PREVENTING CONFIDENTIAL INFORMATION LEAKAGE IN SUPPLY CHAINS THROUGH TRUST-BASED HEURISTIC SUPPLIER SELECTION

NAFISA KHUNDKER

A THESIS

IN

THE DEPARTMENT

OF

CONCORDIA INSTITUTE FOR INFORMATION SYSTEMS ENGINEERING

PRESENTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF APPLIED SCIENCE (INFORMATION SYSTEMS SECURITY)

CONCORDIA UNIVERSITY

MONTRÉAL, QUÉBEC, CANADA

AUGUST 2014

CONCORDIA UNIVERSITY
School of Graduate Studies

This is to certify that the thesis prepared

By: **Naf sa Khundker**

Entitled: **Preventing Conf dential Information Leakage in Supply Chains Through Trust-Based Heuristic Supplier Selection**

and submitted in partial fulf llment of the requirements for the degree of

Master of Applied Science (Information Systems Security)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the f nal examining commitee:

_____ Dr. A. Ben Hamza (Chair)

_____ Dr. Lingyu Wang (Supervisor)

_____ Dr. Yong Zeng (Co-supervisor)

_____ Dr. Anjali Awasthi (CIISE Examiner)

_____ Dr. Akif A. Bulgak (External Examiner (MIE))

Approved _____
Chair of Department or Graduate Program Director

_____ 20 _____ _____

Christopher W. Trueman, Interim Dean
Faculty of Engineering and Computer Science

Abstract

Preventing Conf dential Information Leakage in Supply Chains Through Trust-Based Heuristic Supplier Selection

Naf sa Khundker

In today's global economy, outsourcing has become increasingly popular in design and production. Manufacturers need to share massive information with their suppliers during an outsourcing activity. In the meantime, the manufacturers need to protect conf dential information related to the product for the purpose of intellectual property (IP) protection. In such a context, secure collaboration has become an emerging research topic in global supply chain management. A number of methods were proposed to select secure and eligible suppliers among all potential suppliers involved in a supply chain system to satisfy security requirements and to minimize the cost at the same time. The selection can be performed by assessing suppliers' ability, risk assessment of information leakage, and cost analysis. However, depending on given security requirements, a valid selection of suppliers to meet such requirements may not always be possible to obtain. Moreover, such a selection process is usually very expensive with existing risk assessment algorithms. This thesis addresses both issues by proposing a method which is both secure and eff cient for generating optimal selections of suppliers for a supply chain system. First, we introduce a multi-level trust model of suppliers to address the cases where existing approaches based on a f at trust model will fail to generate any valid supplier selection. To our best knowledge this is the f rst work on formally modeling the level of trust in suppliers. Second, we propose eff cient heuristic algorithms for eliminating insecure selections of suppliers as early as possible in the process such that the need for expensive risk assessment on such selections is avoided. The effectiveness and eff ciency of proposed approaches were analyzed and validated through a case study.

Acknowledgments

In the name of Allah, the Most Gracious and the Most Merciful

Alhamdulillah, all praises to Allah, the most gracious the most giving, for the strengths and His blessing in completing this thesis.

Special appreciation goes to my supervisor, Dr. Lingyu Wang for his inspirational advice and constant support and my appreciation to my co-supervisor, Dr. Yong Zeng for his support and knowledge regarding this topic.

I owe special thanks to my colleagues Mengyuan Zhang, Wenming Liu, William Nzoukou Tankou, Mickael Emirkanian-Bouchard, Mina Khalili, Paria Shirani, Dr. Xiao Guang Deng for their insightful comments in completing the thesis.

I am full of praise for my friends Nazmun Nahar Bhuiyan, Md. Istiaque Shahriar, Rezwana Kursia, Monalia Sadia, Md. Arifuzzaman, Razia Sultana, Anamul Haque for their support, suggestions, inspiration and their help in times of distress and need.

Last but not least, my deepest gratitude goes to my beloved parents Mr. Khundker Ali Nasim and Mrs. Roushan Ara, my husband Rubayat Muntasir, my son Radiyan Muntasir and other members in my family for their endless love, prayers, support and encouragement. Without their great sacrif ce and inspiration, this would not have come to end.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Facing intensive global competition, today many manufacturers outsource their products and services to suppliers for the benef ts of a lower cost and higher f exibility [35]. In a supply chain-based system, a large number of suppliers are usually involved initially where each supplier is eligible to supply one or more products, components of a product, or other tasks within the chain. During outsourcing activity, a focal manufacturer has to share a large amount of product information to its suppliers in order to help them to complete the design and manufacturing tasks. Some of the suppliers are also potential competitors, or serving the competitiors of, the manufacturer. Therefore, a manufacturer must try to conceal any conf dential information about product design from such competitors.

However, even when conf dential information is not directly shared, it may still be acquired through information leakage. Moreover, it is often possible for competitors to gather information from different sources about a product and consequently infer conf dential information out of available non-conf dential information, through relationships existing between information about different parts of a product. Therefore, a manufacturer must use appropriate techniques for sharing information with suppliers in order to protect conf dential information from potential leakages or inferences. There are two seemingly conf icting goals here. First, the manufacturer needs to facilitate outsourcing by sharing suff cient information with suppliers. Second, the manufacturer must also conceal conf dential information from competitors who may infer, e.g., sensitive parameters based on the logical

dependency of information inside a system.

To this end, research already exist on secure supplier selection for the purpose of minimizing information leakage in a supply chain. In this thesis, we improve existing methods in selecting suppliers to minimize information leakage in a supply chain. Specif cally, the general security issues in supply chain in the context of information sharing are described in [48]. In another existing work, Zhang et al. has evaluated the case of information leakage caused by inferences, where sharing information in supply chain may allow information to unintentionally f ow from its owner to the competitor, which is generally known as information leakage by inference [50] . In their research, the authors showed that it is possible to calculate the risk of information leakage in a supply chain, which is related to the amount of information being shared by the manufacturer with the other party, as well as with the logical dependencies that exist among different parts of a product. In a later work, Zhang et al. proposed a method to minimize risk of information leakage in a two-level supply chain, where manufacturer is in one level and suppliers in the other, by selecting optimum suppliers [49]. Building upon such existing work, we focus on proposing a more general and improved solution to the supplier selection problem using the novel concept of trust primitive between the manufacturer and suppliers, as well as eff cient supplier selection methods to minimize the calculation complexity.

1.1 Motivation

Since our solution aims to address limitations of existing work on modeling the risk of information leakage and selecting suppliers [50, 49], we f rst illustrate necessary concepts and techniques in those work that will be required for further discussions. This section focuses on a running example based on a natural gas dryer to illustrate how to select suppliers using the risk information in order to provide motivation to our work.

