2001).

4.1. Approximating the solution space

The solution space is approximated by applying the L-shaped method on each single-period subproblem (6.5) for time periods $t = 1, \ldots, T-2$. The L-shaped method generates iterative solutions in the process of finding an optimal solution. These iterative solutions are feasible and thus serve as columns in the restricted master problem.

The configurations generated by solving a subproblem for a time period t serve multiple functions:

1. They populate the columns associated with this time period in the restricted master problem.

2. They also serve as columns for all other time periods. This results from the relatively complete recourse property of the subproblems, which ensures that a configuration generated for one time period is feasible for all other periods. A prerequisite to using a configuration to populate columns of other periods is to evaluate the impact of this configuration on the optimal values of the second stages of these periods. This entails solving one linear program per demand scenario for each of these periods. Nevertheless, solving these linear programs is more computationally efficient than re-generating these columns from scratch, which involves repeatedly solving the same linear programs on top of a binary first stage problem.

3. They jump start the L-shape procedure for all following subproblems. The parameters needed to generate the Benders cuts associated with these configurations become available through solving these linear programs.

Consequently, as the solution procedure progresses, the required computation effort declines.

Numerous variants of Benders cuts exist. This heuristic adopts the variant suggested by Geoffrion and Graves (1974). This variant prevents solving each iteration of the first stage to optimality. Instead, once a feasible solution for the first stage is found, the procedure immediately moves on to the second stage. The successively imposed Benders cuts on the first stage render all previously found first stage solutions infeasible for future iterations. The procedure terminates when the successively added Benders cuts render the master problem infeasible. The following benefits justify the selection of this variant:

- Using this variant preserves the pure-binary nature of the first stage. Other Benders variants introduce continuous variables in the first stage. The resulting MIP is less convenient to solve than the original pure-binary problem (Geoffrion and Graves, 1974).

- Accumulating Benders cuts improves the approximation of the second stage. The poor approximation of the second stage in the earliest iterations doesn't warrant the effort spent to strictly optimize the first stage (Geoffrion and Graves, 1974).

- The increased number of iteration that this variant might require to converge benefits this heuristic. The iterative solutions provide the columns for the restricted master problem. The greater the number of restricted master problem columns, the higher the chance of achieving the (global) optimal solution of P_c.

4.2. Selecting the best combination of generated configurations

The SPP is constructed as explained in section 3.2. Applying Dijkstra's algorithm (Dijkstra, 1959) to the SPP results in the best reconfiguration schedule throughout the planning horizon. This reconfiguration schedule implies the best configuration per time period for the supply chain network.

Lucas *et al.* (2001) and Mitra *et al.* (2006) solve a comparable SCND problem using a type-I column generation heuristic but the solution time for their restricted master problems is in terms of days. The effectiveness of my heuristic relies on the computational efficiency of Dijkstra's algorithm, which is enabled by the decomposition discussed in section 2. The following section uses computational experiments to illustrate this computational efficiency.

5. Computational experiment

This computational experiment applies this heuristic over three different problem sizes. Table II (section 5 of Chapter V) summarizes the sizes of these problems, which are referred to hereafter as size A, size B and size C. Ten instances are tested per problem size. Each instance is solved three different times to achieve the following:

- Optimize the DEM using CPLEX 11.0.
- Apply the EoH decomposition, which uses CPLEX 11.0 to optimize its subproblems.
- Apply this heuristic, which uses CPLEX 11.0 to solve the nodal subproblems.

All computational experiments were conducted on a quad-core Intel Xeon X5355 processor running at 2.66 GHz with 12 GB RAM. All instances for all problem sizes were allowed to run to completion. The cost discount rate in all instances was 2% and the planning horizon was considered infinite. All other parameters were generated as described by Appendix G.

Tables XV and XVI summarize the results for the instances of problem size A. Tables XVII and XVIII summarize the results for the instances of problem size B. Tables XIX and XX summarize the results for the instances of problem size C.

The following features can be observed in the results:

- For instances of size A, the heuristic's solution time ranges from 1.47% to 5.41% of the DEM time (see Table XVI). For instances of size B, the heuristic's solution time ranges from 2.54% to 6.47% of the DEM time (see Table XVIII). For instances of size C, the heuristic's solution time ranges from 3.91% to 6.82% of the DEM time (see Table XX).

Table XV. Solution values for SMIP instances for a SCND problem of size A

Instance	DEM profit ($ 1,000)	Heuristic profit ($ 1,000)	Optimality gap
1	23,254.2	23,159.0	0.41%
2	366.3	364.1	0.60%
3	884.2	859.5	2.80%
4	3,779.8	3,614.9	4.36%
5	32,291.7	30,984.9	4.05%
6	18,199.6	17,190.3	5.55%
7	111,081.8	109,740.1	1.21%
8	40,776.3	39,426.6	3.31%
9	15,126.2	14,284.0	5.57%
10	26,947.0	25,327.5	6.01%

Table XVI. Solution times for SMIP instances for a SCND problem of size A

Instance	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
1	9.35	0.17	1.81%
2	7.77	0.17	2.23%
3	7.32	0.21	2.93%
4	8.82	0.13	1.47%
5	8.89	0.48	5.41%
6	8.71	0.24	2.71%
7	8.82	0.33	3.69%
8	7.68	0.19	2.53%
9	8.37	0.30	3.57%
10	9.44	0.22	2.37%

Table XVII. Solution values for SMIP instances for a SCND problem of size B

Instance	DEM profit ($ 1,000)	Heuristic profit ($ 1,000)	Optimality gap
1	32,337.1	32,337.1	0.00%
2	269.1	269.0	0.05%
3	13,383.5	13,135.4	1.85%
4	94,770.2	93,712.3	1.12%
5	16,156.2	16,156.2	0.00%
6	1,232.5	1,210.8	1.76%
7	26,512.7	26,118.5	1.49%
8	230,987.9	228,144.0	1.23%
9	11,277.5	11,075.9	1.79%
10	3,388.6	3,388.6	0.00%

Table XVIII. Solution times for SMIP instances for a SCND problem of size B

Instance	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
1	26.4	1.3	4.75%
2	35.4	2.3	6.47%
3	33.1	1.8	5.35%
4	24.2	1.1	4.51%
5	24.4	1.3	5.26%
6	24.7	1.2	4.71%
7	28.3	0.7	2.54%
8	38.5	1.5	3.90%
9	31.2	1.5	4.96%
10	32.1	1.1	3.27%

Table XIX. Solution values for SMIP instances for a SCND problem of size C

Instance	DEM profit ($ 1,000)	Heuristic profit ($ 1,000)	Optimality gap
1	223,465.4	222,514.4	0.43%
2	542,452.6	537,082.3	0.99%
3	7,594.4	7,584.6	0.13%
4	42,325.5	41,552.0	1.83%
5	142,094.8	139,484.7	1.84%
6	6,128.7	6,049.8	1.29%
7	53,607.7	52,713.5	1.67%
8	419,008.4	414,274.4	1.13%
9	38,527.1	37,839.9	1.78%
10	596,366.4	588,061.4	1.39%

Table XX. Solution times for SMIP instances for a SCND problem of size C

Instance	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
1	62.2	4.0	6.43%
2	96.7	4.2	4.34%
3	74.9	4.7	6.34%
4	98.3	6.0	6.11%
5	71.9	4.9	6.82%
6	94.1	4.7	4.99%
7	108.8	4.5	4.13%
8	63.5	2.7	4.25%
9	98.0	4.1	4.22%
10	104.9	4.1	3.91%

- For instances of size A, the optimality gap ranges from 0.41% to 6.01% (see Table XV). For instances of size B, the optimality gap ranges from 0.00% to 1.85% (see Table XVII). For instances of size C, the optimality gap ranges from 0.13% to 1.84% (see Table XIX).

These observations lead to the following conclusions:

- The optimality gap is acceptable for practical applications.
- The heuristic significantly reduces the solution time.
- The heuristic's computational efficiency allows solving problems of larger sizes than is possible using the direct application of MIP tools over their DEMs.

The heuristic's performance for SCND problems modeled as SMIPs encourages extending it to tackle its MSMIPs counterparts, which is the focus of the following chapter.

CHAPTER VII

MULTISTAGE SCND DECOMPOSITION AND HEURISTIC

Chapter IV developed P_c, a deterministic equivalent model (DEM) of a multistage stochastic mixed-integer program (MSMIP) that models the supply chain network design (SCND) problem. To reduce the size of this MSMIP, Chapter V developed an EoH decomposition that pegs the configuration of the last period for which design decisions exist to the optimal configuration for that period. The MSMIP represented by (5.2)–(5.5) results from applying this EoH decomposition to P_c. This chapter develops a practical solution method for this MSMIP. This solution method generalizes the solution procedure developed in Chapter VI for the two-stage special case of this MSMIP.

This generalized solution approach is a type-I column generation procedure, which has proved successful in solving NP-hard problems (Wilhelm, 2001). It rests on decomposing the supply chain network design (SCND) model by scenario-tree node into a conveniently structured master problem and a number of nodal subproblems as section 1 of this chapter explains.

These subproblems are still NP-hard but their (relatively) small-size allows them to be solved more efficiently. The L-shaped method particularly suits solving these subproblems, and its resulting iterative solutions provide the master problem's columns.

Section 1.2 reformulates the master problem into a Leontief substitution flow problem (LSFP). This reformulation exhibits the integrality property, which renders it amenable to efficient solution algorithms. Section 2.4 tailors one of these algorithms to exploit the special structure of the reformulated master problem and thereby further improve solution efficiency.

The results of the computational experiments presented in section 4 show that this heuristic's performance is suitable for realistic applications.

1. Nodal decomposition

To overcome mathematical intractability, the block-angular structure of P_c is exploited by Dantzig-Wolf decomposition (Dantzig and Wolfe, 1961). The resulting master problem links $|\mathcal{N}_B| - |\mathcal{N}_{T-1}|$ otherwise independent nodal subproblems.

1.1. Nodal subproblem

A nodal subproblem seeks the best SCND for a single scenario-tree node. An arbitrary nodal subproblem, P_n, is expressed as follows:

$$\kappa_n = \min\{\mathbf{g_n y_n} + \sum_{m \in \mathcal{D}(n)} \phi_{m|n} \mathbf{h_m z_m} \mid \mathbf{y_n} \in Y_n, \mathbf{D z_m} + \mathbf{B_3 y_n} = \mathbf{d_m}\}. \tag{7.1}$$

Subproblem (7.1) results from relaxing constraint (4.35).

P_n is a two-stage stochastic mixed-integer program (SMIP) in which the first stage, (7.2), selects the operational resources ($\mathbf{y_n}$) at node $n \in \mathcal{N}_B \backslash \mathcal{N}_{T-1}$. Excluding the nodes belonging to the set $\mathcal{N}_{T-1}$ results from the EoH decomposition, which tackles these nodes independently. Given $\mathbf{y_n}$ and $\omega \in \Omega$, the recourse stage, (7.3), selects the production/distribution decisions ($\mathbf{z_m}$) at nodes $m \in \mathcal{D}(n)$.

$$\kappa_n = \min\{\mathbf{g_n y_n} + Q(\mathbf{y_n}) \mid \mathbf{y_n} \in Y_n\}, \tag{7.2}$$

$$Q(\mathbf{y_n}) = E_\omega Q(\mathbf{y_n}, \omega) = \min\{\sum_{m \in \mathcal{D}(n)} \phi_{m|n} (\mathbf{h_m z_m}) \mid \mathbf{D z_m} + \mathbf{B_3 y_n} = \mathbf{d_m}\}. \tag{7.3}$$

P_n is amenable to solution by the L-shaped method (Van Slyke and Wets, 1969) for the reasons explained in section 2.1 of Chapter VI.

1.2. Master problem

The master problem tracks the evolution of the network's configuration throughout the planning horizon under all possible demand scenarios and thereby accounts for network reconfiguration costs. It results from expressing $\mathbf{y_n}, n \in \mathcal{N}_B \backslash \mathcal{N}_{T-1}$, in terms of the members of its feasible solution set as follows:

$$\mathbf{y_n} = \sum_{v \in \mathcal{V}_n} \lambda_n^v \mathbf{y_n^v}, \qquad \sum_{v \in \mathcal{V}_n} \lambda_n^v = 1, \qquad \lambda_n^v \in \{0, 1\} \quad \forall v \in \mathcal{V}_n,$$

where $\{\mathbf{y_n^v} | v \in \mathcal{V}_n\}$ is the set of these solution points, and $\mathcal{V}_n$ is its index set. Likewise, $\sum_{v \in \mathcal{V}_n} \kappa_n^v \lambda_n^v$ substitutes for the contribution of $\mathbf{y_n}$ and $\mathbf{z_m}, m \in \mathcal{D}(n)$, in the objective function; $\kappa_n^v = \mathbf{g_n} \mathbf{y_n^v} + \sum_{m \in \mathcal{D}(n)} \phi_{m|n} \mathbf{h_m} \mathbf{z_m^v}, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$. The resulting master problem, P_M, is expressed as follows:

$$\kappa = \min \sum_{n \in \mathcal{N}_B} \phi_n \mathbf{f_n} \mathbf{x_n} + \sum_{n \in \mathcal{N}_B \backslash \mathcal{N}_{T-1}} \sum_{v \in \mathcal{V}_n} \phi_n \kappa_n^v \lambda_n^v, \tag{7.4}$$

subject to

$$\mathbf{A}\mathbf{x_1} + \sum_{v \in \mathcal{V}_1} \mathbf{B_1} \mathbf{y_1^v} \lambda_1^v = -\mathbf{B_2} \mathbf{y_0}, \tag{7.5}$$

$$\mathbf{A}\mathbf{x_n} + \sum_{v \in \mathcal{V}_n} \mathbf{B_1} \mathbf{y_n^v} \lambda_n^v + \sum_{u \in \mathcal{V}_{a(n)}} \mathbf{B_2} \mathbf{y_{a(n)}^u} \lambda_{a(n)}^u = \mathbf{b} \qquad \forall\, n \in \mathcal{N}_B \setminus (\mathcal{N}_{T-1} \cup \{1\}), \tag{7.6}$$

$$\mathbf{A}\mathbf{x_n} + \sum_{u \in \mathcal{V}_{a(n)}} \mathbf{B_2} \mathbf{y_{a(n)}^u} \lambda_{a(n)}^u = -\mathbf{B_1} \mathbf{y_n} \qquad \forall\, n \in \mathcal{N}_{T-1}, \tag{7.7}$$

$$\sum_{v \in \mathcal{V}_n} \lambda_n^v = 1 \qquad \forall\, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}, \tag{7.8}$$

$$\lambda_n^v \in \{0, 1\} \qquad \forall\, v \in \mathcal{V}_n\, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}. \tag{7.9}$$

In P_M, constraints (7.8) and (7.9) dictate the selection of exactly one configuration per node. Constraint sets (7.5)–(7.7) enforce the reconfiguration actions necessary to transition between the configurations of two successive nodes. The objective function accounts for the reconfiguration costs and the costs associated with the selected configurations.

Traditionally, when a master problem involves binary variables, Dantzig-Wolf decomposition is followed by branch and price (Barnhart *et al.*, 1989), which is computationally prohibitive for SCND problems. The pricing step involves many iterations; each entails solving the NP-hard nodal subproblems. Closing the duality-gap of the master problem further magnifies the computational burden. This entails performing several iterations, each involving the pricing step. The following section circumvents this computational burden using a Leontief substitution flow problem formulation.

2. Leontief substitution flow problem

The Leontief substitution flow problem (LSFP) exhibits the integrality property and is amenable to efficient solution algorithms. To take advantage of these properties, the master problem is reformulated into a LSFP. Section 2.1 explains the logic behind this reformulation. Section 2.2 discusses the ensuing mathematical model. Section 2.3 develops the dual formulation for this LSFP model.

2.1. Equivalent Leontief substitution flow problem

P_M is mainly concerned with selecting a configuration for each node. In contrast, its LSFP reformulation focuses on the reconfiguration actions required to transition between the successive configurations that P_M selects. In other words, the optimal solution of the LSFP reformulation specifies the actions necessary to transition between the configurations described by the optimal solution of P_M. The following paragraphs use Fig. 5 and Fig. 6 to elaborate on the distinctions between P_M and P_L.

Fig. 5 depicts a master problem, P_M. The root node ($n = 0$) represents the original configuration of the supply chain network (at $t = 0$). The boundary nodes ($n_\omega \in \mathcal{N}_{T-1}$) represent the final configurations for the network. A node $n, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$, represents a nodal subproblem P_n. Vertices $v \in \mathcal{V}_n$ drawn inside a node $n, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$, represent nodal feasible solutions, $\{\mathbf{y_n^v} \mid v \in \mathcal{V}_n\}$. Each nodal feasible solution, $\mathbf{y_n^v}$, describes a possible configuration, v, for node n. Due to the strong association between a nodal feasible solution, the supply chain network configuration it describes, and the vertex that represents this solution in the hypergraph, the three terms are used interchangeably hereafter.

The optimal solution of P_M selects exactly one configuration per node such that the total cost is minimized. In Fig. 5, vertices 2, 4, 6, 7, 10, 11, and 14 represent these configurations. The reconfiguration actions required to transition between these configurations are depicted by *hyperarcs* $h_1, h_2, h_3, h_4, h_5, h_6, h_7$ and h_8. A hyperarc is an arc capable of joining more than two vertices. Graphs involving hyperarcs, such as that in Fig. 5, are referred to as *hypergraphs*. For an exposition of hypergraphs, see Koehler *et al.* (1975), Martin *et al.* (1990), and Jeroslow *et al.* (1992).

The proposed LSFP involves selecting the reconfiguration actions that achieve the optimal solution. It assumes that all possible hyperarcs are available (Fig. 6 depicts a subset of these hyperarcs since depicting all of them would make it cumbersome). The LSFP, P_L, achieves the same optimal solution of P_M by selecting a set of hyperarcs such that:

- Exactly one of the hyperarcs emanating from node n is selected, for $n \in \{0\} \cup \mathcal{N}_B \setminus \mathcal{N}_{T-2}$.

- If the selected hyperarc, h_i, emanating from node n leads to vertex $v, v \in$

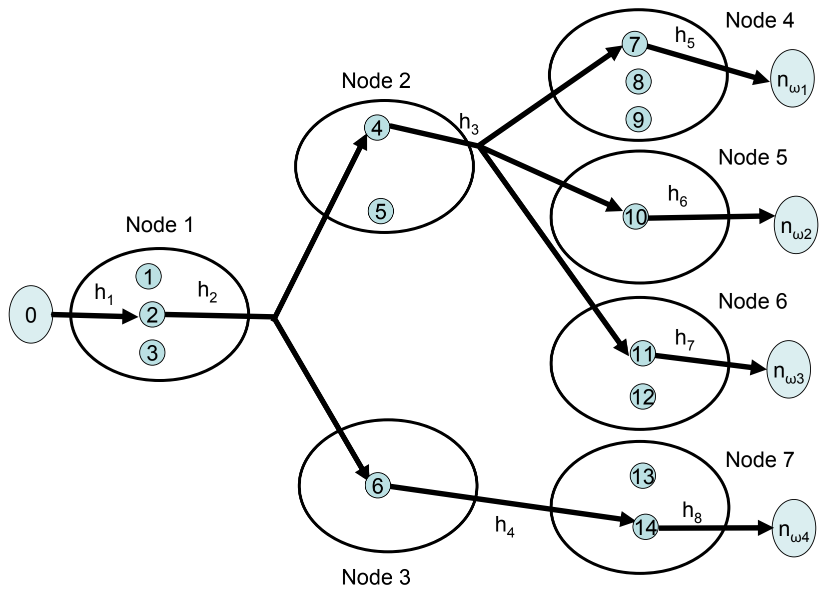

Fig. 5.: Graphical representation of the master problem, P_M.

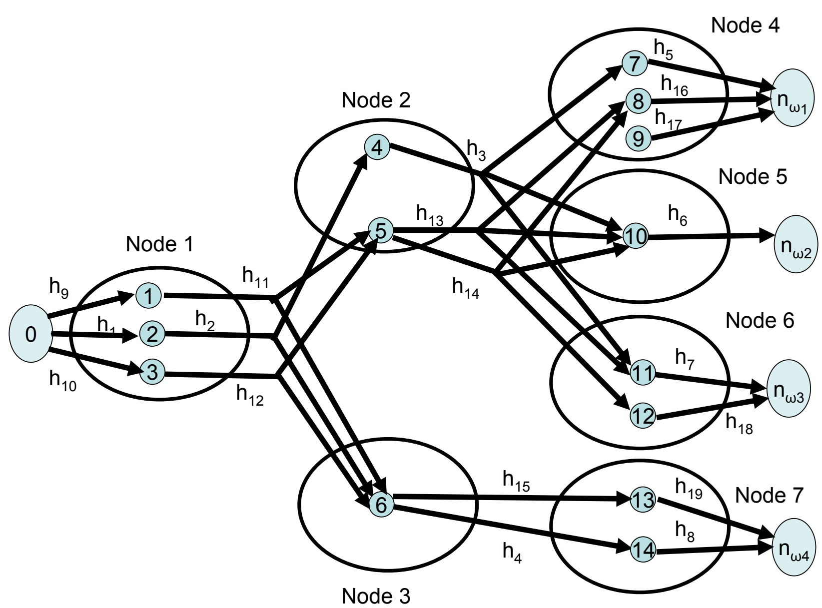

Fig. 6.: Partial graphical representation of the Leontief flow problem, P_L.

$\mathcal{V}_m, m \in \mathcal{D}(n), n \in \{0\} \cup \mathcal{N}_B \setminus \mathcal{N}_{T-3}$, then the selected hyperarc for node m must emanate from vertex v.

Combined, these two conditions are equivalent to selecting exactly one configuration per scenario-tree node.

To ease the mathematical exposition, let $\mathcal{H}$ be the index set of hyperarcs. $\mathcal{H}_v^+$ and $\mathcal{H}_v^-$ are the index sets of hyperarcs leaving and entering vertex $v \in \mathcal{V}$, respectively, where $\mathcal{V} = \cup_{n \in \mathcal{N}_B} \mathcal{V}_n$. Further, $\mathcal{H}_0^- = \mathcal{H}_v^+ = \emptyset$, where $\emptyset$ is the empty set, for $v \in \mathcal{V}_n, n \in \mathcal{N}_T$. Hyperarc h_i emanates from vertex $v_{h_i} \in \mathcal{V} \cup \mathcal{V}_0$. $\mathcal{V}_{h_i}, i \in \mathcal{H}$, denotes the set of vertices to which h_i leads. For example, in Fig. 6, $v_{h_3} = \{4\}$ and $\mathcal{V}_{h_3} = \{7, 10, 11\}$.

