Chapter 15
Stochastic Programming for Supply Chain Planning Under Demand Uncertainty

Abstract In this chapter we focus on stochastic programming for making optimal supply chain planning decisions under uncertainty. In particular, we extend deterministic linear programming for supply-chain planning (SCP) by using stochastic programming to incorporate the issues of demand risk and liquidity risk. Because the resulting stochastic linear programming model is similar to that of Asset-Liability Management (ALM) and because the literature using stochastic programming for ALM is extensive, we survey various modeling and solution choices developed in this literature and discuss their applicability to supply chain planning. This survey forms a basis for making modeling/solution choices in research and in practice to manage the risks of unmet demand, excess inventory and inadequate cash liquidity when demand is uncertain.

15.1 Introduction

In this chapter we consider stochastic programming for making optimal supply chain planning decisions under uncertainty. In particular, we extend the linear programming (LP) model of deterministic supply-chain planning to take demand uncertainty and cash flows into account for the medium term. Because the resulting stochastic LP model is similar to that of Asset-Liability Management (ALM) and because the literature using stochastic programming is extensive, we survey various modeling and solution choices developed in the ALM literature and discuss their applicability to supply chain planning in this chapter. This survey forms a basis for making modeling/solution choices in research and in practice to manage the risks of unmet demand, excess inventory and inadequate cash liquidity when demand is uncertain.

Companies usually manage supply chain risks either at the strategic (long term) or at the tactical (medium term) level (Sodhi and Lee, 2007). In the medium term—typically 12–26 weeks for consumer electronics companies but up to 24 months for

M.S. Sodhi, C.S. Tang, *Managing Supply Chain Risk*,
International Series in Operations Research & Management Science 172,
DOI 10.1007/978-1-4614-3238-8_15,

petrochemical companies—supply-chain planning should incorporate the risks of unmet demand, excess inventory and even liquidity.

We extend deterministic linear programming for supply-chain planning (SCP) by using stochastic programming for demand uncertainty to consider unmet demand and excess inventory and by incorporating cash flows to consider liquidity risk as well. By noting that procurement and production decisions to match supply and demand in (centralized) SCP under different demand scenarios are similar to investment decisions in interest-rate-based securities to match the future cash flows from these securities to the financial institution's future liabilities under different interest-rate scenarios, we look to the ALM literature for modeling perspectives. Indeed, stochastic programming has been extensively used by researchers and practitioners for ALM (Sodhi, 2005b) and we can benefit from the experience of ALM researchers.

Our survey of the ALM (and supply chain) literature from the viewpoint of modeling and solution choices for stochastic programming for SCP prepares the groundwork for researchers and practitioners to consider the use of stochastic programming for managing demand, inventory and liquidity risks associated with SCP for the medium-term. Still, our focus is on stochastic programming and on demand uncertainty: We do not survey heuristics, dynamic programming algorithms, or optimal policies from the inventory management literature. We provide a simple model motivated by a Japanese consumer electronics company for illustration rather than for an instance of a real-life application. We do not address the credit risk of not being able to collect payments from customers, something quite important for liquidity risk. Our survey of the literature, as with the previous Chapters 13 and 14, is indicative rather than exhaustive. Finally, we do not consider decentralized supply chains or adaptation of financial instruments like options that may have application for some of the supply-chain contracts like the QF contracts in Chapter 14.

We motivate the use of stochastic programming as a modeling choice for SCP under uncertainty in Section 15.2. Section 15.3 presents a stochastic program formulation for an idealized supply chain under uncertainty. Section 15.4 presents various modeling choices and Section 15.5 presents solution choices, both motivated by the ALM literature, before the conclusion in Section 15.6.

15.2 Motivation

For the *long* term, SCP with multiple uncertain factors is too broad to tackle with mathematical programming alone and may require scenario planning (Sodhi, 2003). However, modeling solutions have been attempted in at least three ways: (1) through multiple runs of deterministic models (Geoffrion, 1976; Geoffrion and Powers, 1995; Cohen and Lee, 1988), (2) through simulation of deterministic models (Iassinovski et al., 2003), or (3) through stochastic programming (Eppen et al., 1989; Alonso-Ayuso et al., 2003; Swaminathan and Tayur, 1999). By contrast, for the *short* term, operational supply chain risks such as one-off delays in delivery of raw materials are usually dealt with operationally rather than through modeling solutions.

For the *medium* term, the deterministic mathematical programming literature provides models for planning production and distribution based on a single demand forecast. For example, Arntzen et al. (1995) and Camm et al. (1997) developed mixed integer programming models for SCP at Digital Equipment Corporation and for Procter & Gamble, respectively. These deterministic models are solved by using linear programming or heuristics solutions offered by vendors like SAP, i2 and Manugistics (Sodhi, 2001). Typically, a rolling horizon is used so that the forecast as well as the plan are updated every week within the company's enterprise resource planning (ERP) system supported by an advanced planning system (APS) (Sodhi, 2000; 2001). Sometimes, "what-if" forecasts are used in different runs to capture demand uncertainty. However, deterministic modeling even with multiple what-if forecasts is not adequate for managing supply chain risks because such modeling does not take consider the risk pertaining to unmet demand, excess inventory, or inadequate liquidity.

Unmet demand risk (understocking) and inventory risk (overstocking) can be managed by having the "appropriate" level of inventory determined by certain rules, heuristics, or algorithms developed in the inventory-management literature (c.f., Mula et al., 2006; Tarim et al., 2004; and Zipkin 2000). Most stochastic inventory management models focus on determining the optimal ordering and production quantities that minimize the long-term expected cost associated with inventory and unmet demand. However, these stochastic inventory models are inadequate in the sense that they do not consider risks associated with a firm's cash flows for acquiring excessive raw materials when demand is unexpectedly high, and for disposing excessive finished goods (write-offs) when demand is unexpectedly low (as in the case of Cisco's $2 billion write-off in 2001). Researchers have considered multi-echelon inventory theory as a way to analyze SCP under uncertainty (e.g., Porteus 2002; Minner 2003). However, this approach may not be appropriate for dealing with products with short life cycles and it is intractable for complex supply chains with multiple products (Tang 2006a).