The supplier selection process starts with analyzing the product structure tree that shows how different components of the product are related to each other[49]. An example of product structure tree is shown in Figure 1. This self-explanatory example demonstrates

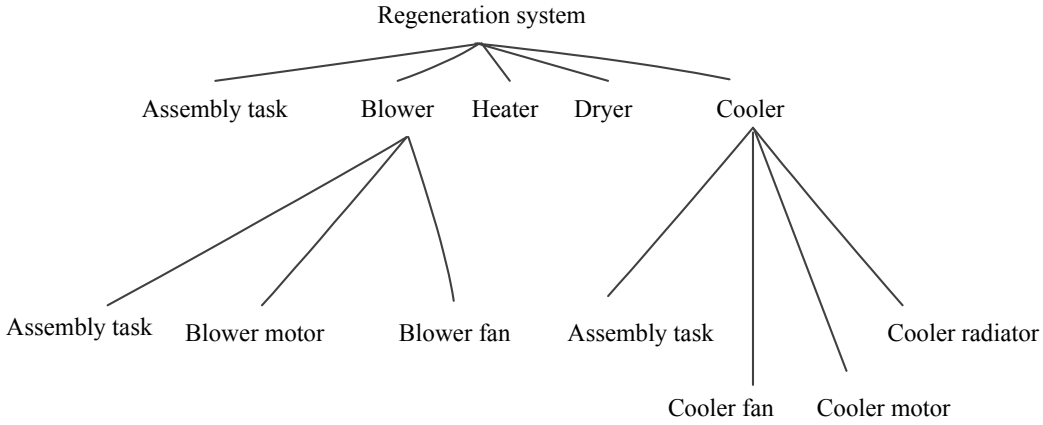

Figure 1: Product structure tree [49]

how a product can be decomposed into its components and further the components in sub-components. According to the authors of [49], when a component is shared with a supplier, all its components should also be shared. Consequently, all the relevant information about these components needs to be shared with that supplier. Therefore, in this section, the example will be based on the components blower, heater, dryer, and cooler, without explicitly mentioning further about the sub-components blower motor, blower fan, cooler fan, cooler motor and cooler radiator as these are considered together with the parent components.

Another important concept is the supplier capability function which shows the suppliers' capabilities of supplying components [49]. The supplier capability function is shown in Table 1.

Table 1: Example of supplier capability function [49]

Supplier s	$F_{sc}(s)$
s_0	n_1, n_4
s_1	n_2, n_3, n_5
s_2	n_2
s_3	n_3
s_4	n_5

Using such information as inputs, all possible allocations can be calculated as shown in Table 2 [49]. In any particular allocation, one or more components are allocated to a supplier. Practically, to allocate the components, the manufacturer will need to share

relevant information with that supplier. Utilizing relevant information about the supplier, the manufacturer can then calculate the risk of information leakage of that supplier for that specif c allocation. In such a way, the manufacturer can calculate the information leakage risk of all suppliers in all allocations[49].

Table 2: Example of Allocation of components [49]

Allocation	n_1	n_2	n_3	n_4	n_5
A_1	s_0	s_1	s_1	s_0	s_1
A_2	s_0	s_1	s_1	s_0	s_4
A_3	s_0	s_1	s_3	s_0	s_1
A_4	s_0	s_1	s_3	s_0	s_4
A_5	s_0	s_2	s_1	s_0	s_1
A_6	s_0	s_2	s_1	s_0	s_4
A_7	s_0	s_2	s_3	s_0	s_1
A_8	s_0	s_2	s_3	s_0	s_4

For a given component and supplier pair, we can pre-def ne a risk threshold which will help to decide whether information about this particular component can be shared with this specif c supplier [49]. If the amount of information leakage is lower than the pre-def ned risk threshold, then the information sharing can be considered as safe. Otherwise if the risk information leakage is higher than the risk threshold, then the information sharing is considered as unsafe and will not allowed. An example depicting the risk of information leakage of the suppliers in all allocations are shown in Table 3 (details regarding the risk calculation can be found in [49] and is obmitted here). The corresponding risk threshold is shown in Table 4.

Finding an optimum allocation requires to satisfy two requirements. First, the allocation should be safe which means all information sharing to each supplier in that allocation should be safe. Second, the allocation should have the least cost. Table 5 shows the cost information of each supplier.

Specif cally, f rst, the risk factor must be taken into consideration and the goal is thus to f nd safe allocations. For this purpose, we compare the risk of information leakage of each supplier with the risk threshold for that specif c supplier. In this way, we f nd for a specif c allocation whether the information sharing with all suppliers is safe. Thus, we can f nd safe

Table 3: Risk of information leakage to suppliers [49]

Allocation	s_1	s_2	s_3	s_4
A_1	100	1.82	2.07	2.77
A_2	9.56	1.82	2.07	3.17
A_3	100	1.82	2.14	2.77
A_4	5.14	1.82	2.14	3.17
A_5	3.04	1.18	2.07	2.77
A_6	3.32	1.18	2.07	3.17
A_7	2.73	1.18	2.14	2.77
A_8	2.93	1.18	2.14	3.17

Table 4: Risk threshold [49]

	s_1	s_2	s_3	s_4
Threshold	5	10	10	10

allocations in which any information sharing with any of the suppliers is safe. Accordingly, we f nd that allocation A_5, A_6, A_7 and A_8 is safe. Note here an important fact about such an approach [49], that is, we have to calculate risk for every possible allocation, which will lead to a prohibitive cost, as we will show later.

Then, we also need to consider the second factor, the cost. Analyzing the information given about cost here in this case, we can see allocation A_5 has the least cost. As a result, our desired optimum allocation is allocation A_5.

Clearly, the above approach [49] is straightforward and effective. The authors give a complete solution covering risk calculation, product decomposition and constructing allocations, and f nally f nding an optimum choice. However, we now take a closer look at this approach and identify several limitations, which we will tackle in the remainder of this thesis.

Table 5: Components ,suppliers, and costs [49]

	s_1	s_2	s_3	s_4
n_2	2	3	100	100
n_3	2	100	3	100
n_5	2	100	100	3

- First of all, the above approach is designed for a two-level supply chain, composed of the fully trusted manufacturer and the equally trusted suppliers. This is a clear limitation since in practice it is not always the case that all suppliers receive exactly the same level of trust.

- Secondly, a component with all its sub-components must be allocated to the same supplier according to the approach. However, from the cost perspective one supplier may not be able to provide the best price for all the sub-components of a component, which means this is a unnecessary requirement and should be relaxed.

- Finally, the number of allocation depends on suppliers' capabilities and the product structure tree. For a supply chain having hundreds of components and/or a large number of suppliers, the number of possible allocations increase exponentially and it quickly becomes prohititive to perform risk calculation on all such allocations.

To illustrate those limitations, we assume the risks and risk thresholds are slightly different, as shown in Table 6 and 7.

