SUPPLIER SELECTION FOR ASSEMBLY COMPONENTS UNDER LEAD TIME UNCERTAINTY

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ZEYNEP KÜREKCİ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

FEBRUARY 2022

Approval of the thesis:

SUPPLIER SELECTION FOR ASSEMBLY COMPONENTS UNDER LEAD TIME UNCERTAINTY

submitted by **ZEYNEP KÜREKCİ** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _______________

Prof. Dr. Esra Karasakal
Head of Department, **Industrial Engineering** _______________

Prof. Dr. Sinan Gürel
Supervisor, **Industrial Engineering, METU** _______________

Examining Committee Members:

Prof. Dr. Pelin Bayındır
Industrial Engineering, METU _______________

Prof. Dr. Sinan Gürel
Industrial Engineering, METU _______________

Prof. Dr. Cem İyigün
Industrial Engineering, METU _______________

Assist. Prof. Dr. Özgen Karaer
Industrial Engineering, METU _______________

Assist. Prof. Dr. Cihan Tuğrul Çiçek
Industrial Engineering, Atılım University _______________

Date: 10.02.2022

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: ZEYNEP KÜREKCİ

Signature :

ABSTRACT

SUPPLIER SELECTION FOR ASSEMBLY COMPONENTS UNDER LEAD TIME UNCERTAINTY

KÜREKCİ, ZEYNEP
M.S., Department of Industrial Engineering
Supervisor: Prof. Dr. Sinan Gürel

February 2022, 46 pages

Supplied components constitute a great majority of companies' total cost and supplier selection is one of the key operations of supply chain management in the competitive world. In this study, our main concern is to start assembly processes of an aircraft company at the planned starting date by receiving components that are required for each process on time. We decide on which components will be ordered from which suppliers under scenario based supplier lead times. The aim is to minimize total cost resulting from penalty costs of assembly processes, holding and procurement costs of components. We propose a linear mixed integer stochastic programming model to solve the problem. Also, we propose a supplier selection algorithm to solve large instances of the problem in reasonable computational times. We conduct an experimental study and examine the performance of the mathematical model and the algorithm.

Keywords: supplier selection, assembly operations, lead time uncertainty, stochastic programming, scenarios

ÖZ

MONTAJ BİLEŞENLERİ İÇİN TESLİM SÜRESİ BELİRSİZLİĞİ ALTINDA TEDARİKÇİ SEÇİMİ

KÜREKCİ, ZEYNEP
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Sinan Gürel

Şubat 2022 , 46 sayfa

Tedarik edilen bileşenler, şirketlerin toplam maliyetlerinin önemli bir kısmını oluşturur ve tedarikçi seçimi, rekabetçi dünyada tedarik zinciri yönetiminin kilit operasyonlarından biridir. Bu çalışmada esas amacımız, bir uçak imalat firmasının montaj süreçlerini, her bir süreç için gerekli olan bileşenleri zamanında tedarik ederek, planlanan başlangıç tarihinde başlatmaktır. Hangi tedarikçilerden hangi bileşenlerin sipariş edileceğine senaryo bazında oluşturulan belirsiz tedarikçi teslim süreleri altında karar veriyoruz. Amaç, montaj süreçlerinin ceza maliyetlerinden, bileşenlerin elde tutma ve tedarik maliyetlerinden kaynaklanan toplam maliyeti en aza indirmektir. Problemi çözmek için doğrusal karışık tamsayılı bir stokastik programlama modeli önerdik. Ayrıca, büyük problem örneklerini daha kısa hesaplama sürelerinde çözmek için bir tedarikçi seçim algoritması oluşturduk. Deneysel bir çalışma yürüterek matematiksel modelin ve algoritmanın performansını inceledik.

Anahtar Kelimeler: tedarikçi seçimi, montaj operasyonu, teslim süresi belirsizliği,

stokastik programlama, senaryolar

ACKNOWLEDGMENTS

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

MRP	Material Requirement Planning
BOM	Bill of Material
SSA	Supplier Selection Algorithm
NSA	Neighbourhood Search Algorithm

CHAPTER 1

INTRODUCTION

Supply chain management is one of the key functions for companies to gain competitive advantage in the globalizing world. Alkahtani et. al. [1], states that supplies, consist of purchased components, materials, etc., constitutes 40% to 60% of the cost of the sold products. Therefore, making the right decision on supplier selection is a compulsory and compelling task since it affects the price, on time delivery and quality performance. Supplier selection decisions have significant influence on supply chain management. In this thesis, we discuss supplier selection problem within the scope of assembly operations.

Our problem is inspired by assembly processes of an aircraft company. This company works on project basis and signs long term contracts with its customers. It provides customers with aircraft, sub-assemblies for aircraft and spare parts, contractually. Production starts based on delivery dates stated in the contracts, therefore; there is no demand uncertainty. Production processes have long lead times resulting from manufacturing, quality control and documentation times since products in aerospace industries require high precision and high quality standards.

In this thesis, we focus on sub-assembly operations of this company. Each sub-assembly requires certain quantity of different components. Lead time for each component and operation is estimated by the company and given as an input to the material requirement planning (MRP) system. Bill of material (BOM) includes hierarchical information for sub-assemblies and their component requirements. BOM explodes according to the delivery date of sub-assembly determined by the contract terms and manufacturing times of each operation. By this way, MRP system gives alerts when to release orders for components, so we do not study on order release date decisions which are made by the MRP system. Orders of the components that used in the same assembly operation are launched at the same time.

The most significant risk for an assembly operation in this sector is to miss the planned delivery dates. The assembly process cannot start until all required components are available. Components are produced in house and some are procured from approved suppliers. Supplied parts constitute a great majority of the component demands; therefore, this study focuses on supplied parts. MRP system provides purchasing engineer with the component list to order. Purchasing engineer selects a supplier for each component in the list.

Purchasing engineer has the list of suppliers, their capacities and historical lead time data. Lead time for a component includes manufacturing, quality control, packaging and transportation times. In other words, lead time is the time between order release date and order arrival date. Suppliers lead times are uncertain. Therefore, at the time of order release, we don't know the exact delivery dates of components.

Each assembly operation should start at the planned starting date but if some components arrive late then it is delayed . We can only launch a sub-assembly operation when all required components are available. Therefore, the latest arrival time of the components determines the process starting time. We assumed that the manufacturing time of an assembly process is fixed so if it cannot start at the planned time, it will be late and penalty cost will occur. Since these are huge and expensive projects, a considerable amount of daily penalty cost are incurred for any delay from the delivery schedule. Also, delays cause loss of reputation. Hence, the company aims to start all assembly operations on time.

As said, due to uncertainty in lead times, components usually do not arrive at the same time. Until assembly process starts, the components that arrive earlier should be stored in inventory so storage cost occurs for those items. Also, storing some of the components is challenging because they may be very big in size. Finding an appropriate place for big components is not always possible. Even if the components are not large in size, all of them are exposed to the risks of damage and lost. That is why, the company also tries to avoid early deliveries.

To sum up, there is a trade off between storage cost and penalty cost. A purchasing engineer do not want components arrive early and also arrive late. Assembly process starting times and component arrival times should be as closer as possible to the planned starting time. The release date of orders are given by MRP system so the purchasing engineer can only decide which component to be procured from which supplier. While making this decision, purchasing engineer has to consider suppliers' capacities and uncertain lead times.

The aim of this study is to develop models and algorithms for supplier selection decisions with uncertain delivery times while minimizing total cost resulting from penalty, storage and purchasing costs. A single objective, scenario based, mixed integer, linear, stochastic programming model and a meta-heuristic algorithm is proposed to solve this problem.

In the literature, supplier selection problem has received significant attention in the last decades. Researchers mostly focus on satisfying component demands of assemblies. Existing studies mostly adapt inventory control approaches to supplier selection decisions. To the best of our knowledge the interdependence of components as described above for our problem, is not studied before. To satisfy component demands of assembly processes and to control inventory levels is important but the components that arrive late, delay the assembly process and cause other components to be hold in inventory. If there is an assembly requirement, the interdependence of components to each other should be also incorporated to the problem. In this thesis, we consider assembly schedules in supplier selection decisions. In Chapter 2, we review the related studies in the literature.

In Chapter 3, we present a scenario based stochastic mixed integer linear mathematical model for the problem. For large instances, IBM CPLEX fails to solve mathematical model in reasonable computational times. Thus, we propose a meta-heuristic algorithm, as well. The algorithm consists of a construction and an improvement phase. Construction phase uses a greedy approach. Improvement phase is a neighbourhood search algorithm, which uses simulated annealing approach. After the mathematical model, the meta-heuristic is explained and pseudo codes are given in Chapter 3.

In Chapter 4, we explain how we generate data, especially lead time data under different scenarios. Then, we discuss computational results for the model and the meta-heuristic algorithm.

Finally, in Chapter 5, we come up with a conclusion and discuss possible further improvements related to our problem.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we review related studies in supplier selection literature and assembly systems with uncertainties. In Section 2.1, we cover supplier selection problems with uncertainties and disruption risks. We devote Section 2.2 to studies on assembly systems.