Another approach has been to consider uncertainty in the context of *decentralized* supply chains. Starting with the seminal work of Clark and Scarf (1960), researchers have considered, for instance, *contracts* (e.g., Cachon 2003; Martínez-de-Albéniz and Simchi-Levi 2005), *channel coordination* (e.g., Gan, Sethi and Yan 2004, 2005), *incentives* using real options (e.g., Miller and Park 2005; Kleindorfer and Wu 2003; Huchzermeier and Cohen 1996; Kogut and Kulatilaka 1994), *capacity investment* (e.g., Birge 2000; Lederer and Mehta 2005), and *R&D investment* (Huchzermeier and Loch 2001). While these decentralized SCP models are not designed for centralized planning for complex supply chains, they may well have implications for such planning and mathematical modeling.

Stochastic programming can be a useful choice for modeling SCP for the medium term when demand is uncertain. It has been applied to electric power generation, telecommunication network planning, and financial planning; c.f., Sen (2001) for the medium term. While recent advances in computational technology encourage more development in stochastic programming, many advantages of stochastic programming as explained by Birge and Louveaux (1997) have not translated into widespread use in practice except primarily in financial applications like ALM.

Stochastic programming for ALM goes back to the late 1960s and its use has been proposed by, among others, Bradley and Crane (1972), Klaassen (1998), Consigli and Dempster (1998), Dattatreya and Fabozzi (1995), Dert (1999), Sodhi (2005b), Zenios and Ziemba (2005), and Ziemba and Mulvey (1999).

The use of stochastic programming for SCP is showing increasing promise. Dormer et al. (2005) illustrate how Xpress-SP—a stochastic programming suite—can be used to solve certain supply chain management problems. Leung et al. (2005) and Sodhi (2005a) present different stochastic programming models for production planning in a supply chain. However, these models focus on production planning and do not deal with liquidity risk stemming from uncertain demand. As such, we seek to encourage the development of stochastic programming models for managing demand, inventory and liquidity risks.

As further motivation, consider the fulfillment process at the digital video camera division of a Japanese consumer-electronics company (Sodhi, 2005a). The company has several regional offices to deal with regional demand from customers that include electronics retail chains with multiple stores, and to coordinate with the company's headquarters for orders and fulfillment. Every week, each customer provides a weekly "order" of a 26-week rolling horizon at the SKU-level to the designated regional office. Each regional office aggregates its customers' orders by week and then sends the aggregate weekly orders to headquarters. Headquarters allocates supplies to these aggregate orders and ships against the current week's aggregate orders from a central warehouse either to customers' warehouses directly or to regional warehouses for final shipment to customers.

To match demand and supply in this company, the planning problem can be modeled as a multi-period network flow model in which the central warehouse is the "source" node and the customers' and regional offices are the "sink" nodes. Sodhi (2005a) presents a *deterministic* linear programming model as well as a *stochastic* programming extension of this problem that is associated with a single-product, three customers, and a planning horizon of $T = 10$ weeks. The deterministic model has 150 decision variables and 100 constraints while the stochastic model, using 2^{10} ($= 1024$) scenarios, has about 1,500 decision variables and 10,000 constraints. The former solves in a fraction of a second and the latter takes about 10 seconds on the same computer. One benefit of the stochastic programming model is that, even with infinite capacity, the model does not recommend fulfilling all of the customer's orders owing to a high risk of excess inventory. *This underscores our belief that we cannot manage demand and supply risks by simply running deterministic models with a rolling horizon because doing so does not allow us to trade off unmet demand and excessive inventory.*

While stochastic programming is an appropriate modeling choice, the computational requirement is a big challenge. For example, suppose we increase the planning horizon T from 10 weeks to 26 weeks. Then the stochastic program for the single-product example described earlier would have over 100 million decision variables and close to 200 million constraints. Therefore, in order to use stochastic programming as a "practical" modeling choice for SCP under uncertainty, one needs to make modeling choices carefully.

15.3 A Stochastic Program for Supply-Chain Planning

15.3.1 A Stochastic Program for Asset Liability Management

Before we formulate a model for SCP under uncertainty, let us review Asset Liability Management (ALM) and examine a model (A) for ALM with interest rate uncertainty (Sodhi 2005b). Let ζ_t be the interest-rate scenario that ends at time t and the scenarios evolve according to a "binary event tree" over time. There are *six* sets of decision variables, the two operational ones being the amount of security i purchased (x) or sold (y). Each security in the portfolio generates cash in each period under each interest-rate scenario, and this cash can be used to meet the bank's liabilities in that period. Securities not sold are carried over as "inventory," and the total value of all securities in the portfolio at the end of the decision horizon T, discounted to the current time is what is sought to be maximized. These six sets of decision variables are:

x_{i,ζ_t} = Amount purchased of (original) principal of security i
y_{i,ζ_t} = Amount sold of principal of security i
h_{i,ζ_t} = Holdings of principal of security i *after* trades
b_{ζ_t} = Amount borrowed at current short interest rate ρ_{ζ_t}, plus a premium Δ
l_{ζ_t} = Lending amount at the current short interest rate ρ_{ζ_t}
L_{ζ_t} = Liability under scenario ζ_t

The parameters that accompany the above decision variables are:

κ_{i,ζ_t} = Cash generated from the principal and interest per unit of security i
π_{i,ζ_t} = Ex-dividend computed price at time t for principal for each unit of security i (same as market price when $t = 0$)
ρ_{ζ_t} = Single-period interest rate; \$1 at time $t \equiv (1+\rho_{\zeta_t})$ in $t+1$
δ_{ζ_t} = Present value of a cash flow of \$1 at time period t in scenario ζ_t
Δ = Premium paid over the short interest rate for single-period borrowing
T_i = Transaction cost per trade dollar associated with trading of security i
T = Planning horizon
l_{-1} = Single-period lending due in current time period ($t = 0$)
b_{-1} = Single-period borrowing from last period due in current time period
$h_{i,-1}$ = Holding of security i at beginning of current time period