Table 6: Risk of information leakage

Allocation	s_1	s_2	s_3	s_4
A_1	100.00	4.04	4.04	4.04
A_2	19.82	4.04	4.04	3.84
A_3	100.00	4.04	3.23	4.04
A_4	3.93	4.04	3.23	3.84
A_5	3.03	3.39	4.04	4.04
A_6	3.23	3.39	4.04	4.04
A_7	3.84	3.39	3.23	4.04
A_8	4.04	3.39	3.23	3.84

Table 7: Risk threshold

	s_1	s_2	s_3	s_4
$Threshold$	4	4	4	4

In this example, none of the allocations are safe when compared to the threshold values. Specif cally, in Table 6 and 7, if we compare the risk threshold with the risk of information

leakage, we can see that there is no safe choice at all, because the risk of information leakage to some suppliers in all allocations are higher than the threshold. Such a case may certainly be unavoidable in practice when the desired thresholds do not leave any safe choices at all.

Under existing approaches, in such a situation, when there is no safe choices left, the manufacturer will naturally have to do all the work by itself or at least take care of some of the tasks. For allocation A_8 in this example, manufacturer has to do the job of s_1 to prevent information leakage in the system. One way to understand this situation is to regard it essentially as a two-level supply chain system, where the manufacturer is regarded as a special supplier who enjoys a special, high level of trust which is higher than all the other suppliers, such that this special supplier can be trusted when others cannot.

Our innovation in this thesis is to generalize such a two level special case into a more general model, where the suppliers may be given different levels of trust. In such a multi-level supplier model, some suppliers are more trustworthy than others. In this case, the manufacture can distribute less critical (sub-)components to suppliers with lower trust levels, and these low trust level suppliers can then hand over their finished components to suppliers with a higher trust level, who in turn will finish the more sensitive work, e.g., refining or assembling these components, for the manufacturer. In such a way, it will also be possible to distribute different sub-components of a component to different suppliers, which eventually will allow optimizing the cost.

Moreover, as we have demonstrated earlier in this chapter, the existing methods of secure selection of suppliers require the calculation of all possible allocations first, which is very time consuming. In this thesis, we aim to make the selection process based on the multi-level trust model more efficient.

Our proposed solution is heuristic. The word 'Heuristic' refers to experience-based techniques for problem solving, learning, and discovery that give a solution which is not guaranteed to be optimal. In our method, when we identify one unsafe assignment, we consider this as an experience and use this experience to find other unsafe assignments and eventually we identify all unsafe assignments and remove them to obtain a set of allocations

where all allocations are safe. Hence all the safe allocations are not necessarily optimum allocation, in the next step we do a cost f ltering to f nd the optimum allocation. In this way our method is heuristic in nature.

1.2 Contributions

In this thesis, we f rst demonstrate several important limitations of the existing work [50, 49] in preventing conf dential information leakage through inferences. We show that, although this method has been successfully developed, validated and applied in aerospace product outsourcing management, it still has several practical limitations that deserve further research efforts. We demonstrate the incomplete protection problem in which given risk thresholds may never be achieved no matter how we partition the product information among suppliers. Furthermore, we demonstrate that the procedure of supplier selection is very time-consuming while considering a full collection of possible allocations. This problem is particularly critical when dealing with a large scale data of suppliers and products in real world.

Consequently, this thesis proposes the novel concept of multi-level supplier model based on different levels of trust in a supply chain. This new concept of different trust levels described later in this thesis will allow manufacturers to distribute parts of components among two or more suppliers for partial manufacturing and ref nement, allowing to assign more crucial components to more trustworthy supplier and vice versa. In this way, we provide a more realistic model of real world scenarios which in turns leads to better security and lower cost. We also aim to improve the eff ciency of supplier selection procedure. For this purpose, we propose a supplier elimination mechanism to heuristically f lter out “unsafe” assignments in order to reduce the computational cost for performing risk assessment. In summary, our contributions are as follows.

1. Introducing the concept of trust level in secure supplier selection in order to more realisticlally model real world supply chains.

2. Designing supplier selection solutions based on the multi-level model of suppliers in order to achieve optimal selection.

3. Proposing algorithms to eff ciently select suppliers so that the selection process becomes cost effective and time eff cient.

1.3 Thesis Organization

This thesis is organized as follows. Chapter 2 reviews the related work. Chapter 3 will introduce the preliminaries and the model. In Chapter 4 the basic building blocks of unsafe assignment identif cation and the idea of identif cation mechanism is presented. Chapter 5 will present three supplier elimination approaches. The effectiveness of the proposed approaches will be discussed in Section 6. The conclusions and future work will be given in Section 7.

Chapter 2

Related Work

Globalization benef ts industries with cost reduction and time saving production, but at the same time, industries that are involved in globalized production face the challenge of securing conf dential information from its competitors. Selecting suppliers can depend on many constrains, for example, risk, cost, quality [23]. Selected suppliers can share information horizontally or vertically (def nitions will be given later) among themselves and with manufacturers. However, the receiver of information will always try to get as much information as possible. On the other hand, the owner of information wants to conceal private information for the sake of business competition. Some shared information can be protected by access control, policies, laws, etc. Nonetheless, there could still be information leakage to the other party. Information leakage occurs due to inference where competitors do not get information directly from the information owner, but can indirectly infer the private information from other more covert channels. Properly partitioning product information can not always provide suff cient protection of the information since shared information unavoidably contains partial core proprietary knowledge and know-how [3].

More specif cally, in supply chains, since some of the suppliers are also potential competitors or may have partnership with other competitors, focal manufacturer's intellectual property (IP) might be leaked during outsourcing activities [50]. This type of leakage of information is usually referred as indirect information f ow. Focal manufacturer thus faces a double-edged challenge to balance "collaboration" and "security" in order to achieve the

best competitiveness in global economy [19]. For the sake of better production manufacturers need to share information as demanded by the suppliers. Practically, suppliers would ask for as much detailed information, including conf dential information, as necessary. However, for security reasons the manufacturer would have to conceal conf dential information. In such a context, secure collaboration in global supply chain management becomes an emergent research topic in recent years [48]. In the following, we review related work in several domains. First, we discuss about information sharing and leakage in supply chain. Then, we focus on existing models and solutions for supplier selection, supply chain risk management, and information leakage in supply chain.

2.1 Information Sharing in Supply Chain

Supply chain collaboration primarily focuses on information sharing among partners to create synergies for competitive advantage. Therefore, most existing research emphasizes on supply chain collaboration process modeling and information sharing [52]. The former provides a mechanism to help partners to collaboratively plan, forecast and manage supply chain activities.

Collaborative Planning, Forecasting and Replenishment model [44] is a representative solution of this issue. This solution is often used to induce collaboration and coordination through information sharing between supply chain partners. Moreover, several researchers have modeled the supply chain collaboration process using various theories and technologies.

For example, Fawcett et al. [12] developed a three-stage implementation model in order to manage the dynamic and changing collaboration process based on organizational theories. According to them, the three stages are “create commitment and SC understanding” , “remove resisting forces to SC collaboration to change culture and practice” , and “continuously improve collaboration capabilities”.