Hypergraphs associate a weight, c_i, with every hyperarc, h_i. Achieving equivalence between the weights of the hyperarcs and the cost of the SCND rests on the following definitions. Let $v_{h_i} \in \mathcal{V}_m$ and $v_n \in \mathcal{V}_{h_i}, n \in \mathcal{D}(m)$. Equations (7.10)–(7.12) specify the required reconfiguration actions to transition from the configuration described by $\mathbf{y}_\mathbf{m}^{\mathbf{v_{h_i}}} = \left(\{y_j^{v_{h_i}}\}_{j \in \mathcal{J}}, \{y_{\ell,q,j}^{v_{h_i}}\}_{\ell \in \{0\} \cup \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J}}\right)$ to the configuration described by $\mathbf{y}_\mathbf{n}^{\mathbf{v_n}} = \left(\{y_j^{v_n}\}_{j \in \mathcal{J}}, \{y_{\ell,q,j}^{v_n}\}_{\ell \in \{0\} \cup \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J}}\right)$.

$$x_{\text{open } j}^{v_{h_i} \to v_n} = \max\{0, y_{\text{open } j}^{v_{h_i}} - y_{\text{open } j}^{v_n}\} \qquad \forall\, j \in \mathcal{J}, \tag{7.10}$$

$$x_{\text{close } j}^{v_{h_i} \to v_n} = \max\{0, -y_{\text{close } j}^{v_{h_i}} + y_{\text{close } j}^{v_n}\} \qquad \forall\, j \in \mathcal{J}, \tag{7.11}$$

$$x_{\ell_1,\ell_2,q,j}^{v_{h_i} \to v_n} = \max\{0, y_{\ell_2,q,j}^{v_{h_i}} + y_{\ell_1,q,j}^{v_n} - 1\} \qquad \forall\, \ell_1 \neq \ell_2 \in \mathcal{L}_q, q \in \mathcal{Q}_j, j \in \mathcal{J}. \tag{7.12}$$

$\mathbf{x}_\mathbf{n}^{\mathbf{v_{h_i}} \to \mathbf{v_n}}$ groups $x_{\text{open } j}^{v_{h_i} \to v_n}$, $x_{\text{close } j}^{v_{h_i} \to v_n}$, and $x_{\ell_1,\ell_2,q,j}^{v_{h_i} \to v_n}$ in a single vector as follows:

$$\mathbf{x}_\mathbf{n}^{\mathbf{v_{h_i}} \to \mathbf{v_n}} = \left(\{x_{\text{open } j}^{v_{h_i} \to v_n}\}_{j \in \mathcal{J}}, \{x_{\text{close } j}^{v_{h_i} \to v_n}\}_{j \in \mathcal{J}}, \{x_{\ell_1,\ell_2,q,j}^{v_{h_i} \to v_n}\}_{\ell_1 \neq \ell_2 \in \{0\} \cup \mathcal{L}_j, q \in \mathcal{Q}_j, j \in \mathcal{J}}\right).$$

Moreover, define the reconfiguration cost between vertex $v_{h_i} \in \mathcal{V}_{a(n)}$ to vertex $v_n \in$

$\mathcal{V}_n$ as follows:

$$f_n^{v_{h_i} \rightarrow v_n} = \mathbf{f_n x_n}^{\mathbf{v_{h_i}} \rightarrow \mathbf{v_n}},$$

where $\mathbf{f_n} = \left(\{f^n_{\text{open } j}\}_{j \in \mathcal{J}}, \{f^n_{\text{close } j}\}_{j \in \mathcal{J}}, \{f^n_{\ell_1,\ell_2,q,j}\}_{\ell_1 \neq \ell_2 \in \{0\} \cup \mathcal{L}_j, q \in \mathcal{Q}_j, j \in \mathcal{J}}\right)$.

Thus, the cost associated with a given hypergraph, h_i, can be calculated by (7.13), where $\kappa_n^{v_n}$ is the cost associated with $\mathbf{y_n^{v_n}}$ as defined by (7.1).

$$c_i = \sum_{v_n \in \mathcal{V}_{h_i}} \phi_n \left(f_n^{v_{h_i} \rightarrow v_n} + \kappa_n^{v_n} \right). \tag{7.13}$$

2.2. Reformulated master problem

Problem P_L seeks the shortest *hyperpath* between the root and boundary nodes of the scenario-tree. A hyperpath, $\boldsymbol{\beta}$, is a subset of hyperarcs that form a tree rooted at the scenario-tree root node and ending at the scenario-tree boundary nodes; $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_H)$ where β_i is a binary variable indicating whether hyperarc $h_i, i \in \mathcal{H}$, takes part of the hyperpath ($\beta_i = 1$) or not ($\beta_i = 0$), and $H = |\mathcal{H}|$. A shortest hyperpath, $\boldsymbol{\beta^*}$, is a hyperpath with minimum weight; i.e., $\mathbf{c}\boldsymbol{\beta^*} \leq \mathbf{c}\boldsymbol{\beta^j}$ for all possible hyperpaths $\boldsymbol{\beta^j}$, where $\mathbf{c} = (c_1, \ldots, c_H)$.

The shortest hyperpath problem is a LSFP and can be expressed by the following linear program, P_L:

$$\kappa_L = \min \sum_{i \in \mathcal{H}} c_i \beta_i, \tag{7.14}$$

subject to

$$\sum_{i \in \mathcal{H}^+_{v_0}} \beta_i = 1, \tag{7.15}$$

$$\sum_{i \in \mathcal{H}^+_v} \beta_i - \sum_{i \in \mathcal{H}^-_v} \beta_i = 0 \qquad \forall\, v \in \mathcal{V}_n, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}, \tag{7.16}$$

$$\beta_i \geq 0 \qquad \forall\, i \in \mathcal{H}. \tag{7.17}$$

The objective function (7.14) minimizes the cost of the selected hyperarcs. The first constraint (7.15) ensures that exactly one unit of flow leaves the root node. Constraint set (7.16) forces a flow leaving the root node to eventually reach the boundary nodes by ensuring conservation of flow at each node (except root and boundary nodes). The nonnegativity constraint set (7.17) suffices to achieve binary variables due to the integrality property of the LSFP.

P_L is a LSFP because it exhibits the following characteristics:

- Elements of the constraint-matrix are either $0, +1$, or -1.
- The constraint-matrix has exactly one positive element per column.
- Elements of the right-hand side are binary.
- The corresponding hypergraph (Fig. 6) is free of *paracycles* due to its underlying scenario-tree structure.

Proposition 3 *The linear program P_L, represented by* (7.14)–(7.17)*, has a binary optimal solution.*

Proof: In vector form, P_L can be expressed as $\kappa_L = \min_{\boldsymbol{\beta} \geq \mathbf{0}} \{\mathbf{c}\boldsymbol{\beta} \mid \mathbf{H}\boldsymbol{\beta} = \mathbf{b}\}$. Matrix $\mathbf{H}$ is pre-Leontief since it has exactly one positive element per column (Veinott, 1968). Furthermore, $\mathbf{H}\boldsymbol{\beta} = \mathbf{b}$ is a pre-Leontief substitution system since $\mathbf{H}$ is pre-Leontief, $\mathbf{b} \geq \mathbf{0}$ and $\boldsymbol{\beta} \geq \mathbf{0}$ (Veinott, 1968). By construction, this system is free of paracycles (due to its underlying scenario-tree structure). Consequently, $\min_{\boldsymbol{\beta} \geq \mathbf{0}} \{\mathbf{c}\boldsymbol{\beta} \mid \mathbf{H}\boldsymbol{\beta} = \mathbf{b}\}$ is a Leontief substitution flow problem, and its optimal solution is integral (Veinott, 1968). Furthermore, this optimal solution is binary since the elements of $\mathbf{b}$ are binary and the elements of $\mathbf{H}$ are either $0, +1$, or -1 (Jeroslow *et al.*, 1992). ■

Hyperarc $h_i, i \in \mathcal{H}$, belongs to the shortest hyperpath if the optimal solution of P_L indicates that $\beta_i^* = 1$. This implies that $\lambda_n^{v_{h_i}} = \lambda_m^{v_m} = 1, v_m \in \mathcal{V}_{h_i}, m \in \mathcal{D}(n)$, in the optimal solution of P_M. This equivalence between the optimal solution of P_M and that of P_L results from the following proposition.

Proposition 4 *The shortest hyperpath problem P_L, represented by* (7.14)–(7.17), *and the master problem P_M, represented by* (7.4)–(7.9), *are equivalent.*

Proof: This proof consists of the following two steps.

1. Show that constraints (7.8) and (7.9) are equivalent to (7.15)–(7.17).

2. Show that $\min\left\{\sum_{n\in\mathcal{N}_B}\phi_n f_n x_n + \sum_{n\in\mathcal{N}_B\setminus\mathcal{N}_{T-1}}\sum_{v\in\mathcal{V}_n}\phi_n\kappa_n^v\lambda_n^v \mid (7.5),(7.6),(7.7)\right\}$ is equivalent to $\min\left\{\sum_{i\in\mathcal{H}} c_i\beta_i \mid \beta_i \geq 0\right\}$.

To show that constraints (7.8) and (7.9) are equivalent to (7.15)–(7.17), consider node $n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$, and its descendants $m \in \mathcal{D}(n)$. $\sum_{u_m\in\mathcal{V}_m}\lambda_m^{u_m} = 1, m \in \mathcal{D}(n)$ by constraint (7.8). This results in the following equation:

$$\prod_{m\in\mathcal{D}(n)}\left(\sum_{u_m\in\mathcal{V}_m}\lambda_m^{u_m}\right) = 1.$$

Multiplying both sides of this equation by λ_n^v for a given $v \in \mathcal{V}_n$ results in the following relationship:

$$\lambda_n^v\prod_{m\in\mathcal{D}(n)}\left(\sum_{u_m\in\mathcal{V}_m}\lambda_m^{u_m}\right) = \lambda_n^v.$$

Expanding the left hand side results into the sum of $\prod_{m\in\mathcal{D}(n)}|\mathcal{V}_m|$ unique terms. Let $\mathcal{H}_v^+$ be an index set, where $\mathcal{H}_v^+ = \{1,\ldots,H_v^+\}$, and let $H_v^+ = \prod_{m\in\mathcal{D}(n)}|\mathcal{V}_m|$. Let β_i be an arbitrary term of this sum; $\beta_i = \lambda_n^v\prod_{m\in\mathcal{D}(n)}\lambda_m^{u_m}$, for an arbitrary $u_m \in \mathcal{V}_m$, for each $m \in \mathcal{D}(n)$. This results in the following equation:

$$\sum_{i\in\mathcal{H}_v^+}\beta_i = \lambda_n^v\prod_{n\in\mathcal{D}(m)}\left(\sum_{u_m\in\mathcal{V}_m}\lambda_n^{u_m}\right) = \lambda_n^v \qquad \forall v\in\mathcal{V}_n, n\in\{0\}\cup\mathcal{N}_B\setminus N_{T-1}. \quad (7.18)$$

Equation 7.19 results from applying a similar logic on node $e = a(n)$, and its descendants $n_g \in \mathcal{D}(e)$, where $n = n_g$ for an arbitrary g.

$$\sum_{i \in \mathcal{H}_v^-} \beta_i = \lambda_n^v \left(\sum_{w \in \mathcal{V}_e} \lambda_e^w \right) \prod_{n_g \in \mathcal{D}(e) \setminus \{n\}} \left(\sum_{v_{n_g} \in \mathcal{V}_{n_g}} \lambda_{n_g}^{v_{n_g}} \right) = \lambda_n^v \qquad \forall v \in \mathcal{V}_n, n \in \mathcal{N}_B. \tag{7.19}$$

Constraint (7.16) results from (7.18) and (7.19). Constraint (7.17) results from defining β_i as the product of binary terms; $\beta_i = \lambda_n^v \prod_{m \in \mathcal{D}(n)} \lambda_m^{u_m}$, for an arbitrary $u_m \in \mathcal{V}_m$, for each $m \in \mathcal{D}(n)$.

Finally, $\lambda_0^1 = 1$ results from the uniqueness of the inherited initial configuration, and constraint (7.15) results from substituting $\lambda_n^v = \lambda_0^1 = 1$ in equation (7.18). This completes the first step of the proof.

The second step of this proof follows from relations (7.21), (7.23), and (7.24), which will be constructed one at a time. In the mathematical program (7.4)–(7.9), $\mathbf{y_0}$ and $\mathbf{y_n}, n \in \mathcal{N}_{T-1}$ are given parameters. An equivalent representation is to express them as decision variables, and define $\mathcal{V}_0$ and $\mathcal{V}_n$ such that $|\mathcal{V}_0| = |\mathcal{V}_n| = 1, n \in \mathcal{N}_{T-1}$. In this case, $\kappa_0^{v_0} = \kappa_n^{v_n} = 0, n \in \mathcal{N}_{T-1}$. The resulting mathematical program, P'_M, is expressed as follows.

$$\kappa = \min \sum_{n \in \mathcal{N}_B} \left(\phi_n \mathbf{f_n} \mathbf{x_n} + \sum_{v \in \mathcal{V}_n} \phi_n \kappa_n^v \lambda_n^v \right),$$

subject to

$$\mathbf{A}\mathbf{x_n} + \sum_{v \in \mathcal{V}_n} \mathbf{B_1} \mathbf{y_n^v} \lambda_n^v + \sum_{u \in \mathcal{V}_{a(n)}} \mathbf{B_2} \mathbf{y_{a(n)}^u} \lambda_{a(n)}^u = \mathbf{b} \qquad \forall\, n \in \mathcal{N}_B, \tag{7.20}$$

$$\sum_{v \in \mathcal{V}_n} \lambda_n^v = 1 \qquad \forall\, n \in \mathcal{N}_B,$$

$$\lambda_n^v \in \{0, 1\} \qquad \forall\, v \in \mathcal{V}_n,\ n \in \mathcal{N}_B.$$

Therefore, by construction,

$$P_M \equiv P'_M. \tag{7.21}$$

P'_M expresses the root and boundary nodes like all other nodes. This circumvents the effort that would have been otherwise required to prove the boundary conditions, (7.5) and (7.7), independently.

For any node $m \in \mathcal{D}(n), n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$, by inspection of constraints (4.4)–(4.15) and equations (7.10)–(7.12), it can be shown that, for a given $u \in \mathcal{V}_n$ and $v_m \in \mathcal{V}_m$, $\mathbf{f_m x_m} = \lambda_n^u \lambda_m^{v_m} \mathbf{f_m x_m^{u \to v_m}} = \lambda_n^u \lambda_m^{v_m} f_n^{u \to v_m}$. This leads to the following result, where $\mathbf{x_m}$ is defined by (7.20) for a given $\lambda_m^{v_m}$ and a given λ_n^u.

$$\lambda_n^u \lambda_m^{v_m} \phi_m \left(f_m^{u \to v} + \kappa_m^{v_m} \right) = \phi_m \left(\mathbf{f_m x_m} + \lambda_m^{v_m} \kappa_m^{v_m} \right).$$

Summing both sides of this equation over all configurations of a node n ($u \in \mathcal{V}_n$) and its descendants $m \in \mathcal{D}(n)$ ($v_m \in \mathcal{V}_{v_m}$), leads to the following equation, where $\mathbf{x_m}$ is defined by (7.20) for a given n.

$$\sum_{m \in \mathcal{D}(n)} \sum_{u \in \mathcal{V}_n} \sum_{v_m \in \mathcal{V}_m} \lambda_n^u \lambda_m^{v_m} \phi_m \left(f_m^{u \to v} + \kappa_m^{v_m} \right) = \sum_{m \in \mathcal{D}(n)} \sum_{v_m \in \mathcal{V}_m} \phi_m \left(\mathbf{f_m x_m} + \lambda_m^{v_m} \kappa_m^{v_m} \right). \tag{7.22}$$

Further, summing both sides of equation (7.22) over nodes $n \in \mathcal{N}_B \cup \{0\} \setminus \mathcal{N}_{T-1}$ leads to the following result, where $\mathbf{x_n}$ is defined by (7.20).

$$\sum_{n \in \mathcal{N}_B \cup \{0\} \setminus \mathcal{N}_{T-1}} \sum_{m \in \mathcal{D}(n)} \sum_{u \in \mathcal{V}_n} \sum_{v_m \in \mathcal{V}_m} \lambda_n^u \lambda_m^{v_m} \phi_m \left(f_m^{u \to v_m} + \kappa_m^{v_m} \right) = \sum_{n \in \mathcal{N}_B} \phi_n \left(\mathbf{f_n x_n} + \sum_{v_n \in \mathcal{V}_n} \lambda_m^{v_m} \kappa_m^{v_m} \right). \tag{7.23}$$

Finally, the last result relates the left-hand-side of (7.23) to the cost of hyperarcs.

$$\sum_{i\in\mathcal{H}_u^+} c_i\beta_i = \lambda_n^u \left(\prod_{v_m\in\mathcal{V}_{h_i}} \lambda_m^{v_m}\right)\left(\sum_{v_m\in\mathcal{V}_{h_i}} \phi_m\left(f_m^{u\to v_m} + \kappa_m^{v_m}\right)\right).$$

The following equation rests on $\sum_{v_m\in\mathcal{V}_m} \lambda_m^{v_m} = 1$ and $\lambda_m^{v_m} \in \{0,1\}$.

$$\sum_{i\in\mathcal{H}_u^+} c_i\beta_i = \sum_{u\in\mathcal{V}_n}\sum_{v_m\in\mathcal{V}_m} \lambda_n^u\lambda_m^{v_m}\phi_m\left(f_m^{u\to v_m} + \kappa_m^{v_m}\right).$$

Summing both sides of this equation over all vertices $u \in \mathcal{V}_n$ and all $n \in \mathcal{N}_B \cup \{0\} \setminus \mathcal{N}_{T-1}$ leads to the following equation.

$$\sum_{i\in\mathcal{H}} c_i\beta_i = \sum_{n\in\mathcal{N}_B\cup\{0\}\setminus\mathcal{N}_{T-1}}\sum_{m\in\mathcal{D}(n)}\sum_{u\in\mathcal{V}_n}\sum_{v_m\in\mathcal{V}_m} \lambda_n^u\lambda_m^{v_m}\phi_m\left(f_m^{u\to v_m} + \kappa_m^{v_m}\right). \tag{7.24}$$

Finally, combining (7.24), (7.23), and (7.21) completes the proof.

■

P_L expresses hyperarcs explicitly. As the number of vertices per node increases, the number of hyperarcs grows exponentially. Consequently, expressing hyperarcs implicitly in terms of their vertices improves computational efficiency. This is attained by the dual formulation of P_L.

2.3. Dual formulation

Let ψ_0 be the dual variable associated with constraint (7.15), and $\psi_v, v \in \mathcal{V}$, be those associated with constraint set (7.16). The dual problem of P_L, P_L^D, can be expressed by the following linear program.

$$\kappa_L^D = \max \psi_0 \tag{7.25}$$

Subject to

$$\psi_{v_{h_i}} - \sum_{v\in\mathcal{V}_{h_i}} \psi_v \le c_i \qquad \forall\, i \in \mathcal{H}_v^+, v \in \mathcal{V}_n, n \in \{0\} \cup \mathcal{N}_B \setminus \left(\mathcal{N}_{T-1} \cup \mathcal{N}_{T-2}\right), \tag{7.26}$$

$$\psi_{v_{h_i}} \leq c_i \qquad \forall\, i \in \mathcal{H}_v^+, v \in \mathcal{V}_n, n \in \mathcal{N}_{T-2}, \tag{7.27}$$

$$\psi_v \text{ unrestricted} \qquad \forall\, v \in \mathcal{V}_n, n \in \{0\} \cup \mathcal{N}_B \setminus \mathcal{N}_{T-1}. \tag{7.28}$$

Constraint sets (7.26) and (7.27) imply that ψ_v is the cost of the shortest hyperpath connecting vertex v to the boundary nodes of the scenario-tree. Consistent with the terminology traditionally used in network problems, ψ_v will be referred to as the *potential* of vertex v. Constraint (7.28) indicates that ψ_v can take negative values, which is consistent with how P_c was constructed (refer to section 4.3 of Chapter IV for a discussion about the likelihood of negative values for $h_{p,j_w,k}$ as a result of combining the revenue per unit of product p and the transportation cost to market k in one parameter). Thus, $\psi_{v_{h_i}}$ will take a negative value whenever $-\sum_{v \in \mathcal{V}_{h_i}} \psi_v \geq c_i$. The objective function (7.25) calculates the cost of the shortest hyperpath. If $\kappa_L^D < 0$, the supply chain is profitable.

2.4. Shortest hyperpath algorithm

Martin *et al.* (1990) show that the optimal solution for P_L^D, $\psi_v^*, v \in \{0\} \cup \mathcal{V}$, can be recursively calculated as follows:

$$\psi_v^* = \min\{c_i + \sum_{u \in \mathcal{V}_{h_i}} \psi_u^* \mid i \in \mathcal{H}_v^+\}. \tag{7.29}$$

Martin *et al.* (1990) also develop a polynomial-time, shortest hyperpath algorithm, which uses an explicit list of hyperarcs as input. As applied to P_L, their algorithm scans the hypergraph backwards (starting from boundary nodes and progressing towards the root node). At each vertex, only hypergraphs emanating from this vertex are examined, and the one associated with the least expensive hyperpath towards the boundary nodes is selected. The complexity of this algorithm is $\mathcal{O}(|\mathcal{H}|)$ since each hyperarc is visited exactly once. Expressed in terms of nodes and vertices, this shortest hyperpath algorithm's complexity is $\mathcal{O}\left(|\mathcal{N}|\widehat{|\mathcal{V}_n|}^{\widehat{|\mathcal{D}(n)|}+1}\right)$, where $\widehat{|\mathcal{D}(n)|} = \max_{n\in\mathcal{N}}\{|\mathcal{D}(n)|\}$ and $\widehat{|\mathcal{V}_n|} = \max_{n\in\mathcal{N}}\{|\mathcal{V}_n|\}$.

This section customizes this shortest hyperpath algorithm to the special structure of P_L and thereby reduces its complexity to $\mathcal{O}(|\mathcal{N}|\widehat{|\mathcal{D}(n)|}\widehat{|\mathcal{V}_n|}^2)$. Specifically, the ability to divide the cost of a hyperarc into independent components allows for constructing fewer hyperarcs. By construction, equation (7.13) expresses the cost associated with a hyperarc as the probability weighted sum of a transition cost from vertex u at node n to each of the vertices at node m, $v_m \in \mathcal{V}_m, m \in \mathcal{D}(n)$. This cost structure allows treating a hyperarc as a virtual collection of arcs, each leading from vertex u at node n to one of the vertices $v_m \in \mathcal{V}_m$ of one of its descendant nodes $m \in \mathcal{D}(n)$.

Instead of enumerating all possible hyperarcs, this treatment allows for constructing the best hyperarc to transition between these configurations. This hyperarc results from selecting from each descendant node m a vertex v_m that produces the least cheap arc between u and v_m. Grouping the least cheap arc for each descendant forms a hyperarc. This implicit representation of a hyperarc reduces the number of evaluated hyperarcs per vertex u from $\prod_{m\in\mathcal{D}(n)}|\mathcal{V}_m|$ to $|\mathcal{D}(n)||\mathcal{V}_m|$.

To apply this concept analytically, equation (7.30) defines the cost $\psi_{u\to m}$ associated with the transition from configuration u_n to the best configuration of node $m \in \mathcal{D}(n)$.

$$\psi_{u\to m} = \min_{\lambda_m^v \in \mathbb{B}} \left\{ \sum_{v\in\mathcal{V}_m} \left(\phi_m\left(f_m^{u\to v} + \kappa_m^v\right) + \psi_v\right)\lambda_m^v \mid \sum_{v\in\mathcal{V}_m} \lambda_m^v = 1 \right\}. \quad (7.30)$$

Proposition 5 establishes the customized recursive relationship for P_L^D.

