5-2018

A multi-stage stochastic programming for lot-sizing and scheduling under demand uncertainty

Zhengyang Hu
Iowa State University, zhengya@iastate.edu

Guiping Hu
Iowa State University, gphu@iastate.edu

A multi-stage stochastic programming for lot-sizing and scheduling under demand uncertainty

Abstract

A stochastic lot-sizing and scheduling problem with demand uncertainty is studied in this paper. Lot-sizing determines the batch size for each product and scheduling decides the sequence of production. A multi-stage stochastic programming model is developed to minimize overall system costs including production cost, setup cost, inventory cost and backlog cost. We aim to find the optimal production sequence and resource allocation decisions. Demand uncertainty is represented by scenario trees using moment matching technique. Scenario reduction is used to select scenarios with the best representation of original set. A case study based on a manufacturing company has been conducted to illustrate and verify the model. We compared the two-stage stochastic programming model to the multi-stage stochastic programming model. The major motivation to adopt multi-stage stochastic programming models is that it extends the two-stage stochastic programming models by allowing revised decision at each period based on the previous realizations of uncertainty as well as decisions. Stability test and weak out-of-sample test are applied to find an appropriate scenario sample size. By using the multi-stage stochastic programming model, we improved the quality of solution by 10–13%.

Keywords

Multi-stage stochastic programming, Lot-sizing and scheduling, Demand uncertainty, Automotive industry

Disciplines

Industrial Engineering | Manufacturing | Operational Research | Systems Engineering

Comments

This is a manuscript of an article published as Hu, Zhengyang, and Guiping Hu. "A Multi-stage Stochastic Programming for Lot-sizing and Scheduling under Demand Uncertainty." *Computers & Industrial Engineering* 119 (2018): 157-166. DOI: 10.1016/j.cie.2018.03.033. Posted with permission.

A Multi-stage Stochastic Programming for Lot-sizing and Scheduling under Demand Uncertainty

Abstract

A stochastic lot-sizing and scheduling problem with demand uncertainty is studied in this paper. Lot-sizing determines the batch size for each product and scheduling decides the sequence of production. A multi-stage stochastic programming model is developed to minimize overall system costs including production cost, setup cost, inventory cost and backlog cost. We aim to find the optimal production sequence and resource allocation decisions. Demand uncertainty is represented by scenario trees using moment matching technique. Scenario reduction is used to select scenarios with the best representation of original set. A case study based on a manufacturing company has been conducted to illustrate and verify the model. We compared the two-stage stochastic programming model to the multi-stage stochastic programming model. The major motivation to adopt multi-stage stochastic programming models is that it extends the two-stage stochastic programming models by allowing revised decision at each period based on the previous realizations of uncertainty as well as decisions. Stability test and weak out-of-sample test are applied to find an appropriate scenario sample size. By using the multi-stage stochastic programming model, we improved the quality of solution by 10% - 13%.

Keywords: Multi-stage stochastic programming, Lot-sizing and scheduling, Demand uncertainty, Automotive industry

1 Introduction

Production planning aims to determine the best allocation of production resources to meet demand over a time period with a limited amount of production capacity. Based on the length, production planning decisions can be categorized into three different terms: long-term, medium-term and short-term. Facility location design and resource allocation are considered as long-term decision making problems. Medium-term planning considers production quantity on a monthly basis, and short-term planning involves making decisions such as day-to-day schedule of activities and job sequencing. In the classical hierarchical decision-making environment, lot-sizing and scheduling decisions take place in the medium-term planning levels that usually span about half an year [1, 2]. Over the entire production horizon, the market and manufacturing configurations might change [3]. Therefore, considering uncertainty and designing a robust production plan are crucial in the lot-sizing and scheduling problem.

The major motivations for this paper can be summarized as follows: First, lot-sizing and scheduling problems have been widely applied in industry. Gupta and Magnusson studied a lot-sizing and scheduling problem confronted by a large manufacturing company that produces sandpaper rolls of different grades or roughness [4]. Bitran and Gilbert reviewed the lot-sizing and scheduling problem using a chemical application. In the field of chemicals, setup cost takes place when it is necessary to scrub out a machine between the production of two products that come from different families [5]. Silva and Magalhaes focused on a lot-sizing and scheduling problem in a company that produces acrylic fiber for textile industry. The problem arises because a changeover occurs between two lots of products due to tool wear [6]. Second, little attention has been paid to stochastic lot-sizing and scheduling problem, especially, multi-stage stochastic lot-sizing and scheduling problem. The Lot-sizing and scheduling problem is an extension of lot-sizing problem which considers production sequence. Harris introduced a single-item lot-sizing model with deterministic static demand. The goal is to minimize overall costs include ordering and inventory [7]. Brahimi et al. reviewed both uncapacitated and capacitated single item lot-sizing problem. Different mathematical formulations and extensions of real world applications are studied [8]. Karimi et al. focused on the lot-sizing problem with multiple items. The capacitated lot sizing problem (CLSP) is mentioned [1]. The problem we focus in this paper is called the capacitated lot-sizing and scheduling problem with sequence dependent setups (CLSD). It is a variation of the CLSP which incorporates dependent setups. Kaczmarczyk considered a lot-sizing and scheduling problem that allows only one setup in each time period. Their formulation includes multi-product and identical parallel machines [9]. Kimms proposed a multi-level lot-sizing and scheduling problem with dynamic demand [10]. Multi-level production means the final product in one stage can be used as raw material in the next stage. However, these papers considered only the deterministic lot-sizing and scheduling problems which may not reflect the reality. This point serves as a major motivation for this research.

Production plan can be highly affected by the various uncertainties such as yield, demand, defective rate, etc. Alem et al. used the lot-sizing and scheduling problem as an application to compare the performances and results of stochastic approach with robust optimization approach. The advantage of each approach was assessed via a Monte Carlo simulation procedure [11]. Rahdar et al. proposed a two-stage trilevel optimization model with a rolling horizon. Demand and lead time uncertainty are studied [12]. Hu and Hu proposed a two-stage stochastic programming approach for the lot-sizing and scheduling problem under demand uncertainty. They proved that the stochastic model outperforms the deterministic model and considering uncertainty is important [13]. Ramaraj et al. studied multiple uncertain parameters using a two-stage stochastic programming model [14]. However, the main drawback of the two-stage stochastic programming technique is it does not take into account the sequential decision making due to the multiple periods in the planning horizon. That is, all of the resource decisions have to be done by the beginning of the second period and no makeup/corrective decision is allowed when new information is revealed [15]. Unlike the two-stage model, the multi-stage stochastic programming model explicitly addresses and incorporates the sequential relationship of the decisions over the multiple periods in the planning horizon [16]. The trade-off is that the computational complexity of the multi-stage stochastic programming model is much higher compared to the two-stage stochastic programming model. Therefore, the problem size that can be solved is limited. Huang and Ahmed compared the two-stage model with multi-stage model and used heuristics to derive the bound for the value of multi-stage stochastic programming

(VMS). The results show that even a feasible solution for the multi-stage model can be much better than the optimal solution for the two-stage model [17].

The major contributions of this study can be summarized as follows: Firstly, we proposed a novel multi-stage stochastic programming model to deal with demand uncertainties. We demonstrated that for a multi-period problem, it is more suitable to use a multi-stage stochastic model. Secondly, stability test is used to identify the best scenario size, which is a significant improvement from the existing literature. Thirdly, we quantitatively measured the improvement of results using multi-stage stochastic programming model. Finally, we provided guidelines to choose the most suitable approach for decision makers based on the results and computational performance.