Zou and Yu [52] built a model driven decision support system to simulate the collaboration process using articial intelligence techniques. Their intelligent process simulation

model works in three layers : Problems input layer, data processing layer and solving output layer. The working principle is divided into these steps: issues recognition and organization management, case selection and model coordination, solution and implementation.

The information f ow in a supply chain varies in a number of ways. It can be vertical information f ow or horizontal information f ow. Vertical information f ow refers to the situation when the competition exits between suppliers and manufacturers where one of them is in upstream and another is in downstream. On the other hand, horizontal competition exists between two retailers or two manufacturers who may not sharing information directly. For the former case, two retailers can be suppliers for the same manufacturer , or for the later case, one retailer can be supplier for both manufacturers. Li [30] addressed this problem in his thesis where a three step information sharing strategy is discussed for this situation.

Goyal [3] showed in his thesis information management under leakage in a supply chain. The thesis f nds that horizontal competition and information sharing that happens between two retailers can be affected by two factors: cost and level of demand uncertainty. Zhang [51] addresses the issue where two retailers are competitors and do not share information with each other. But as their information f ow within the supply chain toward the manufacturer, there is a possibility for unintentional f ow of information between two of them through the manufacturer. In this case, no information is voluntarily shared among potential competitors.

2.2 Supplier Selection Problem

The most critical issue in supplier selection is the criteria and techniques for making the optimal selection. The supplier selection problem has in fact been examined for a long time. Kubat and Yuce [23] considered supplier selection as a decision-making problem with many constraints such as cost, quality, risk and so on. The supplier selection process is done by determining each supplier's weight by identif ed factors, and then the best

supplier is determined. De Boer er al. [8] present a review of decision methods of supplier selection. Keeping both the diversity of situations of selecting suppliers and different phases of supplier selection, their framework reviews the supplier selection methods. They showed that supplier selection models, in a large extent, can be categorized into single-deal and multi-deal. Single-deal where the models select suppliers for one product or a group of products at a time. The other case is based on inter dependencies of products. The other supplier selection models are based on if there is any inventory management over time or not. A third criteria involves techniques used on choices for supplier selection.

Aissaoui et al. [2] focus on the f nal selection stage that consists of determining the best mixture of vendors and allocating orders among them so as to satisfy different purchasing requirements, operations research and computational models. Their proposed decision models includes single or multiple sourcing, criteria, items, periods, objectives, etc. Dickson [10] identif ed 23 vendor selection criteria and ranked them. Quality, delivery, performance history, warranties and claim policies, production facilities and capacity, price, technical capability and f nancial position are found to be the most important in this study. The study was done based on a survey of purchasing agents and managers.

Based on the extensive study of supplier selection problem Ho et al. reviewed the problem of supplier selection and provides evidence that the multi-criteria decision making approaches are better than the traditional cost-based approach [18]. There are individual approaches and integrated approaches for supplier selection approaches. Their study found that multi-criteria based decision making for supplier selection is more effective than the other one. Different from most existing work, in this thesis, we focus on limiting information leakage in supplier selection.

2.3 Supply Chain Risk Management

Supply Chain Risk Management (SCRM) is a growing research area. It is a vital issue in the context of Supply Chain Risk Management (SCRM). Four basic constructors and

critical aspects of SCRM are identif ed by Juttner et al. which are risk source, risk consequence, risk driver and risk mitigating [21]. SCRM is def ned from different aspects in literature. Supply Chain Management (SCM) is considered as “a formal process that involves identifying potential losses, understanding the likelihood of potential losses, and assigning signif cance to these losses” in supply chains [16]. However, it can be also def ned as “the identif cation and management of risk for the supply chain,through acoordinated approach amongst supply chain members,to reduce supply chain vulnerability as a whole” [20].

Supply chain risk management can be categorized in different ways. Depending on the source of risk, according to Lockamy and McCormack [31], the risk in supply chain can be operational, network and external. When the risk comes from internal people, system or processes by which the system operates, then it is an operational risk. The network risk is related to suppliers network. In this case, information can be leaked to a competitor supplier directly or indirectly through other intermediate suppliers. However, external risk arises due to external factors of the system. According to the authors, the methodology using this constrains can facilitate outsourcing decision. Their methodology analyze risk by developing a risk prof le which includes suppliers different risk probabilities and the associated impacts. Moreover this method is also benef cial for assessing the risk of potential suppliers who are under consideration.

In [20], the authors categorize supply chain risk in f ve types. First, the risk from environment which arise from political issues, natural disasters and social uncertainties. The second type of risk occurs from suppliers activities and the general relation of suppliers which they referred as supply risk. The third type is related with logistic f ows and demands associated with supply chain. This kind of risk can be called the demand risk. The next type of risk is process risk. Again, there can be control risk also in supply chain. Besides, there is also potential risk of information leakage by inference. According to the classif cation of risks, many existing work are conducted to propose solutions to control specif c types of risk in supply chain. In particular, closest to our work, Zhang et al. [49] focus on mitigating risk by information leakage by inference.

2.4 Information Leakage in Supply Chain

Information leakage occurs due to various reasons in a supply chain network. Information may be leaked by a manufacturer to a competitor supplier [27]. This type of leakage can occur when two competitors are involved in the same supply chain with a common manufacturer. When they share information with the manufacturer, then through the manufacturer the information can be leaked to the other supplier who is a potential competitor. On the other hand, information can also be leaked by a common supplier supporting two competitor manufacturers [3]. In this case, one common supplier working with two manufacturers can possess their information and through the supplier one manufacturer may potentially gain unauthorized access to his competitor manufacturer.

A review was conducted on secure collaboration in global design and supply chain environment focusing on various issues [48] in information leakage in collaborative development. The authors addressed problems in information access control, information partitioning, and partner trust management for collaborative developments. Conf dential information can be leaked to competitors explicitly or implicitly. Explicit leakage occurs when owner of the information mistakenly shares conf dential information with competitors. Usually, owner of the information can do partitioning on data to separate conf dential and non-conf dential information. The intention behind this kind of partitioning is to separate non-conf dential or public data from private or sensitive data so that the owner can share public data for the purpose of collaborative development and keep conf dential data secure. But sometimes the conf dential information becomes very critical for product development that the owner of the information cannot but has to share the information. Also, sometime when partitioning the data, the owner of information can mistakenly partition and some private data can go into public data space. This type of leakage can be protected through security technologies like access control.

Another type of information leakage, implicit leakage, occurs by inferences through derivation or deduction using relationship among information about identical or related

products. A method for preventing indirect information leakage such as information leakage through inference was proposed which shows that primary holder of information can compute in advance the risk of information leakage by inference and then select suppliers to minimize the risk of information leakage [49].

2.5 Information Leakage Prevention in Supply Chain

In this section, we review several types of existing methods for preventing information leakage in supply chains.

Access Control

Access control ensures that only authorized users can access specif c information. Several access control models have been developed to meet the security requirements of information sharing and collaboration in supply chains [49]. In [28] Leong et al. introduced a product data management system in a multiple workspace environment where each workspace has their own security levels. The user's right depends on the workspace he is working on. Role-based viewing access control grant users specif c roles and the security levels depend on their role. Another type of access control employes role-based access control (RBAC) with cryptographic security. This allows RBAC to incorporate time, scheduling and value added activity to provide security at data set level.