Proposition 5 *The optimal solution for P_L^D can be recursively calculated as follows:*

$$\psi_u^* = \sum_{m \in \mathcal{D}(n)} \psi_{u \to m} \qquad \forall u \in \{0\} \cup \mathcal{V}_n, n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1},$$

where $\psi_{u \to m} = \min_{\lambda_m^v \in \mathbb{B}} \left\{ \sum_{v \in \mathcal{V}_m} \left(\phi_m \left(f_m^{u \to v} + \kappa_m^v \right) + \psi_v \right) \lambda_m^v \mid \sum_{v \in \mathcal{V}_m} \lambda_m^v = 1 \right\}.$

Proof: Martin *et al.* (1990) proves that the optimal solution for P_L^D can be calculated using the recursive relation (7.29). The following equation results from substituting equation (7.13) for c_i in this recursive relation.

$$\psi_u^* = \min_{i \in \mathcal{H}_u^+} \left\{ \sum_{v_m \in \mathcal{V}_{h_i}} \left(\phi_n \left(f_m^{u \to v_m} + \kappa_m^v \right) + \psi_{v_m} \right) \right\}.$$

The following equation results because $\mathcal{V}_m \cap \mathcal{V}_{h_i} = v_m$.

$$\psi_u^* = \min_{\lambda_m^{v_m} \in \mathbb{B}} \left\{ \sum_{m \in \mathcal{D}(n)} \sum_{v_m \in \mathcal{V}_m} \left(\phi_m \left(f_m^{u \to v_m} + \kappa_m^{v_m} \right) + \psi_v \right) \lambda_m^{v_m} \mid \sum_{v_m \in \mathcal{V}_m} \lambda_m^{v_m} = 1 \right\}.$$

By rearranging the summation terms, the following equation results.

$$\psi_u^* = \left\{ \sum_{m \in \mathcal{D}(n)} \min_{\lambda_m^{v_m} \in \mathbb{B}} \sum_{v_m \in \mathcal{V}_m} \left(\phi_m \left(f_m^{u \to v_m} + \kappa_m^{v_m} \right) + \psi_v \right) \lambda_m^{v_m} \mid \sum_{v_m \in \mathcal{V}_m} \lambda_m^{v_m} = 1 \right\}.$$

Finally, by invoking equation (7.30), we get the desired result:

$$\psi_u^* = \sum_{m \in \mathcal{D}(n)} \psi_{u \to m}.$$

■

Fig. 7 summarizes this customized shortest hyperpath algorithm. The algorithm starts by initializing $\psi_u = 0, u \in \mathcal{V}_{T-1}$ and $\psi_{u \to m} = \infty$ for all u and m combinations. Afterwards, it scans the scenario-tree from boundary nodes to root node to recursively calculate vertices' potentials as proposition 5 instructs. $\gamma_u = (\gamma_{u \to n}, n \in \mathcal{D}(m))$ enables tracing the shortest hyperpath, β^*, by storing

```
Data: φ_m, f_m^{u→v}, κ_m^v ∀v ∈ V_m, u ∈ V_n, m ∈ D(n), n ∈ {0} ∪ N_B \ N_{T−1}
Result: ψ_v, v ∈ {0} ∪ V
begin
    foreach u ∈ V_n, n ∈ N_{T−1} do
        ψ_u ⟵ 0;
    end
    foreach t ∈ {T − 2, . . . , 0} do
        foreach u ∈ V_n, n ∈ N_t do
            foreach m ∈ D(n) do
                ψ_{u→m} ⟵ ∞;
                foreach v ∈ V_m do
                    if ψ_{u→m} > φ_m (f_m^{u→v} + κ_m^v) + ψ_v then
                        ψ_{u→m} ⟵ φ_m (f_m^{u→v} + κ_m^v) + ψ_v;
                        γ_{u→m} ⟵ v;
                    end
                end
            end
            ψ_u ⟵ Σ_{m∈D(n)} ψ_{u→m};
        end
    end
end
```

Fig. 7.: Customised shortest hyperpath algorithm.

$v \in \mathcal{V}_{h_i}$, where $h_i \in \beta^*$.

Proposition 6 *The customised shortest hyperpath algorithm converges to the optimal solution of P_M in a finite number of iterations.*

Proof: The algorithm converges in a finite number of iterations since it visits each vertex exactly once, and the number of vertices is finite. Proposition 5 proves that the selected hyperpath is the optimal solution for P_L^D. By strong duality theorem,

$\kappa_L^D = \kappa_L$. Proposition 4 proves that $\kappa_L = \kappa_M$, which completes the proof. ■

The speed and simplicity of this shortest hyperpath algorithm allow solving large master problems. Thereby, this algorithm is a corner stone in the SCND heuristic outlined in the following section.

3. MSMIP heuristic for the SCND problem

The developed solution procedure is a type-I column generation procedure, and thus it consists of two steps:

1. Approximate the solution space of the original problem P_c with a set of feasible configurations.

2. Use a restricted master problem to evaluate the interdependencies among different scenario-tree nodes and select the best combination of generated configurations, one configuration per node.

As such, this procedure doesn't guarantee an optimal solution. However, the results of the computational experiments of section 4 reveal satisfactory performance, which is consistent with the results of type-I column generation procedures for other NP-hard problems (Wilhelm, 2001).

Approximating the solution space is achieved by applying the L-shaped method on each nodal subproblem (7.1) for scenario-tree nodes $n \in \mathcal{N}_B \setminus \mathcal{N}_{T-1}$. The L-shaped method generates iterative solutions in the process of finding a nodal optimal solution. These iterative solutions are feasible and thus serve as columns in the restricted master problem.

As discussed in section 4 of Chapter VI, these iterative solutions serve multiple functions:

1. They populate the columns associated with their node in the restricted master problem.

2. They also serve as columns for all other nodes as a result of the relatively complete recourse property of the subproblems.

3. They jump start the L-shape procedure for all following subproblems.

As a result, as the solution procedure progresses, the required computation effort declines.

Numerous variants of the Benders cuts exist. This heuristic adopts the variant suggested by Geoffrion and Graves (1974) for the reasons explained section 4 of Chapter VI.

The shortest hyperpath is constructed as explained in section 2.4. Due to using a restricted set of vertices per node, the optimality of the resulting solution is not guaranteed. However, the results of the computational experiments presented in the following chapter indicates a performance suitable for solving practical problems.

4. Computational experiments

This computational experiment applies this heuristic over three different problem sizes. Table II (section 5 of Chapter V) summarizes the sizes of these problems, which are refered to hereafter as size A, size B and size C. Ten instances are tested per problem size. Each instance is solved three different times to achieve the followings:

- Optimize the DEM using CPLEX 11.0.

- Apply the EoH decomposition, which uses CPLEX 11.0 to optimize its subproblems.

- Apply this heuristic, which uses CPLEX 11.0 to solve the nodal subproblems.

All computational experiments were conducted on a quad-core Intel Xeon X5355 processor running at 2.66 GHz with 12 GB RAM. Problems of size A were allowed unlimited run time under both the DEM and the heuristic. Because it takes a considerable amount of time to optimize a DEM of the second or third problem size, their optimization was halted after one and two hours, respectively. The solution values listed in their corresponding tables indicate the best solutions achieved within these allotted times. The EoH decomposition always finished within these allotted times. The cost discount rate in all instances was 2% and the planning horizon was considered infinite. All other paramters were generated as described by Appendix G.

Tables XXI and XXII summarize the results for the instances of problem size A. Tables XXIII and XXIV summarize the results for the instances of problem size B. Tables XXV and XXVI summarize the results for the instances of problem size C.

The following features can be observed in the results:

- For instances of size A, the heuristic's solution time ranges from 0.45% to 1.68% of that of the DEM (see Table XXII). For instances of size B where the optimization of the DEM was halted after one hour, the heuristic's solution time ranges from 9% to 26% of that of the DEM (see Table XXIV). For instances of size C where the optimization of the DEM was halted after two hours, the heuristic's solution time ranges from 15% to 31% of that of the DEM (see Table XXVI).

- For instances of size A, the optimality gap ranges from 0.00% to 5.42%, as Table XXI shows. For instances of size B, because the DEM did not reach an optimal solution in the allotted time, an upper bound on the optimality gap is calculated using an upper bound on the profit of the DEM. This upper bound is based on the profit associated with the solution generated by the EoH decomposition and the upper bound on the optimality gap generated for the EoH decomposition. The upper bound on this heuristic's optimality gap ranges from 0.00% to 2.16% (see Table XXIII). For instances of size C, the DEM did not reach an optimal solution within the allotted time and a bound on the EoH was not generated. Thus, an approximation for the optimality gap is calculated based on the EoH solution value (which could be inferior to the DEM solution value). This approximation for the optimality gap ranges from 0.00% to 1.60% (see Table XXV).

- For instances of size B, the heuristic resulted in a 1.22% to 5.07% improvement in profit over that of the DEM, which did not achieve an optimal solution in the allotted time. For instances of size C, the heursitic resulted in a 6.13% to 14.54% improvement in profit over the DEM, which did not achieve an optimal solution within the allotted time.

These observations lead to the following conclusions:

- The optimality gap is acceptable for practical applications.
- The heuristic significantly reduces the solution time.
- The heuristic's computational efficiency allows solving problems of larger sizes, which enables modeling SCND problems more realistically than was possible with previous models.
- The heuristic produces a improvement in profit over the DEM when it is halted after exceeding the allotted runtime.

Table XXI. Solution values for the instances of a MSMIP for a SCND problem of size A

Instance	DEM profit ($ 1,000)	Heuristic profit ($ 1,000)	Optimality gap
1	25,255.4	24,365.4	3.52%
2	396.1	377.8	4.63%
3	934.9	933.4	0.17%
4	3,934.3	3,887.1	1.20%
5	34,302.9	34,296.1	0.02%
6	19,556.7	19,332.1	1.15%
7	122,160.4	115,534.7	5.42%
8	44,106.7	42,671.5	3.25%
9	16,033.8	15,747.9	1.78%
10	29,104.2	29,104.2	0.00%

Table XXII. Solution times for the instances of a MSMIP for a SCND problem of size A

Instance	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
1	64.10	0.29	0.45%
2	46.69	0.24	0.51%
3	37.82	0.41	1.08%
4	49.83	0.16	0.32%
5	43.00	0.52	1.19%
6	52.42	0.06	0.12%
7	28.39	0.33	1.15%
8	61.60	0.20	0.32%
9	57.33	0.05	0.09%
10	35.62	0.60	1.68%

Table XXIII. Solution values for the instances of a MSMIP for a SCND problem of size B

Instance	Upper bound on DEM profit ($ 1,000)	Heuristic profit ($ 1,000)	Upper bound on optimality gap	DEM profit at one hour runtime ($ 1,000)	Heuristic's profit improvement over DEM
1	36,731.4	35,953.2	2.12%	34,776.9	3.27%
2	312.9	306.3	2.11%	302.4	1.27%
3	15,671.9	15,436.0	1.51%	15,071.5	2.36%
4	115,831.1	113,333.7	2.16%	111,946.5	1.22%
5	18,895.5	18,655.7	1.27%	18,311.9	1.84%
6	1,434.5	1,421.3	0.92%	1,375.9	3.19%
7	31,847.1	31,598.5	0.78%	29,995.7	5.07%
8	273,111.9	269,342.3	1.38%	265,016.5	1.61%
9	12,968.8	12,968.8	0.00%	12,522.0	3.45%
10	3,874.3	3,790.5	2.16%	3,628.6	4.27%

Table XXIV. Solution times for the instances of a MSMIP for a SCND problem of size B

Instance	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM	Nodal Subproblems runtime (minutes)	LSFP runtime (minutes)
1	60.0	5.6	9%	5.4	0.2
2	60.0	11.8	20%	11.4	0.4
3	60.0	15.0	25%	14.6	0.4
4	60.1	12.4	21%	11.9	0.5
5	60.0	9.9	16%	9.5	0.4
6	60.0	11.7	19%	11.3	0.4
7	60.1	15.5	26%	15.1	0.4
8	60.0	14.2	24%	13.9	0.3
9	60.0	5.8	10%	5.5	0.3
10	60.1	13.2	22%	12.6	0.6

Table XXV. Solution values for the instances of a MSMIP for a SCND problem of size C

Instance	EoH decomposition profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Approximation for optimality-gap	DEM profit at two hours runtime ($ 1,000)	Heuristic's profit improvement over DEM
1	289,398.7	289,398.7	0.00%	271,662.0	6.13%
2	700,759.4	691,054.3	1.38%	614,758.2	11.04%
3	9,337.6	9,187.9	1.60%	8,284.5	9.83%
4	53,868.1	53,045.0	1.53%	47,209.9	11.00%
5	186,957.2	186,842.0	0.06%	161,587.1	13.52%
6	7,713.8	7,597.0	1.51%	6,754.8	11.09%
7	67,684.5	66,741.0	1.39%	58,864.9	11.80%
8	507,613.4	501,569.6	1.19%	436,309.6	13.01%
9	47,086.6	47,050.8	0.08%	40,212.0	14.54%
10	787,771.6	776,861.8	1.38%	684,766.5	11.85%

Table XXVI. Solution times for the instances of a MSMIP for a SCND problem of size C

Instance	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM	Nodal subproblems runtime (minutes)	LSFP runtime (minutes)
1	120.1	20.8	17%	20.3	0.5
2	120.1	27.5	23%	27.1	0.4
3	120.0	20.5	17%	19.8	0.6
4	120.0	26.9	22%	26.4	0.5
5	120.1	21.9	18%	21.2	0.7
6	120.0	33.2	28%	32.6	0.6
7	120.0	17.7	15%	17.0	0.7
8	120.1	37.5	31%	37.0	0.5
9	120.0	23.9	20%	23.2	0.7
10	120.1	18.8	16%	18.3	0.5

5. Advantage of MSMIP SCND over SMIP

MSMIPs are more computationally demanding than SMIP. This explains the prevelance of SMIP models in SCND and the scarcity of those using MSMIP (see section 1.2 in Chapter II for a review.) However, MSMIPs provide better design decisions due to their ability to benefit from the information gained by the gradual unfolding of uncertainty. But since MSMIPs are hard to solve, little insight exists regarding the size of improvement in solution value due to applying MSMIP.

It appears that Huang and Ahmed (2009) provide the only study of the *value of MSMIP*. The value of MSMIP evaluates the solution quality of an MSMIP relative to its SMIP counterpart. It results from dividing the difference between the optimal solutions of an MSMIP and its SMIP counterpart by the optimal solution of the SMIP (Huang and Ahmed, 2009). Huang and Ahmed (2009) show that a lower bound on the value of MSMIP ranges between 5% to 40% for a capacity expansion problem (when three scenarios emanate from each scenario-tree node). They also demonstrate the this value increases along with the variability in customers' demand, the length of the planning horizon, and the number of scenarios emanating from each scenario-tree node.

The SCND heuristic developed in this chapter enables the study of the value of MSMIP for SCND problems. The following sections discuss the value of MSMIP in SCND problems and the extra computational effort required to solve an MSMIP using this heuristic.

5.1. Value of multistage modeling of SCND

Solving the same instances for both the SMIP and the MSMIP formulations for the SCND problem allows to calculation of the value of MSMIP for SCND problems.

Tables XXVII, XXVIII, and XXIX summarize these values for SCND problems of sizes A, B, and C, respectively.

The following features can be observed in these tables:

- For problem size A, the value of the MSMIP ranges from 4.1% to 10.0%, as shown in table XXVII.
- For problem size B, since the DEM did not achieve an optimal solution within the allotted time, the heuristic solution value is used in lieu of the optimal solution value in calculating the value of MSMIP. This results in a lower bound on the value of MSMIP. This lower bound on the value of the MSMIP ranges from 11.2% to 19.6%, as shown in table XXVIII.
- For prolem size C, the lower bound on the value of MSMIP ranges from 19.7% to 31.5%, as shown in table XXIX.

These observations lead to the following conlusions:

- The value of MSMIP warrants the additional computational effort involved in solving an MSMIP for a SCND problem.
- The value of MSMIP increases as the supply chain network size increases.

The results reported in tables XXVII–XXIX are consistent with the results reported by Huang and Ahmed (2009). Also, the conclusions of the experiment regrading the SCND problem straightly aligns with their conclusions regarding the capacity expansion problem. Huang and Ahmed (2009) also concluded that the value of MSMIP depends on demand variabiliy and the length of the planning horizon but did not study the effect of the size of the supply chain network.

Table XXVII. Value of MSMIP for instances of size A of the SCND problem

Instance	SMIP: DEM profit ($ 1,000)	MSMIP: DEM profit ($ 1,000)	Value of MSMIP
1	23,254.2	25,255.4	8.6%
2	366.3	396.1	8.1%
3	884.2	934.9	5.7%
4	3,779.8	3,934.3	4.1%
5	32,291.7	34,302.9	6.2%
6	18,199.6	19,556.7	7.5%
7	111,081.8	122,160.4	10.0%
8	40,776.3	44,106.7	8.2%
9	15,126.2	16,033.8	6.0%
10	26,947.0	29,104.2	8.0%

Table XXVIII. Value of MSMIP for instances of size B of the SCND problem

Instance	SMIP: DEM profit ($ 1,000)	MSMIP: heuristic profit ($ 1,000)	Lower bound on the value of MSMIP
1	32,337.1	35,953.2	11.2%
2	269.1	306.3	13.8%
3	13,383.5	15,436.0	15.3%
4	94,770.2	113,333.7	19.6%
5	16,156.2	18,655.7	15.5%
6	1,232.5	1,421.3	15.3%
7	26,512.7	31,598.5	19.2%
8	230,987.9	269,342.3	16.6%
9	11,277.5	12,968.8	15.0%
10	3,388.6	3,790.5	11.9%

Table XXIX. Value of MSMIP for instances of size C of the SCND problem

Instance	SMIP: DEM profit ($ 1,000)	MSMIP: heuristic profit ($ 1,000)	Lower bound on the value of MSMIP
1	223,465.4	289,399.0	29.5%
2	542,452.6	691,054.3	27.4%
3	7,594.4	9,187.9	21.0%
4	42,325.5	53,045.0	25.3%
5	142,094.8	186,842.0	31.5%
6	6,128.7	7,597.0	24.0%
7	53,607.7	66,741.0	24.5%
8	419,008.4	501,569.6	19.7%
9	38,527.1	47,050.8	22.1%
10	596,366.4	776,861.8	30.3%

5.2. Additional computational effort incurred by MSMIP

Tables XXII–XXVI and XVI–XX show that, for each problem size, solving the MSMIP using the developed heuristic consumed less time than optimizing the DEM for the SMIP for the same problem. However, SMIPs have been increasingly popular and their solution algorithms have been constantly improving. Fortunately, since the developed SCND heuristic relies on solving several nodal subproblems, each is an SMIP, the advances in solving SMIPs also benefit this heuristic, as the following section elaborates.

6. Solution scalability

This MSMIP heuristic involves two major steps:

1. Approximate the solution space by iteratively generating nodal solutions.
2. Choose a global solution by selecting exactly one nodal solution per nodal subproblem.

The second step entails applying the customized shortest hyperpath algorithm, which is polynomial-time. Therefore, the computational effort involved in implementing this step is scalable to accommodate increasing scenario-tree sizes.

The first step, on the other hand, presents an exponentially computational difficulty as the problem size increases. Many accelerated Benders techniques exist to expedite this step (c.f. Santoso *et al.* (2005) and MirHassani *et al.* (2000)). These techniques have not been integrated in the C++ implementation of this model, and therefore the results of the previous sections did not benefit from the computational efficiencies they enduce.

Furthermore, because the nodal subproblems are SMIPs, they can benefit from any existing or future accelerated solution methodologies for SMIPs. It's important to note that while this heuristic relies on nodal decomposition to decompose the SCND problem into a conveniently structured master problem and smaller subproblems, nothing prevents decomposing the subproblems themselves by either scenario-decomposition or component decomposition. The only requirement is that the solution approach be iterative to provide the master problem columns (which is invariably the case due to the complex nature of these SMIPs). In short, solving a nodal subproblem should not consume more time than solving a SCND problem formulated as a SMIP.

The large number of nodal subproblems can present a computational challenge. This challenge can be mitigated with parallel computing, which this heuristic can exploit since nodal subproblems are independent. Consequently, this heuristic can solve a problem in the same time as any current SMIP model but with a much better solution (see section 5.1).

7. Solution stability

The purpose of multi-period models, such as SCND, is to provide insight for managing an evolving business environment. In doing so, this SCND model achieves the best objective function value by periodically updating a supply chain's configuration (see section 5 of Chapter IV). Consequently, the solutions produced by computational experiments (see section 4) require frequent capacity updates for open facilities (almost every period). However, the technology and location decisions tend to be more stable.

A conflict exists between capitalizing on the benefits of multistage modeling and the stability of a model's solution. On the one hand, managers tend to find the frequent updates to a supply chain's configuration chaotic. On the other hand, a model that updates a supply chain's configuration as frequently as it updates cost and demand parameters results in a superior objective function value. The SCND literature (and its facility location ancestor) has traditionally alleviated this conflict by the following mechanisms:

- Enforce a minimum duration between successive supply chain network updates. This is done by fixing the length of each time-period to this minimum duration.
- Discourage frequent updates by including a penalty to the facility reconfiguration cost.
- Restrict the number of configuration updates for each facility during the planning horizon. This is usually achieved by allowing at most one update during the planning horizon.
- Make frequent capacity expansions counterproductive (in situations when demand is non-decreasing). This is done by reducing the available capacity of a facility that undergoes a configuration update for the time period in which this update takes place.

The developed SCND model does not incorporate any of these mechanisms. Applying the first two mechanisms to improve solution stability is straight-forward. The third mechanism requires updating the shortest hyperpath algorithm to eliminate any path that exceeds the allowable number of updates per facility. This extension is an interesting research question that is addressed in section 3.5 of Chapter IX. The last mechanism seems irreconcilable with the nodal decomposition underlying the SCND heuristic since it introduces further dependencies between each node and its predecessor.

CHAPTER VIII

METHODOLOGY EXTENSION TO OTHER APPLICATIONS

The proposed methodology is rooted in SCND. Nevertheless, it is generic enough that it is amenable various applications beyond SCND. This chapter provide general guidelines on formulating other applications so that they benefit from the solution approach developed for the SCND problem.

1. Characteristics of candidate applications

A candidate application must be amenable to formulation into an MSMIP that exhibits the following properties:

1. Decomposable into a master problem and a set of independent nodal subproblems.

2. Each subproblem is a two-stage stochastic mixed-integer program (SMIP) that is amenable to solution using an iterative solution procedure.

3. A feasible solution for the master problem can be constructed by selecting exactly one feasible nodal solution per nodal subproblem.

The third property disallows constraints excluding specific combinations of feasible nodal subproblems.

2. Example applications

These properties are inherent in diverse applications. Examples include loading plans for container ships where a vessel's stability and its partial unloading and reloading speed at successive calling ports are critical factors. In this case, hyperarcs represent reshuffling containers loaded in pervious ports. Another example involves setting prices for services or products where constantly adjusting prices carries consequences. In this case, a hyperarc's weight represents the cost associated with marketing campaigns required to mitigate the effect of price adjustments between successive periods.