2.1 Supplier Selection with Uncertainties and Disruption Risks

Supplier selection problem is a well studied research topic. There are studies which assume all problem parameters to be deterministic and there are others which consider uncertainties. According to Alkahtani et. al. [1], supplier selection is the process of finding the appropriate suppliers who can provide the buyer with certain products by satisfying quantity, quality, price and on time delivery requirements. Naqvi et. al. [2] provides a literature review for supplier selection problems and order allocation studies. They first summarize literature review papers. Then, they review deterministic studies and the studies that deal with uncertainty in supplier selection problems. Sources of uncertainty are demand, capacity, cost, delivery time, quality and quantity discount. The models with delivery time uncertainties contribute to 7 % of studies with uncertainties (Naqvi et. al. [2]).

Hammami et. al. [3] works with fluctuations of currency exchange rates and price discounts. They work on a global setting. Suppliers are located in different regions and their currency exchange rates are exposed to volatility. Şen et. al. [4] only consider stochastic discount offers. They work on a setting with multiple items, multiple periods, multiple suppliers. Both Hammami et. al. [3] and Şen et. al. [4] propose a scenario-based stochastic model similar to our study.

Some studies address only demand uncertainties. Awasthi et. al. [5] study an order allocation problem with stochastic demands. They assume that the demand has a known density function and suppliers have minimum and maximum order size restrictions. They also consider maximum order size as the maximum capacity for each supplier. Similarly, in our study, suppliers have capacities. These two studies consider price as a single decision criterion. Manerba et. al. [6] developed two seperate models. One of them considers only product demand uncertainty while other considers only product price uncertainty. Suppliers have limited capacity and also offer quantity discounts. The aim is to meet the demand with the lowest possible cost. The only criterion in the objective is price like the studies of Awasthi et. al. [5]. They also apply a branch and cut approach to solve the problem. In this thesis, we consider storage and penalty costs as well as price. We also focus on assembly schedule which must be caught.

Zhang et. al. [7] work with demand, quantity discount and fixed selection cost uncertainties. These uncertain parameters have continuous probability distribution functions. They make supplier selection and order allocation decisions. Their objective function is similar to the one considered in this thesis. They minimize holding, shortage, procurement and fixed ordering costs. Different from our problem, they include fixed ordering costs in the objective. They show that their problem is NP hard and propose a Bender's decomposition algorithm.

Li et. al. [8] assume that both the demand and the supplier capacity are uncertain at the same time. They model these uncertainties via scenarios and closed-form probability distributions. Suppliers have capacities like in our study and also offer business discounts. They propose a two-stage stochastic model and a chance-constrained model with multiple objectives. The aim is to select minimum number of suppliers, minimize the risk of not meeting the demand, maximize quality of components and minimize purchasing cost.

Aggarwal et. al. [9] and Bilsel et. al. [10] consider transportation and variable cost uncertainties in addition to demand and supplier capacity uncertainties. They define variable costs as operating, purchasing and holding costs. Both studies propose chance-constrained models. Bilsel et. al. [10] consider disruption risks while Aggarwal et. al. [9] take lead time uncertainties into consideration. They used goal programming and weighted aggregate function method to solve the problem. Aggarwal et. al. [9] have common approaches with our study.They adapt a scenario based approach and lead times are uncertain in their studies. Assembly operation setting is not within the scope of the neither of these two papers.

Demand and yield are uncertain in some studies like Gurnani et. al. [11], [12] and Anupindi et. al. [13]. According to Gurnani et. al. [11], the final product is composed of two critical parts and supplier delivery yields are not deterministic, that is to say, suppliers may make partial delivery. In our study, suppliers have to make complete delivery. They both consider to order components from different suppliers and order two components as a set from a single/joint supplier. The problem is to determine a target assembly amount of the end product and to satisfy this target with a certain probability by minimizing purchasing, holding and penalty costs. It is an inventory control problem and they do not consider the timing of assembly operations.

Anupindi et. al. [13] modeled lead time uncertainty in a different way. They allocate the demand between two unreliable suppliers. They generate three types of models for uncertainty. In the first model the supplier make complete shipment of an order with probability (p) or ship no component of an order with probability $(1 - p)$. If supplier does not ship the order, it will make complete shipment in the next period. In the second model, supplier send a fraction of an order. It is simply the yield problem. In model three, in addition to model two, supplier will send the remaining amount in the next period. The way of modelling uncertainty is different from our problem and other problems in the literature. We use scenarios to model lead time uncertainty and do not allow partial deliveries. They consider minimizing ordering, holding and penalty costs.

Assembly operation of the final product requires two critical components in Gurnani et. al. [12]. The components can be ordered separately from individual suppliers or in a set from a joint supplier. Similar to Anupindi et. al. [13], they assume that a supplier supplies 100% of the order quantity with probability (p) or supplies nothing with probability $(1 - p)$. However, these studies also do not consider the effect of different arrival times of components and assembly schedule concern. They only consider assembly requirement quantities.

There are studies in which both demand and supplier lead times are uncertain. Abginehchi et. al. [14] propose two inventory replenishment models. In one of them, only supplier lead times are uncertain, while in the second model both demand and supplier lead times are uncertain. When the inventory level drops to the reorder level, new order is given and the order amount is allocated between different suppliers. The decision maker has to decide on the reorder level and order allocation to minimize the expected total cost, including ordering cost, procurement cost, inventory holding cost, and shortage cost. It is an inventory replenishment model and there is no assembly system.
Assellaou et. al [15] solve a supplier selection and order allocation problem. They involve uncertainty in terms of demand, quality and lead time. They assume that demand follows a normal distribution and lead time follows an exponential distribution. They also consider quality performance of suppliers and average defect rate cannot exceed a threshold value. They assume suppliers have limited capacities and model uncertainties via scenarios in the case of our study. They adapt Var and CVar approaches. In the objective function, delay time, disruption risks, total cost of ordering, purchasing delay, percentage of poor quality are minimized and total service level is maximized. They consider demand for each component but they do not consider the timing of an assembly operation and the dependence between components. Also, they consider penalty costs of components not for assembly operations.

Sawik [16] considered supplier selection decisions with delivery time uncertainties. The problem is to find the optimal allocation of demand between unreliable suppliers in a make to order production environment with a risk-averse behavior. Sawik [16] assumes delivery time and quality level uncertainties exist. In his setting, different types of customer products are manufactured by the factory and lots of components are supplied by vendors to assemble the final product. He adapts a scenario based approach by using delivery patterns which are obtained from past data. Maximum acceptable rates are determined for late delivery and average defect rate. Suppliers have limited capacity. It is a single period problem in which the decision maker selects suppliers and allocates demand for each order with the aim of minimizing cost and achieving high quality and reliability. However, he does not consider any assembly setting. He works with both discount and non-discount environments based on the order volume.

Sawik [17] also conducts another supplier selection and order allocation study with delivery time uncertainties. He takes part quality, purchasing cost and on time delivery into consideration. He adapts VaR and CVaR approaches. In neither of the both studies, Sawik considers assembly schedules and component dependence. His focus is to meet the demand for each component by considering cost minimization and on time delivery. However, in our setting, delivery time of each component affects the starting time of an assembly operation. The aim is not only to meet the demand but also to minimize total cost resulting from holding cost of early delivered items, penalty cost for delays in assembly schedules and purchasing cost.

Sawik [18] also considers regional supplier disruption risks in another study. There are suppliers located in the producer's region and outside the producer's region. There is a trade of between disruption risk and cost of supply. Suppliers that are in the same region with the producer have lower disruption risks while having higher price level. Vendors outside the region have higher disruption risks and lower price level. He provides a scenario based stochastic mixed integer programming model for supplier selection and order allocation. He models the problem by using scenarios and solves it with VaR and CVaR approaches. Sawik [19] considers supply chain disruption and delay risks in multi-period dynamic environment .

In this section, we reviewed the studies on supplier selection problems with different types of uncertainties. Many studies modeled uncertainties using scenarios. Some papers generate scenarios by using historical data and some of them use probability distributions. We use historical delivery data of suppliers and as will be discussed in Chapter 4, we showed that delivery time data fits triangular distribution.
Most of the studies considers inventory control for components. To the best of our knowledge, assembly schedules were not considered in supplier selection problems before. There are studies which consider backlogging for a component but not the delay of an assembly process due to late component deliveries. That is why, the next section is devoted to studies with lead time uncertainties in assembly problems.

2.2 Uncertainties in Assembly Systems

An assembly is a combination of different components. There are one level, two level, multi level assembly systems [20]. We give Figure 2.1 to visualise a one level assembly setting.

In this thesis, we consider different assembly operations with planned starting dates. To start the operation, all components must be available in the planned starting date of the operation. Components that arrive earlier than the planned starting date are stored which results in holding cost. Components that arrive later than the planned starting date cause delay. We minimize these holding and penalty costs with purchasing costs.