A trust evaluation method to share information is proposed by Chen et al. [7]. This solution focuses on information sharing between co-workers in virtual project teams to facilitate secure collaboration. This method helps the members in a virtual private team in information sharing decision making. In this method they use a threshold which helps to determine whether to share information. They also employed direct and indirect trust values to enhance the degree of trust in the system. Osborn et al. have provided systematic constructions for various common forms of both of the traditional access control paradigms using the role-based access control (RBAC) models and it was presented that for the mandatory access control simulation, only one administrative role needs to be assumed,

whereas for the discretionary access control simulations, a complex set of administrative roles is required [34].

Access control in a collaborative environment has traditionally relied on models based on digital certicates and the Public Key Infrastructure (PKI) [48]. Welch et al. [43] proposed a f exible approach for grid to manually edit policy databases or credentials issuance using digital certif cate based on authentication and authorization. Access control based information leakage prevention is good for a number of security problems. But it does not help with the problem of information leakage by inference. On the other hand, a lot of work was done on privacy preserving data mining. Methods were proposed to allow a user to modify sensitive inputs in a collaborative development environment without compromising the privacy [1].

Data Partitioning

Data partitioning is suggested in many research work to provide security while sharing information in supply chain or collaborative development. Two or more party can use vertically partitioned data or horizontally partitioned data without breaking the privacy.

Vaidya et al. presented in their work the problem of preserving privacy in vertically partitioned data [42]. Vertically partitioned data refers to a scenario where data is divided into several clusters which are not in downstream or upstream of each other. In this kind of situations, it is possible to obtain information about several partitions using a common parameter that was used by them. The authors presented an association rule mining method to assist in secure data mining on vertically partitioned data based on crypto techniques. Kantarcioglu et al. [22] proposed a method of privacy preserving data mining when the data is partitioned horizontally. Horizontally partitioned data refers to the situation where data are distributed in different sites and at a certain time data is gathered in a center point in order to perform operations on their collection. This situation can lead to loss of privacy as well. Their work propose a solution for mining on horizontally partitioned data.

Rizvi and Haritsa [38] presented a privacy metric and an analytical formula to evaluate security on a distorted database. A method is proposed to provide privacy by creating

ambiguity in data set [14]. In this method, real data set is converted to unreal data sets where the real data set cannot be extracted from unreal data set unless whole unreal data set is available. This makes the it harder for the adversaries to break the privacy of the real data set.

Protecting Privacy by Supplier Selection

A method for preventing indirect information leakage such as information leakage through inference was proposed to show that the primary holder of information can compute in advance the risk of information leakage by inference and then select suppliers to minimize the risk of information leakage [49]. In their work the authors used an example of natural gas dryer which has several components and each component has come parameters which hold information about the component. When the owner of the information wants to outsource any of the components, he needs to share the related parameters with them. So, the problem is converted to selecting suppliers for components such that no supplier can get enough information about the product or a critical part of product. To decide whether to share information, this thesis used a risk threshold which is a benchmark to select suppliers. From the information shared with a supplier the risk of information leakage can be calculated. Then the risk of information leakage is compared to the risk threshold to find out if it is safe to share the information with the supplier. If the risk of information leakage of sharing certain components or groups of components with a supplier is greater, than the risk threshold then it is considered that the sharing of information is not safe. In this way by selecting suppliers such that the risk of information leakage to all suppliers remain below the risk threshold, it can be ensured that the risk of information leakage for the whole system is sufficiently low.

Deng et al.[9] presented an approach of preventing information leakage by product decomposing. In this work, the authors proposed a method to protect the privacy of intellectual property (IP) against indirect leakage. The product decomposition is done based on several principles to ensure IP leakage prevention and cost optimization. Similar types of components are grouped together to outsource. Also, the manufacturing similarity of the

components in a cluster is kept as high as possible. The main intention of clustering components according to similarity is to minimize the cost of production. Another principle is to keep the interaction between components in a cluster as low as possible to minimizing the risk. The solution ultimately depends on the supplier capability of supplying components, since if there is only one supplier to supply a component then it has to be supplied by that supplier and only when there are multiple choices can a decision be made according to the given principles. In contrast to our work, these existing methods are time consuming and have a high complexity because they demand a huge amount of calculations for risk assessement on all possible allocations of the suppliers.

Approximation, Suppression and Generalization

When multiple parties want to compute on shared information without compromising their conf dential information, one solution is approximation [13]. Nonetheless, very often approximation cannot serve the purpose when exact information is required to be shared. In such a case, data and knowledge can be shared without revealing any additional information of each individual database apart from the aggregate result. Li et al. [45] showed that while several parties or private databases interact during data aggregation a decentralized peer-to-peer protocol can provide security for data sharing while minimizing data disclosure. The authors in [32] proposed an arithmetic solution for multiparty computation where several parties are allowed to compute over common data through a trusted third party without revealing private information of any party. This solution relies on cryptographic computation as well as cryptographic communication to provide security in secure multi-party computation (SMC). They employ polynomials for encrypting data and add dummy data at random places to provide security.

In a later work, Mishra et al. proposed a zero hacking protocol with several trusted third parties [33]. The advantage of having several third party is to have the option to select one of them randomly and thus creating an anonymity in the system to increase security. At a specif c time, this protocol, selects one third party among all randomly; the authors claim that it is more effective than with one third party. The main concept behind this idea

is that any single third party does not obtain complete data from a system. They proposed the protocol in four layers. In the first layer, data is split into packets; in the second layer, packets are randomly sent to the anonymizer; in the third layer, the packets are forwarded to randomly selected TTP, and in the last layer, all TTPs select a master TTP and send the packet to it. Pathak et al. [36] proposed zero hacking security protocol using a virtual party which performs computation on encrypted data where the encryption does not affect the result of the computation. While secure multi party computation are required for large scale sensitive data sharing in large scale surveys, this protocol offers a solution to compute data among multi parties without revealing information of any of them.

A protocol for supply chain security is proposed by Atallah et al. [4] where the parties can jointly compute data without breaking privacy. In this method, a supply chain decision can be made using all parties' information without revealing the private information of any of the parties. This ensures that a powerful supplier cannot take advantage to compromise security. In this work, the authors mainly focused on capacity allocation in e-commerce and e-auction. To secure the e-commerce information sharing their protocol supports such environments where the users are honest-but-curious, which means they follow the protocol but also tries to calculate other parties information. For secure e-auction they provide solutions for two scenarios, where all buyers get the same unit price from suppliers, and where they may get different prices. For the first case, every bidder starts with a cryptographic value of their data, thus keeping their confidential information secret. For the later case, the buyers price quantity information is revealed but from this information price or quantity cannot be extracted.