3. Guidelines to formulate and solve candidate applications

The characteristics discussed in section 1 can be induced by following specific steps.

3.1. Variables selection

Three sets of variables need be defined. Each such set consists of one vector of variables per scenario-tree node (i.e., one vector for each combination of time period and scenario realization). The first set represents the state of the system describing the application under consideration. Members of this set are binary. The second describes the recourse actions that can be taken once uncertainty unfolds. Members of this set are continuous variables. The third set describes the necessary actions to transition from one state variable set (of the first variables set) to another. Members of this set are binary.

3.2. Scenario-tree formulation

The scenario-tree node formulation of a problem can be constructed in two steps. First, mathematically formulate each nodal subproblem by a SMIP for which the first-stage decision involves a single period and the second-stage decisions involves a single period. The first-stage decisions represent the state of the system before the unfolding of uncertainty. The second-stage decisions represent the recourse actions that could be implemented after the unfolding of uncertainty.

The second step involves the stage coupling constraints. These constraints establish the interdependencies between scenario-tree nodes by describing the actions necessary to transition for the state of the system in a parent node to that of its descendant. In other words, these constraints establish the relationship between the first and third of variables.

Following these steps results into constraints indexed over scenario-tree nodes.

3.3. Relatively-complete recourse

The relatively complete recourse property refers to the existence of a feasible second stage solution for any feasible first stage solution. This property enables using a solution of one nodal subproblem to jump start another nodal subproblem (see section 3 of Chapter VII for more details regarding the role of relatively-complete recourse in solving the nodal subproblems).

Inducing the relatively-complete recourse property into a second-stage of a SMIP can be done by introducing an extra set of variables. These variables act as surplus or slack variables for constraints that could otherwise be violated. Penalizing these variables in the objective function guide the second-stage to choose a solution that sets these variables to zero over one that doesn't. These variables cary practical meaning depending on the application. For example, in production and distribution applications, a set of variables representing the shortage in fulfilling customers' demand artificially guarantees the feasibility of a constraint mandating that all demand be satisfied.

3.4. Nodal decomposition

By relaxing the stage coupling constraints, the problem decomposes into a number of independent nodal subproblems. Each nodal subproblem is a SMIP enjoying the property of relatively complete recourse.

The master problem re-establishes the interdependencies among these nodal subproblems. The state variables are now decided in the nodal subproblems. Consequently, each state variable is replaced by all its feasible values, each weighted by a binary indicator variable.

3.5. Master problem reformulation

The master problem is reformulated into a Leontief substitution flow system (LSFS). This is best achieved by using a hypergraph. In this hypergraph, a vertex represents a possible solution for a nodal subproblem. A hyperarc represents possible transition between nodal subproblem solutions. Each such transition combine several of the third variable type (as many as the number of descendants for a node).

The weights associated with a hyperarc are carefully chosen to induce equivalence between a transition cost in the hypergraph and a transition cost in the master problem. This is achieved by calculating the weighted sum of the transition costs to each of the descendant nodes that a hyperac leads to and the state variable cost for these nodes, where each cost is weighted by the arc probability of its corresponding node.

3.6. Solution space approximation

Approximating the solution space is achieved by applying an iterative algorithm on each nodal subproblem. If these iterative solutions are feasible (i.e., resulting from a primal algorithm like the L-shaped method) they form vertices for the hypergraph. If they are not feasible (i.e., resulting from a dual algorithm like the Lagrangian relaxation) a feasible solution need to be constructed using these infeasible solution. This process will differ from one application to another. However, usually dual algorithms involves finding such a feasible solution to use its value as a bound. This implies that no additional effort is expended to find these feasible solutions.

The generated iterative solutions serve multiple functions:

1. They populate the columns associated with their node in the restricted master problem.
2. They also serve as columns for all other nodes as a result of the relatively complete recourse property of the subproblems.
3. They jump start the L-shape procedure for all following subproblems, as described in section 4 of Chapter VI.

As a result, as the solution procedure progresses, the required computation effort declines.

3.7. Global solution selection

Once the problem is expressed as an LSFP and its vertices are generated using nodal subproblems, it becomes amenable to solution using the customized shortest hyperpath algorithm summarized in Fig. 7 of Chapter VII. This algorithm constructs a global solution by selecting the best combination of generated nodal solutions.

CHAPTER IX

CONCLUSION

This chapter lists the conclusions, contributions, and future research.

1. Conclusions

This dissertation extends the rich literature of supply chain network design (SCND) both in terms of model scope and solution approach.

The developed SCND model is more realistic than current models. This model mimics the actual problem more closely by including all the followings as shown by sections 2–4 of Chapter IV:

- The location, capacity, and technology attributes of a resource.
- The effect of the economies of scale on the cost structure.
- Multiple products and multiple levels of supply chain hierarchy.
- Stochastic, dynamic, and correlated demand.
- Using the gradually unfolding uncertainty to improve design decisions.

The resulting model is a multistage stochastic mixed-integer program (MSMIP) that has no known practical general purpose solution methodology.

A heuristic procedure has been developed to solve this model. This heuristic has been implemented using C++ and CPLEX 11.0. The results of the conducted computational experiments presented in section 4 of Chapter VII demonstrate that this heuristic produces satisfactory results. Furthermore, this heuristic can handle problems that are no smaller than those tackled by current models as shown by section 5.2 of Chapter VII. Finally, using the gradually unfolding uncertainty provides improved design decisions, which is reflected by the value of this MSMIP over its SMIP counterparts as shown in section 5.1 of Chapter VII.

The foundation underlying this heuristic are the scenario-tree node formulation of the SCND problem, the developed end-of-horizon (EoH) decomposition that preprocesses the model to shrink its size, the nodal decomposition of this formulation, and the reformulation of the resulting master problem into a Leontief flow problem.

The developed EoH decomposition preprocesses an MSMIP to shrink its size and thereby reduces its solution time as shown in section 5 of Chapter V. This EoH decomposition rests on exploiting the traditional treatment of the end-of-horizon effect, which enables the independent optimization of each boundary node (a node belonging to the last level of the scenario-tree for which design decisions apply). This reduction in the MSMIP size is significant due to the scenario-tree structure: the last two level of the scenario-tree constitute a large portion of its nodes. The optimality of the resulting solution (with respect to the original MSMIP) is proved mathematically when the cost discount rate is 0%. When this discount rate exceeds 0%, a bound on the optimality gap is deduced. Computational results of section 5 of Chapter V show that this bound never exceeds 6%. Despite their significantly reduced sizes, the MSMIPs resulting from the EoH decomposition remain beyond current computational technology for models reflecting the sizes of actual business applications.

The developed EoH decomposition is neither SCND nor MSMIP specific; it pertains to any application sensitive to the EoH-effect and to special cases of MSMIP. To demonstrate this versatility, additional computational experiments were conducted for a two-stage stochastic mixed-integer program (SMIP) for this SCND problem. The SMIP results echo those of the MSMIP in terms of solution time and bound on optimality gap as shown in section 5.2 of Chapter V .

A heuristic has been developed to solve the MSMIP resulting form the EoH decomposition as shown in section 3 of Chapter VII. This heuristic rests on the nodal decomposition of the MSMIP. As such, this heuristic provides the first application of nodal decomposition into MSMIP and thereby demonstrates that this decomposition can form the basis of viable MSMIP solution techniques.

The nodal decomposition of this MSMIP results into a conveniently structured master problem that ties many otherwise independent nodal subproblems. These nodal subproblems are still NP-hard but their smaller size renders them easier to solve. The structure of these subproblems makes them amenable to solution via the L-shaped method as shown in section 1.1 of Chapter VII.

The structure of the master problem allows its reformulation into a Leontief substitution flow problem (LSFP) that enjoys the integrality property and is amenable to polynomial time solution algorithms. As shown in section 2 of Chapter VII, this LSFP can be represented graphically using a hypergraph, in which vertices represent the solutions of the subproblems and hyperarcs represent the stage coupling constraints of the MSMIP. This handling of the master problem is novel in MSMIP. Conceptually, it reformulates the master problem to avoid the duality-gap. Technologically, it provides the first application of Leontief substitution flow problems in MSMIP and thereby shows that hypergraphs lend themselves to loosely coupled MSMIPs.

This LSFP is amenable to solution using the polynomial time shortest hyperpath algorithm developed by Martin *et al.* (1990). To exploit the special structure the SCND model provides, this shortest hyperpath algorithm has been customized, as shown in section 2.4 of Chapter VII, and thereby a further reduction in solution time has been achieved.

The developed heuristic relies on pulling together the solution algorithm for the subproblems and that for the master problem. The principal mechanism of this heuristic is that of type-I column generation procedures. The iterative nodal solutions generated by the L-shaped method provide the restricted master problem columns. The selection of the best solution among the generated columns relies on the customized shortest hyperpath algorithm. By generating only a subset of all feasible columns, this heuristic does not guarantee an optimal solution. Nevertheless, the computational results indicate satisfactory results.

The results of the computational experiment of section 4 of Chapter VII show that this heuristic produces remarkable savings in solution time as compared to the direct application of MIP techniques over the DEM. As a result, this heuristic enables tackling problem sizes that current technology fails to handle. The resulting optimality gap never exceeds 6% and tends to shrink as the problem size increases.

This heuristic is an improvement over current models as shown in section 5 of Chapter VII. First, it can benefit from most acceleration techniques for current models, as shown in section 5.2 of Chapter VII. Second, the results of the computational experiments show that the sub-optimal solutions generated by this heuristic are always superior to the optimal solutions resulting from their SMIP counterparts as shown in section 5.1 of Chapter VII. Furthermore, these results also show that the value of MSMIP (over SMIP) ranges from 4% to 31.5% of the SMIP optimal solution values. These figures are consistent with the results reported by a previous study involving the value of MSMIP in capacity expansion problems (Huang and Ahmed, 2009). The computation results also show that the value of MSMIP tend to increase as the problem size increases.

This heuristic still applies when the SCND problem is modeled as a SMIP. In this case, the restricted master problem is reformulated into a shortest path problem as shown in section 3.2 of Chapter VI. Computational results for this SMIP indicate a 88% reduction in solution time as compared to its DEM and an 13% bound on the optimality-gap.

2. Contributions

The contributions of this research fall into two fields: supply chain management and stochastic programming. In supply chain management, this research contributes the following:

1. Model the SCND problem more realistically than current models, and provide a practical solution approach useful for business applications.

2. Provide computational results that show the value of the multistage modeling of stochastic SCND problems over the traditional two-stage models.

3. Develop a practical solution approach for the SCND problem when modeled as a two-stage mixed integer stochastic program (SMIP).

In stochastic programming, this research will contribute the following:

1. Provide the first application of nodal decomposition in MSMIP, showing that nodal decomposition lends itself to loosely coupled MSMIPs, and therefore can form the basis of a viable solution approach; and provide the first computational results involving the application of nodal decomposition in MSMIP, illustrating the usefulness of this decomposition to MSMIPs.

2. Provide the first application of Leontief flow problems in stochastic programming, and show that hypergraphs lend themselves to multistage stochastic programs.

3. Develop an End-of-Horizon Decomposition that significantly reduces the solution time for (two-stage and multi-stage) stochastic programs. This decomposition is SCND-independent and therefore applies to any stochastic program that models a problem sensitive to the end-of-horizon effect.

4. Extend the developed formulation and solution methodology to applications beyond SCND problems by establishing guidelines to formulate a MSMIP such that its decomposition results in conveniently structured master problem and describing a methodology to reformulate the master problem into an easy to solve, integral, Leontief flow problem.

3. Future research

This SCND model and heuristic can be deployed to investigate and potentially resolve several SCND persisting concerns. This section explores this prospect.

3.1. Location-inventory problem

Traditionally, SCND has excluded inventory considerations when evaluating location decisions (Owen and Daskin, 1998). Likewise, the scope of inventory planning models do not extend beyond the relationships between stocking and demand points (Melo *et al.*, 2009). This decoupling is not unreasonable especially since location selection and inventory planning belong to different hierarchical planning levels and that each is already a complex task.

Nevertheless, in some applications inventory implications can discredit an otherwise optimal location decision (Owen and Daskin, 1998). Examples include situations when safety stocks are sizable and associated with dominating costs, such as in international sourcing and distribution networks.

Effective models that integrate location decisions and inventory planning remain elusive. Risk pooling introduces a nonlinear cost term resulting in a "nasty combinatorial optimization problem" (Shu *et al.*, 2005).

The SCND model and heuristic developed in this dissertation have the potential to account for this cost. This heuristic compiles the weights of hyperarcs (or arcs in the case of the shortest path problem) before associating them with binary variables. This makes these weights amenable to including nonlinear cost terms. By exploiting this feature, a clever formulation and decomposition of the location-inventory problem can build on this SCND model and heuristic.

3.2. Global supply chain network design

Unique uncertainties such as exchange rates influence design decisions for global supply chains (Melo *et al.*, 2009). Considering combinations of different demand realizations and different exchange rates dramatically increases the number of possible scenarios, which makes this problem much harder to solve.

With some extensions, the SCND heuristic can effectively tackle this problem. In the SCND heuristic, solving the shortest hyperpath problem is computationally efficient and can handle the implications of including global aspects of a supply chain network. Providing the shortest hyperpath problem with its hypergraph with vertices is the computationally taxing part. Generating these vertices involves solving a SMIP problem for each node of the scenario-tree. The number of these nodes increases as the number of scenarios does. Applying the sample average approximation (SAA) technique can alleviate this exceedingly computational burden.

The SAA reduces the computational burden by repeatedly solving a reduced-sized problem involving a subset of the scenarios. This method has been usefully implemented to solve subsets of the SCND problems (c.f. Schutz *et al.* (2007) and

Santoso *et al.* (2005)). However, these applications involve scenario formulations and scenario decompositions. Adapting the SAA to nodal formulation and decomposition on which this SCND heuristic relies is an interesting research question.

3.3. Robust supply chain network design

Klibi *et al.* (2010) defines a robust supply chain network as one capable of creating value under any plausible future scenario. As such, aversion to performance variability under the different plausible scenarios is a cornerstone of the robust design for supply chains. Invariably, minimizing performance variability involves objective functions more complex than the traditional expected value of discounted cash flows (Klibi *et al.*, 2010). Snyder and Daskin (2007) discuss several objective functions that enable robust models.

The shortest hyperpath problem provides an opportunity to include more complex objective functions. The weight of each hyperarc is computed prior to solving this problem. Computing this weight currently accommodates the expected value objective function of the SCND model but could be extended to emphasis risk aversion. An example is to include a measure of SCND performance variation for the vertices to which a hyperarc leads. Properly designing the constituents of a hyperarc's weight leads to a risk-averse hyperpath and is an attractive research direction.

3.4. Solution's value assessment in dynamic problems

In models that adopt a rolling horizon strategy, only the solution of their first period is implemented. Decisions for later periods are revised as the planning horizon is updated and the problem resolved. Therefore, two feasible solutions that share the same first period decisions but are otherwise different have the exact practical implications and thus are practically equivalent.

A measure that captures the effects of rolling a planning horizon is needed. Objective functions that emphasize the entire horizon, such as the expected value of discounted cash flows, fail to reflect this equivalence. On the other hand, a greedy algorithm that neglects the long-term implications of immediate decisions can produce poor solutions and is unsuitable for problems involving sizable upfront investments, such as the SCND problems.

The structure of shortest hyperpath reformulation of the master problem of this SCND model can help explore new measures of performance suitable for rolling horizons. Devising a cost structure that assigns the same value for all paths leading out each vertex can provide a starting point to eliminate the difference in objective function values for practically equivalent solutions.

3.5. Methodology extension to other applications

Although the proposed methodology is rooted in SCND, it is generic enough and amenable to applications beyond SCND. Identifying applications that stand to benefit the most and tailoring the heuristic to suit them is a promising research area.

A candidate application must have a formulation with the following properties:

1. A formulation decomposable into a master problem and a set of nodal subproblems exists.

2. An iterative solution procedure to solve the nodal subproblems exists.

3. A feasible solution for the master problem can be constructed by selecting exactly one feasible nodal solution per nodal subproblem.

These properties are inherent in diverse applications. Examples include loading plans for container ships where a vessel's stability and its partial unloading and reloading speed at successive ports are critical factors. In this case, hyperarcs represent reshuffling containers loaded in previous ports. Another example involves setting prices for services or products where constantly adjusting prices carries consequences. In this case, a hyperarc's weight represents the cost associated with marketing campaigns required to mitigate the effect of price adjustments between successive periods.

Furthermore, this heuristic can be updated to fit other applications that do not possess all those properties. For example, the third property disallows constraints excluding specific combinations of feasible nodal subproblems. However, accommodating such constraints is straight-forward if they involve only a parent node and/or its immediate descendants. In this case, the weight of the specific hyperarc joining the infeasible combination of vertices can be set to infinity. Constraints that exclude combining specific vertices of distant nodes are trickier to accommodate and require further investigation. Devising a cost structure that penalizes the entire hyperpath containing the offending vertices can resolve this issue.

3.6. SCND heuristic extension into an optimization algorithm

Many directions to extend this SCND heuristic into an algorithm that guarantees an optimal solution exist. Generating additional hyperarcs that have the potential to improve the solution is a common goal among these directions. Directly generating these hyperarcs can be computationally burdensome since it involves solving a subproblem that combines multiple scenario-tree nodes. In contrast, generating vertices that enable constructing a combination of promising hyperacs carry more potential.

Generating these vertices involves either identifying conditions that guide the search for promising vertices that can improve the incumbent shortest hyperpath or conditions that prevent the generation of unpromising vertices that could not possibly improve it.

One way to generate promising vertices on the fly involves a type-II column generation procedure (Wilhelm, 2001). In this case, information gained from the incumbent shortest hyperpath guides the search for new vertices. The difficulty lurking in this direction emerges from the dependency of a hyperarc's weight on all the vertices it joins. Isolating the effect of each vertex on a hyperarc's cost can be challenging yet very effective if attained.

One way to hinder the generation of unpromising vertices involves excluding them from the feasible space of nodal subproblems using exclusion cuts. Deriving these cuts requires deducing a strong lower bound on the total weight of a feasible shortest hyperpath.

REFERENCES

Ahmed, S. and Garcia, R. (2004). Dynamic capacity acquisition and assignment under uncertainty, *Annals of Operations Research* **124**: 267–283.

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

Ahmed, S. and Sahinidis, N. (2003). An approximation scheme for stochastic integer programs arising in capacity expansion, *Operations Research* **51**: 461–471.

Alonso-Ayuso, A., Escudero, L. F. and Ortuno, M. T. (2003). BFC, a branch-and-fix coordination algorithm framework for solving some types of stochastic pure and mixed 0-1 programs, *European Journal of Operational Research* **151**: 503–519.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. and Vance, P. H. (1989). Branch-and-price: column generation for solving huge integer programs, *Operations Research* **46**(3): 316–329.

Bienstock, D. and Shapiro, J. F. (1988). Optimizing resource acquisition decisions by stochastic programming, *Management Science* **34**: 215–229.

Burnson, R. (2009). 2009 state of logisitcs: collaboration time, *Logistics Management* **48**(7): 24–27.

C2ER (2008). *ACCRA Cost of Living Index: 2007 Annual Average Data*, Vol. 40, The Council for Community and Economic Research (C2ER), Arlington, VA.

Carøe, C. C. and Schultz, R. (1999). Dual decomposition in stochastic integer programming, *Operations Research Letters* **24**(1-2): 37–45.

Chandra, C. and Gragis, J. (2007). *Supply Chain Configuration*, Springer, New York.

Chopra, S. (2003). Designing the distribution network in a supply chain, *Transportation Research E* **39**: 123–140.

Chopra, S. and Meindle, P. (2007). *Supply Chain Management: Strategy, Planning and Operation*, Prentice-Hall, Upper Saddle River, NJ.

Dantzig, G. B. and Wolfe, P. (1961). The decomposition algorithm for linear programs, *Econometrica* **29**: 767–777.

Davis, T. (1993). Effective supply chain management, *Sloan Management Review* **34**: 35–46.

Dentcheva, D. and Römisch, W. (2004). Duality gaps in nonconvex stochastic optimization, *Mathematical Programming* **101**(3): 515–535.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs, *Numerische Mathematik* **1**: 269–271.

Dupačová, J., Consigli, G. and Wallace, S. W. (2000). Scenarios for multistage stochastic programs, *Annals of Operations Research* **100**: 25–53.

Eppen, G., Martin, R. K. and Schrage, L. (1989). A scenario approach to capacity planning, *Operations Research* **37**: 517–527.

Escudero, L. (2009). On a mixture of the fix-and-relax coordination and lagrangian substitution schemes for multistage stochastic mixed integer programming, *Top* **17**: 5–29.

Escudero, L., Garin, A., Merino, M. and Perez, G. (2009). BFC-MSMIP: an exact branch-and-fix coordination approach for solving multistage stochastic mixed 0-1 problems, *Top* **17**: 99–122.

Farahani, R. Z., SteadieSeifi, M. and Asgari, N. (2010). Multiple criteria facility location problems: a survey, *Applied Mathematical Modeling* **34**: 1689–1709.

Geoffrion, A. M. and Graves, G. (1974). Multicommodity distribution system design by Benders decomposition, *Management Science* **20**(5): 822–844.

Goetschalckx, M. and Fleischmann, B. (2008). Strategic network planning, *in* H. Stadler and C. Kilger (eds), *Supply Chain Management and Advanced Planning: Concepts, Models, Software and Case Studies*, Springer, New York, pp. 117–131.

Gupta, D., Gerchak, Y. and Buzacott, J. A. (1992). The optimal mix of flexible and dedicated manufacturing capacities: hedging against demand uncertainty, *International Journal of Production Economics* **28**(3): 309–319.

Hoyland, K. and Wallace, S. W. (2001). Generating scenario trees for multistage decision problems, *Management Science* **47**: 295–307.

Huang, K. and Ahmed, S. (2009). The value of multistage stochastic programming in capacity planning under uncertainty, *Operations Research* **57**: 893–904.

Hübner, R. (2007). *Strategic Supply Chain Management in Process Industries*, Springer-Verlag, Berlin.

Jeroslow, R. G., Martin, K., Rardin, R. L. and Wang, J. (1992). Gainfree Leontief substitution flow problems, *Mathematical Programming* **57**: 375–414.

Kamath, K. and Pakkala, T. (2002). A Bayesian approach to a dynamic inventory model under an unknown demand distribution, *Computers & Operations Research* **29**: 403–422.

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

Koehler, G. J., Whinston, A. B. and Wright, G. P. (1975). *Optimization over Leontief Substitution Systems*, North Holland, Amsterdam, the Netherland.

Liu, M. L. and Sahinidis, N. V. (1996). Optimization in process planning under uncertainty, *Industrial & Engineering Chemistry Research* **35**: 4154–4165.

Louveaux, F. V. and Peeters, D. (1992). A dual-based procedure for stochastic facility location, *Operations Research* **40**(3): 564–573.

Lucas, C., Messina, E. and Mitra, G. (1996). Risk and return analysis of a multi-period strategic planning problem, *in* L. Thomas and A. Christers (eds), *Stochastic Modeling in Innovative Manufacturing*, Springer, Berlin, pp. 81–96.

Lucas, C., MirHassani, S., Mitra, G. and Poojari, C. (2001). An application of lagrangian relaxation to a capacity planning problem under uncertainty, *Journal of Operational Research Society* **52**: 1256–1266.