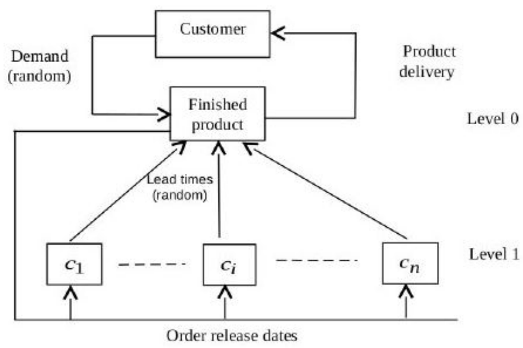

Figure 2.1: One Level Assembly System

BOM includes information about component requirements for each operation in each level of the assembly system. Planned lead times are input to MRP system for each component and assembly operation. According to demand and requirement of the final product, BOM explodes by using lead times of every single operation. By this way component requirements and assembly starting times are determined. Dolgui et. al. [21] provides a literature review on supply planning with uncertainties in MRP environment which considers all studies until 2007. Uncertainties may arise in different matters such as demand uncertainties, lead time uncertainties, etc. In order to mitigate the effects of these uncertainties, safety stock, safety lead time, planned lead time and lot sizing policies are studied.

There are lots of studies conducted with stochastic lead times in assembly systems. Lead time may be uncertain for production and procurement time of components or operation time of assemblies. Companies can use safety stocks or safety lead times to cope with uncertain lead times; however, holding inventory is a costly solution. Especially storage of big components is difficult and costly. Using safety lead time also causes holding extra inventory when parts are produced or procured before the required date. That is why, most studies concentrate on order release dates, in other words, planned lead time optimization to minimize inventory holding and back-ordering costs.

Most studies for planned lead time optimization in assembly systems are inspired by Yano [22], who focused on two and three level assembly systems. Hnaien et. al. [23] also worked on two level systems under lead time uncertainties to find the release dates for the components at level two. Both consider multi objective approaches and also apply genetic algorithm to solve the problem. Jamshidi et.al. [24], considers lead times are not certain in a two level assembly system. The aim is to find optimal order release dates for the components at level two. They also consider the trade of between expected holding cost and back ordering cost by applying Pareto approach in their multi objective problem. They also propose a hybrid genetic algorithm.

Hnaien et. al. [25] devote their study to find optimal order release dates and order quantities for each type of product. They worked in a single period, one level assembly setting with uncertain demand of final product and uncertain lead times of suppliers. Both demand and lead times have a known discrete probability distribution. There are n types of components and n suppliers. Each type of item is supplied by a unique supplier which is known a priori. There is no supplier selection decision. The objective minimizes total cost consists of holding, tardiness, surplus and lost sales costs. They also use branch and bound algorithm to solve the model. Ben-Ammar et. al. [26] work on a one level, multi period inventory model, as well.
Dolgui et. al. [27] focus on finding the optimal values for planned lead times which are parameter values of MRP system with the aim of minimizing expected holding and backlogging costs. Production lead time of each item has its own probability distribution. Supply capacity is not finite. They studied a one level assembly system with newsboy model approach and multi level assembly system with Markov chain approach. By adopting similar assumptions with this study, Ben-Ammar et. al. [28] work on a multi level assembly system with uncertain lead times of components. Lead

time has a known discrete probability distribution and demand is known. The aim is to find optimal order release date for the components at the lowest level of assembly to minimize expected total cost which includes holding cost of components, back ordering and holding cost of the final product.
Ben-Ammar et. al. [29] present another paper on multi level assembly system problem. They decide when to launch the orders of components at the last level in BOM with uncertain lead times of all processes. Demand of final product is deterministic. Lead times have a known discrete probability distribution and covers both processing times and additional times depending on load, capacity constraints, transportation etc. In our study, lead times include production, transportation and material handling times, as well. They decompose the multi-level assembly system into several multi-level linear chains. They used newsboy approach by this way. A branch and bound algorithm is also proposed to achieve the solution in relatively shorter times.

The study of Chu et. al. [30] is devoted to ordering instant and order timing decisions. Components that needed in assembly of the final product are provided by suppliers. Suppliers are selected priory and it is known which item is bought from which supplier. Lead times are uncertain and random with a known continuous distribution. They decide ordering instant by minimizing inventory and backlogging costs. They make trials with different continuous probability distributions of lead times.

Louly et. al. [31] also focus on planned lead time optimization in multi level assembly systems with a different approach. A supply reliability coefficient is assigned to each supplier. Planned lead time multiplication with supplier coefficient gives safety lead times. They also construct a branch and bound algorithm. Optimal planned lead times for level two items are determined by Tang et. al. [32]. They also get safety lead times.

Song et. al. [33] incorporates both lead time and demand uncertainties into their work. They consider holding cost of components, penalties for tardiness of assembly operations, overage and underage costs for satisfying the demand. Two level assembly system is investigated for only one type of finish product. There is no supplier selection decision in the problem, it is decided on when and how much to order for each component. Both demand and lead time have discrete probability distribution.

They also proposes three different solution procedures and compare them to each other which are news vendor, mean lead time and mean demand heuristics [33].

To the best of our knowledge, assembly system studies in the literature mostly make decisions on order release times. Studies mentioned above do not consider supplier selection decisions. Suppliers of components are determined priory and given as parameter to the models. In our problem, MRP system determines order release times and we decide on which component will be ordered from which supplier.

To sum up, in the literature, studies exists for supplier selection and assembly system problems with lead time uncertainties. However, supplier selection problems do not consider assembly settings and schedules. They ignore the dependence of components to each other and the delays caused by the components that arrive late. There are studies that catch the dependence of components and considers assembly systems but they work on order release times.

In this thesis, we make supplier selection decisions by considering assembly processes of an aircraft company. We consider assembly system and schedule that the latest arrival of components determines the assembly starting time and the components arrive before the assembly starting time have to be stored until the process starts. Our study brings supplier selection and assembly system problems together. This thesis brings an assembly system perspective to supplier selection literature.

CHAPTER 3

PROBLEM DEFINITION AND PROPOSED SOLUTION APPROACHES

In this chapter, we first give a detailed description of the problem.Then, we introduce the mathematical model and finally describe the solution algorithm.

3.1 Problem Definition

This study focuses on a supplier selection problem for assembly operations in an aircraft company. The company produces sub-assemblies for aircraft. It makes long term contracts with its customers. These contracts determine the production plan of sub-assemblies.

Producing sub-assemblies require manufacturing or procuring components and then assembling them to get the final product. The cost and on time delivery performance of the company is affected by its suppliers' performance. Assembly operation for each product requires different amounts of components and each assembly operation is considered as an independent one level assembly system.

BOM includes all hierarchical information of assembly operations and their component requirements. Also, lead times of components and manufacturing times of assembly operations are estimated by the company and given as an input to the MRP system. Then, planned starting time for assembly operations and order release times for components are determined by backward propagation. By this way, the company has a predetermined assembly schedule based on customer contracts and assembly times. For a given assembly process and its scheduled time, components of the process must be ordered. Order release times of components are determined by distracting lead time values given as an input to MRP system from the planned assembly starting time. Since company's estimation on lead times of supplied components is valid and the same for all components, orders of components used in the same assembly process are released at the same time. In other words, the company's policy is to order all components of an assembly process at the same time. MRP system notifies purchasing engineer with providing a list of components to be ordered for each assembly operation.

If different lead time values are estimated for different components, their order release times would be different. This time, the model would be a multi period model. In each period, MRP system gives a component list that should be ordered. Also, capacities of suppliers would be updated at each period after selecting suppliers.

The company has designated suppliers. We make operational decisions by choosing between these specific suppliers when opening an order. These suppliers manufacture components usually by processing metals. Each supplier can produce a number of components and each component can be produced by different suppliers.

At a given time point, the purchasing department has to order several components for several sub assemblies. We have to choose which component will be ordered from which supplier. Then, we release the orders. A component can be supplied by only one supplier. If a purchase order is launched for a component, whole demand of that component is produced by a single supplier. Due to the nature of the products in aerospace industries, components require high precision and quality standards. These are not standard products so set-up times and costs are considerably high. Hence, allocating an order between different suppliers is costly and time consuming.

Moreover, partial shipment is not allowed. If there is any defective item, the supplier has to reproduce it and is responsible for complete shipment of an order. Therefore, lead time is also affected by the quality performance of the supplier. Lead times include manufacturing times, packaging and transportation times and quality performance. In other words, lead time is the time between order release time and the time the order is received by the company.

The ERP system in the company uses planned lead times for each component. However, actual lead times are uncertain. Each component may have different lead times and each supplier has different delivery time performance. We model lead time uncertainties via scenarios. Each scenario is generated independently by observing past delivery data of suppliers.

All required components must be available to start an assembly process. Then, actual lead time for a component is critical in an assembly process. If a component is delivered late, then the related assembly process must be delayed. This may cause penalty costs to be paid to the customer. Since aircraft is an expensive and high precision product, daily penalty costs are high in aviation industry. Delays also cause loss in company's reputation.

Late deliveries of components cause problems and penalty cost; however, early deliveries also cause problems. Components delivered before the assembly starts have to be stored in certain physical conditions until the process starts. These components can be very large in size. Hence, storage operations require considerable effort and cost. Also, there is a risk of damage and loss for the stored components.

Furthermore, the company monitors available capacities of its suppliers. This means, number of components that can be assigned to a supplier is limited. Purchasing engineer is aware of the available capacity at a supplier before releasing an order and release orders by taking available capacities into account. By this way, capacity of suppliers are not exceeded.

The problem is, given a set of components to be procured for a set of assembly processes and a set of suppliers, to select a supplier for each component while considering scheduled assembly starting times, lead time uncertainties, capacities and dependence between the components of each assembly process by minimizing expected total cost, consisting of penalty, storage and procurement costs.