The stochastic program associated with this ALM model can be formulated as problem (A):

$$\text{(A)} \quad \max\ 2^{-T} \sum \zeta_T \delta_{\zeta_T} \left\{ \sum_i \left[\pi_{i,\zeta_T} h_{i,\zeta_T} \right] + l_{\zeta_T} - b_{\zeta_T} \right\}, \qquad \text{subject to:}$$

$$\sum_i \kappa_{i,\zeta_t} h_{i,\zeta_{t-1}} + l_{\zeta_{t-1}}(1+\rho_{\zeta_{t-1}}) + b_{\zeta_t} + \sum_i (1-T_i)\pi_{i,\zeta_t} y_{i,\zeta_t}$$

$$\sum_i (1+T_i)\pi_{i,\zeta_t} x_{i,\zeta_t} - l_{\zeta_t} - b_{\zeta_{t-1}}(1+\rho_{\zeta_{t-1}} + \Delta) = L_{\zeta_t} \quad \forall \zeta_t$$

$$h_{i,\zeta_t} + y_{i,\zeta_t} - x_{i,\zeta_t} - h_{i,\zeta_{t-1}} = 0 \quad \forall i, \forall \zeta_t$$

$$x_{i,\zeta_t}, y_{i,\zeta_t}, h_{i,\zeta_t}, l_{\zeta_t}, b\zeta_t, = 0 \quad \forall i, \forall \zeta_t$$

for all $t = 0, \ldots, T$ (for all three sets of constraints).

There are two sets of constraints besides the non-negativity ones: cash balance and inventory holdings. The first set specifies the net cash that needs to be raised in order to match the liability L_{ζ_t} under scenario ζ_t. Cash comes from: (a) the holding of security i from period $t-1$; (b) the principal and interest from lending in period $t-1$; (c) the amount borrowed in period t; and (d) the sale of securities in period t. Cash is spent on: (a) purchasing securities in period t (along with transaction costs); (b) lending in period t; and (c) the principal plus interest associated with the amount borrowed in period $t-1$. The second set corresponds to the inventory balancing equation of each security i in period t.

15.3.2 A Stochastic Program for Supply Chain Planning

Consider the electronics company example illustrated in Section 15.2 along with consideration of its cash flows. Take an idealized situation with a manufacturer with a single plant and a single warehouse that faces customer demand. The plant purchases inputs from a supplier, converts these inputs into finished goods, and transports all finished goods to the warehouse. The procurement, conversion, and transportation lead times are known; however, future demand for finished goods is uncertain. Our focus is on the planning of materials and cash flows over a planning horizon T (Fig. 15.1). Material flows comprise inputs from the supplier to the plant and finished goods from the plant to the warehouse and then onto the customers. Information flows are orders from the manufacturer to the supplier. Cash flows include the financial transactions between the bank and the manufacturer including the (delayed) payments from the manufacturer to the supplier and from the customers to the manufacturer.

To formulate this SCP model as a stochastic program S, we need to model uncertain demand by specifying various demand scenarios with certain probabilities. (We shall discuss ways to model other types of uncertainties in Section 15.4.1.) Let ζ_t be a demand scenario for period t and ζ_{t-1} its "parent" scenario so that the evolution of demand scenario over time follows a "binary" event tree (c.f., Luenberger, 1998) in which each scenario ζ_t emerges with two "child" scenarios ζ_{t+1} with equal probability. Thus, the model with T periods has 2^T scenarios. Note that a binary event tree for generating demand scenarios is only one way of representing uncertainty

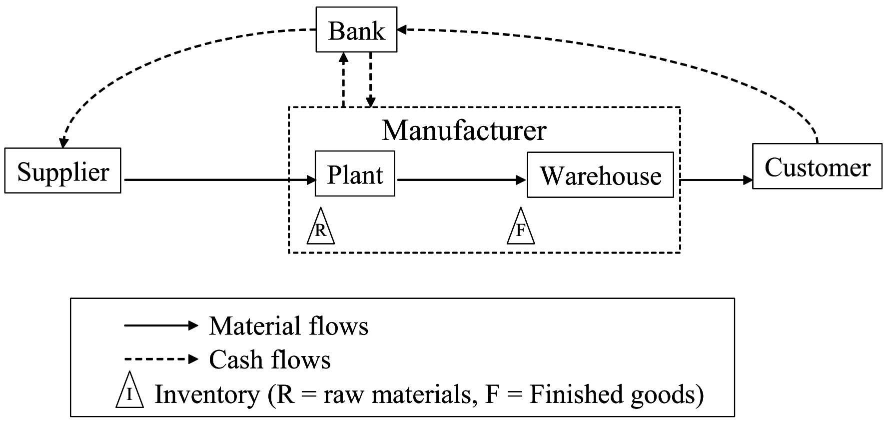

Fig. 15.1 Material and cash flows in a supply chain

with a single source of uncertainty that is motivated by the way uncertain interest rate is modelled in the ALM model. We describe a stochastic process for generating a binary event tree and the demand scenarios such a tree entails in Section 15.3.4.

In the SCP model, there is the planning horizon T as before, and another horizon T_E that exceeds T by at least the sum of the lead times of the two material flows and the lead time for cash settlement on the sell side. The extended horizon enables us to handle the awkward end conditions created by lead times. Our SCP model has *nine* sets of ***decision variables***, the three operational ones being the amount of an input i purchased (x), the amount of a finished product produced (y), and the amount of the product sold (z). These nine set of decision variables are:

$x_{i,t}$ = Amount of input i to be purchased at the beginning of period t (independent of scenario)
$y_{j,t}$ = Amount of product j to be produced at the beginning of period t (independent of scenario)
z_{j,ζ_t} = Amount of product j to be sold in period t
h_{i,ζ_t} = Inventory level of input i at the end of the period t
g_{j,ζ_t} = Inventory level of product j at the end of the period t
b_{ζ_t} = Amount borrowed in period t at interest rate $\rho_t + \Delta$
$b^p_{\zeta_t}$ = Single-period borrowing over a limit B at a higher rate $\rho_t + \gamma\Delta$, where $\gamma > 1$. (Hence, risk aversion to excessive borrowing is captured by the parameter γ.)
l_{ζ_t} = Amount of cash in account during period t earning interest at rate ρ_t
u_{j,ζ_t} = Unmet demand for product j at the end of period t