A novel and efficient protocol is proposed by Cachin [6] which facilitates two parties bargaining with the help of a third party where no one reveals the private information rather uses a combination of encryption and hiding assumption. This protocol is also useful for internet-based bargaining and also can serve as a building block for secure auction protocol. This work shows that secure bargaining is possible between two parties where no party learns about the strategy of other party. This is achieved by repeatedly using this protocol to bargain until both parties are satisfied. In addition, researchers have proposed various

solutions to this problem, such as using access control, suppression or generalization of information, using documentation standards, implementing laws and policies to enhance secure collaboration. Data are usually divided into two categories: conf dential and non-conf dential. Non-conf dential data are shared among other parties in supply chains to serve the manufacturing task, and the conf dential information is either supressed (kept conf dential) or shared in generalized forms [41]. This method however is not always applicable to real problems because sometimes the information that are categorized as conf dential may be critical for product development and manufacturer has no choice but to share the exact information (e.g., dimensions of a part), which is the case of this thesis.

2.6 Trust

Trust managements deal with the relationships between collaborating parties involved in data sharing based on predef ned trust policies and principles. Over the past years, many researchers tried to def ne trust in different ways. Griff n [15] def ned trust as "an attitude displayed in solutions where a person is relying on another person, a person is risking something of value, and/or a person is attempting to achieve a desired goal". According to Schurr and Ozanne [40] trust is "the belief that a party's word or promise is reliable and that a party will fulf ll his/her obligations in an exchange relationship". Trust has ebeen def ned as a broad range of concepts over the time, and hence, it is reaonable to say that there are multiple factors involved in trust management. Fawcett et al. [11] consider trust to consist in two dimensions: benevolence and capacity, and the case study shows that although benevolence underlies trust in personal relationship, it does not really exist among companies; rather, supply chain trust is capability-based. According to the research, two types of capabilities are needed: performance capability and relationship commitment capability. However, Handf eld proposed [17] that, managers who are serious about improving supply chain responsiveness should work towards building greater levels of trust with key-input suppliers, and explore opportunities for collocation and information sharing on a regular basis.

Kwon [24] attempt to f nd factors that affect the level of trust, and in their research they presented that trust between partners in a supply chain is highly associated with both sides' specif c asset investments and behavioral uncertainty. The former affects trust positively and the later negatively. In a later work, Kwon [25] also found that a partner's reputation in the market has a strong positive impact on the trust-building process, whereas a partner's perceived conf ict creates a strong negative impact on trust. Sahay et al. [39] focus on the natural and crucial role played by trust in long-term relationships and tries to integrate a number of different perspectives to develop a framework along with three issues: ways the term "trust" is used, factors leading to trusting behavior in the customer-supplier relationship, and the effect of trust on the behavior of a customer and a supplier. In [5], only authorized entities are permitted to execute on the system. This method discussed about organizational knowledge management used for intellectual assets protection.

Yu and Winslett [46] discussed about trust negotiation between two parties. Here, they attempt formalizing the concept of negotiation protocols, strategies, and inter operations. This method can be applied for online collaboration of sensitive information. To facilitate their ideas, they presented a trust negotiation architecture to show the interoperability between strategies occurs when they are from the same family. Pinkas proposed a protocol that solves the fairness problem [37] using a secure protocol for collaboration which utilizes signature for verif cation. Sometimes data has inherently coherent nature. Unlike previous methods where the overhead is very high, this solution generates more eff cient outputs, and it does not need a third party to be involved. But the number of rounds needed increases proportionally to the security parameters, and also it requires blind signatures.

In such cases, the authors in [26] bring a solution by creating virtual trust domains. They propose a decentralized approach for grid environments. In an untrusted network of shared computers, a virtual trust domain can be created with the help of visualization of network and visualization of operating system as well as simple public key infrastructure certif cates. They implemented this idea using vanilla IPsec stack and OS virtualization mechanism implemented in the native OS [26]. As a result on a single physical host, multiple virtual HIPernet was created and any of these can be possibly use for secure allocation

for remote access.

Chapter 3

The Model

This chapter introduces the basic model of supply chain, essential component sets, supplier capabilitiy, allocation and assignment, risk, safe/unsafe assignment, and f nally trust level.

3.1 Conceptual Model of Supply Chain

We f rst introduce a conceptual model of supply chains with one focal manufacturer and *n* suppliers [50]. Denote the manufacturer with s_0, let $S = \{s_1, s_2, ..., s_n\}$ be a set of suppliers. The manufacturer s_0 develops a product, which consists of a set of components. A component may also consist of sub-components as an assembly. Each supplier $s_i \in S$ has the capabilities of producing particular components.

In the two-level supply chain, the focal manufacturer s_0 is the holder of conf dential information and attempts to prevent its leakage to supplier s_1 by inferences. Supplier s_1 is an inferrer who tries to acquire the conf dential information I_0 protected by the holder s_0. Supplier s_1 may obtain its knowledge of s_0's conf dential information through three sources: initial knowledge K_0, shared information I_s, and inferences. In a previous work [50], the knowledge of information is modeled as probability distributions. In this case, the manufacturer s_0 can estimate supplier s_1's knowledge through inferences K_s, and can evaluate the risk of his conf dential information leakage to supplier s_1 by a set of algorithms [49].

3.2 Essential Component Sets

In order to describe a product and the hierarchical relations between its components, Zeng and Gu [47] previously proposed a product structure tree. We review the concept here. A node of the product structure tree represents a product or a component. An edge, connecting two component nodes, represents a parent-child relationship between them. A node can be denoted as $n(k, i_k, j_{k-1}), k = 1, 2, ...$, if the node is at the i_k^{th} position in the k^{th} layer and its parent node is at the j_{k-1}^{th} position in the $(k-1)^{th}$ layer. All the nodes together constitute a product structure tree, which is denoted as T, and the collection of all the nodes of T are denoted as N_T, with the root node of T denoted as $r(T) = n(1, 1, 0)$, when $k = 1$.

Intuitively, an essential component set refers to a set of nodes in which all the components together form a partition on the leaf nodes of the product structure tree [49]. Note that the the nodes in an essential component set are not necessarily on the same level, and there are usually many essential component sets for a given product.

Definition 1. *(Essential Component Set (ECS)) In order to mathematically represent an essential component set (ECS), these functions will be firstly defined as follows.*

1. *node of a sub-tree:* $N(n) = N_{T'}$*, where* $n \in N_T$*,* T' *is a sub-tree of* T *and* $r(T') = n$*;*
2. *leaf of a sub-tree:* $LN(n)=n'$*, where* n' *is a leaf node in* $(k+1)^{th}$ *level.*

For any set of nodesl $N \subseteq N_T$*,* N *is an essential component set of tree* T*, if it satisfies:* $\forall n_i, n_j \in N, i \neq j, LN(n_i) \cup LN(n_j) = \phi$ *and* $\bigcup_{n_i \in N} LN(n_i) = LN(n(1, 1, 0))$.