In the next section, 3.2, we give the mathematical formulation of the problem.

3.2 Mathematical Model

We propose a linear stochastic mixed integer programming model. We first give the notation for the problem. Then, we will propose a model that solves the defined supplier selection problem.

Sets:

I : Set of components.

J : Set of suppliers.

K : Set of assembly processes.

S : Set of scenarios.

I_k : Set of components required for assembly process k.

Parameters:

T_k : planned starting time for assembly process k.

h_i : daily storage cost for component i.

c_k : penalty cost for production delay of assembly process k per day.

p_{ij} : price paid for component i when purchased from supplier j.

cap_j : capacity of supplier j (number of parts).

q_i : required quantity of component i.

d_{ij}^s : delivery time of component i when purchased from supplier j under scenario s.

p_s : probability of scenario s.

Decision Variables:

$$X_{ij} = \begin{cases} 1, & \text{if component i is purchased from supplier j,} \\ 0, & \text{otherwise.} \end{cases}$$

ST_k^s : actual starting time of assembly process k under scenario s.

H_i^s : storage time for item i under scenario s.

ST_k^s represents the assembly process starting time. An assembly either starts at the planned starting date or starts when its latest component is delivered. ST_k^s is calculated as below:

$$ST_k^s = \max\left(T_k, \max_{i \in I_k}\left(\sum_{j \in J} d_{ij}^s \cdot X_{ij}\right)\right) \qquad \forall s \in S, k \in K$$

which can be linearized by the following inequalities:

$$\begin{aligned} ST_k^s \geq & \sum_{j \in J} d_{ij}^s \cdot X_{ij} && \forall s \in S, k \in K, i \in I_k \\ ST_k^s \geq & T_k && \forall s \in S, k \in K \end{aligned}$$

We explain the mathematical model in the next section.

Model:

$$min \sum_{s \in S} \sum_{i \in I} h_i \cdot p^s \cdot q_i \cdot H_i^s + \sum_{s \in S} \sum_{k \in K} c_k \cdot p^s \cdot (ST_k^s - T_k) + \sum_{i \in I} \sum_{j \in J} p_{ij} \cdot q_i \cdot X_{ij} \quad (3.1)$$

st.

$$\begin{aligned}
ST_k^s &\geq \sum_{j \in J} d_{ij}^s \cdot X_{ij} && \forall s \in S, k \in K, i \in I_k && (3.2) \\
ST_k^s &\geq T_k && \forall s \in S, k \in K && (3.3) \\
H_i^s &= ST_k^s - \sum_{j \in J} d_{ij}^s \cdot X_{ij} && \forall s \in S, k \in K, i \in I && (3.4) \\
\sum_{j \in J} X_{ij} &= 1 && \forall i \in I && (3.5) \\
\sum_{i \in I} X_{ij} \cdot q_i &\leq cap_j && \forall j \in J && (3.6) \\
X_{ij} &\in \{0, 1\} && \forall i \in I, j \in J && (3.7) \\
ST_k^s &\geq 0 && \forall k \in K, s \in S && (3.8) \\
H_i^s &\geq 0 && \forall i \in I, s \in S && (3.9)
\end{aligned}$$

Objective function, (3.1), consists of three terms. The first term is the storage cost of components. The second term is the penalty cost for assembly process. Finally, in the third part, purchasing cost for components is given. Constraints (3.2) and (3.3) together determine assembly process starting time. They guarantee that an assembly process cannot start before the planned starting date and before its latest component arrives. Constraint (3.4), gives the storage time for each component under each scenario. Constraint (3.5) guarantees that each component is assigned to exactly one supplier. Finally, constraint (3.6) ensures that capacity of a supplier is not exceeded. Constraints (3.8) and (3.9) are non negativity constraints for ST_k^s, process starting time, and H_i^s, holding time, respectively.

In this problem, the main focus is on the actual starting times of assembly processes. It is affected by component arrival times. Our preliminary computations show that solution time of the model gets longer with the increase in the number of components per an assembly process. Number of scenarios and suppliers also affect the computation time. Also, capacity constraints for suppliers make it harder to solve the problem. Therefore, we design a heuristic algorithm to solve large problem instances in reasonable times. In the next section, we will present the proposed heuristic algorithm.

3.3 Supplier Selection Algorithm (SSA)

We propose a heuristic algorithm which makes supplier selection decisions for a given set of components required by assembly processes already scheduled. There are a number of available suppliers with given capacities. As discussed, a number of scenarios are generated for lead times. We propose SSA to solve this problem in reasonable times.

SSA consists of two main phases as construction and improvement. In construction phase, SSA finds an initial solution for the problem. For this phase, we developed two different construction algorithms. After finding an initial solution, SSA implements a neighbourhood search procedure to find improved solutions and hopefully an optimal solution.

The neighbourhood search procedure is based on simulated annealing meta-heuristic. SSA can generate the neighbourhood of a solution by using two operators: 1-move and 2-swap. 1-move operator achieves a neighbour solution by changing supplier of a component. 2-swap operator finds a new solution by exchanging two components between two suppliers.

In neighbourhood search phase, SSA explores the neighbourhood of a solution and tries to find a better solution. If found, SSA moves to that solution and explores its neighbourhood this time. Using the simulated annealing approach, SSA may move to a worse solution, as well. In the end, SSA outputs the best solution found so far.

In addition to the notation used in mathematical formulation, below we define additional notation that will be used in SSA.

Additional Sets, Parameters

P : Set of component supplier pairs. In other words, a partial or complete solution to the problem. If all components are not assigned to a supplier we named the solution P as partial solution. If suppliers of all components are determined, then solution P is a complete solution.

P': A neighbour solution.

I_P : Set of components assigned to a supplier in a partial solution P.

$(ij)_P$: Component supplier pairs in solution P.

$d^s_{(ij)_P}$: Delivery time of component i from its assigned supplier j in solution P under scenario s.

$p_{(ij)_P}$: Price of component i from its assigned supplier j in solution P.

Notation

$\bar{D}_{ij}$: expected earliness/tardiness for a component supplier pair of assembly process k under scenario s.

$$\bar{D}_{ij} = \sum_s p_s \cdot \left| \underset{(i \in I_k)}{d^s_{ij}} - T_k \right| \qquad \forall i \in I, j \in J \tag{3.10}$$

Equation (3.10) calculates the expected deviation of lead times for component i which is ordered from supplier j under scenario s from planned starting time of assembly process k. It first take the difference between lead time and planned starting time then get the expectation of this value over scenarios. We get expected deviation for (i, j) pairs by this way.

$\bar{D}_i$: average of expected earliness/tardiness for component i over all suppliers.

$$\bar{D}_i = \left(\sum_j \bar{D}_{ij} \right) / |j| \qquad \forall i \in I \tag{3.11}$$

We get average of expected earliness/tardiness for each component i by (3.11) and use that value to sort components in the construction phase of the algorithm.

D^{max}_{ij} : maximum earliness/tardiness for a component supplier pair of assembly process k under scenario s.

$$D_{ij}^{max} \quad = \quad \max_{s} \left| \underset{(i \in I_k)}{d_{ij}^{s}} - T_k \right| \qquad \forall i \in I, j \in J \tag{3.12}$$

Equation (3.12) again calculates the deviation of lead times from planned starting time of assembly operations under each scenario and get the largest of them over scenarios. We get the largest deviation or worst case of scenarios for (i, j) pairs by this way.

$\bar{D}_i^{max}$: average of the maximum earliness/tardiness for a component i over all suppliers.

$$\bar{D}_i^{max} \quad = \quad \left(\sum_j D_{ij}^{max} \right) /|j| \qquad \forall i \in I \tag{3.13}$$

Equation (3.13), calculates the average of the largest deviation over suppliers and gives the largest deviation for all components, separately. That value is used to sort components in the construction phase of the algorithm, as well.

$(ST_k^s)^P$: expected starting time for assembly process k under scenario s in solution P.

$$(ST_k^s)^P \quad = \quad \max \left(T_k, \max_{i \in I_k \cap I_P} d_{(ij)_P}^{s} \right) \qquad \forall k \in K, \forall s \in S \tag{3.14}$$

$(ST_k^s)^P$ represents process starting time of assembly process k for solution P. It cannot start before the planned process starting time or before all components arrive.

$(H_i^s)^P$: holding time for component i in solution P under scenario s.

$$(H_i^s)^P \quad = \quad \max \left(0, (ST_k^s)^P - d_{(ij)_P}^{s} \right) \qquad \forall i \in I, \forall s \in S \tag{3.15}$$

Holding time is calculated for all components for solution P in (3.15). Components that arrive earlier than the assembly starting time are held in the inventory. The components that arrive at the process starting time are not held in inventory.

$Cost^P$: expected cost of solution P.

$$Cost^P = \sum_{i \in I_P} \sum_{s \in S} h_i \cdot q_i \cdot p^s \cdot (H_i^s)^P + \sum_{k \in K_P} \sum_{s \in S} c_k \cdot p^s \cdot (ST_k^s)^P - T_k)) + \sum_{i \in I_P} \sum_{jinJ} p_{(ij)_P} \cdot q_i \tag{3.16}$$

$Cost^{P'}$: : expected cost of solution P'.