The first two sets of variables, x and y, represent procurement and production plans, respectively, for the next so many periods. Unlike the other decision variables, these are independent of demand scenario ζ_t because we have to make these plans now for a certain time horizon. Alternatively, we could have modeled these variables as scenario-independent variables up to a certain time horizon and from then on as scenario-dependent. However, there seems little lost by having these variables scenario-independent and much gained by having the number of x and y decision variables grow linearly rather than exponentially with the number of periods. Although procurement and production plans are usually obtained using deterministic modeling in MRP, ERP, or APS technology, the solution produced is different from that obtained by a stochastic model because the latter hedges these decisions for risk that a deterministic model cannot.

Our stochastic programming model has the following **parameters**:

$\kappa_{j,\zeta_t} =$ Selling price for product j(net of any selling costs)
$\pi_{i,\zeta_t} =$ Unit price of input i in period t

Note: Both selling price of product j and purchasing price of input i are assumed to be known functions of demand. Therefore, these two parameters are scenario-dependent (or, we can model these more compactly as node-dependent in relation to the demand event tree).

$\rho_t =$ Single-period interest rate at time t; \$1 at time $t \equiv \$(1+\rho_t)$ at time $t+1$
$\delta_t =$ Present value of a cash flow of \$1 at time period t
$\alpha_{ji} =$ Amount of input i needed to make one unit of product j (i.e., bill of materials)
$\beta =$ Fraction of unmet demand that is backordered for next period, $0 \leq \beta \leq 1$
$D_{j,\zeta_t} =$ Demand for finished product j in scenario ζ_t
$\Delta =$ Premium paid over the short rate for single-period borrowing; strictly positive
$L_s^M =$ Material-flow lead time for supplier to fulfill order (i.e., supply lead time)
$L_p^M =$ Material-flow lead time for manufacturer to convert raw materials to finished goods (i.e., production lead time)
$L_s^C =$ Cash-flow lead time for manufacturer to pay supplier following receipt of supplies (i.e., account payable lead time)
$L_p^C =$ Cash-flow lead time for customer to pay manufacturer following sale of finished goods (i.e., account receivable lead time)
$T =$ Planning horizon (with periods in weeks or months)
$T_E =$ Extended horizon with $T_E \geq T + L_s^M + L_p^M + L_p^C$
$B =$ Borrowing limit
$C_t =$ Joint production capacity across all finished products in period t.

We also need parameters to reflect the inventories at hand or already "in the pipeline" for the coming weeks. These **initial values** are denoted by the following:

$l_{-1} =$ Initial amount of cash in account at the beginning of current time period $(t=0)$
$b_{-1} =$ Amount borrowed in the last period that is due in the current time period
$h_{i,-1} =$ Inventory level of input i at the end of last period
$g_{j,-1} =$ Inventory level of product j at the end of last period

$x_{i,-L_s^M},\dots,x_{i,-1} =$ Outstanding orders of input i (to be received)
$y_{j,-L_p^M},\dots,y_{j,-1} =$ Work in process inventory of product j (to be completed)
$z_{j,-L_p^C},\dots,z_{j,-1} =$ Amounts (in units) of product j in corresponding to accounts receivable.

There are 2^T scenarios for the binary event tree that we have chosen as our way of representing demand uncertainty. Each scenario ζ_t occurs with probability 2^{-T} and the expected present value of the net cash at period T is $2^{-T}\delta_T \sum \zeta_T[l_{\zeta_T} - b_{\zeta_T}]$. Suppose our objective is to maximize this quantity (there may be other objective functions as we discuss later in Section 15.4.4). Then we can formulate this SCP model as the following stochastic program (S):

$$\text{(S)} \qquad \max 2^{-T}\delta_T \sum_{\zeta_T}[l_{\zeta_T} - b_{\zeta_T}], \qquad \text{subject to:} \tag{15.1}$$

$$\begin{aligned}
&\sum_j \kappa_{j,\zeta_{t-L_p^C}} z_{j,\zeta_{t-L_p^C}} + l_{\zeta_{t-1}}(1+\rho_{t-1}) \\
&\quad + b_{\zeta_t} + b^p_{\zeta_t} - \sum_i \pi_{i,\zeta_{t-L_s^C}} x_{i,\zeta_{t-L_s^C}} \\
&-l_{\zeta_t} - b_{\zeta_{t-1}}(1+\rho_{t-1}+\Delta) - b^p_{\zeta_{t-1}}(1+\rho_{t-1}+\gamma\Delta) = 0 \quad \forall \zeta_t
\end{aligned} \tag{15.2}$$

$$h_{i,\zeta_t} + \sum_j y_{j,t}\alpha_{ji} - x_{i,t-L_s^M} - h_{i,\zeta_{t-1}} = 0 \quad \forall i, \forall \zeta_t \tag{15.3}$$

$$g_{j,\zeta_t} + z_{j,\zeta_t} - y_{j,t-L_p^M} - g_{j,\zeta_{t-1}} = 0 \quad \forall j, \forall \zeta_t \tag{15.4}$$

$$z_{j,\zeta_t} + u_{j,\zeta_t} - \beta u_{j,\zeta_{t-1}} = D_{j,\zeta_t} \quad \forall j, \forall \zeta_t \tag{15.5}$$

$$b_{\zeta_t} \le B \quad \forall \zeta_t \tag{15.6}$$

$$\sum_j y_{j,t} \le C_t \quad \forall t \tag{15.7}$$

$$x_{i,t}, y_{j,t}, z_{j,\zeta_t}, h_{i,\zeta_t}, g_{j,\zeta_t}, l_{\zeta_t}, b_{\zeta_t}, b^p_{\zeta_t}, u_{j,\zeta_t} \ge 0 \quad \forall j, \forall i, \forall \zeta_t \tag{15.8}$$

for all $t = 0,\dots,T_E$ for all constraints (15.2)–(15.8).