Martel, A. (2005). The design of production-distribution networks: a mathematical programming approach, *in* J. Geunes and P. M. Pardalos (eds), *Supply Chain Optimization*, Springer, New York, pp. 265–305.

Martin, R. K., Rardin, R. L. and Cambell, B. A. (1990). Polyhedral characterization of discrete dynamic programming, *Operations Research* **38**: 127–138.

Melo, M., Nickel, S. and da Gama, F. S. (2005). Dynamic multi-commodity capacitated facility location: a mathematical modeling framework for strategic supply chain planning, *Computers & Operations Research* **33**: 181–208.

Melo, M., Nickel, S. and da Gama, F. S. (2009). Facility location and supply chain management—a review, *European Journal of Operational Research* **196**: 401–412.

Min, H. and Zhou, G. (2002). Supply chain modeling: past, present and future, *Computers & Industrial Engineering* **43**: 231–249.

MirHassani, S. A., Lucas, C., Mitra, G. and Poojari, C. (2000). Computational solution of a capacity planning model under uncertainty, *Parallel Computing Journal* **26**: 511–538.

Mitra, G., Poojari, C. and Sen, S. (2006). Strategic and tactical planning models for supply chain: an application of stochastic mixed integer programming, *in* G. Appa, L. Pelsoulis and H. P. Williams (eds), *Handbook on Modeling for Discrete Optimization*, Vol. 88, Springer, New York, pp. 227–264.

Mo, Y. and Harrison, T. P. (2005). A conceptual framework for robust supply chain design under demand uncertainty, *in* J. Geunes and P. M. Pardalos (eds), *Supply Chain Optimization*, Springer, New York, pp. 243–263.

Owen, S. H. and Daskin, M. S. (1998). Strategic facility location: a review, *European Journal of Operational Research* **111**: 423–447.

Paquet, M., Martel, A. and Desaulniers, G. (2004). Including technology selection decisions in manufacturing network design models, *International Journal of Computer Integrated Manufacturing* **17**(2): 117–125.

Peidro, D., Mula, J. and Poler, R. (2009). Quantitative models for a supply chain planning under uncertainty: a review, *International Journal of Advanced Manufacturing Technology* **43**: 400–420.

Rand-McNally (2006). *Rand-McNally's 2006 Commercial Atlas & Marketing Guide*, Vol. 1, Rand-McNally & Company, Chicago, IL.

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

Santoso, T. S., Ahmed, S., Goetschalckx, M. and Shapiro, A. (2005). A stochastic approach for supply chain network design under uncertainty, *European Journal of Operational Research* **167**: 96–115.

Schmidt, G. and Wilhelm, W. (2000). Strategic, tactical and operational decisions in multi-national logistics networks: a review and discussion of modelling issues, *International Journal of Production Research* **7**: 1501–1523.

Schultz, R. (2009). Comments on: On a mixture of the fix-and-relax coordination and lagreangean substitution schemes for multistage stochastic mixed integer programming, *Top* **17**: 35–36.

Schultz, R., Stougie, L. and van der Vlerk, M. (2003). Stochastic programming with integer variables, *Mathematical Programming* **97**: 285–309.

Schutz, P., Stougie, L. and Tomasgard, A. (2007). Stochastic facility location with general long-run costs and convex short-run costs, *Computers and Operations Research* **35**(9): 2988–3000.

Schütz, P., Stougie, L. and Tomasgard, A. (2008). Stochastic facility location with general long-run costs and convex short-run costs, *Computers & Operations Research* **35**(9): 2988–3000.

Sen, S. (2005). Algorithms for stochastic mixed-integer programming models, *in* K. Aardal, J. Nemhauser and R. Weismantel (eds), *Handbooks in Operations Research and Management Science: Discrete Optimization*, Elsevier, Amsterdam, pp. 515–558.

Shapiro, J. F. (2007). *Modeling the Supply Chain*, Duxbury Thomson Learning, Belmont, CA.

Shu, J., Teo, C.-P. and Shen, Z. M. (2005). Stochastic transportation-inventory network design problem, *Operations Research* **53**(1): 48–60.

Silva, E. F. and Wood, R. K. (2006). Solving a class of stochastic mixed-integer programs with branch and price, *Mathematical Programming* **108**: 395–418.

Snyder, L. V. (2006). Facility location under uncertainty: a review, *IIE Transactions* **38**: 537–554.

Snyder, L. V. and Daskin, M. S. (2007). Models for reliable supply chain network design, *in* A. Murray and T. Grubesic (eds), *Critical Infrastructure: Reliability and Vulnerability*, Springer, Berlin, pp. 257–289.

Tsiakis, P., Shah, N. and Pantelides, C. (2001). Design of a multiechelon supply chain network under demand uncertainty, *Industrial and Engineering Chemistry Research* **40**: 3585–3604.

Van Roy, T. and Erlenkotter, D. (1982). A dual-based procedure for dynamic facility location, *Management Science* **28**: 1091–1105.

Van Slyke, R. and Wets, R. (1969). L-shaped linear programs with application to optimal control and stochastic programming, *SIAM Journal on Applied Mathematics* **17**: 638–663.

Veinott, A. F. (1968). Extreme points of Leontief susbstitution systems, *Linear Algebra and Its Applications* **1**: 181–194.

Verter, V. (2002). An integrated model for facility location and technology acquisition, *Computers & Operations Research* **29**: 583–592.

Verter, V. and Dincer, M. (1992). An integrated evaluation of facility location, capacity acquisition, and technology selection for designing global manufacturing strategies, *European Journal of Operational Research* **60**: 1–18.

Wilhelm, W. (2001). A technical review of column generation, *Optimization and Engineering* **2**(2): 159–200.

Wilhelm, W., Liang, D., Rao, B., Warrier, D., Zhu, X. and Bulusu, S. (2005). Design of international assembly systems and their supply chains under NAFTA, *Transportation Research Part E* **41**: 467–493.

APPENDIX A

COMPUTATIONAL RESULTS FOR THE EOH DECOMPOSITION OF MSMIPS FOR THE FIRST PROBLEM SIZE

Table XXX. Solution values for MSMIP instance A2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	396.1	396.1	0.00%	0.00%
2%	100	375.4	375.1	0.10%	0.23%
2%	50	261.3	260.9	0.13%	0.29%
2%	20	146.5	146.2	0.19%	0.31%
2%	10	84.7	84.5	0.21%	0.46%
2%	5	52.5	52.4	0.24%	0.53%
5%	infinity	162.1	161.9	0.11%	0.44%
5%	100	160.2	160.0	0.12%	0.45%
5%	50	139.3	139.0	0.18%	0.58%
5%	20	91.8	91.6	0.22%	0.62%
5%	10	63.7	63.5	0.30%	0.65%
5%	5	49.4	49.2	0.35%	0.67%
10%	infinity	63.1	62.2	1.46%	3.70%
10%	100	62.6	61.6	1.60%	3.76%
10%	50	60.8	59.6	1.95%	4.08%
10%	20	54.7	53.6	2.10%	4.49%
10%	10	43.3	42.1	2.95%	4.59%
10%	5	34.2	33.1	2.97%	4.74%
20%	infinity	47.4	46.2	2.46%	4.00%
20%	100	45.6	44.4	2.60%	4.07%
20%	50	43.2	42.0	2.95%	4.13%
20%	20	42.2	40.9	3.10%	4.21%
20%	10	30.5	29.5	3.15%	4.81%
20%	5	24.1	23.3	3.25%	4.95%
50%	infinity	4.7	4.6	3.05%	5.22%
50%	100	4.5	4.4	3.17%	5.31%
50%	50	4.4	4.3	3.20%	5.73%
50%	20	4.4	4.3	3.20%	5.84%
50%	10	4.3	4.1	3.80%	5.91%
50%	5	4.2	4.1	3.91%	5.98%

Table XXXI. Solution times for MSMIP instance A2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	46.69	0.48	1.03%	4.98
2%	100	60.72	0.48	0.79%	6.70
2%	50	26.88	0.47	1.76%	5.18
2%	20	28.00	0.47	1.67%	5.76
2%	10	53.98	0.48	0.89%	6.40
2%	5	29.67	0.47	1.57%	5.74
5%	infinity	40.58	0.44	1.07%	6.59
5%	100	51.15	0.43	0.85%	7.08
5%	50	43.28	0.43	1.00%	6.51
5%	20	51.96	0.45	0.86%	7.44
5%	10	34.95	0.43	1.24%	7.21
5%	5	43.97	0.44	1.00%	8.25
10%	infinity	55.46	0.50	0.91%	7.33
10%	100	50.08	0.49	0.99%	6.00
10%	50	26.52	0.50	1.90%	5.08
10%	20	48.56	0.50	1.03%	8.78
10%	10	50.42	0.50	0.99%	5.86
10%	5	43.06	0.51	1.17%	6.52
20%	infinity	41.58	0.58	1.40%	3.54
20%	100	30.10	0.58	1.94%	4.74
20%	50	36.12	0.58	1.60%	5.39
20%	20	43.16	0.58	1.34%	4.70
20%	10	27.16	0.59	2.16%	4.93
20%	5	26.01	0.59	2.25%	3.33
50%	infinity	70.23	0.49	0.70%	4.34
50%	100	50.46	0.48	0.96%	6.50
50%	50	32.23	0.50	1.54%	5.49
50%	20	74.94	0.49	0.65%	6.43
50%	10	28.52	0.49	1.73%	4.57
50%	5	50.18	0.49	0.98%	5.28

Table XXXII. Solution values for MSMIP instance A3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	934.9	934.9	0.00%	0.26%
2%	100	862.0	862.0	0.00%	0.26%
2%	50	594.3	594.3	0.00%	0.36%
2%	20	356.2	356.2	0.02%	0.37%
2%	10	199.3	199.0	0.16%	0.41%
2%	5	131.0	130.7	0.21%	0.48%
5%	infinity	349.9	349.4	0.13%	0.40%
5%	100	349.6	349.1	0.13%	0.41%
5%	50	325.6	325.1	0.17%	0.57%
5%	20	226.7	226.2	0.22%	0.67%
5%	10	163.1	162.5	0.32%	0.70%
5%	5	103.6	103.0	0.49%	0.82%
10%	infinity	158.7	158.4	0.21%	1.41%
10%	100	156.0	155.6	0.24%	1.49%
10%	50	153.9	153.4	0.34%	1.55%
10%	20	134.3	133.5	0.60%	1.58%
10%	10	112.1	111.2	0.79%	1.72%
10%	5	81.7	81.0	0.96%	1.82%
20%	infinity	100.8	100.0	0.79%	2.25%
20%	100	94.1	93.3	0.86%	2.50%
20%	50	94.0	93.1	0.90%	2.53%
20%	20	93.5	92.4	1.15%	2.79%
20%	10	55.0	54.4	1.26%	2.92%
20%	5	54.1	53.3	1.58%	2.98%
50%	infinity	10.2	10.1	1.38%	3.40%
50%	100	10.2	10.0	1.39%	3.47%
50%	50	10.1	10.0	1.59%	3.60%
50%	20	10.1	9.9	2.07%	3.82%
50%	10	10.1	9.9	2.27%	3.55%
50%	5	9.5	9.3	2.73%	3.53%

Table XXXIII. Solution times for MSMIP instance A3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	37.82	1.16	3.08%	7.76
2%	100	41.40	1.22	2.94%	6.71
2%	50	75.54	1.48	1.96%	5.79
2%	20	59.23	1.44	2.44%	7.18
2%	10	62.98	1.82	2.89%	6.39
2%	5	65.60	1.35	2.06%	6.19
5%	infinity	82.30	2.48	3.01%	8.29
5%	100	75.31	2.15	2.86%	8.87
5%	50	71.39	2.20	3.08%	8.19
5%	20	77.70	2.29	2.95%	8.59
5%	10	42.39	2.56	6.05%	8.71
5%	5	37.29	2.08	5.57%	8.10
10%	infinity	50.09	1.67	3.33%	6.85
10%	100	51.45	1.88	3.65%	7.12
10%	50	66.57	1.66	2.50%	7.04
10%	20	67.54	1.41	2.09%	7.22
10%	10	23.20	1.39	6.00%	6.92
10%	5	21.44	1.24	5.78%	6.88
20%	infinity	75.33	1.44	1.91%	7.51
20%	100	67.07	1.65	2.46%	7.77
20%	50	61.01	1.59	2.60%	7.75
20%	20	88.67	1.55	1.75%	7.10
20%	10	66.63	1.79	2.68%	7.35
20%	5	79.64	1.05	1.32%	6.99
50%	infinity	32.60	1.43	4.40%	4.83
50%	100	33.72	1.85	5.50%	4.57
50%	50	36.18	1.62	4.48%	4.52
50%	20	38.27	1.73	4.52%	4.76
50%	10	38.02	1.98	5.22%	4.21
50%	5	25.63	1.40	5.46%	4.02

Table XXXIV. Solution values for MSMIP instance A4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	3,934.3	3,934.3	0.00%	0.00%
2%	100	3,364.2	3,364.2	0.00%	0.08%
2%	50	2,394.5	2,393.2	0.05%	0.15%
2%	20	1,317.8	1,316.8	0.08%	0.19%
2%	10	871.2	870.4	0.09%	0.23%
2%	5	526.4	525.8	0.11%	0.26%
5%	infinity	1,512.5	1,499.7	0.85%	1.15%
5%	100	1,490.1	1,475.4	0.99%	1.76%
5%	50	1,314.3	1,298.5	1.20%	1.86%
5%	20	977.3	964.7	1.29%	2.10%
5%	10	646.8	638.3	1.32%	2.36%
5%	5	448.3	442.3	1.34%	2.41%
10%	infinity	756.9	747.8	1.20%	2.39%
10%	100	701.6	692.5	1.30%	2.57%
10%	50	644.0	633.8	1.58%	3.38%
10%	20	545.4	535.2	1.87%	3.62%
10%	10	438.0	429.5	1.93%	3.94%
10%	5	325.9	319.6	1.96%	3.98%
20%	infinity	444.6	435.6	2.04%	3.98%
20%	100	427.5	418.7	2.05%	3.99%
20%	50	403.4	394.8	2.13%	4.02%
20%	20	369.4	361.5	2.14%	4.24%
20%	10	235.3	230.0	2.24%	4.27%
20%	5	218.0	211.9	2.80%	4.76%
50%	infinity	42.0	40.9	2.53%	4.31%
50%	100	40.5	39.4	2.72%	4.35%
50%	50	39.1	38.0	2.81%	4.49%
50%	20	38.7	37.6	2.88%	4.69%
50%	10	38.5	37.3	3.02%	4.87%
50%	5	38.4	37.2	3.11%	5.15%

Table XXXV. Solution times for MSMIP instance A4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	49.83	0.35	0.70%	5.37
2%	100	49.18	0.35	0.71%	5.67
2%	50	45.17	0.35	0.76%	5.57
2%	20	47.64	0.34	0.72%	5.12
2%	10	52.53	0.35	0.67%	5.81
2%	5	27.08	0.34	1.26%	5.60
5%	infinity	39.64	0.36	0.90%	5.40
5%	100	49.40	0.36	0.72%	5.81
5%	50	104.15	0.36	0.34%	5.41
5%	20	59.53	0.37	0.62%	5.74
5%	10	30.20	0.36	1.18%	5.05
5%	5	29.17	0.36	1.24%	5.61
10%	infinity	51.30	0.37	0.72%	3.16
10%	100	50.53	0.36	0.71%	3.18
10%	50	50.73	0.37	0.72%	3.74
10%	20	20.90	0.37	1.75%	3.43
10%	10	54.58	0.36	0.67%	3.83
10%	5	39.28	0.37	0.94%	3.47
20%	infinity	49.35	0.37	0.74%	4.57
20%	100	45.26	0.37	0.81%	4.34
20%	50	45.70	0.36	0.79%	5.00
20%	20	45.70	0.36	0.79%	4.40
20%	10	24.29	0.37	1.51%	4.26
20%	5	49.81	0.37	0.74%	4.99
50%	infinity	21.89	0.36	1.65%	6.15
50%	100	23.72	0.35	1.49%	6.27
50%	50	21.08	0.36	1.72%	5.88
50%	20	22.52	0.36	1.58%	6.12
50%	10	20.43	0.36	1.76%	5.37
50%	5	22.10	0.36	1.62%	5.42

Table XXXVI. Solution values for MSMIP instance A5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	34,302.9	34,302.9	0.00%	0.00%
2%	100	29,785.7	29,785.7	0.00%	0.00%
2%	50	22,155.1	22,155.1	0.00%	0.10%
2%	20	12,298.1	12,293.6	0.04%	0.14%
2%	10	7,482.9	7,466.7	0.22%	0.71%
2%	5	4,708.5	4,683.1	0.54%	0.72%
5%	infinity	12,699.1	12,699.1	0.00%	0.00%
5%	100	12,634.6	12,634.6	0.00%	0.10%
5%	50	11,766.1	11,760.2	0.05%	0.13%
5%	20	8,497.2	8,488.7	0.10%	0.40%
5%	10	5,798.5	5,781.3	0.30%	1.81%
5%	5	3,874.2	3,843.6	0.79%	1.88%
10%	infinity	5,579.7	5,579.7	0.00%	0.00%
10%	100	5,581.2	5,578.4	0.05%	0.16%
10%	50	5,569.8	5,556.7	0.24%	0.37%
10%	20	4,979.9	4,952.0	0.56%	1.17%
10%	10	3,917.8	3,874.6	1.10%	1.18%
10%	5	2,851.9	2,817.0	1.22%	1.78%
20%	infinity	3,658.2	3,620.3	1.03%	1.56%
20%	100	3,526.0	3,485.0	1.16%	1.58%
20%	50	3,482.8	3,429.9	1.52%	1.96%
20%	20	3,416.4	3,359.6	1.66%	2.14%
20%	10	2,025.9	1,989.6	1.79%	2.21%
20%	5	1,986.4	1,949.7	1.85%	2.55%
50%	infinity	375.0	366.7	2.21%	4.19%
50%	100	373.4	364.4	2.41%	4.23%
50%	50	367.2	357.7	2.60%	4.32%
50%	20	366.6	356.9	2.66%	4.62%
50%	10	362.8	353.1	2.68%	4.67%
50%	5	338.8	329.3	2.78%	4.87%

Table XXXVII. Solution times for MSMIP instance A5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	43.0	1.7	3.98%	9.2
2%	100	43.1	1.7	3.97%	9.2
2%	50	40.8	1.7	4.15%	9.6
2%	20	34.9	1.7	4.81%	9.0
2%	10	45.3	1.7	3.78%	9.4
2%	5	37.4	1.7	4.49%	8.9
5%	infinity	43.1	1.6	3.73%	3.9
5%	100	41.9	1.6	3.82%	4.3
5%	50	37.5	1.6	4.28%	4.2
5%	20	33.4	1.6	4.90%	4.1
5%	10	21.5	1.6	7.45%	4.4
5%	5	34.7	1.6	4.68%	4.1
10%	infinity	42.5	1.8	4.15%	7.3
10%	100	44.8	1.7	3.89%	7.4
10%	50	38.3	1.8	4.61%	7.4
10%	20	55.5	1.8	3.17%	7.5
10%	10	47.6	1.8	3.69%	7.2
10%	5	35.1	1.8	5.04%	7.0
20%	infinity	51.9	1.9	3.75%	8.0
20%	100	37.5	2.0	5.20%	8.3
20%	50	33.9	1.9	5.71%	8.2
20%	20	44.7	1.9	4.33%	8.3
20%	10	24.0	2.0	8.16%	8.0
20%	5	43.5	2.0	4.49%	8.1
50%	infinity	42.4	1.7	4.12%	8.6
50%	100	40.0	1.7	4.30%	8.6
50%	50	35.5	1.8	4.93%	8.5
50%	20	45.6	1.7	3.79%	8.6
50%	10	34.3	1.7	5.07%	8.8
50%	5	44.2	1.7	3.92%	8.6

APPENDIX B

COMPUTATIONAL RESULTS FOR THE EOH DECOMPOSITION OF MSMIPS FOR THE SECOND PROBLEM SIZE

Table XXXVIII. Solution values for MSMIP instance B2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at one hour runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM	Bound on optimality gap
2%	infinity	302.4	311.3	2.8%	0.5%
2%	100	261.4	270.3	3.3%	0.6%
2%	50	190.9	201.8	5.4%	0.7%
2%	20	110.1	112.4	2.0%	0.7%
2%	10	65.6	68.7	4.5%	0.7%
2%	5	41.7	43.5	4.3%	0.8%
5%	infinity	110.3	115.8	4.7%	0.8%
5%	100	112.2	114.5	2.0%	1.0%
5%	50	105.0	109.7	4.2%	1.2%
5%	20	73.4	76.7	4.2%	1.3%
5%	10	55.1	57.3	3.8%	1.3%
5%	5	32.7	34.6	5.5%	1.3%
10%	infinity	51.5	52.9	2.6%	1.5%
10%	100	47.7	50.6	5.6%	1.6%
10%	50	50.2	51.9	3.3%	1.7%
10%	20	49.7	51.2	2.8%	1.7%
10%	10	33.1	34.8	4.9%	1.8%
10%	5	27.2	28.2	3.7%	1.8%
20%	infinity	37.6	39.5	4.7%	2.0%
20%	100	33.1	35.1	5.6%	2.0%
20%	50	33.5	34.7	3.4%	2.1%
20%	20	29.9	31.2	4.3%	2.3%
20%	10	17.1	17.8	4.1%	2.4%
20%	5	16.0	16.8	4.6%	2.4%
50%	infinity	6.1	6.3	3.9%	3.4%
50%	100	3.9	4.0	2.2%	3.5%
50%	50	3.9	4.0	2.7%	3.6%
50%	20	3.9	4.0	3.5%	3.7%
50%	10	3.8	3.9	3.7%	3.8%
50%	5	2.9	3.0	2.8%	4.1%

Table XXXIX. Solution times for MSMIP instance B2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM	Solution time for bound on gap (minutes)
2%	infinity	60.0	26.8	45%	116.0
2%	100	60.0	27.8	46%	117.0
2%	50	60.0	29.8	50%	111.8
2%	20	60.0	32.6	54%	116.9
2%	10	60.0	31.7	53%	118.4
2%	5	60.1	32.4	54%	118.6
5%	infinity	60.1	29.9	50%	84.4
5%	100	60.0	30.3	51%	86.2
5%	50	60.1	29.2	49%	81.4
5%	20	60.1	34.6	58%	97.6
5%	10	60.1	23.1	38%	90.9
5%	5	60.1	36.9	61%	89.5
10%	infinity	60.0	27.1	45%	87.5
10%	100	60.0	38.8	65%	78.2
10%	50	60.0	31.1	52%	77.6
10%	20	60.0	36.0	60%	82.8
10%	10	60.1	40.6	68%	86.3
10%	5	60.1	31.9	53%	77.7
20%	infinity	60.0	34.9	58%	108.3
20%	100	60.0	43.4	72%	110.3
20%	50	60.1	42.0	70%	102.2
20%	20	60.1	42.1	70%	114.7
20%	10	60.1	35.0	58%	117.2
20%	5	60.0	31.4	52%	141.0
50%	infinity	60.0	25.5	42%	81.8
50%	100	60.0	34.9	58%	86.1
50%	50	60.0	31.6	53%	82.2
50%	20	60.1	32.5	54%	88.1
50%	10	60.0	30.4	51%	83.2
50%	5	60.0	33.1	55%	82.3