Total cost for solution P consists of three main parts which are holding, penalty and procurement costs.

$P_+(i,j)$: solution P plus (i,j) assignment.

$\triangle^P(i,j)$: the amount of cost increase that would result if i is assigned to supplier j given a partial solution P.

$$\triangle^P(i,j) \quad = \quad Cost^{P+(i,j)} - Cost^P \tag{3.17}$$

Cost increase with an additional (i,j) pair to the solution P is represented in (3.17). In the next section we give the details of construction algorithm.

3.3.1 Construction Algorithm

Construction phase of the SSA is built by adapting a greedy approach. The aim is to minimize the difference between the planned starting time and the actual starting time of each assembly process.

At first, we list the assembly processes in descending order of penalty cost c_k and named this list as L_{ap}. Then, we sort components of each process in descending order according to $\bar{D}_i$ and named this list as L_{ik}. We assign components to suppliers according to the sequence of these lists. The aim of this sorting approach is to allocate the capacities of suppliers for the components that cause the largest delay in planned starting time of assembly processes according to $\bar{D}_i$.

Construction algorithm takes an assembly process from the list L_{ap} and in each iteration assigns a component of this process to a supplier according to list L_{ik}. After all components of the process are assigned, the algorithm moves to the next process. The steps of the algorithm are given below.

Construction Algorithm

Step 1: Initialize $P = \oslash, Cost^P = 0$

Step 2: Sort processes in decreasing order of c_k and create assembly process list L_{ap}

Step 3: Sort parts in decreasing order of $\bar{D}_i, i \in I_k$ and create component list of process k L_{ik}

Step 4:

For each $k \in L_{ap}$ do

For each $i \in L_{ik}$ do

For each $j \in J$ do

Calculate $\triangle^P(i,j)$ as $Cost^{P_+(i,j)} - Cost^P$

Check if $P_+(i,j)$ satisfies supplier capacity constraints

end

Choose the feasible (i,j) with $\min \triangle^P(i,j)$

Update P, $Cost^P$, L_{ik} and L_{ap}

$P = P_+(i,j)$

$Cost^P = Cost^{P_+(i,j)}$

$L_{ik} = L_{ik} \backslash i$

end

$L_{ap} = L_{ap} \backslash k$

end

Step 5: Return P, $Cost^P$

At the beginning of the algorithm, no supplier is selected for any component. We initialize the solution set as a null set and name it as P. Also, since there is no assignment, cost of the current solution is initialized as zero. In step 2, assembly processes are sorted in descending order according to penalty cost c_k and listed in L_{ap}. In step 3, components of each assembly process are sorted in descending order according to average expected deviation from planned starting date $(\bar{D}_i)$ and listed in L_{ik}. We first sort processes by penalty cost parameters c_k since our main concern is start an assembly process on the planned time, practically. After that we sort the components based on the deviation from the planned starting time of the assembly process $(\bar{D}_i)$ by considering both early and late deliveries of components. Then we select suppliers in Step 4. This step takes each assembly processes one by one from the list L_{ap} and takes each component i required for the assembly processes k from the list L_{ik}. Then start selecting a supplier for each component i. The supplier which gives the smallest $\triangle^P(i,j)$ is selected. Then, P, $Cost^P$ are updated and the algorithm moves to the next component in L_{ik}. If all components in L_{ik} are assigned to a

supplier then the next assembly process in L_{ap} is considered. After assigning all components of all assembly processes, the algorithm returns the final assignment and total cost resulting from this assignment in Step 5.
We propose two different construction algorithms with different sorting approaches. They differ only in Step 3. In the first algorithm, expected average deviation from the planned starting date of assembly processes $\bar{D}_i$ is used to sort components. In the second algorithm, expected average value of the maximum deviations from the planned starting date of assembly processes $\bar{D}_i^{max}$. Instead of $\bar{D}_i$, $\bar{D}_i^{max}$ is used for Step 3 of the second construction algorithm.

The output given by Step 5 is used as an initial solution by the improvement phase of SSA. Details of this phase, neighborhood search algorithm, is discussed in the next section.

3.3.2 Neighborhood Search Algorithm

Neighborhood search algorithm (NSA) starts with the initial solution generated by the construction algorithm. It tries to find an optimal solution for the problem by applying neighborhood search based on simulated annealing meta-heuristic. We used two operators to generate the neighborhood of a solution: 1-move and 2-swap.
1-move operator generates a neighborhood by changing supplier of a component. We apply this neighborhood generation method for all components.
2-swap operator generates neighborhood by exchanging suppliers of two components that assigned to different suppliers. We apply this neighbourhood generation method for each component pair assigned to different suppliers in the current solution.
We work on these neighbourhoods by applying simulated annealing meta-heuristic by Kirkpatrick [34]. It is inspired by the cooling of metals and uses cooling rate and temperature parameters. The characteristic of this meta heuristic is that it allows to select a neighbour solution with worse cost value than the current solution. Because, there is a possibility to find a better solution from the neighbourhood of this worse solution.
After generating the neighbourhood of the current solution with one of the operators, we select the one with minimum cost value. If this value is smaller than the current

solution's cost value, we update the current solution according to this neighbour solution. Then, generate new neighbourhood around this solution.

If the best neighbour solution is worse than the current solution, there is a chance to update current solution as this solution with a random probability. The probability of selecting a worse solution is getting smaller in each iteration. This probability is controlled by temperature and cooling rate parameters.

Steps of the algorithm are given below:

Step 1: Initialize P and $Cost^P$ to the solution found in construction phase.

Step 2: Initialize the following parameters.

$temperature = Cost^P$, $coolingrate = 0.05$, $result = 1$, $q = 0$

Step 3:

Do while $(q < result)$

Generate neighborhood by using the operator 1-move.

Calculate $Cost^{P'}$ for all neighbour solution P'.

Choose a feasible solution P' with min $Cost^{P'}$

If $Cost^{P'}$ > $Cost^P$ Then

$result$ = $Exp((Cost^{P'} - Cost^P)\backslash temperature)$

$temperature$ = $temperature \cdot (1 - coolingrate)$

$q = rand()$

Update P as P', $Cost^P$ as $Cost^{P'}$

Else

$result = 1$, $q = rand()$

Update P as P', $Cost^P$ as $Cost^{P'}$

end

Step 4: Return P, $Cost^P$

Step 5: Update P and $Cost^P$ according to the result of Step 4.

Step 6: Same as Step 2.

Step 7: Repeat Step 3 by using 2-swap operator.

Step 8: Return P, $Cost^P$

NSA improves the solution taken from construction phase of SSA. In Step 1, NSA takes the solution and cost of the construction algorithm as an initial solution. Parameters of simulated annealing meta heuristic are initialized in Step 2. Then we apply the meta heuristic in Step 3. In each iteration in while loop of Step 3, a neighbourhood is generated by 1-move operator. Cost of each neighbour solution P' is calculated as $Cost^{P'}$. Then, the neighbour solution P' with minimum cost is selected.

If this selected solution's cost value $Cost^{P'}$ is larger than the current solution's cost value $Cost^{P}$, simulated annealing is applied. The probability of selecting a worse solution is calculated as $result$ by using $temperature$ and $coolingrate$ parameters. It is compared by a random probability of q. $temperature$ is getting lower at each iteration and $result$ is also getting lower, consequently. Then, the probability of selecting a worse solution in terms of total cost is getting lower.

If cost value $Cost^{P'}$ of P' is lower than the current cost value $Cost^{P}$, then we update the current solution P according to P' and cost value $Cost^{P}$ according to $Cost^{P'}$. Step 3 stops when $result$ is smaller than or equal to the random probability q.

In Step 4, algorithm returns the solution and cost. Then, Step 5 uses these values as current solution and cost values. Parameters of simulated annealing meta heuristic are initialized in Step 6, same as Step 2. Step 7 is the same as the Step 3. Instead of 1-move operator, neighbourhood is generated by 2-swap operator in this step. It improves the improved solution by Step 3. Finally, NSA return the last solution and cost value in Step 8.

The performance of the mathematical model and SSA are discussed in the next Chapter.

CHAPTER 4

COMPUTATIONAL STUDY

In this chapter, we present the results of computational experiments we carried out to test the performance of the mathematical model and SSA. We solved randomly generated problem instances by using the mathematical model and implementing SSA. First, we present how we designed the experiments. Then, we provide parameters and results.

4.1 Experimental Set up

We realized in our preliminary runs that the number of components, suppliers, processes and scenarios affects the solution time. Therefore, we select them as experimental factors. Also, storage and penalty costs are defined as experimental factors, as well since there is a trade of between them. Table 4.1, includes information about experimental design parameters. The first column shows which parameters are selected as experimental factors, while the second column shows their high and low values. p_i shows the nominal price of a component. Storage and penalty costs are built based on nominal prices of components. They are uniformly generated by using average nominal price value of components.
We have six different experimental factors and two levels for each. In total, we have $2^6 = 64$ different experimental setting. We generate 5 replications for each setting so we get 320 randomly generated problem instances.

All parameters and their values are given in 4.1.