Note that the objective function discounts the value of cash at time T while the time periods in the constraints go up to the extended horizon T_E. Constraints (15.2) represents the balance of cash flows for each demand scenario as follows similar to the constraints for ALM. Cash flows in from accounts receivable based on the sale of the $z_{j,\zeta_{t-L_p^C}}$ units of product j earlier due in period t; from the principal and interest associated with the cash in account in period $t-1$; and from amounts borrowed at the base and at the higher rate in term t. Cash flows out to the accounts payable of the $x_{i,t-L_s^C}$ units of input i purchased earlier that is due in period t; to the cash in account in period t; and to the principal and interest associated with the amounts borrowed in period $t-1$ at the two different rates. As with the ALM model, constraints (15.3) and (15.4) correspond to inventory balancing at the beginning and at the end of each period in each scenario: in this case we have inputs $i$and product j, respectively instead of securities. The remaining constraints are different from those in the ALM model. Constraint (15.5) ensures that sales do not exceed demand (plus the fraction of backordered demand). Constraint (15.6) specifies the “soft” borrowing limit B

above which it is more expensive to borrow. Constraint 15.7) specifies the joint production capacity constraint.

Notice that stochastic program (S) is feasible and bounded. Feasibility can be established by setting the decision variables x and y to zero for all t. Boundedness can be established by noting that cash is generated only from sales, which are limited by the demand. Moreover, production is limited by the joint production capacity.

15.3.3 Comparison of SCP Model with the ALM Model

The two models (A) and (S) presented in Section 15.3.1 and 15.3.2 are quite similar. The model (A) has interest-rate scenarios and (S) has demand scenarios. Both have "material" flows: (S) has physical units being ordered or produced while (A) has the number of securities being purchased or sold. As already indicated, the constraints (15.2)–(15.4) for balancing cash and inventory in (S) have their counterpart for (A). We can impose a soft limit B on single-period borrowing (i.e., constraint (15.6) for program (A) as well.

However, the backorder constraint (15.5) and the joint-production capacity constraint (15.7) are relevant to SCP only. Moreover, the variables for SCP are slightly different in the sense that the purchase decision vector x_i and the production decision vector y_j in (S) do not depend on the scenarios owing to lead times; however, the buy/sell decision of each security i in (A) is scenario-specific for ALM. Hence, the number of decision variables associated with x and y in program (S) grows only linearly, while the number of decision variables grows exponentially with the number of periods for the ALM model. This makes it easier to solve (S).

A fundamental difference lies in the generation of scenarios. In ALM, the scenarios for interest rate should, as a set, (1) match the yield curve, (2) provide the same period-on-period returns for all securities to prevent arbitrage, and (3) discounted cash flows of securities should match market prices for a chosen set of securities like treasuries. Such a restriction does not apply to (S). However, this difference applies only to the generation of scenarios and not to the models themselves or to solution techniques for these models.

There are operational differences that do affect modeling. For ALM, the volume of buying and selling is unrestricted (at least in the model A) whereas for SCP, selling is limited by the demand. Production capacity or joint-production capacity in (S) is another difference as is conversion from inputs to finished product. Finally, there are lead times for material and cash flows in (S)—such lead times do not have an equivalent in ALM as buy/sell decisions of securities are immediately executed.

Overall, the programs (S) and (A) are similar in terms of types of decision variables, types of constraints, and the coefficient matrix structure of the constraints. Therefore, we can discuss the applicability of modeling and solution choices for ALM to SCP.

15.3.4 Demand-Scenario Generation: An Example

For the SCP model (S), the demand for product j in period t under scenario ζ_t (i.e., D_{j,ζ_t}) can be generated according to a binary event tree (i.e., each scenario ζ_t in period t has two child scenarios in period $t+1$) as follows. Consider modifying an autoregressive model of first order, i.e., AR(1), in the following manner (Sodhi 2005a): Given an unbiased forecast $\mu_j[t]$, the D_{i,ζ_t} follows the stochastic process $D_{j,\zeta_0} - \mu_j[0] = 0$, and

$$D_{j,\zeta_t} - \mu_j[t] = \theta_j(D_{j,\zeta_{t-1}} - \mu_j[t-1]) + \varepsilon_t$$

for $t \geq 1$, where θ_j, $|\theta_j| \leq 1$, is the product-dependent auto-correlation coefficient. Notice that the above process for D_{j,ζ_t} will be reduced to the standard AR(1) process when $\mu_j[t]$ is a constant over time. However, to model certain effects such as season-related effect or product-lifecycle effect, we can make $\mu_j[t]$ depend on time. To be consistent with the findings of Lee, So, and Tang (2000) regarding the consumer-packaged goods and the electronics industries, we can assume $\theta_j > 0$. The error ε_t is i.i.d. across time periods and equals ν or $-\nu$ with equal probability for some positive scalarν; hence, $E(\varepsilon_t) = 0$ and $\mathrm{Var}(\varepsilon_t) = \nu^2/4$ for all t. We shall discuss other mechanisms in the literature for generating demand scenarios in Section 15.4.1.

15.3.5 Risk Measures Consideration

Besides the optimal solution to program (S), the decision maker should evaluate the goodness of the solution by considering various risk measures. We propose three risk measures pertaining to SCP under uncertainty, adapting the Value-at-Risk (VaR) measures from financial risk management. The first risk measure is called Demand-at-Risk (DaR)—a measure of unmet demand at any given period t. For example, by considering the distribution of the unmet demand of product j in period t (i.e., u_{j,ζ_t}), let $\mathrm{DaR}_t(p)$ be the "critical value" that satisfies: $\mathrm{Prob}\{u_{j,\zeta_t} > \mathrm{DaR}_t(p)\} = p$; i.e., there is a probability p that the unmet consumer demand u_{j,ζ_t} exceed $\mathrm{DaR}_t(p)$ in period t. Using a similar setup, we can create a measure called Inventory-at-Risk (IaR) that measures the inventory level g_{j,ζ_t} exceeding a "certain threshold" in any given period (i.e., $\mathrm{Prob}\{g_{j,\zeta_t} > \mathrm{IaR}_t(p)\} = p$.) Finally, we can define a measure called Borrowing-at-Risk (BaR) that measures $b^p_{\zeta_t}$ (the extent of excessive borrowing above the limit B); i.e., $\mathrm{Prob}\{b^p_{\zeta_t} > \mathrm{BaR}_t(p)\} = p$.