The remainder of this paper is organized as follows: problem statement and model formulations for both deterministic model and multi-stage stochastic programming model are presented in section 2. The numerical results and model comparisons are reported in section 3. Finally, conclusions, limitations, and future works are discussed in section 4.

2 Model formulation

The deterministic model and multi-stage stochastic programming model are introduced in this section. In a lot-sizing and scheduling problem, each time slot typically represents a week or a month while the overall production horizon is usually no longer than half an year [18, 19]. We aim to find the best production decisions such that the overall cost is minimized. The deterministic model is presented followed by the multi-stage stochastic programming model in which demand uncertainty is considered. There are two types of decisions in the multi-stage model: regular time production decisions and recourse decisions. The regular time production decisions need to be determined at the beginning of each time slot while the recourse decisions include overtime production, inventory, and backlogs are made after the realization of uncertainty in the current stage.

2.1 Problem statement

The problem we address in this paper can be described as follows: manufacturers acquire raw material from up-stream suppliers and produce final products for downstream plants or customers. Orders can be placed at the beginning of each month. According to resource availability, decision makers need to design a good production plan so that the costs can be minimized. Two different resource capacities are: time capacity on the machine and production quantity limitation. Unmet demand can be fulfill later since backlog is allowed. Decision variables include regular time production, overtime production, production sequence, inventory and backorder. The regular time production is limited by both the time capacity on the machine and resource availability. Overtime production is proportional to the regular time production. Production sequence is really critical because setup is sequence dependent and can be carried over to the following period. In other words, different production sequences will result in different resource requirements. Inventory and backorder can then be evaluated.

In the deterministic model, parameters are fixed and known. In the stochastic model, demand is uncertain and represented by scenarios. In the stochastic model, regular time production and production sequence need to be determined in the presence of uncertainty while overtime production decision are made after uncertainty is realized. Scenario sample size analysis and weak out-of-sample stability test aim to identify a good scenario sample size. The analysis of the two-stage stochastic programming model demonstrates the importance of considering uncertainty. The comparisons between two-stage and multi-stage stochastic programming models include computation time and objective value.

2.2 Mathematical notations

The mathematical notations for the deterministic model are included in Table 1. These parameters are fixed and known in the deterministic model. The parameters and variables in the stochastic model are scenario-based and the means of those parameters are same as the deterministic model.

2.3 Model assumptions

The assumptions are listed as follows:

Table 1: Notations for the deterministic model [13]

Subscripts		
i	$1, 2 \cdots N$	Material index
j	$1, 2 \cdots N$	Material index
t	$1, 2 \cdots T + 1$	Time period index
Parameters		
$d_{i,t}$	Demand of material i at time t	
h_i	Holding cost of each material i for one time period	
b_i	backorder cost of each material i for one time period	
cap_t	Time capacity on the machine at time t	
p_i	Manufacturing time of each material i	
p_i^r	Regular time manufacturing cost of each material i	
p_i^o	Overtime manufacturing cost of each material i	
$q_{i,t}$	The maximum regular time batch size of product i at time t	
$sc_{i,j}$	Cost when there is a setup changeover from material i to material j	
$st_{i,j}$	Setup time from material i to material j	
α	Ratio of regular manufacturing quantity and overtime manufacturing quantity	
N	Number of material families	
Decision Variables		
$I_{i,t}$	Inventory quantity of material i at the end of time t	
$B_{i,t}$	Backorder quantity of material i at the end of time t	
$X_{i,t}$	Regular time production quantity of material i during time slot t	
$O_{i,t}$	Overtime production quantity of material i during time slot t	
$Y_{i,j,t}$	Binary variable takes value 1 if there is a setup changeover from material i to material j during time slot t	
$Z_{i,t}$	Binary variable takes value 1 if setup of material i carried over from previous time slot to time slot t	
$V_{i,t}$	Sequence of production in time period t. It takes value from 1 to N	

- Inventory and backlog are allowed which indicate demand does not need to be fulfilled all the time. The initial values of inventory and backlog are assumed to be zero.

- Demand is time independent, so the realization of demand in current stage does not depend on the previous realization.

- The uncertain demand is realized at the beginning of each period, inventory and backorder levels will be measured at the end of each production period.

- The regular time production and overtime production are resources limited. The former has time and batch size capacities while the latter only has batch size limitation.

- The regular time production and overtime production share the same setup. Since the actual demand is realized after production started, the overtime production serves as the recourse for the baseline production.

- A setup is required between products from different families. In addition, a setup can be carried over between two consecutive production periods. Therefore, the last setup in one period will be the first default setup in the following period.

2.4 Deterministic model

The deterministic model aims to minimize the overall system costs, including regular time production cost, overtime production cost, setup cost, inventory cost, and backlog cost. The first and second terms in the objective function are the regular time production cost and overall setup changeover cost, respectively. It should be noted that there is no setup between products from the same family. The third term is the overtime production cost. The last two terms are the overall inventory holding cost and the backlog cost, respectively. There are three possible cases for the inventory and the backlog costs. First, they all equal to zero meaning current demand is met and no extra product is manufactured. Second, inventory is positive and backlog is zero indicating current demand is met and extra products are manufactured for future demand. Third, inventory is zero and backlog is positive showing that the production capacity is not sufficient to satisfy the demand requirement. The last production time is not included in this model because we add it as a dummy period.

$$\min \zeta = \sum_{i=1}^{I} \sum_{t=1}^{T} p_i^r * X_{i,t} + \sum_{i=1}^{I} \sum_{i \neq j} \sum_{j=1}^{J} \sum_{t=1}^{T} sc_{i,j} * Y_{i,j,t}$$
$$+ \sum_{i=1}^{I} \sum_{t=1}^{T} p_i^o * O_{i,t} + \sum_{i=1}^{I} \sum_{t=1}^{T} h_i * I_{i,t} + \sum_{i=1}^{I} \sum_{t=1}^{T} b_i * B_{i,t} \tag{1}$$

2.4.1 Constraints of deterministic model

Constraints (2) and (3) are product flow conservation constraints. The total amount of production plus inventory from the previous time period equal to the total demand plus the current inventory. The inventory in these two constraints can be either the extra inventory or the backlog demand that can be fulfilled later. There is no inventory coming into the first time period as we assume that initial inventory and backlog are both zero.

$$X_{i,t} + O_{i,t} = d_{i,t} + I_{i,t} - B_{i,t} \qquad \forall i, t = 1 \tag{2}$$

$$I_{i,t-1} - B_{i,t-1} + X_{i,t} + O_{i,t} = d_{i,t} + I_{i,t} - B_{i,t} \qquad \forall i, t = 2 \cdots T + 1 \tag{3}$$

Constraint (4) restricts that the regular time production quantity will not exceed the maximum regular time production quantity which is $q_{i,t}$. Recall that only one setup is allowed in each product family indicating $Z_{i,t} + \sum_{j \neq i}^{J} Y_{j,i,t} \leq 1$. If both terms are zero, then product i cannot be manufactured in time period t. If $Z_{i,t} = 1$ and $\sum_{j \neq i}^{J} Y_{j,i,t} = 0$, then material i will be the first product on the assembly line. If $Z_{i,t} = 0$ and $\sum_{j \neq i}^{J} Y_{j,i,t} = 1$, then material i will be manufactured after material j.