Table 4.1: Experimental Design Factors

Factor	Levels	
	Low	High
Number of components ($\|I\|$)	100	200
Number of suppliers ($\|J\|$)	10	20
Number of assembly processes ($\|K\|$)	5	10
Number of scenarios ($\|S\|$)	10	100
Storage cost (h_i)	$U\left(5 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\| \cdot 365}\right), 10 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\| \cdot 365}\right)\right)$	$U\left(10 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\| \cdot 365}\right), 20 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\| \cdot 365}\right)\right)$
Penalty cost (c_k)	$U\left(1 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\|}\right), 4 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\|}\right)\right)$	$U\left(2 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\|}\right), 8 \cdot \left(\frac{\sum_{i \in I} p_i}{\|I\|}\right)\right)$

Number of components $|I|$ is 100 and 200, respectively. Number of suppliers $|J|$ is 10 and 20, respectively. There are two levels for number of assembly processes $|K|$ as 5 and 10 processes. Also, number of scenarios is 10 and 100, as low and high level, respectively.

Nominal price p_i is uniformly generated as $U(10, 180)$. The company has an estimation for price levels of components. Nominal prices are the expected prices for components by the company. However, the actual prices differ from one supplier to another. Therefore, p_{ij} is defined as procurement cost per component i of supplier j and it is generated based on nominal price of components by the equation 4.1. Price value of dependable suppliers, have lower fluctuation in terms of lead time, is higher.

$$p_{ij} \quad = \quad p_i + (\frac{100}{|J|}) \cdot (|J| - 1) \qquad \forall j \in J \tag{4.1}$$

Daily storage cost per component, h_i, is defined as two levels. It is based on nominal price value of components, as shown in Table 4.1. It is calculated by finding average of nominal price values and dividing this value to 365 to find daily holding cost. Also, daily penalty cost per assembly process, c_k,is also based on nominal price value of components and defined as two levels in Table 4.1. Their high and low value coefficients are determined to keep the trade of between procurement cost, expected holding cost and penalty cost.

Required quantity of a component, q_i, is uniformly distributed as $U(5, 10)$.

Capacity, cap_j, is calculated according to total requirement quantity of all components by equation 4.2. Total requirement quantity is divided by the number of suppliers n_j. Since this is a very strict constraint and may result in infeasibility, we multiply it with

1.5 similar to Sawik [16] who multiplies with 2 in his study.

$$cap_j = 1.5 \cdot \sum_i q_i / n_j \tag{4.2}$$

Planned starting day for an assembly process, T_k, is 60. Probability of each scenario, p_s, is 0.1 and 0.01 respectively. Each scenario is equally likely to occur; therefore, the probability of a scenario is 0.1 for 10 scenarios and 0.01 for 100 scenarios.
Lead time of supplier j for component i under scenario s, d_{ij}^s, has triangular distribution. It's upper, lower and mean values for 10 suppliers are given in Table 4.2 and for 20 suppliers are given in Table 4.3.

Table 4.2: Triangular Distribution Parameters for 10 Suppliers

Supplier	a	b	c
1	10	100	35
2	15	115	30
3	10	110	35
4	20	120	50
5	20	120	35
6	10	200	45
7	20	280	50
8	10	290	50
9	25	300	50
10	20	300	60

We name customers as reliable if the deviation of lead times are small. In other words, the difference between the upper and lower values of triangular distribution parameters are smaller. We sort and index the suppliers from the most reliable to least reliable. The data set is sorted accordingly.

Table 4.3: Triangular Distribution Parameters for 20 Suppliers

Supplier	a	b	c	Supplier	a	b	c
1	10	100	30	11	10	200	40
2	15	100	35	12	10	200	45
3	15	115	30	13	20	280	45
4	15	115	35	14	20	280	50
5	10	110	30	15	10	290	45
6	10	110	35	16	10	290	50
7	20	120	40	17	25	300	45
8	20	120	50	18	25	300	50
9	20	120	30	19	20	300	55
10	20	120	35	20	20	300	60

4.1.1 Generating Scenarios

Lead time and scenario generation deserves more discussion. We analyze the real lead time data for 10 suppliers. We get the data for the last 2 years since some factors such as capacities, work loads, number of employees are susceptible to change. We take order release dates and order delivery dates for each supplier and each order. The difference between the order release date and order delivery date gives us the lead time for that order. Lead times are measured in days. After getting lead times for all orders of suppliers, we plot a histogram to visualize the distribution and make interpretations about it. The histogram of one of the suppliers is given in Figure 4.1.

The most frequent two lead time values are 30 days then 40 days. We can say that this supplier is likely to complete an order until 40 days by looking at this histogram. However, we need to make an assumption about the distribution of this data to generate lead time data of the supplier. The distribution looks like a triangular distribution. We apply Chi-Square Test to prove that the distribution fits to triangular distribution. There are two types of Chi-Square test. One of them is used to observe the dependence of two different data to each other. One of them is goodness of fit test. If an experimenter has a data and idea about its distribution, he/she can use Chi Square test

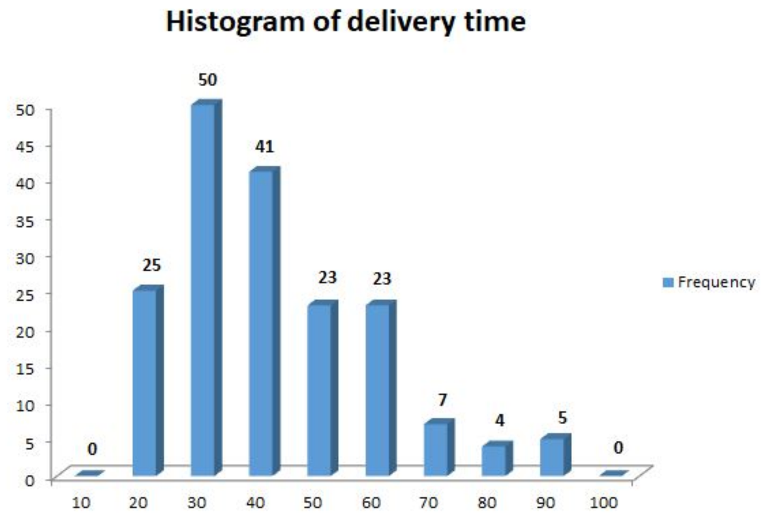

Figure 4.1: Histogram for delivery data of one of the suppliers

to prove that the hypothesis is true or false. Hence we apply Chi-Square goodness of fit test. According to histogram, our hypothesis is that delivery data fits to triangular distribution between 10 and 100. The most frequent value for this data is selected as 35. Then, we prove that our hypothesis is correct and our distribution fits to triangular distribution by applying Chi Square goodness of fit test.

We apply this test to 10 suppliers separately. Lead time data of all suppliers fit to triangular distribution. The upper, lower and peak values is different from each other. Hence, lead time data of each supplier for each scenario is generated independently by using triangular distribution according to upper, lower and mean values. The more scenarios, the better we can represent uncertainty, but in this experimental study, we work with a maximum of 100 scenarios.

Some studies in the literature apply discrete or continuous probability functions to generate lead times and scenarios. The most similar ones to other study in terms of scenario generation are Li et. al. [8] and Sawik [16]. Li et. al. [8], apply both normal and triangular distributions to generate scenarios. Sawik [16], generate delivery patterns by using historical delivery data of suppliers. Sawik [16] uses h different delivery patterns and the probability of occurrence of delivery pattern h is 1 /h. Different from his approach we do not take the past delivery data as it is. We analyse the data and see that it fits to triangular distribution then we generate delivery data according to triangular distribution. We generate $|S|$ scenarios by using these distributions.

4.2 Results

In this section, we give the experimental results of the mathematical model and SSA. Before discussing the experimental results, we provide some outputs of the model with only procurement cost concern without considering holding and penalty costs also with considering holding, penalty and procurement costs. We give the costs of the models in the first four columns. We provide holding time of components and lateness time of assemblies in the last two columns. In the model with holding, penalty and procurement costs, we represent the assembly setting. This is the model we provide in this thesis. In the second row, we try to represent a problem with only price concern not an assembly setting. We set storage and penalty cost parameters as zero in this model.

Procurement cost in the second row is smaller than the first row. Since our only concern is procurement cost, the model tries to minimize it. Also holding and lateness times are larger because their parameter values are zero and we do not have any concern as minimizing holding time or lateness time. However, we do not want to get large holding and lateness times. Our model shorten holding and lateness times by incorporating assembly system to the problem.

Table 4.4: Comparison with Assembly System

	Obj	Holding Cost	Penalty Cost	Procurement Cost	Holding Time	Lateness Time
Model with holding and penalty costs	39446	6819	4926	27700	19	5
Model with only procurement cost	118128	26558	69949	21621	74	71

According to generated parameters and experimental factors given at the beginning of this chapter, the mathematical model is solved by using cplex libraries in Java. We used a time limit of 3600 cpu seconds. The results of the model are given below in Table 4.5. The first four columns show the levels of experimental factors. The fifth column presents the average CPU time of the model in seconds. The sixth column shows the average gap. The maximum of gaps in an experimental setting is shown in the next column. In the last column, the number of optimal solutions is given by calculating percentages in each experimental setting. To illustrate, optimal solution is found in all instances generated for 100 components and 10 scenarios within the given time limit. Then percentage of optimal solutions is 100%.