Example 1. *For example, we consider the natural gas dryer depicted in Figure 1. According to the product structure tree,* $N = \{Assembly\ task, Dryer, Cooler, heater, Blower\}$ *is one essential component set.*

For a valid distribution of tasks or product parts among suppliers in a supply chain, every element in an essential component should be distributed to an eligible supplier. A valid distribution of tasks or product parts to suppliers is commonly termed as an "allocation".

3.3 Supplier Capability

In order to describe an allocation, two functions are def ned below. We use supplier capability function F_{sc} to describe a supplier's capability to supply components, and supplier capability set to measure how much information is shared with a supplier in an allocation.

Def nition 2. *(Supplier Capability Function) For a product structure tree T, a supplier capability function F_{sc} (s) returns a set of components $N \subseteq N_T$ which supplier s can supply.*

Example 2. *(Supplier Capability Function) Supplier capability function gives the set of component N which a supplier s is able to supply. Table 8 shows an example of supplier capability function. In the table, for instance, supplier s_0 is capable of supplying component n_1 and component n_4, s_1 capable of n_1 and component n_4, etc. We will follow this running example for all the discussions in this thesis.*

Table 8: Supplier capability function

Supplier s	$F_{sc}(s)$
s_0	n_1, n_4
s_1	n_1, n_4
s_2	n_2
s_3	n_3
s_4	n_4, n_{41}
s_5	n_{11}
s_6	n_{12}, n_{42}
s_7	n_{41}
s_8	n_{42}
s_9	n_{43}

Def nition 3. *(Supplier Capability Set) Suppliers capability set is the power set of each supplier's capability function, denoted $\rho(F_{sc}(s))$, $\forall s \in S$.*

Example 3. *(Supplier Capability Set) From table 8 we can see that supplier s_0 is capable of supplying component n_1 and component n_4. So, the supplier capability set for supplier s*

would include n_1, n_4, n_1, n_4 and ϕ. The signif cance of def ning the Supplier Capability Set is that Supplier Capability Function only lists the components that a supplier can supply, but Supplier Capability Set lists sets of components that can be assigned to a supplier during product distribution. This helps to know how much information is shared with a particular supplier, as detailed later. The complete suppliers capability set can be found later in this thesis, in table 28.

3.4 Allocation and Assignment

Allocation and assignments refers to the distribution of components among suppliers. Here we def ne both terms and explain them with examples.

Allocation is a mapping of suppliers and components that shows a particular possible distribution of the components to participating suppliers. Intuitively, an allocation satisf es following conditions. First, it maps an essential component set (ECS) to a set of suppliers. Second, every component mapped to a supplier must be within that supplier's capabilities.

Def nition 4. *(Allocation) T is a product structure tree, N_T is the set of all nodes of T, $N' \subset N_T$. A mapping F_a: $N' \rightarrow S$ is called an allocation, if it satisf es:*

1. N' is an ECS;
2. if $\exists s$, $F_a(n)$=s, then $n \in F_{sc}(s)$, where $F_{sc}(s)$ is supplier capability function;

Example 4. *(Allocation) Table 9 shows the allocation of the regeneration system of natural gas dryer described in [29, 49]. It shows all the possible allocations for the given supplier capabilities shown in table 8. Here, for instance, in allocation A_1, component n_1 is assigned to supplier s_0, component n_2 assigned to supplier s_2, n_3 assigned to s_3, n_4 assigned to s_0, n_{11} assigned to s_5, n_{12} assigned to s_6, n_{41} assigned to s_4, n_{42} assigned to s_8, and n_{43} assigned to s_9. The important property of any allocation is that it must be based on an ECS, such that all the essential components are distributed to suppliers, whereas it may or may not require all the suppliers.*

Table 9: Allocation

Allocation	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_1	s_0	s_2	s_3	s_0	s_5	s_6	s_4	s_8	s_9
A_2	s_0	s_2	s_3	s_0	s_5	s_6	s_4	s_6	s_9
A_3	s_0	s_2	s_3	s_0	s_5	s_6	s_7	s_8	s_9
A_4	s_0	s_2	s_3	s_0	s_5	s_6	s_7	s_6	s_9
A_5	s_0	s_2	s_3	s_1	s_5	s_6	s_4	s_8	s_9
A_6	s_0	s_2	s_3	s_1	s_5	s_6	s_4	s_6	s_9
A_7	s_0	s_2	s_3	s_1	s_5	s_6	s_7	s_8	s_9
A_8	s_0	s_2	s_3	s_1	s_5	s_6	s_7	s_6	s_9
A_9	s_0	s_2	s_3	s_4	s_5	s_6	s_4	s_8	s_9
A_{10}	s_0	s_2	s_3	s_4	s_5	s_6	s_4	s_6	s_9
A_{11}	s_0	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_{12}	s_0	s_2	s_3	s_4	s_5	s_6	s_7	s_6	s_9
A_{13}	s_1	s_2	s_3	s_0	s_5	s_6	s_4	s_8	s_9
A_{14}	s_1	s_2	s_3	s_0	s_5	s_6	s_4	s_6	s_9
A_{15}	s_1	s_2	s_3	s_0	s_5	s_6	s_7	s_8	s_9
A_{16}	s_1	s_2	s_3	s_0	s_5	s_6	s_7	s_6	s_9
A_{17}	s_1	s_2	s_3	s_1	s_5	s_6	s_4	s_8	s_9
A_{18}	s_1	s_2	s_3	s_1	s_5	s_6	s_4	s_6	s_9
A_{19}	s_1	s_2	s_3	s_1	s_5	s_6	s_7	s_8	s_9
A_{20}	s_1	s_2	s_3	s_1	s_5	s_6	s_7	s_6	s_9
A_{21}	s_1	s_2	s_3	s_4	s_5	s_6	s_4	s_8	s_9
A_{22}	s_1	s_2	s_3	s_4	s_5	s_6	s_4	s_6	s_9
A_{23}	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_{24}	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_6	s_9

Based on the Def nition of allocation, we give the def nition of an assignment below. Assignment refers to a component or a set of components that is assigned to a supplier in an allocation.

Def nition 5. *(Assignment) A mapping A: $S \rightarrow 2^{N'}$ is called an assignment, if it satisf es that s is mapped to $n \in N'$ if and only if $F_a(n)$=s for all $s \in S$ and $n \in N'$.*

Example 5. *(Assignment) Table 10 shows the Assignments of our running example. For instance, in the f rst allocation, component set $\{n_1, n_4\}$ is assigned to supplier s_0, no component assigned to supplier s_1, component n_2 assigned to supplier s_2, component n_3 assigned to supplier s_3, component n_{41} assigned to supplier s_4, component n_{11} assigned to supplier s_5, component n_{12} assigned to supplier s_6, no component assigned to supplier s_7,*

component n_{42} assigned to supplier s_8, and component n_{43} assigned to supplier s_9.

In fact, assignment is a subset of allocation. If we arrange the allocation from supplier view it gives us assignment for all suppliers.