$$X_{i,t} \leq q_{i,t} * (Z_{i,t} + \sum_{j \neq i}^{J} Y_{j,i,t}) \qquad \forall i, t \tag{4}$$

Total machine time capacity, denoted by cap_t, is the maximum regular time resource on the machine. Constraint (5) ensures that the total time for the regular production and setup changeover time cannot go beyond the total machine time capacity. Constraint (6) sets a capacity limit on the overtime production quantity. Typically, $\alpha * X_{i,t}$ puts a production quantity capacity on the overtime production [20].

$$\sum_{i=1}^{I} p_i * X_{i,t} + \sum_{i=1}^{I} \sum_{i \neq j \; j=1}^{J} st_{i,j} * Y_{i,j,t} \leq cap_t \qquad \forall t \tag{5}$$

$$O_{i,t} \leq \alpha * X_{i,t} \qquad \forall i, t \tag{6}$$

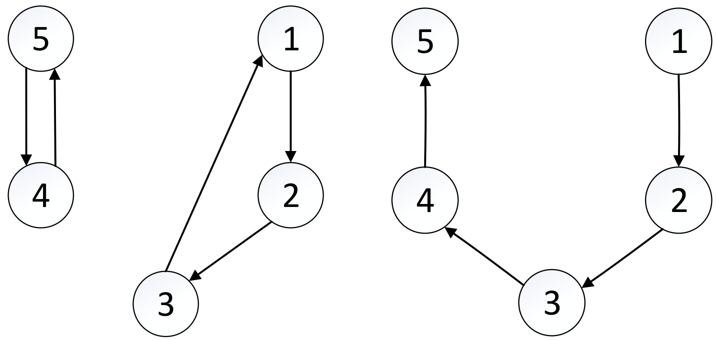
Constraint (7) states at the beginning of each time period, a setup is carried over from the previous time period. Constraint (8) states that the setup flow going into material i equals to the one coming out of it. One easy example will be producing the same product during the entire production horizon meaning $Z_{i,t} = Z_{i,t+1} = 1$ and all of the Y variables are zero because there is no setup changeover.

$$\sum_{i=1}^{I} Z_{i,t} = 1 \qquad \forall t \tag{7}$$

$$Z_{i,t} + \sum_{j \neq i}^{J} Y_{j,i,t} = Z_{i,t+1} + \sum_{j \neq i}^{J} Y_{i,j,t} \qquad \forall i, t = 1 \cdots T \tag{8}$$

Constraint (9) requires that no production activity is allowed in the last dummy period except that the setup is carried over from the previous time period. Constraint (10) is one of the subtour elimination constraints which has been widely applied in the traveling salesman problem. It enforces that there is only a single tour covering all the given nodes and no disjointed tours are allowed. Figure (1a) shows an example of subtour. A disjointed tour (1-2-3-1) is not allowed in the Traveling Salesman Problem (TSP). Constraint (10) makes sure that each product will be visited once and only once. The feasible route in Figure (1b) assigns $V_{1,t} = 1$, $V_{2,t} = 2, \cdots, V_{5,t} = 5$.

There are various modeling approaches to avoid subtours. Smith and Ritzman avoid subtours using the finish time of each product [21], while Knut used production sequence [22]. As shown in Figure (1a), a subtour containing sequence 1-2-3-1 assigns $V_{2,t} = V_{1,t} + 1$, $V_{3,t} = V_{2,t} + 1$ and



(a) An example of subtour (b) An example of feasible route

Figure 1: A subtour and feasible solution to the traveling salesman problem

$V_{1,t} = V_{3,t} + 1$. Since we cannot assign two different values to one decision variable, constraint (10) can avoid any subtours and return a solution with each node being visited exactly once.

$$X_{i,t} = 0 \qquad \forall i, t = T + 1 \tag{9}$$

$$V_{j,t} \geq V_{i,t} + 1 - N * (1 - Y_{i,j,t}) \qquad \forall i, j \neq i, t \tag{10}$$

2.5 Multi-stage stochastic programming model

In this study, variables $X_{i,t}$, $Y_{i,j,t}$, $Z_{i,t}$, and $V_{i,t}$ are the baseline production decisions which involve the regular time production quantity and sequence of production. Decision variables $O_{i,t,s}$, $I_{i,t,s}$, and B_{its} are recourse decisions [13]. Demand is the uncertain factor under investigation since it is among the most common uncertain factors in the production design problem [23]. Uncertainties are usually represented with discrete probabilistic scenarios since continuous distributions are computationally challenging to implement in the model [24]. We use a number of scenarios, i.e, $S = \{\mu_1, \cdots, \mu_s\}$ and corresponding probability ν_s to represent original distribution [20]. Each realization represents the demand in that particular time period while the series of realizations disclose the evolution of uncertain demand. In each time period, multiple realizations will be generated to capture the statistical properties of continuous distribution.

The multi-stage stochastic programming model aims to design a production planning with uncertain demand considered explicitly. All of the variables in the deterministic model need to be slightly changed by adding a scenario index s which have probability ν_s. The multi-stage stochastic programming model is formulated as follows:

$$\min \zeta' = \sum_{s=1}^{S} \nu_s * (\sum_{i=1}^{I}\sum_{t=1}^{T} p_i^r * X_{i,t,s} + \sum_{i=1}^{I}\sum_{i\neq j}\sum_{j=1}^{J}\sum_{t=1}^{T} sc_{i,j} * Y_{i,j,t,s}$$
$$+ \sum_{i=1}^{I}\sum_{t=1}^{T} p_i^o * O_{i,t,s} + \sum_{i=1}^{I}\sum_{t=1}^{T} h_i * I_{i,t,s} + \sum_{i=1}^{I}\sum_{t=1}^{T} b_i * B_{i,t,s}) \tag{11}$$

$$X_{i,t,s} + O_{i,t,s} = d_{i,t,s} + I_{i,t,s} - B_{i,t,s} \qquad \forall i, t = 1, s \tag{12}$$

$$I_{i,t-1,s} - B_{i,t-1,s} + X_{i,t,s} + O_{i,t,s} = d_{i,t,s} + I_{i,t,s} - B_{i,t,s} \qquad \forall i, t = 2 \cdots T+1, s \tag{13}$$

$$X_{i,t,s} \leq q_{i,t} * (Z_{i,t,s} + \sum_{j\neq i}^{J} Y_{j,i,t,s}) \qquad \forall i, t, s \tag{14}$$

$$\sum_{i=1}^{I} p_i * X_{i,t,s} + \sum_{i=1}^{I}\sum_{i\neq j}\sum_{j=1}^{J} st_{i,j} * Y_{i,j,t,s} \leq cap_t \qquad \forall t, s \tag{15}$$