Table 4.5: Results of The Mathematical Model

$\lvert I\rvert$	$\lvert J\rvert$	$\lvert K\rvert$	$\lvert S\rvert$	CPU Times	Avg Gap (%)	Max Gap (%)	Opt (%)
100	10	5	10	14.3	0	0	100
100	10	5	100	1806.0	0.43	4.55	75
100	10	10	10	8.0	0	0	100
100	10	10	100	2688.8	0.41	2.20	65
100	20	5	10	19.3	0	0	100
100	20	5	100	3106.8	1.14	2.72	20
100	20	10	10	6.3	0	0	100
100	20	10	100	3600.0	1.12	3.02	5
200	10	5	10	119.4	0	0	100
200	10	5	100	3600.0	1.65	5.36	0
200	10	10	10	38.3	0	0	100
200	10	10	100	3385.1	0.87	3.65	10
200	20	5	10	616.4	0	0	100
200	20	5	100	3600.0	3.04	4.93	0
200	20	10	10	92.0	0	0	100
200	20	10	100	3600.0	1.89	4.08	0
	Average				0.66	1.91	60.94

According to Table 4.5, we have 16 different experimental setting. The model finds feasible solution in all them and find optimal solution in 8 settings when $|S|$ is 10. The model cannot reach an optimal solution before 3600 CPU seconds in 4 settings. As said above, the model reach optimal solution in 8 settings; therefore, gap values are zero for these settings. In total, average gap is 0.66 %. Maximum gap is zero for the same 8 settings as, well and average of maximum gap is 1.91 %.

The model find optimal solution in 60.94 % of the instances. It finds optimal solution in 71 % of the instances for 100 components and 51 % of the instances for 200 components.

The number of scenarios is an important factor. If the number of scenarios increase the model hardly finds optimal solutions for 100 components and cannot reach the optimal solution for 200 components.

We see from the Table 4.5 that these 4 experimental parameters have an effect on the performance of the model. If the number of components, $|I|$, increases CPU times and gap values increase. The model can reach the optimal solution in most of the instances when the number of components is 100. However, it cannot reach the optimal solution in 3 of the experimental settings when the number of components is 200. The same situation is valid for the number of suppliers, $|J|$, as well. If $|J|$ increases CPU times and gap values increase. The percentage of the instances that the model finds optimal solutions decrease if $|J|$ increase. We can observe this by looking at 100 components cases.

The number of assembly process is an important experimental factor. If $|K|$ value increase, the data size gets larger and the problem gets harder. If the number of assembly processes increase, there are more processes to be considered. Then, an increase in terms of CPU times and gap values may be expected. However, this is not the case most of the time. Because, the model also gets easier since the number of components per an assembly process decreases if the number of assembly processes increase with the same number of components. Therefore, the gap values and CPU times may decrease even though the increase in the number of assembly processes. To illustrate, we can look at 200 components, 10 suppliers, 100 scenarios setting. When the number of assembly processes increase from 5 to 10, CPU times decrease from 3600 seconds to 3385 seconds and the percentage of optimal solutions increase from 0 to 10.

The experimental factor with the greatest impact on CPU times and gap values is the number of scenarios, $|S|$. CPU times dramatically increase if we increase $|S|$ from 10 to 100. Gap values also increase and the model cannot find any optimal solutions when $|I|$ is 200. It also cannot find optimal solution in all of the instances when $|I|$ is 100.

Cost values of the mathematical model for 200 components and 20 suppliers are given in table 4.6. First six columns show the experimental parameter values. We provide average cost values for each experimental setting in the following columns. There is a trade of between holding cost and penalty cost. When holding cost increase penalty cost decrease. When holding cost decrease penalty cost increase. There are two levels for holding and penalty cost parameters: high and low. Procurement cost is not an experimental factor.

Table 4.6: Average Costs of the Model

n_i	n_j	n_k	n_s	h_i	c_k	Obj	Holding Cost	Penalty Cost	Procurement Cost
200	20	5	10	High	Low	167358	73899	6099	87359
200	20	5	10	High	High	172626	71535	9532	91559
200	20	5	10	Low	Low	121635	31682	7465	82488
200	20	5	10	Low	High	129033	30563	10703	87767
200	20	5	100	High	Low	310182	128246	38879	143057
200	20	5	100	High	High	349723	132890	78090	138742
200	20	5	100	Low	Low	223752	61308	44305	118139
200	20	5	100	Low	High	267036	56665	80608	129763

Average objective value increase in all settings with the increase in the number of scenarios. Penalty cost increases if the number of processes increases because penalty cost incurred per an assembly process. Also, holding cost decrease since there is a trade of between penalty cost and holding cost. Average holding cost is high for high level of h_i and low for low level of h_i. The same thing is valid for penalty cost, as well. Average penalty cost high for high level of c_k and low for low level of c_k. Holding cost and penalty cost are inversely proportional. When holding cost increase, penalty cost decrease and vice versa.

The mathematical model cannot find an optimal solution for large instances within the given time limit of 3600 cpu seconds. We conduct the same experimental study for SSA to see its performance for large instances. Results of the two versions of SSA, with different construction phase, based on the same experimental setting is given in Table 4.7.

CPU times, average and minimum of gap values are given in the table. Gap is calculated by using the difference between the objective value found by the model and SSA. It is calculated as the following.

$$\left(\frac{Cost^{SSA} - Cost^{Model}}{Cost^{Model}}\right) \cdot 100 \tag{4.3}$$

As mentioned in Chapter 3, the only difference between the two versions of SSA is sorting in the construction phase. We can see from Table 4.7 that the second version of SSA is faster than the first version for all settings. We cannot make the same interpretation for gap values. In some settings the first version is better while the second

Table 4.7: Results of The SSA

				First Version			Second Version		
$\lvert I\rvert$	$\lvert J\rvert$	$\lvert K\rvert$	$\lvert S\rvert$	CPU Time	Min Gap	Avg Gap	CPU Time	Min Gap	Avg Gap
100	10	5	10	23.28	0.61	5.35	19.70	0.89	5.05
100	10	5	100	177.12	-3.67	2.57	150.09	-3.58	2.91
100	10	10	10	27.71	0.16	4.13	20.30	1.65	4.62
100	10	10	100	182.17	-2.19	1.98	148.05	-1.38	2.62
100	20	5	10	42.42	1.18	4.68	38.55	1.75	4.99
100	20	5	100	398.39	-2.23	3.24	308.93	-2.90	3.31
100	20	10	10	49.86	2.22	5.13	36.62	2.02	5.97
100	20	10	100	457.53	-2.10	3.24	306.62	-2.18	2.54
200	10	5	10	260.19	1.65	4.99	201.11	1.12	5.18
200	10	5	100	1631.17	-3.33	1.02	1374.98	-3.76	0.90
200	10	10	10	261.26	1.84	4.20	230.23	1.37	4.02
200	10	10	100	1840.36	-1.84	2.22	1359.91	-1.39	2.21
200	20	5	10	409.67	0.095	3.99	354.01	0.61	3.81
200	20	5	100	3600	-2.44	-0.98	3600	-3.20	-1.15
200	20	10	10	438.88	0.54	3.96	389.55	1.14	4.06
200	20	10	100	3600	-2.42	-0.83	3600	-2.19	-0.84

better in some. We select the second version of SSA to compare with the model by looking at the CPU times because we developed these algorithms to solve the problem faster than the mathematical model.

Mathematical model is good at solving the problem with 10 scenarios; hence, CPU times of the model is smaller than SSA when $|S|$ is 10. Also, SSA cannot find better solutions when the number of scenarios is 10 since the model can find the optimal solution for all of the instances. Average and minimum gap values are positive and larger when compared to the settings with 100 scenarios. The improvement in terms of CPU times in the results of SSA can be easily noticed when $|S|$ is 100. It is fifteen times faster than the model when $|I|$ is 20, $|J|$ is 10 and ten times faster than the model when $|I|$ is 20, $|J|$ is 20. SSA is three times faster than the model when $|I|$ is 200 and $|S|$ is 100. The model cannot find optimal solution for this setting and runs with 3600 seconds time limit. CPU times of SSA also starts increasing when the number of components is 200 and scenarios is 100 because the problem size is very large. If $|J|$ is 20, CPU times of SSA exceed 3600 seconds and the average is around 4000 seconds. We set a time limitation for this instance not to run the SSA longer than the mathematical model. Although CPU times are large, SSA can find better solutions than the model within 3600 seconds in average within 3600 seconds for 200 components, 20 suppliers and 100 scenarios case. To conclude, SSA can be used with 3600 seconds time limitation instead of the mathematical model.

SSA also spends a considerable amount of time to solve the problem with large instances when $|I|$ is 200, $|J|$ is 20, $|K|$ is 10 and $|S|$ is 100. SSA spends most of the time with 2-swap operator since the algorithm search a larger neighbourhood with this operator. If we run SSA with only 1-move operator, we can improve the solution time values but we have to sacrifice solution quality. The CPU times and gap values between the mathematical model and SSA is given in Table 4.8. In the first three columns, gap values are given for construction algorithm, 1-move and 2-swap operators, respectively. These gap values are calculated by using equation 4.3 given above. The last two columns show CPU times spent until 2-swap operator and with 2-swap operator, respectively. In the first row, average gap values are shown while in the second row minimum gap values are given. After analyzing these gap values, we saw how effective 2-swap operator to improve the solution quality and we did not give up with that operator.