Table XL. Solution values for MSMIP instance B2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at one hour runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM	Bound on optimality gap
2%	infinity	15,071.5	15,562.2	3.2%	0.7%
2%	100	12,954.8	13,498.1	4.0%	0.8%
2%	50	9,768.3	10,020.1	2.5%	0.9%
2%	20	5,516.6	5,550.9	0.6%	1.1%
2%	10	3,225.2	3,368.9	4.3%	1.1%
2%	5	2,072.4	2,102.5	1.4%	1.2%
5%	infinity	5,514.2	5,760.4	4.3%	0.8%
5%	100	5,672.5	5,715.3	0.7%	1.1%
5%	50	5,174.1	5,321.6	2.8%	1.3%
5%	20	3,760.9	3,834.5	1.9%	1.4%
5%	10	2,508.6	2,607.5	3.8%	1.4%
5%	5	1,688.3	1,721.8	1.9%	1.5%
10%	infinity	2,469.2	2,529.8	2.4%	1.4%
10%	100	2,466.0	2,521.0	2.2%	1.4%
10%	50	2,423.5	2,506.7	3.3%	1.5%
10%	20	2,159.4	2,226.6	3.0%	1.6%
10%	10	1,687.4	1,730.6	2.5%	1.6%
10%	5	1,189.5	1,233.7	3.6%	1.8%
20%	infinity	1,666.0	1,700.7	2.0%	2.0%
20%	100	1,529.0	1,599.1	4.4%	2.1%
20%	50	1,534.0	1,556.9	1.5%	2.2%
20%	20	1,478.1	1,510.4	2.1%	2.2%
20%	10	857.5	881.9	2.8%	2.3%
20%	5	810.6	835.0	2.9%	2.4%
50%	infinity	165.8	169.4	2.2%	3.2%
50%	100	163.9	167.5	2.2%	3.6%
50%	50	165.7	167.2	0.9%	3.7%
50%	20	162.1	162.9	0.5%	3.7%
50%	10	152.4	155.6	2.0%	3.8%
50%	5	139.2	145.3	4.2%	4.1%

Table XLI. Solution times for MSMIP instance B3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM	Solution time for bound on gap (minutes)
2%	infinity	60.0	26.3	44%	113.4
2%	100	60.0	20.6	34%	114.6
2%	50	60.0	29.1	49%	113.0
2%	20	60.1	28.8	48%	120.0
2%	10	60.0	26.5	44%	116.6
2%	5	60.0	27.0	45%	119.5
5%	infinity	60.1	19.8	33%	113.1
5%	100	60.0	27.4	46%	118.9
5%	50	60.1	20.2	34%	112.8
5%	20	60.0	22.6	38%	116.8
5%	10	60.1	23.8	40%	115.4
5%	5	60.0	25.7	43%	112.3
10%	infinity	60.0	28.9	48%	115.4
10%	100	60.0	32.5	54%	111.3
10%	50	60.0	26.2	44%	113.1
10%	20	60.1	27.6	46%	119.9
10%	10	60.0	22.9	38%	112.7
10%	5	60.1	31.5	52%	117.4
20%	infinity	60.0	24.7	41%	111.4
20%	100	60.1	27.3	45%	114.2
20%	50	60.1	26.9	45%	115.7
20%	20	60.0	12.4	21%	119.1
20%	10	60.1	24.9	41%	114.6
20%	5	60.0	26.8	45%	112.6
50%	infinity	60.0	27.6	46%	113.8
50%	100	60.0	30.6	51%	121.0
50%	50	60.1	25.7	43%	117.8
50%	20	60.1	25.0	42%	117.2
50%	10	60.1	28.3	47%	117.4
50%	5	60.0	27.9	46%	116.8

Table XLII. Solution values for MSMIP instance B4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at one hour runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM	Bound on optimality gap
2%	infinity	111,946.5	115,251.9	2.9%	0.5%
2%	100	97,365.7	99,990.0	2.6%	0.6%
2%	50	71,376.1	74,239.1	3.9%	0.6%
2%	20	39,815.2	41,135.2	3.2%	0.7%
2%	10	24,213.6	24,936.7	2.9%	0.7%
2%	5	15,086.5	15,601.3	3.3%	0.8%
5%	infinity	40,546.6	42,648.9	4.9%	1.2%
5%	100	40,305.9	42,363.8	4.9%	1.3%
5%	50	37,315.3	39,430.9	5.4%	1.3%
5%	20	26,961.1	28,411.1	5.1%	1.4%
5%	10	18,208.4	19,316.6	5.7%	1.4%
5%	5	12,047.5	12,760.7	5.6%	1.4%
10%	infinity	17,925.0	18,735.6	4.3%	1.4%
10%	100	17,906.9	18,696.0	4.2%	1.4%
10%	50	17,744.5	18,577.2	4.5%	1.7%
10%	20	15,881.2	16,498.9	3.7%	1.7%
10%	10	12,310.4	12,831.3	4.1%	1.9%
10%	5	8,831.6	9,130.2	3.3%	1.9%
20%	infinity	11,752.1	12,242.0	4.0%	2.1%
20%	100	11,352.4	11,819.2	3.9%	2.2%
20%	50	11,179.3	11,552.3	3.2%	2.2%
20%	20	10,734.9	11,205.6	4.2%	2.2%
20%	10	6,300.5	6,546.9	3.8%	2.5%
20%	5	5,873.2	6,205.3	5.4%	2.5%
50%	infinity	1,231.3	1,256.7	2.0%	2.9%
50%	100	1,212.4	1,239.4	2.2%	3.0%
50%	50	1,192.9	1,225.3	2.6%	3.0%
50%	20	1,164.6	1,188.5	2.0%	3.0%
50%	10	1,123.6	1,151.0	2.4%	3.2%
50%	5	1,039.7	1,060.9	2.0%	3.3%

Table XLIII. Solution times for MSMIP instance B4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM	Solution time for bound on gap (minutes)
2%	infinity	60.1	19.4	32%	48.1
2%	100	60.1	23.4	39%	50.0
2%	50	60.1	21.9	36%	46.6
2%	20	60.1	21.7	36%	50.6
2%	10	60.1	20.2	34%	45.8
2%	5	60.0	22.1	37%	47.5
5%	infinity	60.1	17.9	30%	53.0
5%	100	60.1	20.3	34%	52.2
5%	50	60.1	20.8	35%	52.6
5%	20	60.0	20.3	34%	50.5
5%	10	60.1	20.4	34%	49.3
5%	5	60.1	19.5	32%	49.3
10%	infinity	60.1	21.6	36%	70.6
10%	100	60.1	22.8	38%	75.6
10%	50	60.1	23.1	38%	75.9
10%	20	60.0	22.1	37%	72.4
10%	10	60.1	22.1	37%	73.7
10%	5	60.0	21.7	36%	76.6
20%	infinity	60.1	24.5	41%	68.2
20%	100	60.0	23.8	40%	65.3
20%	50	60.1	25.2	42%	67.6
20%	20	60.1	24.7	41%	64.2
20%	10	60.1	26.3	44%	66.2
20%	5	60.1	26.8	45%	66.6
50%	infinity	60.1	21.7	36%	57.9
50%	100	60.0	21.5	36%	53.9
50%	50	60.0	21.8	36%	52.0
50%	20	60.2	20.0	33%	52.1
50%	10	60.0	19.9	33%	54.3
50%	5	60.1	21.7	36%	51.7

Table XLIV. Solution values for MSMIP instance B5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at one hour runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM	Bound on optimality gap
2%	infinity	18,311.9	18,744.3	2.31%	0.08%
2%	100	15,863.6	16,256.6	2.42%	0.09%
2%	50	11,779.0	12,070.8	2.42%	0.16%
2%	20	6,549.6	6,686.8	2.05%	0.22%
2%	10	3,937.3	4,052.3	2.84%	0.29%
2%	5	2,412.9	2,532.1	4.71%	0.40%
5%	infinity	6,689.4	6,934.7	3.54%	0.11%
5%	100	6,721.6	6,884.3	2.36%	0.12%
5%	50	6,254.5	6,408.4	2.40%	0.14%
5%	20	4,505.7	4,617.6	2.42%	0.15%
5%	10	3,024.0	3,136.3	3.58%	0.19%
5%	5	2,005.3	2,072.8	3.25%	0.28%
10%	infinity	3,795.5	3,957.8	4.10%	0.23%
10%	100	3,500.3	3,642.9	3.92%	0.25%
10%	50	2,905.5	3,015.1	3.63%	0.26%
10%	20	2,619.8	2,677.6	2.16%	0.32%
10%	10	1,995.6	2,082.0	4.15%	0.37%
10%	5	1,423.9	1,481.1	3.86%	0.39%
20%	infinity	2,094.6	2,186.2	4.19%	1.22%
20%	100	2,113.6	2,160.0	2.15%	1.27%
20%	50	1,796.7	1,873.7	4.11%	1.35%
20%	20	1,753.1	1,817.5	3.54%	1.36%
20%	10	1,019.1	1,059.0	3.77%	1.46%
20%	5	961.0	1,002.9	4.17%	1.71%
50%	infinity	194.8	201.3	3.23%	2.42%
50%	100	192.9	198.1	2.61%	2.54%
50%	50	188.8	193.7	2.55%	2.63%
50%	20	185.3	190.5	2.74%	2.64%
50%	10	176.9	184.9	4.34%	2.63%
50%	5	162.1	168.1	3.61%	2.97%

Table XLV. Solution times for MSMIP instance B5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	60.0	18.9	31%	55.1
2%	100	60.0	19.3	32%	56.3
2%	50	60.0	18.5	31%	58.1
2%	20	60.1	20.7	35%	59.9
2%	10	60.1	21.8	36%	58.0
2%	5	60.1	26.7	44%	56.9
5%	infinity	60.1	21.8	36%	71.5
5%	100	60.0	20.6	34%	71.9
5%	50	60.0	19.1	32%	73.0
5%	20	60.1	19.6	33%	75.2
5%	10	60.0	21.4	36%	71.0
5%	5	60.1	20.8	35%	72.3
10%	infinity	60.1	27.5	46%	57.8
10%	100	60.0	24.0	40%	54.5
10%	50	60.1	19.3	32%	56.2
10%	20	60.1	22.8	38%	58.3
10%	10	60.0	26.2	44%	52.9
10%	5	60.0	20.9	35%	57.2
20%	infinity	60.0	29.7	49%	65.0
20%	100	60.0	22.8	38%	63.1
20%	50	60.1	31.5	52%	67.9
20%	20	60.1	27.4	46%	68.8
20%	10	60.0	29.4	49%	66.6
20%	5	60.1	28.9	48%	66.8
50%	infinity	60.0	23.7	40%	36.6
50%	100	60.1	23.7	39%	35.2
50%	50	60.0	23.4	39%	34.5
50%	20	60.0	23.6	39%	32.9
50%	10	60.0	27.4	46%	32.4
50%	5	60.0	26.4	44%	38.5

APPENDIX C

COMPUTATIONAL RESULTS FOR THE EOH DECOMPOSITION OF MSMIPS FOR THE THIRD PROBLEM SIZE

Table XLVI. Solution values for MSMIP instance C2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at two hours runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM
2%	infinity	614,758.2	700,759.4	12.3%
2%	100	508,612.0	591,275.2	14.0%
2%	50	375,176.4	448,309.7	16.3%
2%	20	232,037.5	268,083.2	13.4%
2%	10	135,681.5	160,810.7	15.6%
2%	5	85,274.5	99,380.4	14.2%
5%	infinity	265,429.6	304,394.6	12.8%
5%	100	262,980.4	303,540.0	13.4%
5%	50	225,624.2	262,785.6	14.1%
5%	20	166,367.2	190,783.6	12.8%
5%	10	102,738.9	123,066.8	16.5%
5%	5	80,859.5	95,034.1	14.9%
10%	infinity	140,166.7	166,371.1	15.8%
10%	100	132,149.0	158,339.5	16.5%
10%	50	120,497.4	140,447.9	14.2%
10%	20	102,132.3	122,908.3	16.9%
10%	10	79,514.2	97,240.9	18.2%
10%	5	71,212.9	84,067.4	15.3%
20%	infinity	99,713.0	116,242.3	14.2%
20%	100	76,847.8	94,294.8	18.5%
20%	50	75,719.1	89,284.2	15.2%
20%	20	71,041.4	84,443.4	15.9%
20%	10	41,476.5	49,610.9	16.4%
20%	5	35,775.5	42,098.5	15.0%
50%	infinity	47,240.9	53,978.8	12.5%
50%	100	46,585.5	52,246.5	10.8%
50%	50	43,844.0	49,169.7	10.8%
50%	20	36,708.2	42,222.0	13.1%
50%	10	33,231.5	37,156.0	10.6%
50%	5	27,017.2	30,440.1	11.2%

Table XLVII. Solution times for MSMIP instance C2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
2%	infinity	120.1	65.2	54%
2%	100	120.1	51.9	43%
2%	50	120.1	57.5	48%
2%	20	120.1	62.5	52%
2%	10	120.1	47.1	39%
2%	5	120.0	44.4	37%
5%	infinity	120.1	48.9	41%
5%	100	120.0	46.3	39%
5%	50	120.1	69.5	58%
5%	20	120.0	50.4	42%
5%	10	120.1	41.9	35%
5%	5	120.0	58.0	48%
10%	infinity	120.1	61.9	52%
10%	100	120.0	53.1	44%
10%	50	120.1	43.3	36%
10%	20	121.1	68.5	57%
10%	10	120.0	56.8	47%
10%	5	120.0	63.3	53%
20%	infinity	120.1	64.6	54%
20%	100	120.1	82.3	68%
20%	50	120.0	76.5	64%
20%	20	120.1	64.9	54%
20%	10	120.1	65.2	54%
20%	5	120.1	62.1	52%
50%	infinity	120.0	72.5	60%
50%	100	120.0	68.0	57%
50%	50	120.1	71.2	59%
50%	20	120.1	73.0	61%
50%	10	120.1	73.9	62%
50%	5	120.1	72.9	61%

Table XLVIII. Solution values for MSMIP instance C3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM profit at two hours runtime ($ 1,000)	Heuristic solution profit ($ 1,000)	Improvement over DEM
2%	infinity	8,284.5	9,337.6	11.3%
2%	100	7,086.6	8,098.3	12.5%
2%	50	5,248.1	6,012.6	12.7%
2%	20	2,933.2	3,330.5	11.9%
2%	10	1,771.6	2,021.7	12.4%
2%	5	1,110.2	1,264.2	12.2%
5%	infinity	3,121.2	3,514.8	11.2%
5%	100	2,973.9	3,429.7	13.3%
5%	50	2,803.1	3,194.5	12.3%
5%	20	1,960.3	2,300.5	14.8%
5%	10	1,320.3	1,562.8	15.5%
5%	5	903.4	1,034.8	12.7%
10%	infinity	1,367.4	1,523.4	10.2%
10%	100	1,312.3	1,514.2	13.3%
10%	50	1,287.2	1,504.7	14.5%
10%	20	1,175.0	1,336.4	12.1%
10%	10	890.2	1,039.2	14.3%
10%	5	655.2	740.9	11.6%
20%	infinity	885.4	997.2	11.2%
20%	100	844.5	957.4	11.8%
20%	50	839.4	935.3	10.2%
20%	20	823.2	905.9	9.1%
20%	10	462.3	529.7	12.7%
20%	5	431.1	500.1	13.8%
50%	infinity	90.4	102.2	11.5%
50%	100	89.9	101.3	11.2%
50%	50	88.9	100.5	11.5%
50%	20	86.7	96.2	9.9%
50%	10	86.0	94.9	9.3%
50%	5	78.4	86.8	9.7%

Table XLIX. Solution times for MSMIP instance C3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	DEM solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/DEM
2%	infinity	120.0	57.7	48%
2%	100	120.0	63.7	53%
2%	50	120.0	51.8	43%
2%	20	120.0	57.6	48%
2%	10	120.1	56.8	47%
2%	5	120.0	54.1	45%
5%	infinity	120.1	37.7	31%
5%	100	120.0	29.8	25%
5%	50	120.1	42.7	36%
5%	20	120.0	39.1	33%
5%	10	120.0	34.3	29%
5%	5	120.0	59.2	49%
10%	infinity	120.0	48.6	40%
10%	100	120.1	57.9	48%
10%	50	120.0	57.3	48%
10%	20	120.1	41.3	34%
10%	10	120.0	47.6	40%
10%	5	120.0	42.7	36%
20%	infinity	120.0	55.6	46%
20%	100	120.1	45.2	38%
20%	50	120.0	61.4	51%
20%	20	120.1	54.6	45%
20%	10	120.1	58.8	49%
20%	5	120.0	55.3	46%
50%	infinity	120.0	53.5	45%
50%	100	120.0	51.0	42%
50%	50	120.0	57.4	48%
50%	20	120.1	44.6	37%
50%	10	120.1	42.9	36%
50%	5	120.0	40.4	34%

APPENDIX D

COMPUTATIONAL RESULTS FOR THE EOH DECOMPOSITION OF SMIPS FOR THE FIRST PROBLEM SIZE

Table L. Solution values for SMIP instance A2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	366.3	366.3	0.0%	0.1%
2%	100	350.1	350.1	0.0%	0.0%
2%	50	240.6	240.6	0.0%	0.0%
2%	20	137.5	136.9	0.4%	1.8%
2%	10	78.9	77.7	1.5%	3.9%
2%	5	48.3	47.3	1.9%	4.8%
5%	infinity	149.6	149.6	0.0%	0.0%
5%	100	146.7	146.7	0.0%	0.0%
5%	50	127.5	126.4	0.9%	1.9%
5%	20	85.3	83.9	1.6%	3.3%
5%	10	58.1	56.7	2.4%	4.8%
5%	5	46.4	45.0	2.9%	5.8%
10%	infinity	57.8	57.8	0.0%	0.0%
10%	100	58.4	58.4	0.0%	0.6%
10%	50	56.4	55.8	1.1%	2.5%
10%	20	50.8	49.6	2.4%	4.7%
10%	10	39.8	38.6	2.9%	5.7%
10%	5	31.2	30.2	3.4%	6.6%
20%	infinity	43.5	43.2	0.5%	1.2%
20%	100	42.7	42.1	1.5%	3.0%
20%	50	40.1	39.2	2.2%	4.5%
20%	20	39.1	37.9	3.0%	6.1%
20%	10	27.9	26.9	3.6%	7.3%
20%	5	21.9	21.0	4.0%	8.0%
50%	infinity	4.5	4.4	0.9%	1.8%
50%	100	4.1	4.1	1.9%	3.9%
50%	50	4.1	4.0	2.8%	5.3%
50%	20	4.3	4.0	5.6%	9.2%
50%	10	4.0	3.8	5.9%	11.1%
50%	5	4.0	3.7	5.7%	12.3%

Table LI. Solution times for SMIP instance A2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	7.77	0.41	5%	3.38
2%	100	3.77	0.40	11%	1.64
2%	50	3.91	0.42	11%	1.70
2%	20	4.00	0.34	8%	1.74
2%	10	10.45	0.32	3%	4.54
2%	5	8.72	0.40	5%	3.79
5%	infinity	6.91	0.58	8%	3.00
5%	100	6.66	0.59	9%	2.89
5%	50	8.59	0.56	6%	3.73
5%	20	3.83	0.54	14%	1.66
5%	10	3.89	0.51	13%	1.69
5%	5	3.57	0.59	17%	1.55
10%	infinity	4.84	0.81	17%	2.11
10%	100	6.67	0.71	11%	2.90
10%	50	5.17	0.82	16%	2.25
10%	20	8.29	0.81	10%	3.60
10%	10	6.34	0.84	13%	2.75
10%	5	4.71	0.77	16%	2.05
	infinity				
20%	1000000	10.72	0.45	4%	4.66
20%	100	7.10	0.45	6%	3.09
20%	50	9.87	0.70	7%	4.29
20%	20	8.84	0.75	8%	3.84
20%	10	7.16	0.70	10%	3.11
20%	5	6.06	0.72	12%	2.64
50%	infinity	7.79	0.31	4%	3.39
50%	100	8.89	0.30	3%	3.87
50%	50	7.69	0.38	5%	3.34
50%	20	7.58	0.32	4%	3.29
50%	10	8.45	0.33	4%	3.67
50%	5	7.43	0.33	4%	3.23

Table LII. Solution values for SMIP instance A3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	884.2	884.2	0.0%	0.0%
2%	100	799.7	799.7	0.0%	0.1%
2%	50	542.0	542.0	0.0%	0.1%
2%	20	330.5	330.5	0.0%	0.1%
2%	10	180.0	175.6	2.4%	2.9%
2%	5	121.3	117.1	3.4%	6.6%
5%	infinity	322.6	322.6	0.0%	0.1%
5%	100	314.6	314.6	0.0%	0.1%
5%	50	301.5	299.6	0.6%	0.8%
5%	20	204.7	201.3	1.6%	2.6%
5%	10	147.2	143.3	2.7%	4.9%
5%	5	94.0	91.0	3.2%	6.7%
10%	infinity	145.7	145.7	0.0%	0.1%
10%	100	144.0	144.0	0.0%	0.2%
10%	50	141.3	138.3	2.1%	2.5%
10%	20	122.8	119.6	2.6%	5.3%
10%	10	101.4	98.2	3.1%	6.4%
10%	5	76.2	73.5	3.6%	7.5%
20%	infinity	93.0	93.0	0.0%	0.1%
20%	100	89.4	88.3	1.3%	1.6%
20%	50	88.0	85.6	2.7%	4.5%
20%	20	86.4	83.3	3.6%	7.1%
20%	10	51.5	49.6	3.8%	8.2%
20%	5	50.2	48.0	4.4%	9.1%
50%	infinity	9.2	9.1	1.4%	1.7%
50%	100	9.2	9.1	1.5%	3.4%
50%	50	9.2	8.9	2.9%	5.0%
50%	20	9.2	8.9	3.6%	7.2%
50%	10	9.2	8.8	3.9%	8.4%
50%	5	8.9	8.5	5.0%	9.9%

Table LIII. Solution times for SMIP instance A3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	7.32	0.63	8.6%	4.43
2%	100	4.67	0.60	12.9%	2.09
2%	50	8.05	0.55	6.9%	1.71
2%	20	5.37	0.53	9.9%	2.25
2%	10	8.59	0.54	6.3%	2.93
2%	5	8.02	0.57	7.1%	2.99
5%	infinity	7.41	0.82	11.0%	2.70
5%	100	7.40	0.84	11.4%	3.93
5%	50	9.35	0.73	7.8%	3.52
5%	20	10.03	0.79	7.9%	1.21
5%	10	8.65	0.74	8.6%	1.12
5%	5	6.63	0.84	12.7%	1.49
10%	infinity	5.47	0.85	15.5%	1.90
10%	100	7.38	0.94	12.7%	2.89
10%	50	5.76	1.04	18.1%	2.23
10%	20	7.20	0.83	11.5%	1.95
10%	10	7.09	0.80	11.3%	1.24
10%	5	8.28	0.85	10.3%	3.80
20%	infinity	9.62	0.62	6.4%	3.42
20%	100	14.66	0.59	4.0%	2.55
20%	50	6.03	0.60	10.0%	3.09
20%	20	9.09	0.64	7.0%	3.23
20%	10	8.47	0.63	7.4%	4.08
20%	5	10.98	0.60	5.5%	4.16
50%	infinity	8.62	0.79	9.2%	1.04
50%	100	10.52	0.71	6.8%	4.98
50%	50	7.12	0.78	10.9%	4.59
50%	20	10.26	0.60	5.8%	1.77
50%	10	8.90	0.62	7.0%	1.68
50%	5	6.35	0.61	9.6%	3.41