$$O_{i,t,s} \leq \alpha * X_{i,t,s} \qquad \forall i, t, s \tag{16}$$

$$\sum_{i=1}^{I} Z_{i,t,s} = 1 \qquad \forall t, s \tag{17}$$

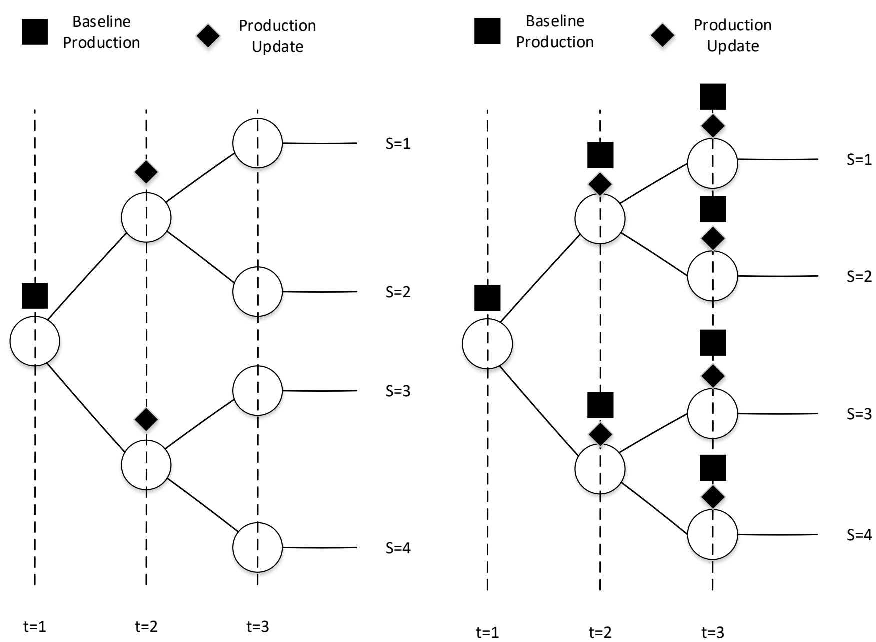
$$Z_{i,t,s} + \sum_{j\neq i}^{J} Y_{j,i,t,s} = Z_{i,t+1,s} + \sum_{j\neq i}^{J} Y_{i,j,t,s} \qquad \forall i, t = 1 \cdots T, s \tag{18}$$

$$X_{i,t,s} = 0 \qquad \forall i, t = T+1, s \tag{19}$$

$$V_{j,t,s} \geq V_{i,t,s} + 1 - N * (1 - Y_{i,j,t,s}) \qquad \forall i, j \neq i, t, s \tag{20}$$

Objective function (11) along with constraints (12) - (20) are based on the deterministic model by adding the scenario index s to the equations in the deterministic model. The only difference is that baseline production can be determined at the beginning of each time period given previous realizations of uncertainty. It should be noted that there are T time periods in the model and production process. The reason to include last dummy time period is to capture the setup carried over from T to $T+1$. We assume that there is no demand or production in the last time period. If $T+1$ is not added, then constraints (8) and (18) will be violated. Besides constraints (11) - (20), we need an additional type of constraint in the multi-stage model, called non-anticipativity constraint. Before explaining the non-anticipativity constraint, we want to first distinguish the multi-stage stochastic model from the two-stage model. The comparison between the two models are shown in Figures (2a) and (2b). Assuming there is a stochastic production problem with three time periods and four scenarios. The positions where we place squares are the time points that we need to determine for the baseline production plan, while the locations of diamonds are the time points that we need to decide for possible updates/recourse for the baseline production. Figure (2a) is a two-stage stochastic programming model in which all of the baseline production decisions need to be determined at the beginning of production horizon $t = 1$ without having any information of uncertainty, and the production updates can be made after the realization of uncertainty in the first period. Figure (2b) is a multi-stage stochastic programming model in which the baseline production can be determined at the beginning of each time period based on the previous information. Clearly, the multi-stage stochastic programming model has a larger decision space and its baseline production decisions as well as production updates are allowed based on the previous realizations and decisions.

An important requirement of the non-anticipative decision process is that the baseline decisions taken at any points do not depend on future realizations of uncertainty, but it is impacted by the previous realizations of uncertainty as well as the knowledge of previous decisions [25]. We use Figure (2b) to illustrate the functionality of a non-anticipativity constraint in the multi-stage stochastic programming model. At the beginning of $t = 1$, no information is revealed, so the baseline production for $t = 1$ should be identical across all scenarios. During the first time period, uncertainty is revealed to be one of the two outcomes. In Figure (2b), scenario 1 and 2 share one outcome while scenario 3 and 4 share the other outcome at time $t = 2$. Given the information in the first time period, the baseline production decisions should be identical at $t = 2$ for $s = 1, 2$ and the same principle applies for $s = 3, 4$ at $t = 2$.



(a) A two-stage model (b) A multi-stage model

Figure 2: Comparison of two-stage and multi-stage models

As mentioned before, continuous distribution is computationally challenging to implement. Therefore, the uncertainty was approximated with multiple sceanrios. The goal of this process is to simplify the problem as well as capture the statistical properties of original distribution [26]. In this paper, the planning horizon has multiple periods which significantly increases the number of scenarios. A scenario reduction procedure has been implemented to identify a representative subset of scenario so that essential features and computational tractability can be maintained [27]. The details of moment matching and scenario reduction techniques are discussed in the detail in the case study section.

3 Case study

In order to demonstrate and validate the multi-stage stochastic programming model proposed in this paper, we apply a case study to a braking equipment manufacturing plant located in Italy. An analysis shows that disturbances affect both upstream and downstream manufactories. Hence, a robust production design is required to balance between production profit and customer satisfaction [28, 29]. In this case study, the manufacturing plant collects two different types of raw material, P_1B and P_2B, and produces three final braking productsP_1, P_2 and P_3. The overall production horizon for the lot-sizing and scheduling problem is usually shorter than half an year, while each time slot is commonly on the weekly or monthly basis [18, 19].

3.1 Data sources

The case study in this paper is a single-level, multi-product, and multi-period stochastic programming problem. Single-level means there is no semi-finished product. Probability density functions of demand are fitted with historical data. Demands of three final products P_1, P_2, and P_3 follow Weibull distribution. We assume that the demand is both product and period independent [30, 31]. The details of statistical properties are shown in Table 2.

Table 2: Statistical properties of monthly demand [13]

Properties	P_1	P_2	P_3
PDF	Weibull	Weibull	Weibull
Scale	518	38	169
Shape	1.51	2.76	2.27
Mean	467.25	33.82	149.70
Variance	99422	175.4231	4877.8
Skewness	1.06	0.25	0.47
Kurtosis	4.35	2.78	2.98

Production sequences can make significant impact on overall production cost as changeovers are sequence dependent. Therefore, it becomes essential to identify the optimal production plan. Setup changeover time and manufacturing time are listed in Table 3 and 4, respectively. Setup cost can be derived by multiplying setup time with a constant factor [32, 33].

Inventory costs and time capacities are included in Table 4 and 5, respectively. Gnoni et al.

Table 3: Setup changeover time (mins/setup) [13]

	P_1	P_2	P_3
P1	0	270	90
P2	180	0	270
P3	90	180	0

Table 4: Costs and manufacturing time for different products

	P_1	P_2	P_3
Manufacturing time (mins/unit)	6	6.6	7.2
Inventory cost ($/unit month)	0.16	0.15	0.38
Regular production cost ($/unit)	254.08	254.08	254.08

claimed that time capacity is the bottleneck and critical resource in production [32]. The regular time production costs are shown in Table 4. The overtime manufacturing cost and backlog cost are based on the regular manufacturing cost [20, 34]. Maximum overtime production quantity is setup to 20% of regular production quantity as large overtime allowance reduces efficiency and increases the chance of injury [20].