These three risk measures can be computed from the output of our stochastic program (S) because we know the probability distributions unmet demand, excess inventory and liquidity for any time period corresponding to the optimal solution. Note that we can reduce the DaR at the expense of increasing IaR or BaR or vice versa. So, one needs to balance these risk measures. However, all three risk measures can be reduced if uncertainty or the lead times were to be reduced.

15.4 Modeling Choices

To model SCP, we look for guidance in the ALM literature for modeling choices including: representation of uncertainty, time periods and length of the decision horizon, the objective function, and constraints. Modeling choices are important not only to convince managers of the efficacy of the model but also for solution considerations.

15.4.1 Representation of Uncertainty

The SCP model (S) relies on an auxiliary demand model to provide demand scenarios that evolve in the form of a binary event tree with either of any pair of child nodes being equally probable. Only demand uncertainty is taken into account similar to the ALM model (A) where only interest-rate uncertainty is considered. However, our model can incorporate other uncertain factors if we can handle a (much) larger number of scenarios computationally. Scenarios can be used to capture one or more sources of uncertainty. We have considered demand scenarios with only one underlying source of uncertainty in (S) but we could alternatively consider correlated demand of multiple products, uncertain production yields, unreliable delivery times and/or uncertain supply, at least in principle, using a (very) large number of scenarios.

The binary event tree, as described in Section 15.3.4, is a simple way for generating demand scenarios. However, in the ALM model, the binary event tree for interest rate uncertainty is built using the yield curve and the prices of treasury options by assuming a "risk-neutral" world with "risk-neutral" probabilities as opposed to objective or subjective ones. All 2^T scenarios are included in the ALM model so that the number of periods T cannot be very large in practice (c.f., Black et al., 1990; Ho and Lee, 1986; and Heath et al., 1990).

To model demand uncertainty, we need to first determine the underlying factors that affect product demand (e.g., market conditions, product novelty, etc.) and then use an appropriate stochastic process to generate scenarios for the demand of multiple products for multiple time periods. To do so, we can adopt some of the research development in ALM. Specifically, for interest-rate scenario generation using multiple factors, Mulvey (1996) uses the two-factor model of Brennan and Schwartz to generate a sample of interest-rate scenarios. Hull (2009) discusses different interest-rate models using both one- and two-factor. Bradley and Crane's (1972) generate interest-rate scenarios by using an n-ary tree with any number of underlying factors.

While our current model, as depicted in program (S), assumes that the demands for different products in each time period are independent of each other, we may need to construct more realistic scenarios in which these demands are correlated. ALM researchers have created event trees that use the covariance between across asset classes such as stocks, bonds and real estate although not with individual assets. Kouwenberg (2001) and Gaivorinski and de Lange (2000) create event trees for the ALM model with selected asset *classes* (stocks, bonds, real estate, etc.). They match the means and the covariance matrix of the joint continuous probabilities obtained from history (or future expectations) of returns to those obtainable from the event tree as recommended by Hoyland and Wallace (2001) by solving a nonlinear problem that penalizes differences in the two sets of means and covariance matrices. Consigli and Dempster (1998) use UK data from 1924–1991 comprising third-order autoregressive equations to generate annuals returns (and hence probability distributions) for similar asset classes: ordinary shares, fixed-interest irredeemable bonds, bank deposits, index-linked securities, and real estate. Dert (1999) follows a similar approach for Dutch pension funds with a "vector" autoregressive model for wage inflation, price inflation, cash, stocks, property, bonds, and GNP. Pflug et al. (2000) use principal component analysis on historical data to extract factors of uncertainty that drive asset returns and interest rates and use these to create scenario trees by matching statistical properties. Mulvey and Shetty (2004) generate ALM scenarios with multiple economic factors.

15.4.2 Decision variables

Our SCP model as formulated as program (S) has scenario-specific decision variables z, h, g, l, and b that correspond to the current scenario ζ_0 at $t = 0$ and to the future scenarios ζ_t, $0 < t \leq T_E$. However, in contrast to the ALM model, the planning decision variables x and y can be modeled as scenario-independent as explained earlier. If there are specific *time-fence* restrictions on revising the planned values in subsequent periods, these restrictions can be easily incorporated into our model (S) using initial values for x and y up to the time fence.

15.4.3 Time Periods and the Decision Horizon

Our model is motivated by a SCP problem arising from such industries as consumer electronics or consumer-packaged goods where it makes sense to consider time periods in weeks. If the time periods are "too long", we will have fewer time periods but will lose the granularity needed for operations. If the time periods are "too short", then the number of time periods and consequently the number of scenarios, constraints, and variables will increase, impacting solution tractability. For instance, when $T = 6$ periods, the number of scenarios is only 128 but when $T = 26$ periods, the number of scenarios exceeds 67 million. Therefore, we need to consider the length of the time-interval carefully.

In the ALM literature, some researchers have proposed the use of successive time periods of increasing length to reduce the number of periods T (e.g., Cariño et al., 1994; 1998ab). The idea of increasing length of time periods successively is supported by commercial supply chain software such as that from SAP or i2 as is the case with commercial ALM software, e.g., that from PeopleSoft, now part of Oracle. For a period of six months in our SCP model, we could have $T = 6$ with two periods of one week, one period of a fortnight, two periods of one month each, and one period of one quarter although doing so makes things trickier in a rolling horizon model with lead times. Therefore we may consider 20–30 equal periods of weeks in the electronics industry and months in the petrochemical industry.

15.4.4 Objective function

The objective of our stochastic program (S) is to maximize the expected present value of the net cash (cash minus borrowed amount) in period T. We can easily include the value of the assets in period T (as in the ALM model) by adding the asset value of the inputs and finished goods inventory at liquidation prices as well as the present value of accounts receivable.