Table LIV. Solution values for SMIP instance A4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	3,779.8	3,779.8	0.0%	1.4%
2%	100	3,215.5	3,209.9	0.2%	1.6%
2%	50	2,275.8	2,269.0	0.3%	1.8%
2%	20	1,183.0	1,178.0	0.4%	2.1%
2%	10	775.9	772.6	0.4%	2.2%
2%	5	499.2	495.7	0.7%	2.4%
5%	infinity	1,427.2	1,426.0	0.1%	1.4%
5%	100	1,336.3	1,329.6	0.5%	1.9%
5%	50	1,199.8	1,182.2	1.5%	3.4%
5%	20	934.2	906.6	3.0%	5.7%
5%	10	627.3	601.3	4.2%	8.5%
5%	5	401.8	383.5	4.5%	9.9%
10%	infinity	710.2	708.1	0.3%	1.6%
10%	100	666.5	661.1	0.8%	2.5%
10%	50	618.4	602.2	2.6%	4.8%
10%	20	530.4	510.7	3.7%	7.7%
10%	10	421.6	402.8	4.5%	9.6%
10%	5	294.3	278.4	5.4%	11.2%
20%	infinity	430.2	425.0	1.2%	2.5%
20%	100	409.7	399.0	2.6%	5.1%
20%	50	384.3	374.0	2.7%	6.6%
20%	20	332.8	317.6	4.6%	8.6%
20%	10	210.0	200.2	4.7%	10.7%
20%	5	206.2	193.8	6.0%	11.1%
50%	infinity	39.5	38.5	2.4%	3.8%
50%	100	38.8	37.7	3.0%	6.7%
50%	50	36.7	35.4	3.5%	7.8%
50%	20	36.2	34.1	5.7%	10.5%
50%	10	34.4	32.4	5.8%	12.8%
50%	5	34.4	32.1	6.8%	12.9%

Table LV. Solution times for SMIP instance A4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	8.8	0.6	7%	3.94
2%	100	6.3	0.6	10%	1.54
2%	50	5.3	0.6	12%	1.66
2%	20	9.6	0.8	8%	3.94
2%	10	7.8	0.7	8%	2.36
2%	5	7.2	0.6	8%	1.10
5%	infinity	8.2	0.7	8%	0.84
5%	100	9.6	0.7	7%	3.32
5%	50	8.5	0.7	8%	0.79
5%	20	7.8	0.7	8%	1.39
5%	10	9.6	0.8	8%	3.00
5%	5	7.6	0.6	8%	2.77
10%	infinity	10.2	0.6	6%	2.31
10%	100	5.8	0.6	10%	1.92
10%	50	6.0	0.6	10%	1.26
10%	20	7.2	0.6	8%	1.96
10%	10	8.5	0.7	8%	1.40
10%	5	9.7	0.7	7%	1.92
20%	infinity	9.7	0.8	8%	3.21
20%	100	13.0	0.8	6%	3.20
20%	50	9.9	0.8	8%	0.90
20%	20	14.2	1.1	8%	3.81
20%	10	9.5	0.8	8%	3.96
20%	5	9.1	0.8	9%	3.19
50%	infinity	5.7	0.6	10%	1.33
50%	100	5.8	0.6	11%	2.30
50%	50	7.5	0.6	8%	0.89
50%	20	5.1	0.6	12%	0.94
50%	10	3.9	0.5	14%	0.46
50%	5	3.9	0.6	15%	1.21

Table LVI. Solution values for SMIP instance A5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	32,291.7	32,235.3	0.2%	0.3%
2%	100	27,600.8	27,524.0	0.3%	0.5%
2%	50	20,651.7	20,589.0	0.3%	0.6%
2%	20	11,529.2	11,186.4	3.0%	3.3%
2%	10	6,995.5	6,758.5	3.4%	6.4%
2%	5	4,413.7	4,215.8	4.5%	7.9%
5%	infinity	11,863.3	11,847.1	0.1%	0.2%
5%	100	11,720.5	11,658.7	0.5%	0.7%
5%	50	10,893.7	10,760.4	1.2%	1.8%
5%	20	7,820.2	7,611.3	2.7%	3.9%
5%	10	5,330.8	5,130.7	3.8%	6.5%
5%	5	3,657.3	3,489.1	4.6%	8.4%
10%	infinity	5,295.3	5,275.7	0.4%	0.5%
10%	100	5,251.9	5,204.5	0.9%	1.4%
10%	50	5,245.3	5,114.0	2.5%	3.5%
10%	20	4,641.5	4,491.0	3.2%	5.8%
10%	10	3,632.8	3,490.7	3.9%	7.2%
10%	5	2,655.4	2,529.6	4.7%	8.7%
20%	infinity	3,632.6	3,591.6	1.1%	1.2%
20%	100	3,349.7	3,271.1	2.3%	3.5%
20%	50	3,199.0	3,095.3	3.2%	5.6%
20%	20	3,187.8	3,028.6	5.0%	8.3%
20%	10	1,884.0	1,788.4	5.1%	10.1%
20%	5	1,879.7	1,763.5	6.2%	11.3%
50%	infinity	356.1	348.1	2.2%	2.3%
50%	100	346.0	336.4	2.8%	5.1%
50%	50	341.6	329.2	3.6%	6.4%
50%	20	340.7	323.8	5.0%	8.6%
50%	10	337.3	319.9	5.1%	10.2%
50%	5	317.6	297.0	6.5%	11.7%

Table LVII. Solution times for SMIP instance A5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	8.89	1.03	12%	1.63
2%	100	6.52	1.03	16%	1.98
2%	50	6.77	1.09	16%	1.68
2%	20	4.61	0.96	21%	1.69
2%	10	5.12	1.27	25%	1.35
2%	5	9.11	0.73	8%	1.33
5%	infinity	9.11	1.12	12%	4.08
5%	100	9.84	1.17	12%	4.85
5%	50	6.93	0.71	10%	2.00
5%	20	8.01	1.26	16%	2.20
5%	10	5.15	1.30	25%	2.07
5%	5	6.58	0.77	12%	2.92
10%	infinity	5.51	1.15	21%	1.63
10%	100	6.65	1.23	19%	1.57
10%	50	8.15	1.23	15%	3.61
10%	20	15.39	0.74	5%	2.44
10%	10	9.83	1.05	11%	3.63
10%	5	5.35	0.74	14%	1.61
20%	infinity	8.28	1.26	15%	2.97
20%	100	9.46	1.03	11%	3.95
20%	50	6.10	1.31	21%	1.67
20%	20	5.20	1.00	19%	1.96
20%	10	9.02	0.80	9%	3.51
20%	5	7.57	0.72	9%	1.74
50%	infinity	8.13	1.16	14%	3.37
50%	100	6.48	1.26	19%	3.22
50%	50	7.59	0.70	9%	3.53
50%	20	7.33	0.80	11%	2.52
50%	10	9.57	0.68	7%	1.72
50%	5	6.43	1.19	18%	2.99

APPENDIX E

COMPUTATIONAL RESULTS FOR THE EOH DECOMPOSITION OF SMIPS FOR THE SECOND PROBLEM SIZE

Table LVIII. Solution values for SMIP instance B2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	269.1	269.0	0.0%	0.1%
2%	100	217.2	217.1	0.0%	0.1%
2%	50	163.7	163.5	0.1%	0.2%
2%	20	101.6	101.4	0.2%	0.4%
2%	10	57.9	56.6	2.1%	3.3%
2%	5	38.6	37.6	2.4%	4.6%
5%	infinity	105.6	105.6	0.0%	0.1%
5%	100	102.3	101.9	0.4%	0.5%
5%	50	91.2	90.4	0.9%	1.3%
5%	20	65.0	63.7	1.9%	2.8%
5%	10	45.5	44.3	2.6%	4.6%
5%	5	31.1	30.2	2.8%	5.5%
10%	infinity	44.9	44.8	0.1%	0.2%
10%	100	44.6	44.4	0.5%	0.7%
10%	50	44.1	43.3	1.7%	2.3%
10%	20	43.6	42.6	2.3%	4.1%
10%	10	29.8	28.9	2.8%	5.2%
10%	5	25.1	24.2	3.4%	6.3%
20%	infinity	35.3	35.0	0.7%	0.8%
20%	100	31.7	31.2	1.7%	2.4%
20%	50	31.1	30.4	2.3%	4.0%
20%	20	27.5	26.6	3.5%	5.9%
20%	10	14.7	14.2	3.7%	7.3%
20%	5	14.5	13.8	4.4%	8.2%
50%	infinity	5.1	5.1	1.5%	1.6%
50%	100	3.7	3.6	1.8%	3.4%
50%	50	3.5	3.4	2.7%	6.6%
50%	20	3.2	3.1	3.6%	7.3%
50%	10	3.2	3.1	4.7%	8.0%
50%	5	2.6	2.5	5.3%	9.6%

Table LIX. Solution times for SMIP instance B2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	35.4	8.2	23%	25.4
2%	100	33.6	8.5	25%	10.8
2%	50	42.2	10.7	25%	24.8
2%	20	40.1	8.0	20%	12.9
2%	10	35.3	10.1	29%	15.8
2%	5	33.9	8.2	24%	22.0
5%	infinity	36.6	8.8	24%	10.7
5%	100	40.1	9.9	25%	13.3
5%	50	34.2	9.0	26%	11.6
5%	20	35.3	9.1	26%	24.0
5%	10	35.9	7.1	20%	20.4
5%	5	40.7	9.9	24%	28.7
10%	infinity	42.9	8.7	20%	10.9
10%	100	29.0	7.3	25%	10.8
10%	50	39.6	7.6	19%	30.4
10%	20	36.7	7.7	21%	33.3
10%	10	29.6	7.5	25%	27.2
10%	5	39.3	7.7	20%	13.6
20%	infinity	33.5	8.4	25%	21.8
20%	100	24.4	6.5	27%	9.0
20%	50	40.2	7.8	20%	19.9
20%	20	35.5	6.9	19%	9.3
20%	10	41.7	6.4	15%	26.0
20%	5	41.1	6.7	16%	40.4
50%	infinity	34.7	8.5	24%	8.7
50%	100	40.1	9.4	23%	17.6
50%	50	40.8	8.7	21%	8.7
50%	20	39.0	7.8	20%	18.9
50%	10	46.0	10.8	23%	24.5
50%	5	43.3	9.3	22%	22.3

Table LX. Solution values for SMIP instance B3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	13,383.5	13,383.5	0.0%	0.0%
2%	100	12,403.1	12,403.1	0.0%	0.4%
2%	50	8,837.4	8,837.4	0.0%	0.5%
2%	20	4,851.7	4,851.7	0.0%	1.0%
2%	10	3,004.0	2,987.1	0.6%	3.1%
2%	5	1,868.6	1,807.3	3.3%	6.4%
5%	infinity	4,953.9	4,953.9	0.0%	0.0%
5%	100	4,989.0	4,989.0	0.0%	0.4%
5%	50	4,684.9	4,684.9	0.0%	1.0%
5%	20	3,430.4	3,430.4	0.0%	1.6%
5%	10	2,347.1	2,318.3	1.2%	5.6%
5%	5	1,578.3	1,524.2	3.4%	5.9%
10%	infinity	2,175.6	2,175.6	0.0%	0.2%
10%	100	2,307.2	2,307.2	0.0%	0.7%
10%	50	2,284.8	2,284.8	0.0%	1.1%
10%	20	1,988.1	1,922.2	3.3%	5.1%
10%	10	1,489.0	1,438.5	3.4%	5.7%
10%	5	1,128.6	1,088.3	3.6%	6.1%
20%	infinity	1,462.6	1,462.6	0.0%	0.3%
20%	100	1,383.8	1,383.8	0.0%	1.7%
20%	50	1,370.2	1,366.4	0.3%	2.8%
20%	20	1,366.2	1,319.2	3.4%	6.0%
20%	10	762.5	734.9	3.6%	6.9%
20%	5	752.2	716.9	4.7%	7.0%
50%	infinity	145.7	145.7	0.0%	0.8%
50%	100	147.6	145.7	1.3%	2.2%
50%	50	147.1	141.7	3.7%	4.3%
50%	20	142.8	135.0	5.4%	7.1%
50%	10	140.0	132.1	5.6%	10.4%
50%	5	125.9	118.2	6.1%	12.3%

Table LXI. Solution times for SMIP instance B3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	33.1	5.3	16%	26.2
2%	100	41.7	4.2	10%	12.5
2%	50	34.1	5.0	15%	24.2
2%	20	42.1	5.8	14%	19.6
2%	10	33.8	6.8	20%	17.3
2%	5	23.4	4.4	19%	13.4
5%	infinity	27.8	5.1	18%	18.1
5%	100	34.8	5.4	16%	14.9
5%	50	32.9	5.7	17%	15.8
5%	20	34.7	5.9	17%	23.8
5%	10	41.8	9.2	22%	20.9
5%	5	35.5	9.3	26%	20.3
10%	infinity	25.1	5.5	22%	12.5
10%	100	28.7	4.7	16%	9.6
10%	50	26.5	5.1	19%	13.6
10%	20	33.8	8.1	24%	20.3
10%	10	41.9	9.5	23%	11.7
10%	5	35.6	8.1	23%	19.0
20%	infinity	39.2	5.1	13%	19.1
20%	100	38.6	6.4	17%	19.6
20%	50	41.7	10.3	25%	12.1
20%	20	40.2	5.9	15%	10.0
20%	10	28.7	5.4	19%	17.9
20%	5	27.0	5.5	20%	17.2
50%	infinity	39.3	4.7	12%	25.3
50%	100	36.0	5.2	14%	10.5
50%	50	26.8	5.4	20%	11.6
50%	20	32.8	6.1	19%	16.7
50%	10	25.5	5.2	20%	9.1
50%	5	31.0	7.2	23%	15.8

Table LXII. Solution values for SMIP instance B4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	94,770.2	94,769.7	0.0%	0.0%
2%	100	86,788.2	86,788.2	0.0%	0.0%
2%	50	67,659.5	67,617.2	0.1%	0.2%
2%	20	35,962.3	35,874.7	0.2%	0.4%
2%	10	23,793.6	23,603.3	0.8%	1.3%
2%	5	14,655.1	14,442.6	1.5%	2.7%
5%	infinity	36,024.4	36,024.2	0.0%	0.0%
5%	100	35,312.3	35,311.7	0.0%	0.0%
5%	50	33,121.9	33,021.7	0.3%	0.4%
5%	20	25,716.0	25,537.7	0.7%	1.2%
5%	10	17,624.6	17,342.6	1.6%	2.8%
5%	5	11,421.6	11,152.4	2.4%	4.7%
10%	infinity	16,805.9	16,801.0	0.0%	0.0%
10%	100	15,754.2	15,606.6	0.9%	1.2%
10%	50	15,606.0	15,388.3	1.4%	3.2%
10%	20	15,508.9	15,276.3	1.5%	4.3%
10%	10	11,792.8	11,518.2	2.3%	5.1%
10%	5	7,616.0	7,391.7	2.9%	6.3%
20%	infinity	11,039.2	11,037.9	0.0%	0.0%
20%	100	10,674.0	10,556.6	1.1%	1.3%
20%	50	9,662.1	9,507.5	1.6%	3.2%
20%	20	9,652.4	9,478.6	1.8%	4.1%
20%	10	5,897.0	5,749.6	2.5%	6.2%
20%	5	5,447.1	5,261.9	3.4%	7.1%
50%	infinity	1,156.3	1,143.6	1.1%	1.3%
50%	100	1,139.2	1,126.1	1.1%	2.7%
50%	50	1,097.1	1,077.4	1.8%	3.9%
50%	20	1,046.6	1,018.3	2.7%	4.8%
50%	10	1,001.8	965.8	3.6%	8.6%
50%	5	906.4	851.1	6.1%	11.6%

Table LXIII. Solution times for SMIP instance B4 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	24.2	6.3	26%	17.4
2%	100	40.2	10.1	25%	28.4
2%	50	47.8	12.0	25%	31.0
2%	20	44.3	11.1	25%	25.7
2%	10	41.2	10.4	25%	20.9
2%	5	33.1	8.4	25%	23.2
5%	infinity	23.9	6.2	26%	10.7
5%	100	25.1	6.5	26%	10.6
5%	50	38.0	9.6	25%	32.2
5%	20	45.5	11.4	25%	27.7
5%	10	29.1	7.5	26%	16.3
5%	5	43.8	11.0	25%	25.7
10%	infinity	37.5	9.5	25%	18.4
10%	100	36.7	9.3	25%	33.3
10%	50	43.2	10.8	25%	26.4
10%	20	41.5	10.4	25%	31.7
10%	10	28.0	7.2	26%	20.2
10%	5	30.8	7.9	26%	20.3
20%	infinity	32.2	8.2	25%	10.3
20%	100	26.8	6.9	26%	10.5
20%	50	40.2	10.1	25%	25.6
20%	20	36.9	9.3	25%	15.1
20%	10	23.5	6.1	26%	14.6
20%	5	41.9	10.5	25%	28.1
50%	infinity	45.6	11.4	25%	32.0
50%	100	47.8	11.9	25%	29.6
50%	50	36.2	9.2	25%	17.0
50%	20	36.6	9.3	25%	20.6
50%	10	45.8	11.5	25%	38.7
50%	5	47.0	11.8	25%	35.1

Table LXIV. Solution values for SMIP instance B5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	16,156.2	16,156.2	0.0%	0.0%
2%	100	13,709.9	13,709.9	0.0%	0.2%
2%	50	10,238.2	10,238.2	0.0%	0.3%
2%	20	5,716.9	5,716.9	0.0%	2.0%
2%	10	3,489.2	3,412.5	2.2%	5.2%
2%	5	2,161.5	2,109.5	2.4%	5.9%
5%	infinity	5,962.9	5,962.9	0.0%	0.0%
5%	100	5,940.9	5,940.9	0.0%	0.3%
5%	50	5,570.0	5,570.0	0.0%	1.2%
5%	20	4,188.1	4,149.3	0.9%	3.1%
5%	10	2,873.3	2,766.7	3.7%	5.6%
5%	5	1,801.9	1,733.4	3.8%	6.7%
10%	infinity	3,524.8	3,524.8	0.0%	0.2%
10%	100	3,087.1	3,087.1	0.0%	0.7%
10%	50	2,621.3	2,621.3	0.0%	2.2%
10%	20	2,296.7	2,231.2	2.9%	4.9%
10%	10	1,799.7	1,715.2	4.7%	6.1%
10%	5	1,280.1	1,219.5	4.7%	6.8%
20%	infinity	1,875.9	1,875.9	0.0%	0.6%
20%	100	1,852.6	1,848.6	0.2%	2.5%
20%	50	1,722.6	1,670.9	3.0%	5.0%
20%	20	1,611.0	1,557.2	3.3%	5.7%
20%	10	955.5	903.5	5.4%	7.3%
20%	5	848.7	802.2	5.5%	8.0%
50%	infinity	180.8	180.8	0.0%	1.2%
50%	100	178.7	175.4	1.8%	3.9%
50%	50	172.3	165.3	4.0%	6.2%
50%	20	161.1	151.0	6.3%	6.4%
50%	10	159.0	148.5	6.6%	10.6%
50%	5	148.8	138.5	6.9%	11.0%

Table LXV. Solution times for SMIP instance B5 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	24.4	5.9	24%	15.0
2%	100	41.5	10.2	25%	36.0
2%	50	38.6	9.4	24%	11.7
2%	20	37.0	9.5	26%	19.3
2%	10	43.9	9.7	22%	11.8
2%	5	39.8	9.5	24%	16.9
5%	infinity	36.1	5.9	16%	16.9
5%	100	19.3	5.0	26%	16.6
5%	50	39.6	5.7	14%	18.8
5%	20	42.4	7.4	18%	20.3
5%	10	37.7	5.1	14%	26.7
5%	5	41.2	7.9	19%	14.2
10%	infinity	25.5	5.6	22%	20.1
10%	100	32.5	6.2	19%	21.0
10%	50	37.2	7.9	21%	10.5
10%	20	32.3	5.5	17%	31.1
10%	10	31.3	5.3	17%	19.9
10%	5	48.4	10.5	22%	13.5
20%	infinity	33.6	6.4	19%	11.4
20%	100	40.7	8.2	20%	24.8
20%	50	41.7	10.1	24%	24.3
20%	20	37.7	10.2	27%	33.6
20%	10	34.7	8.1	23%	15.4
20%	5	21.3	5.2	24%	20.2
50%	infinity	31.8	4.3	13%	17.3
50%	100	40.5	4.3	11%	29.0
50%	50	29.6	4.9	17%	17.1
50%	20	32.0	4.7	15%	27.4
50%	10	39.6	4.1	10%	29.1
50%	5	36.0	5.0	14%	32.3

APPENDIX F

COMPUTATIONAL RESULTS FOR THE EOH DECOMPOSITION OF SMIPS FOR THE THIRD PROBLEM SIZE

Table LXVI. Solution values for SMIP instance C2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	542,452.6	542,452.6	0.0%	0.0%
2%	100	488,641.0	488,641.0	0.0%	0.0%
2%	50	374,580.0	374,580.0	0.0%	0.0%
2%	20	225,521.4	223,942.7	0.7%	1.7%
2%	10	120,306.6	118,863.0	1.2%	2.0%
2%	5	77,864.6	76,930.2	1.2%	2.1%
5%	infinity	247,895.4	245,912.2	0.8%	0.8%
5%	100	243,072.0	241,127.4	0.8%	1.0%
5%	50	196,711.4	194,744.3	1.0%	1.2%
5%	20	153,638.1	151,640.8	1.3%	3.3%
5%	10	103,284.4	101,838.4	1.4%	3.5%
5%	5	76,011.0	74,566.8	1.9%	4.1%
10%	infinity	135,011.5	133,796.3	0.9%	1.7%
10%	100	134,031.5	132,691.2	1.0%	1.6%
10%	50	106,917.3	105,634.3	1.2%	3.2%
10%	20	90,795.3	89,615.0	1.3%	3.3%
10%	10	71,147.4	69,795.6	1.9%	4.5%
10%	5	63,776.0	62,436.7	2.1%	4.8%
20%	infinity	86,009.4	84,633.3	1.6%	3.1%
20%	100	76,122.6	74,904.6	1.6%	4.0%
20%	50	73,625.1	72,373.4	1.7%	4.1%
20%	20	64,186.0	63,030.7	1.8%	4.8%
20%	10	37,571.4	36,744.8	2.2%	5.0%
20%	5	34,429.4	33,637.5	2.3%	5.2%
50%	infinity	45,198.5	44,384.9	1.8%	4.0%
50%	100	40,772.4	39,997.7	1.9%	4.1%
50%	50	36,890.0	36,004.6	2.4%	4.9%
50%	20	32,321.3	31,157.7	3.6%	5.1%
50%	10	27,586.2	26,565.5	3.7%	8.3%
50%	5	22,518.5	21,572.7	4.2%	9.2%