Table 5: Time capacities on the machine (mins) [13]

Month	Capacity
1	6087
2	5367
3	6087
4	6087
5	4407
6	4407

Identifying the optimal production quantity as well as sequence are two critical decisions in production problems [35, 36]. Park did a sensitivity analysis to explore the impact of production capacity resource on production decisions [37]. Hu and Hu tested 4 different regular time production resources [13]. In this study, similar experiment settings have been employed.

3.2 Scenario generation and reduction

Representing uncertain parameters with continuous distributions has proven to be computationally challenging for a stochastic model [38]. A common way to simplify and approximate the continuous distribution is to discretize it with a number of realizations. This process is called scenario generation. Scenario size increases dramatically as the number of time horizons increase which affects the tractability of the solution. Therefore, it is common to select a subset of representative scenarios from the entire set. This process is known as scenario reduction [38].

3.2.1 Scenario generation technique

Scenario generation technique is briefly reviewed in this section. Ψ includes all the statistical properties we want to consider in the model. In this study, ψ belongs to the set Ψ which includes the first four moments. ω_ψ is the weight for statistical property ψ which measures the importance of matching mathematical expression [27]. $f_\psi(\pi, Pr)$ represents the mathematical expression for each ψ, and VAL_ψ is the input parameter for ψ. The goal of this model is to generate the discrete realizations π_ψ with probabilities Pr_ψ so that the squared differences between mathematical expression and given input is minimized. For example, if we want to approximate a normal distribution, then Ψ contains statistical properties such as mean and variance. VAL_ψ is the given mean/variance of the normal distribution as a input parameter. $f_\psi(\pi, Pr)$ is the mathematical expression for mean/variance which can be expressed as $\sum_\psi \pi_\psi Pr_\psi$ or $\sum_\psi Pr_\psi * (\pi_\psi - \sum_\psi \pi_\psi Pr_\psi)^2$. The objective function (21) aims to minimize the overall weighted squared distance between the specified value of the statistical property and the value of the mathematical expression. An objective value of zero means that the discrete realizations match with the specified statistical property perfectly. Constraints (22) and (23) state that the probability of all realizations should add up to 1 and be positive.

$$\min_{\pi, Pr} \sum_{\psi \in \Psi} \omega_\psi * (f_\psi(\pi, Pr) - VAL_\psi)^2 \tag{21}$$

$$\sum Pr * M = 1 \tag{22}$$

$$Pr \geq 0 \tag{23}$$

In this paper, we consider the first 4 moments: mean, variance, skewness and kurtosis. A

Table 6: Four scenario trees

Scenario Tree τ_1				
Realizations	Probability	Demand of P_1	Demand of P_2	Demand of P_3
1	0.158	7.466	57.583	179.142
2	0.242	444.582	39.492	55.517
3	0.3	448.167	30.808	162.662
4	0.12	1207.037	10.72	293.763
5	0.18	439.649	25.749	133.073
Scenario Tree τ_2				
Realizations	Probability	Demand of P_1	Demand of P_2	Demand of P_3
1	0.15	358.495	30.561	37.734
2	0.332	487.503	30.561	142.587
3	0.227	486.63	55.11	142.963
4	0.11	1226.625	9.457	128.929
5	0.181	33.365	30.548	276.923
Scenario Tree τ_3				
Realizations	Probability	Demand of P_1	Demand of P_2	Demand of P_3
1	0.162	18.949	57.58	104.528
2	0.2	502.759	23.08	52.293
3	0.2	524.753	37.113	181.643
4	0.11	1225.85	10.74	297.157
5	0.328	377.621	34.361	162.487
Scenario Tree τ_4				
Realizations	Probability	Demand of P_1	Demand of P_2	Demand of P_3
1	0.193	40.745	30.865	167.317
2	0.25	511.049	30.069	56.874
3	0.22	431.713	30.866	134.101
4	0.11	1226.769	9.462	297.223
5	0.227	448.113	55.12	180.546

non-linear objective function allows to reset the initial values and execute the model until a good solution is obtained. We assume that the demand of final material are both period and product independent [3, 30, 31]. Multi-product and multi-period scenario trees are generated in this paper. Three products, six time periods and four statistical properties lead to $|\Psi| = 72$ specified statistical properties. The minimum number of realizations in each period is four, and we choose to create five realizations in each period since we need to balance the trade-off between the quality of the solution and the complexity of the problem. The GAMS (General Algebraic Modeling System) is used to solve this non-linear optimization problem. Due to the fact that there can be multiple optimal solutions, we created four different scenario trees each has 5^6 scenarios in order to compare and validate the results. All the scenario trees have objective values of zero implying that the discrete realizations have a perfect match with the specified properties of continuous distribution, and satisfactory results are reached [26]. A summary of scenario trees is constructed in Table 6. It should be noted that we only include the realizations in the first period since the demand is period-independent.

3.2.2 Scenario reduction technique

Each scenario tree we generated has 5^6 scenarios and solving a NP-Hard problem with this amount of scenarios becomes computationally intractable. Therefore, we adopted scenario reduction to reduce the computational complexity. There are two types of scenario reduction techniques, one is called fast forward selection (FFS) and the other one is called backward selection (BS). The FFS outperforms the BS when the size of selected scenarios is no more than 25% of the size of original scenarios. FFS is used in this paper as sample size after reduction is approximately 1% of original scenario size. We decided to keep different scenario sample sizes and test the stability as well as the quality of scenario reduction.

Table 7: Notations for FFS

S	Scenario set
μ_s	Scenario s
ν_s	The probability of scenario s
$L(\cdot)$	Nonnegative function L_2-norm
$D_{k,l}^{[\eta]}$	Distance between scenario k and scenario l at iteration η
$WD_k^{[\eta]}$	Overall weighted distance of scenario k at iteration η
$U^{[\eta]}$	Set of unselected scenarios up to iteration η
Φ	Set of selected scenarios after reduction

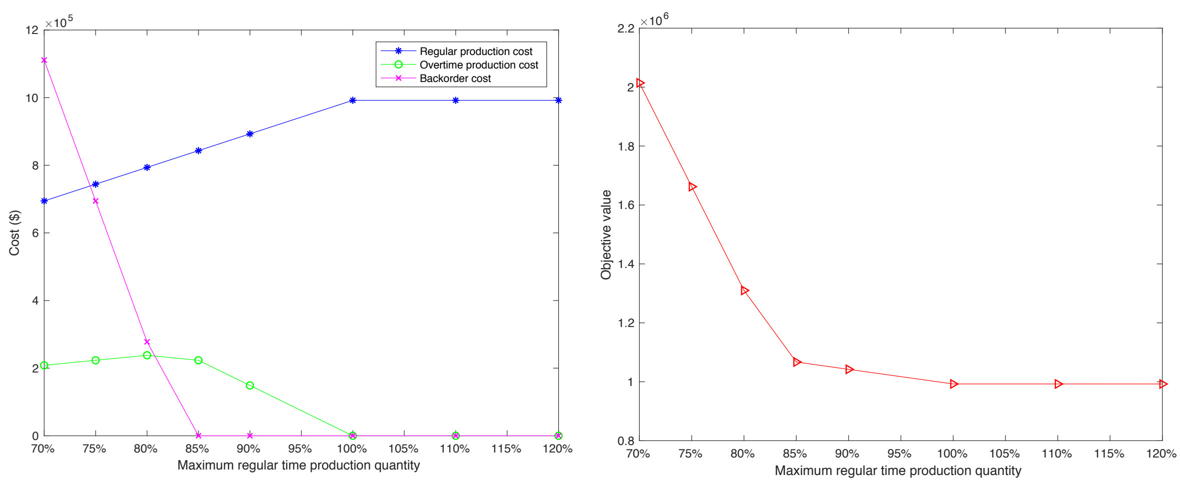
Algorithm 1 Fast Forward Selection Algorithm (FFS)