Table 4.8: Time and Gap Analysis for SSA

	Construction Gap	1-Move Gap	2-Swap Gap	Time Until 2-Swap	2-Swap Time
Average	6.03	2.36	-0.84	1200	2400
Minimum	1.11	-1.85	-2.19	1200	2400

CHAPTER 5

CONCLUSION

In this thesis, we study on a supplier selection problem for the component requirements of assembly operations in an aircraft company.

All components of an assembly operation may not arrive exactly at the planned starting date of the operation. Components delivered after the planned starting time of an assembly process cause delays in the completion of the assembly process. Then, a penalty cost occurs. Also, components delivered before the actual starting time of the assembly process have to be stored until the assembly process starts. Then, storage cost occurs. Storing these components is costly and it is hard to handle the storage since they are usually large in size. We also take procurement cost into consideration. We propose a linear mixed integer stochastic model to solve the problem.

Supplier selection problem attracts a lot of attention in the last decades and there are lots of studies. Both deterministic and stochastic problems are available in the literature. There are studies that consider assembly requirements. However, they consider only the required quantities of components. Penalty costs are component based. Most of them are inventory control problems. To the best of our knowledge, they missed the dependence of components to each other and have not any assembly operation concern.

We design an experimental setting to discuss the performance of our mathematical model. We see that the model solves practical size problems with small number of scenarios. For large number of scenarios, it achieves reasonable gaps. Then, we design a meta-heuristic, SSA, to solve the problem in reasonable times.

SSA is not really fast when the number of components, suppliers and scenarios are at the high level in the experimental setting. However, it succeeds to find better solutions in that setting. This is important because we need high number of scenarios to represent uncertainty in a realistic way.

As a future study, another additional experimental design can be conducted. Also, SSA can be improved since it is not really fast. We do not consider quantity discounts in this study. It may taken into consideration in further studies, as well.

REFERENCES

[1] M. Alkahtani and H. Kaid, "Supplier selection in supply chain management: a review study," *International Journal of Business Performance and Supply Chain Modelling*, vol. 10, no. 2, pp. 107–130, 2018.

[2] M. A. Naqvi and S. H. Amin, "Supplier selection and order allocation: a literature review," *Journal of Data, Information and Management*, pp. 1–15, 2021.

[3] R. Hammami, C. Temponi, and Y. Frein, "A scenario-based stochastic model for supplier selection in global context with multiple buyers, currency fluctuation uncertainties, and price discounts," *European Journal of Operational Research*, vol. 233, no. 1, pp. 159–170, 2014.

[4] A. Şen, H. Yaman, K. Güler, and E. Körpeoğlu, "Multi-period supplier selection under price uncertainty," *Journal of the Operational Research Society*, vol. 65, no. 11, pp. 1636–1648, 2014.

[5] A. Awasthi, S. Chauhan, S. Goyal, and J.-M. Proth, "Supplier selection problem for a single manufacturing unit under stochastic demand," *International Journal of Production Economics*, vol. 117, no. 1, pp. 229–233, 2009.

[6] D. Manerba, R. Mansini, and G. Perboli, "The capacitated supplier selection problem with total quantity discount policy and activation costs under uncertainty," *International Journal of Production Economics*, vol. 198, pp. 119–132, 2018.

[7] J.-l. Zhang and J. Chen, "Supplier selection and procurement decisions with uncertain demand, fixed selection costs and quantity discounts," *Computers & Operations Research*, vol. 40, no. 11, pp. 2703–2710, 2013.

[8] L. Li and Z. B. Zabinsky, "Incorporating uncertainty into a supplier selection problem," *International Journal of Production Economics*, vol. 134, no. 2, pp. 344–356, 2011.

[9] R. Aggarwal and S. Singh, “Chance constraint-based multi-objective stochastic model for supplier selection,” *The International Journal of Advanced Manufacturing Technology*, vol. 79, no. 9, pp. 1707–1719, 2015.

[10] R. U. Bilsel and A. Ravindran, “A multiobjective chance constrained programming model for supplier selection under uncertainty,” *Transportation Research Part B: Methodological*, vol. 45, no. 8, pp. 1284–1300, 2011.

[11] H. Gurnani, R. Akella, and J. Lehoczky, “Supply management in assembly systems with random yield and random demand,” *IIE transactions*, vol. 32, no. 8, pp. 701–714, 2000.

[12] H. Gurnani, R. Akella, and J. Lehoczky, “Optimal order policies in assembly systems with random demand and random supplier delivery,” *IIE transactions*, vol. 28, no. 11, pp. 865–878, 1996.

[13] R. Anupindi and R. Akella, “Diversification under supply uncertainty,” *Management science*, vol. 39, no. 8, pp. 944–963, 1993.

[14] S. Abginehchi and R. Z. Farahani, “Modeling and analysis for determining optimal suppliers under stochastic lead times,” *Applied Mathematical Modelling*, vol. 34, no. 5, pp. 1311–1328, 2010.

[15] H. Assellaou, B. Ouhbi, and B. Frikh, “Multi-objective programming for supplier selection and order allocation under disruption risk and demand, quality, and delay time uncertainties,” *International Journal of Business Analytics (IJBAN)*, vol. 5, no. 2, pp. 30–56, 2018.

[16] T. Sawik, “Single vs. multiple objective supplier selection in a make to order environment,” *Omega*, vol. 38, no. 3-4, pp. 203–212, 2010.

[17] T. Sawik, “Supplier selection in make-to-order environment with risks,” *Mathematical and Computer Modelling*, vol. 53, no. 9-10, pp. 1670–1679, 2011.

[18] T. Sawik, “Joint supplier selection and scheduling of customer orders under disruption risks: Single vs. dual sourcing,” *Omega*, vol. 43, pp. 83–95, 2014.

[19] T. Sawik, "Selection of a dynamic supply portfolio in make-to-order environment withrisks," *Computers & Operations Research*, vol. 38, no. 4, pp. 782–796, 2011.

[20] O. B. Ammar, F. Hnaien, H. Marian, and A. Dolgui, "Optimization approaches for multi-level assembly systems under stochastic lead times," in *Metaheuristics for Production Systems*, pp. 93–107, Springer, 2016.

[21] A. Dolgui and C. Prodhon, "Supply planning under uncertainties in mrp environments: A state of the art," *Annual Reviews in Control*, vol. 31, no. 2, pp. 269–279, 2007.

[22] C. A. Yano, "Setting planned leadtimes in serial production systems with tardiness costs," *Management science*, vol. 33, no. 1, pp. 95–106, 1987.

[23] F. Hnaien, X. Delorme, and A. Dolgui, "Multi-objective optimization for inventory control in two-level assembly systems under uncertainty of lead times," *Computers & operations research*, vol. 37, no. 11, pp. 1835–1843, 2010.

[24] S. Fallah-Jamshidi, N. Karimi, and M. Zandieh, "A hybrid multi-objective genetic algorithm for planning order release date in two-level assembly system with random lead times," *Expert Systems with Applications*, vol. 38, no. 11, pp. 13549–13554, 2011.

[25] F. Hnaien, A. Dolgui, and D. D. Wu, "Single-period inventory model for one-level assembly system with stochastic lead times and demand," *International Journal of Production Research*, vol. 54, no. 1, pp. 186–203, 2016.

[26] O. Ben-Ammar, B. Bettayeb, and A. Dolgui, "Optimization of multi-period supply planning under stochastic lead times and a dynamic demand," *International Journal of Production Economics*, vol. 218, pp. 106–117, 2019.

[27] A. Dolgui and M.-A. Ould-Louly, "A model for supply planning under lead time uncertainty," *International Journal of Production Economics*, vol. 78, no. 2, pp. 145–152, 2002.

[28] O. B. Ammar, H. Marian, D. Wu, and A. Dolgui, "Mathematical model for supply planning of multi-level assembly systems with stochastic lead times," *IFAC Proceedings Volumes*, vol. 46, no. 9, pp. 389–394, 2013.

[29] O. Ben-Ammar, A. Dolgui, and D. D. Wu, “Planned lead times optimization for multi-level assembly systems under uncertainties,” *Omega*, vol. 78, pp. 39–56, 2018.

[30] C. Chu, J.-M. Proth, and X. Xie, “Supply management in assembly systems,” *Naval research logistics (NRL)*, vol. 40, no. 7, pp. 933–949, 1993.

[31] M.-A. Louly, A. Dolgui, and F. Hnaien, “Optimal supply planning in mrp environments for assembly systems with random component procurement times,” *International Journal of Production Research*, vol. 46, no. 19, pp. 5441–5467, 2008.

[32] O. Tang and R. W. Grubbström, “The detailed coordination problem in a two-level assembly system with stochastic lead times,” *International journal of production economics*, vol. 81, pp. 415–429, 2003.

[33] J.-S. Song, C. A. Yano, and P. Lerssrisuriya, “Contract assembly: Dealing with combined supply lead time and demand quantity uncertainty,” *Manufacturing & Service Operations Management*, vol. 2, no. 3, pp. 287–296, 2000.

[34] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated annealing,” *science*, vol. 220, no. 4598, pp. 671–680, 1983.