Besides maximizing expected present value, *minimizing cost* has been used in many deterministic tactical SCP models with penalties for not meeting demand and costs for holding inventory. When the purchasing price for inputs and the selling price for finished goods are *not* available (unlike the assumption for (S), we can change the objective function in our model (S) from maximizing the expected present value of net cash to minimizing the total penalty associated with over-stocking and under-stocking. This situation occurs in the ALM model as well. While it is common to maximize the expected present value of net cash position, Dert (1999) has considered a different ALM model in which the goal is to minimize the expected costs of funding a defined-benefit pension fund.

Minimizing the probability of an adverse event—running out of cash, not meeting demand, or having excessive inventory—has been captured in our risk measures such as BaR, DaR, and IaR as defined in the previous section. Similar risk measures have been proposed for ALM and other financial models as a way to control risk. For tactical supply chain management this could translate into the likelihood of more than a certain quantity of backorders or surplus respectively (refer to chance constraints below as well).

For model (S), the linear objective function is justified when the decision maker is actually risk-neutral or when his risk-aversion is captured implicitly by penalizing any borrowings above a threshold value B (or a series of such thresholds with increasingly higher penalties) from the expected present value. Modeling risk aversion explicitly is challenging. One objective function is to *maximize risk-averse utility* $u(\cdot)$ as a function of the *subjective* probability distribution $f(\cdot)$ of "wealth" over terminal scenarios ζ_T (i.e., w_{ζ_T}). In this case, the objective function becomes: $\max u(f(w_{\zeta_T}))$. A simplified version is the expected utility function $\sum_{\zeta_T} \phi_{\zeta_T} u_1(w_{\zeta_T})$ where ϕ_{ζ_T} is the probability of the scenario (Klaassen, 1997). Any utility-based objective entails nonlinear functions, e.g., expected log value of excess horizon return (Worzel et al., 1994). Kallberg and Ziemba (1983) discuss the relevance of different forms of concave utility functions and show that they all give similar results if the average risk aversion is the same. Others attempted to avoid nonlinearity by adding constraints to model risk-aversion (Bradley and Crane 1972; Kusy and Ziemba 1986; Cariño, et al., 1994, 1998a, 1998b). Such constraints—soft or hard—can be used with a linear objective function (e.g., Kusy and Ziemba (1986).

15.4.5 Constraints

One modeling choice consideration is how to use (S) to generate output that can be used as input or guidance for a more detailed but deterministic model. Thus, we can argue for a "simple" model (S) that suppresses many practical constraints but incorporating uncertainty, and then accompany this simple model by examining a detailed model with all the constraints but without incorporating uncertainty.

For stochastic programming models, the so-called *non-anticipatory constraints* arise from the requirement that for scenarios sharing past events up to a certain time, the decisions must be the same at this time (e.g., Birge and Louveaux, 1997). In (S), we have used scenarios in the form of a scenario tree so the non-anticipatory constraints are implicitly embedded in the event tree.

In the development of the ALM model, some researchers (e.g., Dert, 1999; and Drijver et al., 2000) use chance constraints to limit the probability of not meeting the liabilities, i.e., under-funding the pension fund. For SCP, not meeting the liabilities is analogous to not meeting demand. However, according to Kusy and Ziemba (1986), such ALM models with chance constraints tend to have theoretical problems in multi-period situations. Therefore, one needs to be cautious about imposing additional chance constraints in our SCP model.

15.5 Solution-Technique Choices

Our SCP model (S) is simple to understand. However, solving it directly is computationally challenging because it has more than 100 million constraints and a similar number of decision variables for a single-product SCP problem with a 26-period planning horizon. Still, for a single product, the distribution-only model (i.e., no production) solves in only about 5 seconds for a 10-period problem on a 1.2 GHz Pentium laptop with 512 Meg memory using XPRESS-MP$^{\text{TM}}$ software (Sodhi 2005a). This observation suggests that there is hope to solve such models directly, possibly taking into account its special network structure and using solution methods as proposed, for instance, by Mulvey and Vladimirou (1992). In addition to seeking direct methods for solving (S), we should consider other solution techniques including decomposition, sampling of scenarios, aggregation, and combined simulation/optimization methods that have been proven to be effective for solving program (A) for the ALM model. Because (S) has a coefficient matrix similar to that of program (A) for the ALM model, the solution techniques for (A) may be effective for solving (S).

15.5.1 Decomposition

When the number of inputs and/or the number of finished goods are large, our stochastic program (S) could become intractable. However, one can always decompose the joint-production capacity constraint (7) using Benders decomposition by finished product (c.f., Benders 1962). This decomposition approach will enable us to decompose problem (S) into sub-problems, each of which corresponds to a single finished product j. As noted in Sodhi (2005a), this single-product sub-problem can be solved efficiently. In this case, we could think of using this heuristic decomposition approach as a practical way to manage demand risk in SCP. When solving the stochastic program (A) for the ALM model, Benders decomposition has generated some promising results: Gondzio and Kouwenberg (2001) use Benders decomposition and an efficient model generator to solve problems arising from a Dutch pension funds application with $T = 6$ and up to 13^6 (≥ 4.8 million) scenarios on a parallel computer with 16 processors. Still, to temper our optimism, Mulvey and Shetty (2004) describe challenges in solving for even a modest number of scenarios (4096) on a 128-processor machine; they describe the use of interior-point and parallel Cholesky methods as well as new ways to reduce the number of floating point operations. Dert (1999) refers to an iterative heuristic for a chance-constrained ALM model that tackles only one or two time periods per iteration. Thus, despite the merits of decomposition, the large number of scenarios means that researchers have to approximate uncertainty in some way for optimal solution.