1: **function** $(S, \mu_s, \nu_s, \eta = 1)$ ▷ η is number of iteration
2: **for** $k = 1 \cdots s \in S$ **do**
3: **for** $l = 1 \cdots s \in S$ **do**
4: $D_{k,l}^{[1]} = L(\mu_k, \mu_l)$
5: **end for**
6: $WD_k^{[1]} = \sum_l \nu_l D_{k,l}^{[1]}$
7: **end for**
8: $k_1 = \arg\min_{k \in 1,\ldots s \in S} WD_k^{[1]}$ $U^{[1]} = 1 \cdots s \backslash k_1$ $\Phi = \Phi + k_1$
9: $\eta = \eta + 1$
10: **for** $k, l \in U^{[\eta-1]}$ **do**
11: $D_{k,l}^{[\eta]} = min[D_{k,l}^{[\eta-1]}, D_{k,k_1}^{[\eta-1]}]$
12: **end for**
13: $WD_k^{[\eta]} = \sum_{l \in U^{[\eta-1]}} \nu_l D_{k,l}^{[\eta]}$ for k $\in U^{[\eta-1]}$
14: $k_\eta = \arg\min_{k \in U^{[\eta-1]}} WD_k^{[\eta]}$ $U^{[\eta]} = U^{[\eta-1]} \backslash k_\eta$ $\Phi = \Phi + k_\eta$
15: **while** selected sample size is not enough **do**
16: run line 9 - line 14
17: **end while**
18: **for** $k \in S \backslash \Phi$ **do**
19: l $= \arg\min_{l \in \Phi} D_{k,l}$ $\nu_l = \nu_l + \nu_k$
20: **end for**
21: **end function**

S is a scenario set in which s = 1 $\cdots$ S. Each scenario can be represented by μ_s which has probability ν_s. L function measures the euclidean distance between two different scenarios. For each scenario, we calculate the overall weighted distance to the rest of scenarios, which is $WD_k^{[\eta]}$. $U^{[\eta]}$ contains all of the unselected scenario up to iteration η. It should be noted that when $\eta = 1$, all of the scenarios are unselected and Φ is empty. In detail, we measure the euclidean distance between each pair of scenario k and l where k,l $\in$ S. The overall weighted euclidean distance is stored in $WD_k^{[\eta]}$. Then we find the scenario with minimum overall distance and remove it from the unselected scenario set U. Next, distance matrix is updated because of the scenario we removed. Next scenario is selected using the same approach until enough scenarios are selected. After scenario selection, we assign the probability of those unselected scenario to the closest selected scenario. The pseudo-code of FFS is included in algorithm 1. Feng and Ryan studied five different sample sizes 10, 20, 30, 50, 100 [38]. We decide to keep 10, 15, 20, 30, 40, 80, 120 and 150 scenarios after reduction and reasons is two-fold: First, we want to test how scenario sample size affects the objective value. Second, we want to check if the results become stable as scenario sample size increases. One interesting observation is that for any i, j ∈ {10, 15, 20, 30, 40, 80, 120, 150}, if we let λ_i indicates the scenario sample with size i, then $\lambda_i \subset \lambda_j \ \forall i < j$. For example, the scenario sample with size 30 is a proper subset of the scenario sample with size 40. This is due to that FFS algorithm is a construction process. Therefore, the sample under the larger size scenario is built upon the sample with the smaller size.

3.3 Analysis for the deterministic case

Results of the deterministic model are shown in Figures (3a) and (3b). The objective value decreases as the maximum batch size increases because we have more regular time production resources. All of the production activity can be done in the regular time when the maximum batch size is 100% of the mean demand. Backorder exists when the maximum batch size is smaller than 85% of the mean demand. We have two different production capacities in this paper, which are the maximum batch size capacity and the maximum time capacity. When the maximum batch size is small, constraint (4) is the binding constraint and that is why objective value changes dramatically as we change $q_{i,t}$. When the maximum batch size is large enough, the overall cost becomes stable since the binding constraint becomes constraint (5), that is, we are running out of production time resource. Utilization of machine time ranges from 65% to 100% depending on the maximum batch size.



(a) Costs in the deterministic model

(b) Objective value in the deterministic model

Figure 3: Results of the deterministic model

Different maximum batch sizes result in different production sequences, but those production sequences have similar setup cost, which means the maximum batch size only affects the production schedule not the setup cost. One of the production sequences is shown in Table 8. It should be noted that the last setup in one period becomes as the first setup in the next period since we assumed that setup can be carried over from period to period. In order to save setup changeover cost, setups are typically saved and reused in the following time period . Note that the setup cost decreases when the maximum batch size changes from 110% to 120% indicating that extra products have been produced ahead of time in order to balance between inventory cost and setup cost. Carrying extra products increases inventory cost, but it can be justified with huge setup cost. In this case, producing and carrying extra products for future demand become beneficial.

Table 8: An optimal production sequence

Products	T_1	T_2	T_3	T_4	T_5	T_6
1	1	2	3	1	2	2
2	3	1	2	3	1	3
3	2	3	1	2	3	1

3.4 Analysis for the stochastic case

Four different scenario trees τ_1 to τ_4 were generated using moment matching technique as detailed in section 3.2.1. The objective values equal to zero in the moment matching method indicating that those scenarios match the continuous distribution perfectly. Scenario reduction is used to select a subset which has a good representation of the original scenario set. In order to compare the solution for the two-stage model with the one for the multi-stage model, we need to find a reasonable scenario sample size. Solving the multi-stage stochastic programming problem to optimality is usually computationally intractable, so we decide to conduct stability test using the two-stage model and then compare the solution with the multi-stage model. The stability test aims to find a good scenario sample size such that the objective value is stable. Intuitively, when the scenario sample size is small, we only keep the scenarios in the central of the set and omit other scenarios that is far from center. On the other hand, as we increase the scenario sample size, the representativeness improves and problem becomes more complicated. Balancing the quantity of scenarios and computational complexity is an important step. The details of the relationship between sample size and objective value are included in Figure 4.