Table LXVII. Solution times for SMIP instance C2 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	96.7	21.4	22%	49.1
2%	100	63.2	15.4	24%	43.9
2%	50	66.4	15.0	23%	39.6
2%	20	62.2	17.1	27%	36.7
2%	10	68.7	20.7	30%	49.1
2%	5	68.0	11.4	17%	54.4
5%	infinity	84.4	13.6	16%	57.5
5%	100	67.3	15.9	24%	53.5
5%	50	71.9	14.0	20%	58.5
5%	20	72.4	21.4	30%	60.2
5%	10	88.1	20.0	23%	62.0
5%	5	75.7	15.0	20%	52.6
10%	infinity	81.9	14.9	18%	41.4
10%	100	89.9	22.1	25%	46.8
10%	50	73.6	12.9	17%	42.2
10%	20	77.5	19.0	25%	40.8
10%	10	68.9	19.8	29%	39.0
10%	5	66.6	12.7	19%	47.8
20%	infinity	65.0	19.8	31%	73.0
20%	100	88.7	21.5	24%	75.1
20%	50	82.1	19.5	24%	69.3
20%	20	88.4	18.4	21%	72.1
20%	10	61.3	15.1	25%	70.9
20%	5	66.9	18.1	27%	75.1
50%	infinity	68.1	16.3	24%	40.1
50%	100	82.6	17.3	21%	40.9
50%	50	71.0	17.3	24%	49.0
50%	20	58.1	13.4	23%	47.8
50%	10	69.3	21.2	31%	50.7
50%	5	61.2	18.5	30%	43.2

Table LXVIII. Solution values for SMIP instance C3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution profit ($ 1,000)	Heuristic solution profit ($ 1,000)	Optimality gap	Bound on optimality gap
2%	infinity	7,594.4	7,594.4	0.0%	1.4%
2%	100	5,969.7	5,969.7	0.0%	2.0%
2%	50	4,738.1	4,643.4	2.0%	2.9%
2%	20	2,757.1	2,670.7	3.1%	3.5%
2%	10	1,575.5	1,516.7	3.7%	4.8%
2%	5	1,017.2	977.9	3.9%	4.9%
5%	infinity	2,833.9	2,760.9	2.6%	2.8%
5%	100	2,760.5	2,682.0	2.8%	3.1%
5%	50	2,366.7	2,296.5	3.0%	4.2%
5%	20	1,796.0	1,736.0	3.3%	4.7%
5%	10	1,211.7	1,166.5	3.7%	5.2%
5%	5	784.1	750.7	4.3%	5.3%
10%	infinity	1,160.3	1,118.7	3.6%	4.0%
10%	100	1,213.9	1,166.5	3.9%	4.1%
10%	50	1,161.0	1,115.0	4.0%	4.8%
10%	20	991.5	945.4	4.6%	5.2%
10%	10	860.8	818.6	4.9%	5.9%
10%	5	611.9	577.8	5.6%	6.0%
20%	infinity	784.0	752.6	4.0%	5.3%
20%	100	702.1	672.2	4.3%	5.5%
20%	50	772.4	738.8	4.3%	6.2%
20%	20	700.0	665.8	4.9%	6.3%
20%	10	389.5	368.4	5.4%	6.3%
20%	5	369.1	348.1	5.7%	6.5%
50%	infinity	80.9	77.6	4.2%	3.9%
50%	100	81.0	77.3	4.6%	7.5%
50%	50	81.7	77.9	4.7%	8.7%
50%	20	70.9	66.8	5.8%	9.3%
50%	10	79.9	74.9	6.2%	11.6%
50%	5	73.3	68.5	6.6%	11.8%

Table LXIX. Solution times for SMIP instance C3 under various combinations of discount rates and EoH approximations

Discount rate	End-of-horizon approximation (years)	Optimal solution time (minutes)	Heuristic solution time (minutes)	Solution time ratio: heuristic/optimal	Solution time for bound on gap (minutes)
2%	infinity	74.9	22.4	30%	51.1
2%	100	87.2	27.7	32%	55.7
2%	50	86.5	29.0	34%	51.3
2%	20	76.3	23.2	30%	52.8
2%	10	70.0	19.7	28%	65.8
2%	5	68.7	18.9	28%	31.6
5%	infinity	82.2	26.6	32%	62.8
5%	100	79.4	25.0	31%	57.1
5%	50	70.0	19.7	28%	42.2
5%	20	75.8	23.0	30%	49.8
5%	10	61.2	20.8	34%	38.4
5%	5	74.1	22.0	30%	68.1
10%	infinity	105.8	26.6	25%	65.0
10%	100	65.3	17.0	26%	46.4
10%	50	107.2	27.4	26%	49.3
10%	20	70.9	20.2	28%	37.7
10%	10	63.7	22.5	35%	40.9
10%	5	70.3	19.9	28%	30.7
20%	infinity	82.4	20.1	24%	42.3
20%	100	75.1	25.5	34%	66.8
20%	50	77.8	23.0	30%	39.1
20%	20	68.3	18.7	27%	33.7
20%	10	77.8	24.1	31%	63.3
20%	5	70.6	20.0	28%	53.3
50%	infinity	68.6	18.9	27%	44.7
50%	100	86.9	29.2	34%	68.4
50%	50	70.0	19.7	28%	39.9
50%	20	82.5	26.8	32%	56.9
50%	10	46.8	12.5	27%	37.6
50%	5	43.3	11.5	27%	36.5

APPENDIX G

PARAMETERS GENERATION FOR COMPUTATIONAL EXPERIMENTS

The generation of test problem instances follows three steps. First, the candidate sites for the supply chain network are randomly selected. Second, parameters for first node of the scenario-tree ($n = 1$) are randomly generated. Third, all other nodes ($n \neq 1, n \in \mathcal{N}$) generate their parameters by varying around those of their immediate ancestor.

The generated parameters fall in three categories: supply chain network candidate locations, markets' demand and capacities of suppliers and of manufacturing and distribution facilities, and cost parameters. The following sections explain how these parameters are generated.

1. Supply chain network

Generating locations for the entities of a supply chain network involves randomly selecting a *major trading center* (MTC) for each market, supply-region, candidate manufacturing site, and candidate distribution site. A MTC is a city that serves as a primary center of wholesaling, distribution, banking, and other specialized services for at least two *basic trading areas* (Rand-McNally, 2006). A basic trading area relies on a nearby city (a basic trading center) for shopping goods purchases (Rand-McNally, 2006). The USA has 50 MTCs (Rand-McNally, 2006); examples include New York City, Los Angeles, Chicago, Houston, Dallas, and San Antonio, among others.

The same MTC can house at most one entity of each hierarchical level of the supply chain network; for example, the same MTC can be the location for a market, a supplier, a manufacturing facility, and/or a distribution facility. Generating cost and demand parameters uses the following attributes for a MTC:

- A market ability index (MAI), which describes a market's potential relative to other markets of the USA (Rand-McNally, 2006). MAI takes into consideration many attributes of a market, such as the total disposable income of its inhabitants, its total annual retail sales, and its total population (Rand-McNally, 2006). MAI plays an essential role in the generation of demand/supply amounts for markets/supplier-zones.

- A cost of living index (CLI), which measures the relative cost for consumer goods and services in 303 areas of the USA (C2ER, 2008). The average for all participating places equals 100, and each participant's index is read as a percentage of the average for all places (C2ER, 2008). CLI plays an essential role in generating cost parameters.

Moreover, Rand-McNally (2006) mileage chart provides the distances between every two MTCs. These distances are based on the routes usually followed by freight trucks.

2. Markets' demand generation

A base demand is randomly generated using a discrete uniform distribution with given upper and lower bounds. The demand per product-family per market for each node of the scenario-tree is generated using the following equation:

$$d_{p,k}^{n} = \rho_n \times \rho_{k,n} \times \rho_{p,k,n} \times d_{\text{base}},$$

where $d_{p,k}^{n}$ is the demand of product-family p at market k for node n, ρ_n is a nodal multiplier and is defined in section 2.1, $\rho_{k,n}$ is a market multiplier and is defined in section 2.2, $\rho_{p,k,n}$ is a product-family multiplier and is defined in section 2.3, and d_{base} is the base demand.

2.1. Nodal multiplier

For the first scenario-tree node, $n = 1$, the nodal multiplier equals 1 ($\rho_1 = 1$). For all other nodes, the nodal multiplier, $\rho_n > 0$, describes the total demand for node n relative to its parent's, $\rho_{a(n)}$. Using $\rho_{a(n)}$ to define ρ_n produces dependent demand scenarios.

ρ_n results from multiplying its parent's nodal multiplier, $\rho_{a(n)}$, with a random number generated from a log-normal distribution $\lambda(\mu_n, \sigma)$, where μ_n is the expectation and σ is the standard deviation. The following factors motivate choosing a log-normal distribution to generate demand parameters:

- This distribution suits modeling economic uncertain variables such as demand (Kamath and Pakkala, 2002).
- Log-normal distributions preserve the nonnegativity of demand parameters.
- Other researchers have successfully used this distribution to generate the uncertain demand parameters of their SCND models (c.f. Huang and Ahmed (2009) and Santoso *et al.* (2005)).

The value of the standard deviation is held constant for all node, $\sigma = 0.2$. The value of μ_n depends on the scenario leading to node n. Three scenarios emanate from each node (except leaf nodes). The first scenario represents an increasing demand, $\mu_n = 1.5$. The second scenario represents a stable demand, $\mu_n = 1$. The third scenario represents a declining demand, $\mu_n = 0.5$. These choices are inspired by Huang and Ahmed (2009).

2.2. Market multiplier

The market multiplier, $\rho_{k,n}$, describes the share of each market k of the total demand for node n. For the first node, $\rho_{k,1}$ is generated as follows.

$$\rho_{k,1} = \frac{\text{MPI}_k}{\sum_{k \in \mathcal{K}} \text{MPI}_k} \qquad \forall\, k \in \mathcal{K},$$

where MPI_k is the market ability index for market k.

For all other scenario-tree nodes, $\rho_{k,n}$ is generated using their parents' market multiplier, $\rho_{k,a(n)}$, and a number $u_{n,k}$ between 0.8 and 1.2 randomly chosen from a uniform distribution;

$$\rho_{k,n} = \frac{u_{n,k}\rho_{k,a(n)}}{\sum_{k \in \mathcal{K}} u_{n,k}\rho_{k,a(n)}} \qquad \forall\, k \in \mathcal{K}, n = \{2, \ldots, N\}.$$

2.3. Product-family multiplier

The product-family multiplier $\rho_{p,k,n}$ indicates the share of each product-family $p \in \mathcal{P}$ of the total demand in a given market $k \in \mathcal{K}$. The following three steps assign the values of the product-family multiplier for the first scenario-tree node, $\rho_{p,k,1}$:

1. For each market, product-families are randomly classified into three categories with respect to their demand levels:

    - Product-families associated with low-demand
    - Product-families associated with medium-demand
    - Product-families associated with high-demand

2. A weight of 1 is assigned to each low-demand product-family, a weight of 2 is assigned to each medium-demand product-family, and a weight of 4 is assigned to each high-demand product-family.

3. The product-family multiplier $\rho_{p,k,1}$ results from dividing the weight assigned to product-family k by the total weights for all product-families.

For all other scenario-tree nodes, $\rho_{p,k,n}$ is generated using their parents' product-family multiplier as follows.

$$\rho_{p,k,n} = \frac{u_{p,n,k}\rho_{p,k,a(n)}}{\sum_{p\in\mathcal{P}} u_{p,n,k}\rho_{k,a(n)}} \qquad \forall\, p \in \mathcal{P}, k \in \mathcal{K}, n = \{2,\dots,N\},$$

where $u_{p,n,k}$ is a number between 0.8 and 1.2 chosen randomly from a uniform distribution.

3. Suppliers' capacity generation

A base total suppliers' capacity value (d_{supply}) is randomly generated by multiplying the base demand, d_{base}, by a number randomly selected from a uniform distribution with a lower bound equals 0.8 and an upper bound equals 1.5. A supplier's capacity per product-family for each node of the scenario-tree is generated using the following equation:

$$d^n_{p,s} = \rho_n \times \rho_{s,n} \times \rho_{p,s,n} \times d_{\text{supply}},$$

where $d^n_{p,s}$ is the capacity of supplier $s \in \mathcal{S}$ with respect to product-family $p \in \mathcal{P}$ at node $n = \{2,\dots,N\}$, ρ_n is a nodal multiplier, $\rho_{s,n}$ is a supplier's location multiplier, and $\rho_{p,s,n}$ is a product-family multiplier. The value of these multipliers are assigned using the same procedure applied for their market's demand counterparts (section 2.1–section 2.3).

4. Generation of capacity levels for manufacturing and distribution facilities

Manufacturing and distribution facilities are treated the same way. Therefore this section uses the generic term *facility* to refer to either a manufacturing or distribution facility.

Each technology q can process a subset of the product families ($\mathcal{P}_q \subseteq \mathcal{P}$). Moreover, each technology comes in a given number of capacity levels, $\ell \in \mathcal{L}_q$. The rates at which a technology q processes the different product families, $p \in \mathcal{P}_q$, depend on this technology's capacity level, $\ell \in \mathcal{L}_q$. The following rules guide the assignment of these processing rates:

1. At the maximum capacity level available for technology q ($\ell_{\max} \in \mathcal{L}_q$), the processing rates must be sufficient to satisfy the demand for all products at the first scenario-tree node, $\sum_{p\in\mathcal{P}_q}\sum_{k\in\mathcal{K}} d^1_{p,k}$.

2. At the minimum capacity level available for technology q ($\ell_1 \in \mathcal{L}_q$), the processing rates must be sufficient to satisfy a portion of the demand at the first scenario-tree node. This portion equals $\frac{1}{|\mathcal{J}_r|}$ for manufacturing facilities and $\frac{1}{|\mathcal{J}_w|}$ for distribution facilities. In other words, installing the smallest technology level in every candidate site must provide enough aggregate capacity to satisfy all markets' demand for all the products this technology can produce.

3. All other capacity levels are assigned processing rates between those of the minimum and maximum capacity levels such that the difference between every two successive levels' output is constant.

$c_{p,\ell,q}$ is the portion of capacity level ℓ of technology q required to process one product of family $p \in \mathcal{P}$; i.e., $c_{p,\ell,q}$ is the inverse of the processing rate of level ℓ of technology q for product p. For each technology q, the values for $c_{p,\ell,q}$ are assigned using the following steps. First, a number $u_{p,\ell,q}$ between 1 and 10 is randomly selected from a uniform distribution for each p and q combination, where $p \in \mathcal{P}_q$; i.e., $u_{p,\ell_1,q} = \cdots = u_{p,\ell_{\max},q} = u_{p,\ell,q}$. Second, consistent with the first processing rates' assignment rule, the following equation assigns the values for $c_{p,\ell_{\max},q}$.

$$c_{p,\ell_{\max},q} = \frac{u_{p,\ell,q}\left(\sum_{k\in\mathcal{K}} d^1_{p,k}\right)}{\sum_{p\in\mathcal{P}_q} u_{p,\ell,q}\left(\sum_{k\in\mathcal{K}} d^1_{p,k}\right)}.$$

Third, consistent with the second processing rates' assignment rule, the following equation assigns the values for $c_{p,\ell_1,q}$.

$$c_{p,\ell_1,q} = c_{p,\ell_{\max},q}|\mathcal{J}_f|,$$

where $\mathcal{J}_f = \mathcal{J}_r$ in case of manufacturing facilities, and $\mathcal{J}_f = \mathcal{J}_w$ in case of distribution facilities. Finally, $c_{p,\ell,q}, \ell \neq \ell_1, \ell \neq \ell_{\max}, \ell \in \mathcal{L}_q$, is assigned at constant intervals between $c_{p,\ell_1,q}$ and $c_{p,\ell_{\max},q}$ to satisfy the third processing rates' assignment rule.

5. Cost parameters generation

Cost parameters fall into seven categories:

- Variable processing cost ($h^n_{p,\ell,q,j}$) per product of family $p \in \mathcal{P}$ for capacity level $\ell \in \mathcal{L}_q$ of technology $q \in \mathcal{Q}_j$ in site $j \in \mathcal{J}$.

- Raw material and subassemblies procurement cost ($\varphi^n_{p,s}$) per product of family p from supplier-zone $s \in \mathcal{S}$.

- Revenue ($\varphi^n_{p,k}$) per product of family p sold at market $k \in \mathcal{K}$.

- Transportation cost (h_{p,j_1,j_2}) per product of family p between a source j_1 and a destination j_2.

- Fixed operating cost ($g^n_{\ell,q,j}$) for the facility operating at site $j \in \mathcal{J}$ using technology $q \in \mathcal{Q}_j$ at capacity level $\ell \in \mathcal{L}_q$.

- Retooling cost ($f^n_{\ell_1,\ell_2,q,j}$) incurred to upgrading or downgrading the capacity level of technology q from level ℓ_1 to level ℓ_2 at site j.

- The cost to open a new facility at site j ($f^n_{\text{open } j}$) and the cost to close the facility at site j ($f^n_{\text{close } j}$).

The variable processing cost ($h^n_{p,\ell,q,j}$) per product of family $p \in \mathcal{P}_q$ for capacity level $\ell \in \mathcal{L}_q$ of technology $q \in \mathcal{Q}_j$ in site $j \in \mathcal{J}$ is calculated by the following steps. First, for each technology q and product-family p combination, a random per unit processing cost ($h^1_{p,q}$) is randomly selected between 1 and 2 from a uniform distribution. Second, this cost is adjusted to reflect the cost of living index for the location j of this facility. Third, the present value of this adjusted cost is calculated for each node n. The following equation reflects these steps.

$$h^n_{p,\ell,q,j} = \pi(h^1_{p,q} \times \text{CLI}_j),$$

where CLI_j is the cost of living index for location j and $\pi(.)$ returns the present value (at the first time period).

The raw material and subassemblies procurement cost ($\varphi^n_{p,s}$) per product of family p from supplier-zone $s \in \mathcal{S}$ is calculated in two steps. First, for each product family, a random procurement cost is selected between 1 and 2 from a uniform distribution. Second, this procurement cost is adjusted to reflect the cost of living index for the location of supplier s. Third, the present value of the adjusted procurement cost is calculated for each node n.

The revenue ($\varphi^n_{p,k}$) per product of family p sold at market $k \in \mathcal{K}$ is assigned by the following three steps. First, for each product family, a random revenue is selected between $10\widehat{h^1_{p,\ell,q,j}}$ and $20\widehat{h^1_{p,\ell,q,j}}$, where $\widehat{h^1_{p,\ell,q,j}} = \max_{\ell\in\mathcal{L}_q, q\in\mathcal{Q}_j, j\in\mathcal{J}}\{\frac{h^1_{p,\ell,q,j}}{\text{CLI}_j}\}$. Second, adjust this revenue to reflect the cost of living index for the location of market k. Third, the present value of the adjusted revenue is calculated for each node n.

The transportation cost (h_{p,j_1,j_2}) per product of family p between a source j_1 and a destination j_2. It is calculated using the following five steps. First, for each product $p \in \mathcal{P}$, a random number representing the transportation cost per unit per mile is randomly selected between 0.002 to 0.005 from a uniform distribution. Second, for every source j_1 and destination j_2, the total trip cost is calculated by multiplying this random per mile cost by the distance between j_1 and j_2. Third, the trip cost is adjust using the average of the cost of living indices of locations j_1 and j_2. Fourth, the present value of this adjusted trip cost is calculated for each node n. Fifth, if this trip originate at a supplier s, the raw material and subassemblies procurement cost, $\varphi^n_{p,s}$, is added to the transportation cost. Sixth, if market k is the destination of this trip, the revenue per unit, $\varphi^n_{p,k}$, is subtracted from the transportation cost (resulting in a negative value).

Fixed operating cost ($g^n_{\ell,q,j}$) for the facility operating at site $j \in \mathcal{J}$ using technology $q \in \mathcal{Q}_j$ at capacity level $\ell \in \mathcal{L}_q$ is calculated using the following steps. First, an initial value equal to $\sum_{p \in \mathcal{P}_q} \frac{h^1_{p,\ell,q,j}}{|\mathcal{P}_q| \times c_{p,\ell,q}}$ is assigned. Note that $h^1_{p,\ell,q,j}$ is already adjusted to reflect the living cost index of location j. Second, to include the economies of scale in operating cost, this initial value is multiplied by $1.1^{(\ell_{\max}-\ell)}$. For example, for a technology q where $|\mathcal{L}_q| = 4$, $g^n_{\ell_2,q,j}$ is multiplied by $1.1^{(4-2)} = 1.1^2$. Third, the present value is calculated for each node n.

The values for retooling costs dependent on whether this retooling results in upgrading or downgrading a facility's capacity. The cost to upgrade the capacity of technology q from level ℓ_1 to level ℓ_2 is calculated in three steps. First, a number u_q between 1 and 10 is selected from a uniform distribution. Second, an initial value equal to $u_q \left(1.1^{(\ell_2-\ell_1)}\right) \left(g^1_{\ell_2,q,j} - \frac{g^1_{\ell_1,q,j}}{1.1^{(\ell_2-\ell_1)}}\right)$ is assigned. Third, the present value is calculated for each node n. The cost to downgrade the capacity of technology q from level ℓ_1 to level ℓ_2 follows the same steps with two exceptions. First, u_q

is selected between 0.5 and 2 to reflect the cost of removing equipment from this facility and costs associated with reducing its workforce. Second, use the absolute value for the initial retooling cost since in this case $g^1_{\ell_2,q,j} < g^1_{\ell_1,q,j}$.

The cost to open a new facility at site j ($f^n_{\text{open } j}$) is calculated in three steps. First, a number u_j between 1 and 3 is selected from a uniform distribution. Second, an initial value equal to $\frac{u_k \sum_{q \in \mathcal{Q}_j} g^1_{\ell_{\max},q,j}}{|\mathcal{Q}_j|}$ is assigned. Third, the present value is calculated for each node n. Note that $f^n_{\text{open } j}$ does not include the value of assets that are likely to be recouped when this site is closed; it only includes the cost to customize this site to suit the function of this facility.

The cost to close the facility at site j ($f^n_{\text{close } j}$) is calculated in three steps. First, a number u_j between 0.5 and 2 is selected from a uniform distribution. Second, an initial value equal to $\frac{u_k \sum_{q \in \mathcal{Q}_j} g^1_{\ell_{\max},q,j}}{|\mathcal{Q}_j|}$ is assigned. Third, the present value is calculated for each node n. Note that $f^n_{\text{close } j}$ only includes the cost to restore a site to its original condition and to properly dispose of all removed items.

VITA

Ayman Hassan Ragab received his Bachelor of Science and Master of Science degrees in production engineering from Alexandria University, Egypt, in 1995 and 1998, respectively. He recieved his Ph.D. in industrial engineering from Texas A&M University, College Station, TX, in 2010. His research interests include supply chain management and mathematical programming.

Mr. Ragab can be reached at Texas Transportation Institute, 1919 Research Parkway, College Station, TX 77834. His email is ayman@tamu.edu.