Besides using Benders decomposition for solving (S), we should consider applying some of the following decomposition schemes that have been proven to be effective for solving program (A). For example, Bradley and Crane (1972) obtain sub-problems using decomposition that can be solved efficiently. Kusy and Ziemba (1986) use the algorithm developed by Wets (1983) for a stochastic linear program (LP) with fixed recourse, restricting uncertainty to that of deposit flows. Birge (1982) provides a solution method to tackle the large size of multistage stochastic LP's in general. Building upon the work of Kusy and Ziemba (1986) and that of Birge (1982), Cariño et al. (1994, 1998ab) use Bender's decomposition to solve problems that have up to 6 periods, 7 asset classes, and 256 scenarios for a Japanese insurance company. Mulvey and Vladimirou (1992) adopt a generalized network structure for a multi-period asset allocation problem with different classes of bonds, equities, and real estate. They solve problems up to 8 periods, 15 asset classes, and 100 scenarios using the progressive hedging algorithm developed by Rockafellar and Wets (1991).

There are other ways to decompose the problem especially for parallel computation. One approach is to relax the afore-mentioned non-anticipatory constraints so that we can treat the scenarios independently from each other and thus decompose the stochastic model. All the scenarios can be solved in parallel if a parallel computer were available. This approach enables one to exploit parallel computing for solving multi-stage stochastic problems as in program (S). Nielsen and Zenios (1993) report significant savings in computational effort by using parallel algorithms for solving stochastic programs developed by others including Rockafellar and Wets (1991) and Mulvey and Ruszcynski (1995).

15.5.2 Aggregation

Recall from Section 15.4 that the number of demand scenarios generated from a binary event tree grows exponentially in the number of periods. As a way to reduce the number of scenarios, aggregation appears to be a reasonable way to approximate the uncertainty itself. In SCP models, it is common to aggregate finished goods with similar demand patterns, production lead times, or product costs, etc. as an "aggregated product" when dealing with thousands of finished goods. Hence, we can consider using aggregation as a way to reduce the size of our stochastic program (S). Besides aggregating finished goods, one may consider aggregating customers with similar demand patterns or suppliers with similar supply costs or lead times. Two innovative aggregation schemes for reducing the size of stochastic program (A) for the ALM model have been proposed and analyzed by Klaassen (1998). These two aggregation schemes call for aggregating different scenarios into an "aggregated scenario" and for aggregating different time periods into an "aggregated time period." Owing to the similarity between stochastic programs (S) and (A), it is likely these aggregation schemes are applicable to our SCP model.

Birge (1985) obtains a single aggregated scenario by aggregating the rows and columns of the LP coefficient matrix for the stochastic model using weights corresponding to the probability distribution of the random variables to determine error bounds for the expected value problem. His aggregation scheme and the corresponding results apply to the case when only the right hand side of the constraints is stochastic. This aggregation may be applicable to our SCP problem when only the demand D_{j,ζ_t} of each product j on the right hand side of constraint (5) is stochastic. For the case when all coefficients in the constraint matrix are stochastic as well, one may consider using the aggregation technique developed by Wright (1994) that generalizes Birge's (1985) work. Moreover, our stochastic program (S) for SCP under demand uncertainty is essentially a stochastic LP that has a finite number of (discrete) scenarios and is therefore an ordinary LP. Therefore, we can aggregate scenarios by aggregating columns first (Zipkin, 1980a) and then aggregating rows later (Zipkin, 1980b). See Sodhi and Tang (2011) for such an aggregation to support a sales-and-operations planning process.

15.5.3 Sampling Scenarios

The restrictions that apply to the set of scenarios for ALM do not apply to the set of scenarios for SCP as we explained earlier. Therefore, we do not need to use all scenarios from a binary event tree for our SCP model. For instance, we could use simulation to generate a sample of scenarios using one or more factors of uncertainty. However, we could get poor quality solutions depending on how the demand scenarios were generated (Kouwenberg, 2001). Moreover, the solution in each run can be quite different from the previous one with the same input because the number of randomly-generated scenarios may be only a tiny percentage of a large population of very diverse scenarios. This makes comparison of different runs with different inputs (e.g., forecasts) difficult. To reduce the variance of scenarios generated in different runs, Mulvey and Thorlacius (1999) among others use antithetic sampling scenarios for different interest rates, exchange rates, stock returns, and inflation rates.

Besides sampling scenarios, one can combine simulation and optimization to get a better representation of uncertainty (e.g., Seshadri et al., 1998; Sen, 2001).

15.6 Conclusion

We have presented a stochastic programming formulation for a simple SCP model that involves the purchase of inputs from a supplier, their conversion to finished goods at a single plant, and the eventual stocking and selling of these goods from a single warehouse facing uncertain demand. We have also compared the models in these two domains, drawing out similarities and differences. We have argued that the ALM literature leads the SCP literature in its use of stochastic programming and is therefore a useful guide for researchers and practitioners wishing to develop stochastic programming models for SCP.

One conclusion from this survey is that no matter how fast computers become or how much solver technology improves over time, stochastic programming will remain as much of an art as it is a science. As such, modeling choices and solution choices have to be made carefully, matching these choices to the objectives and being able to compromise on the objectives. We hope our survey will help researchers and practitioners to do so.

There is also a cautionary lesson we need to draw from the ALM literature. In practice, there is much emphasis on Monte Carlo simulation to “stress test” an existing or a proposed ALM portfolio. Regardless of the sophistication of scenarios, the simulation-only approach is simplistic compared to dynamic stochastic optimization. Still, it may satisfy managers because simulation is more easily understood by them than stochastic optimization. It is quite conceivable then that for tactical SCP practice too, popular risk management software in the future might use Monte Carlo simulation with the goal being a report on the projected risk performance of a particular procurement and production plan under a variety of what-if scenarios. These scenarios could be computer-generated or hand-crafted to consider demand, inventory, supplier lead-time, and capacity variations—see Nagali et al. (2008) for a procurement risk application at Hewlett-Packard.

In light of such simulation-based approaches, selling the superiority of optimization-based risk-management approaches to managers can be a challenge. One solution may be simple stochastic programming models that can work hand-in-hand with deterministic models in extended ERP/APS systems (e.g., SAP system with Advanced Planning and Optimization or i2's Supply Chain Planner) to provide risk-adjusted plans—see Sodhi and Tang (2011) for an example of this approach.