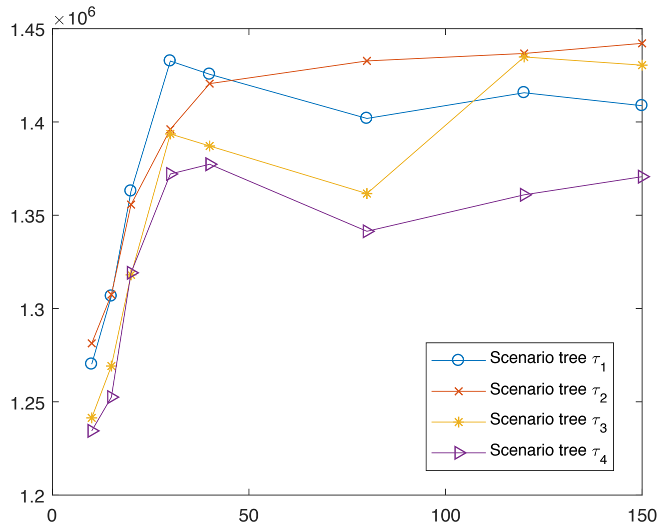


Figure 4: Scenario sample size stability test

The horizontal axis is the number of scenarios after reduction and the vertical axis is the objective values. At the beginning, the objective value has an increasing trend implying that scenario sample sizes like 10, 15, and 20 do not have a really good representation of the original distribution. For example, we may lose some extreme large values or extreme small values when the scenario sample size is small since the focus on is the centroid of the set. As the sample size increases, the mathematical model starts incorporating a more accurate representation of the continuous distribution. In conclusion, when the sample size is smaller than 30, all of scenario trees agree that sample size is insufficient due to increasing trend in the objective value. But when the sample size is bigger than 30, four scenario trees start having different behaviors since different scenario trees cover different aspects of the distribution. We decide to select 30 as the sample size. In addition, when we change the scenario sample size from 30 to 150, range of the objective value is really stable although the result of an individual tree changes randomly. The randomness comes from the fact that for a multi-period tree, one cannot simply compare solutions from different trees, as the nodes beyond the root do not coincide [27]. The reason why variation, in Figure 4, increases as we enlarge the sample size is that four different scenario trees have similar representation of the centroid of the continuous distribution but other representations like variance, skewness, and kurtosis are slightly different. Hence, when scenario sample size is large, different scenario trees will have different realizations. Next, we conducted weak out-of-sample stability test which is defined as follows:

$$f(\Lambda_i, \tau_j) \approx f(\Lambda_j, \tau_i)$$

Λ_i includes all the baseline production decisions at root for scenario tree τ_i. The purpose of this test is to verify whether the scenario sample size we pick in the stability test is good enough. If the scenario tree is weak out-of-sample stable, we should get approximately the same optimal objective values when we solve one scenario tree with the root decisions fixed to the value we get from another tree. The details of weak out-of-sample stability test are included in the Table 9. The biggest difference can be found by applying the root decisions Λ_4 to the scenario tree τ_1, and vice versa. Since the gap is less than 5%, we claim that our scenario sample size stability test is valid.

Table 9: Weak out-of-sample stability test

	τ_1	τ_2	τ_3	τ_4
Λ_1	1,432,658	1,397,574	1,402,861	1,372,923
Λ_2	1,434,616	1,396,084	1,403,776	1,372,938
Λ_3	1,451,086	1,414,833	1,393,577	1,386,877
Λ_4	1,434,973	1,396,948	1,403,319	1,372,113

Comparison of different models can be conducted after obtaining the scenario sample size. Expected value of perfect information (EVPI) measures how much money the perfect information worths and value of stochastic solution (VSS) implies the difference between the deterministic model and stochastic model. Clearly, large EVPI and VSS indicate it is critical to consider uncertainty. EVPI_τ range from 195,684 to 211,513 which are approximately 15% of the two-stage stochastic objective value and VSS_τ range from 210,057 to 256,162 which are approximately

15% to 18% of the two-stage stochastic objective values. Significant EVPI and VSS clearly show that the two-stage stochastic programming model outperforms the deterministic model and considering the uncertainty is necessary. Value of multi-stage stochastic programming (VMS) and relative value of multi-stage stochastic programming (RVMS) are used to measure the improvement of solution by using multi-stage stochastic programming. Clearly, VMS and RVMS are non-negative since multi-stage stochastic program is a relaxation of the two-stage stochastic program.

$$VMS = RP^{TS} - RP^{MS} \qquad RVMS = \frac{RP^{TS} - RP^{MS}}{RP^{TS}}$$

RP^{TS} and RP^{MS} denote the optimal objective values of the two-stage model and of the multi-stage model.

However, multi-stage stochastic programming comes at the expense of solving a much larger and more difficult optimization model and obtains an optimal solution is computationally intractable, we consider the following lower bound:

$$VMS \geq RP^{TS} - RP_F^{MS} \qquad RVMS \geq \frac{RP^{TS} - RP_F^{MS}}{RP^{TS}}$$

Where RP_F^{MS} is a feasible solution to the multi-stage stochastic programming problem. The details of comparison can be found in Table 10. By applying the multi-stage stochastic programming in the field of semiconductor tool production, Huang and Ahmed reported their RVMS varies from less than 5% to around 70% depending on the setup [17]. In order to make a fair comparison, we stop the models at a point where the computation times are around 24 hours and no big improvement in the optimality gaps. Our RVMS values are slightly larger than 10% meaning that, given 24 hours decision making time, we can improve our decision quality by more than 10%.

Note that VMS measures the difference between RP^{TS} and RP^{MS}, and this value is also the difference between the expected value of perfect information to the two-stage model (EVPI^{TS}) and to the multi-stage model (EVPI^{MS}). EVPI measures how much money a decision maker is willing to pay for the perfect information in the future. A big EVPI means uncertainty is worth to be incorporated into the decision making process. In this case study, our VMS value is the

Table 10: Comparison of two-stage and multi-stage objective values

	RP^{TS}	RP_F^{MS}	VMS_{Lower}	RVMS_{Lower}	Optimality gap
τ_1	1,432,658	1,286,643	146,015	10.2%	0.71%
τ_2	1,396,084	1,254,663	141,421	10.1%	0.34%
τ_3	1,393,577	1,213,515	180,062	12.9%	1.56%
τ_4	1,372,113	1,225,938	146,175	10.7%	0.45%

amount of savings when perfect information becomes available, which is about 10% of the total cost. It should be noted that implementing the multi-stage stochastic programming model over the two-stage model is worthwhile due to 10% cost reduction and the 20-hr computation time is manageable for a production planning horizon of 6 months

4 Conclusion

This paper aims to design a multi-stage stochastic programming model to deal with demand uncertainty. A manufacturing plant in the automotive industry has been analyzed in the case study. Scenario generation and reduction techniques have been used to generate scenarios and reduce scenario sample size. Stability test was conducted to examine whether the scenario sample has good representation or not. Results of the two-stage stochastic programming model indicate the importance of considering uncertainty. Improvement in the objective value using multi-stage model has been analyzed.

Fast Forward Selection is used for scenario reduction to ensure good representation of the probabilistic distribution. Based on the sampling stability test, scenario size was kept at 30 after reduction. Compared the deterministic model with the two-stage stochastic programming model, EVPI and VSS are 15% and 18% of the objective value, respectively. It indicates the importance of considering uncertainty. VMS and RVMS are measured to compare the two-stage model with the multi-stage model. Significant VMS indicates big EVPI gap which means significant cost reduction when the multi-stage stochastic model is used for production planning. A big RVMS implies a non-negligible percentage difference in the objective value between the two-stage model and the multi-stage model. We calculated the lower bounds on VMS and RVMS due to the complexity of the multi-stage stochastic programming and thus optimality may not be able to achieve. Results show that the quality of solution can be improved by approximately 10% using the multi-stage stochastic model instead of the two-stage stochastic model.

In summary, this paper presents a multi-stage stochastic programming model to study the lot-sizing and scheduling problem under uncertainty. However, our research has following limitations. Firstly, demand is assumed to be product and period independent. This assumption can be invalid in reality. For example, demand of some automobile parts can heavily depends on the historical data. Secondly, multiple uncertain factors can be studied in our future works as we only focus on one of them. Thirdly, we subjectively determine that the result of scenario sample size is stable when the changing of the objective value is less than 5% which can be a big gap in some other fields. Lastly, heuristics can be designed due to high computational complexity of the multi-stage stochastic programming model. Those limitations should be addressed in our future research.

References

[1] Karimi B., Ghomi S.F., and Wilson J.M. The capacitated lot sizing problem: a review of models and algorithms. *Omega*, 31(5):365–378, 2003.

[2] Özdamar L. and Birbil Ş.İ. Hybrid heuristics for the capacitated lot sizing and loading problem with setup times and overtime decisions. *European Journal of Operational Research*, 110(3):525–547, 1998.

[3] Keyvanshokooh E., Ryan S.M., and Kabir E. Hybrid robust and stochastic optimization for closed-loop supply chain network design using accelerated benders decomposition. *European Journal of Operational Research*, 249(1):76–92, 2016.

[4] Gupta D. and Magnusson T. The capacitated lot-sizing and scheduling problem with sequence-dependent setup costs and setup times. *Computers & Operations Research*, 32(4):727–747, 2005.

[5] Bitran G.R. and Gilbert S.M. Sequencing production on parallel machines with two magnitudes of sequence dependent setup costs. Technical report, Massachusetts Institute of Technology (MIT), Sloan School of Management, 1989.

[6] Silva C. and Magalhaes J.M. Heuristic lot size scheduling on unrelated parallel machines with applications in the textile industry. *Computers & Industrial Engineering*, 50(1):76–89, 2006.

[7] Harris F.W. How many parts to make at once. *Operations Research*, 38(6):947–950, 1990.

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

[9] Kaczmarczyk W. Proportional lot-sizing and scheduling problem with identical parallel machines. *International Journal of Production Research*, 49(9):2605–2623, 2011.

[10] Kimms A. *Multi-level lot sizing and scheduling: methods for capacitated, dynamic, and deterministic models*. Springer Science & Business Media, 2012.

[11] Alem D., Curcio E., Amorim P., and Almada-Lobo B. A computational study of the general lot-sizing and scheduling model under demand uncertainty via robust and stochastic approaches. *Computers & Operations Research*, 90:125–141, 2018.

[12] Rahdar M., Wang L., and Hu G. A tri-level optimization model for inventory control with uncertain demand and lead time. *International Journal of Production Economics*, 2017.

[13] Hu Z. and Hu G. A two-stage stochastic programming model for lot-sizing and scheduling under uncertainty. *International Journal of Production Economics*, 180:198–207, 2016.

[14] Ramaraj G., Hu Z., and Hu G. A two-stage stochastic programming model for production planning in a kitting facility under demand and yield uncertainties. In *IIE Annual Conference. Proceedings*, pages 1520–1525. Institute of Industrial and Systems Engineers (IISE), 2017.

[15] Hu Z., Balaskovits A., and Hu G. Multi-stage lot-sizing and production scheduling under demand uncertainty. In *IIE Annual Conference. Proceedings*, pages 1054–1059. Institute of Industrial and Systems Engineers (IISE), 2017.

[16] Li Y.P., Huang G.H., and Nie S.L. An interval-parameter multi-stage stochastic programming model for water resources management under uncertainty. *Advances in Water Resources*, 29(5):776–789, 2006.

[17] Huang K. and Ahmed S. The value of multistage stochastic programming in capacity planning under uncertainty. *Operations Research*, 57(4):893–904, 2009.

[18] Guimarães L., Klabjan D., and Almada-Lobo B. Modeling lotsizing and scheduling problems with sequence dependent setups. *European Journal of Operational Research*, 239(3):644–662, 2014.

[19] Drexl A. and Kimms A. Lot sizing and scheduling - survey and extensions. *European Journal of Operational Research*, 99(2):221–235, 1997.

[20] Zhang X., Prajapati M., and Peden E. A stochastic production planning model under uncertain seasonal demand and market growth. *International journal of production research*, 49(7):1957–1975, 2011.

[21] Smith-Daniels V.L. and Ritzman L.P. A model for lot sizing and sequencing in process industries. *The International Journal of Production Research*, 26(4):647–674, 1988.

[22] Haase K. Capacitated lot-sizing with sequence dependent setup costs. *Operations-Research-Spektrum*, 18(1):51–59, 1996.

[23] Mula J., Poler R., Garcia-Sabater J.P., and Lario F.C. Models for production planning under uncertainty: A review. *International journal of production economics*, 103(1):271–285, 2006.

[24] Escudero L.F. and Kamesam P.V. On solving stochastic production planning problems via scenario modelling. *Top*, 3(1):69–95, 1995.

[25] Dupacová J., Bertocchi M., and Moriggia V. Testing the structure of multistage stochastic programs. *Computational Management Science*, 6(2):161, 2009.

[26] Høyland K. and Wallace S.W. Generating scenario trees for multistage decision problems. *Management Science*, 47(2):295–307, 2001.

[27] Heitsch H. and Römisch W. Scenario reduction algorithms in stochastic programming. *Computational optimization and applications*, 24(2-3):187–206, 2003.

[28] Svensson G. A conceptual framework for the analysis of vulnerability in supply chains. *International Journal of Physical Distribution & Logistics Management*, 30(9):731–750, 2000.

[29] Haag E. and Vroom R.W. The application of step in the automotive supply chain. *Computers in industry*, 31(3):223–234, 1996.

[30] Bassok Y. and Akella R. Ordering and production decisions with supply quality and demand uncertainty. *Management Science*, 37(12):1556–1574, 1991.

[31] Kopcso D.P. and Nemitz W.C. Learning curves and lot sizing for independent and dependent demand. *Journal of Operations Management*, 4(1):73–83, 1983.

[32] Gnoni M.G., Iavagnilio R., Mossa G., Mummolo G., and Di Leva A. Production planning of a multi-site manufacturing system by hybrid modelling: A case study from the automotive industry. *International Journal of production economics*, 85(2):251–262, 2003.

[33] James R.J. and Almada-Lobo B. Single and parallel machine capacitated lotsizing and scheduling: New iterative mip-based neighborhood search heuristics. *Computers & Operations Research*, 38(12):1816–1825, 2011.

[34] do Rego J.R. and de Mesquita M.A. Demand forecasting and inventory control: A simulation study on automotive spare parts. *International Journal of Production Economics*, 161:1–16, 2015.

[35] Jamal A.M.M., Sarker B.R., and Mondal S. Optimal manufacturing batch size with rework process at a single-stage production system. *Computers & Industrial Engineering*, 47(1):77–89, 2004.

[36] Sarker B.R. and Parija G.R. An optimal batch size for a production system operating under a fixed-quantity, periodic delivery policy. *Journal of the Operational Research Society*, 45(8):891–900, 1994.

[37] Park* Y.B. An integrated approach for production and distribution planning in supply chain management. *International Journal of Production Research*, 43(6):1205–1224, 2005.

[38] Feng Y. and Ryan S.M. Scenario construction and reduction applied to stochastic power generation expansion planning. *Computers & Operations Research*, 40(1):9–23, 2013.