Table 10: Assignment

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_1	n_1,n_4	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_2	n_1,n_4	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_3	n_1,n_4	ϕ	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_4	n_1,n_4	ϕ	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_9	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{11}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{17}	ϕ	n_1, n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{18}	ϕ	n_1,n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{19}	ϕ	n_1,n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{20}	ϕ	n_1,n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{21}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{23}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

3.5 Risk

In this thesis we consider risk as the probability of information leakage to supplier. However, we consider that risk in supply chain can occur from two sources: Direct information leakage- when information in mistakenly shared and indirect information leakage when information is not explicitly shared but supplier can infer the information. In this thesis, we

assume that the risk caused by direct information leakage has been addressed by manufacturer and still some information leakage occurs by means of indirect information leakage that is by inference. Zhang et al. formulated the problem of information leakage caused by inference [49] and devised a quantitative approach to evaluate the risk of information leakage caused by inferences when a given amount of information is shared [50].

For a component n which includes a set of information P_I, the information leakage risk of allocating component n to supplier s, denoted as r_{ns}, is def ned in Eq. 1 below.

$$r_{ns} = \sum_{i=1}^{I} P(p_i|(p_i, P_I)) \times C(p_i) \tag{1}$$

where $P(p_i|(p_i, P_I))$ is the probability of leakage when sharing information p_i. $C(p_i)$ is the consequence if information $p_i, p_{\in} P_I$ is leaked [49]. In this thesis, we consider $C(p_i) = 1$, if p_i is conf dential; otherwise, $C(p_i) = 0$.

3.6 Safe/Unsafe Assignment

Given an assigned supplier s and component n, we assume a threshold t_{ns}. This risk threshold is def ned according to supply chain specialist's expertise and professional experience. If the risk of that component n being leaked to supplier s is lower than t_{ns}, then we consider the corresponding assignment A_{sn} "safe" (A_{sn}=1). Otherwise, we consider the assignment A_{sn} "unsafe" (A_{sn}=0). In other words, A_{sn}=1, if and only if $\forall n, s, r_{ns} \leq t_{ns}$.

Identif cation of unsafe and safe assignments is important because it helps to f lter out unsafe assignments such that we can only need to choose the most cost effective allocations among the safe ones, instead of considering all possible allocations, which is prohibitive. Our focus is to f nd how to f nd the optimum allocation employing the least computational effort. To do that, we would present rules and mechanisms for identifying unsafe allocations and for eliminating them. In this thesis, we assume that if it is safe to allocate one component to a supplier, then it is also safe to allocate or share information about its child components to the same supplier.

3.7 Trust Level

As we have mentioned, in our model, we extend the existing two-level supply chain model to multi-level trust-based model. Intuitively, all the suppliers will be assigned a trust level by the manufacturer, which is an indication of how much conf dence the manufacturer has in that particular supplier in terms of sharing conf dential information with the supplier. In other words, a trust level indicates how safe it is to share conf dential information with a supplier. If the trust level is high, then it may be "safe" to share more product information even if the risk of doing so is higher. For a supply chain system, the manufacturer is primary owner of all conf dential and non-conf dential information and hence the manufacturer has the highest level of trust. A supplier at a higher trust level is eligible to share more sensitive information through being assigned more sophisticated components than a supplier at lower trust level. That is, this supplier would be assigned a higher risk threshold than other suppliers with lower trust levels.

Def nition 6. *(Trust level) For any supplier S, a function $T(s)$ ($a \leq T(s) \leq b$) def nes the risk level of S where a and b are two given non-negative numbers representing the lowest and highest levels of trust among all suppliers. For any two suppliers S_i and S_j, if $T(s_i) < T(s_j)$ is true, then we say that supplier s_i is less trustworthy than s_j and we assume that $t_{ns_i} \leq t_{ns_j}$ holds for any component n.*

In a two level supply chain, the manufacturer is in one level and all other suppliers are in another level which means they all have the same level. In reality this is usually not the case. For example, suppose, one supplier with a good reputation and longtime business relationship with a manufacturer will have higher trust level compared to a supplier who is new and unknown to the manufacturer. In the previous model these two suppliers go into the same level which makes it impractical for the manufacturer to compare them based on risk threshold only. In our model we try to shape this problem by assigning trust level to the suppliers. Consequently, by assigning a high trust level to the known and trusted supplier we propose a more practical solution to this problem. For example, in Table 6 and Table 7, we see that the given thresholds cannot be satisf ed by any of the allocations.

However, if we assume supplier s_1 has a higher trust level than others, then consequently it will enjoy a higher risk threshold which means some of the allocations might become safe now. It should be noted that, our solution does not recommend to trust a random supplier in order to find a solution in the case shown in table 6 and table 7. Rather, our goal is to shape the problem to fit real life situation and try to find a solution. In the example shown in table 6 and table 7 there is no safe choices. Because the analysis is only done based on comparing risk of information leakage with risk threshold. Using practical experience, if trust levels can be applied in that situation we may find that there exist some supplier who has a higher trust level than others. Consequently, that supplier with higher trust level can be allowed to share information or allocate component even the risk calculated is greater than risk threshold. It is worth mentioning that, as the experience and reputation changes from time to time the indication of trust level should be updated in specific time duration, for example, in every 3 years.

Chapter 4

Unsafe Assignment Identif cation

In this chapter, we describe a series of methods for identifying unsafe assignments proactively in order to avoid the unnecessary computational efforts involved in conducting risk assessments for such unsafe assignments in the supplier selection process.

4.1 Scalability of Supplier Selection

In a typical supplier selection scenario [49], the manufacturer has a list of components, and for each component the manufacturer has a list of potential suppliers who are capable of supplying that component. In this way, the manufacturer has a number of options to distribute the work among suppliers. The maximum number of possible ways for such a distribution can be estimated as n = (the number of suppliers who can supply component 1) * (the number of suppliers who can supply component 2) * ... * (the number of suppliers who can supply component n). In the worst case, the maximum number of allocations for n components and s suppliers is s^n. For instance, the number of maximum possible allocations is $3^5 = 243$ when each of the 5 components could be supplied by all 3 suppliers; but if there is one more supplier, then the maximum number of allocations will become $4^5 = 1024$. That is, the number of possible allocations will likely increase exponentially with the number of suppliers.

Figure 2 demonstrates such an increase in the number of allocations as the number of

suppliers increases. Initially the number of suppliers is 5 and the number of component is 3. Then we increase number of suppliers from 3 to 14. The number of allocations increases in a scale of 0 to 500 thousand. Of course, this is the worst case scenario where every supplier can be assigned to every component. In average cases, the increase will be slower. However, considering that in most realistic cases, the number of components and suppliers may be signif cantly larger than those in this example, the increase in the number of allocations may still be signif cant enough that, for any realistic number of suppliers and components, it would be infeasible to f rst enumerate all possible allocations, and conduct risk assessment on them, before we can f nd the optimal solution, which is the approach adopted by existing work [49].

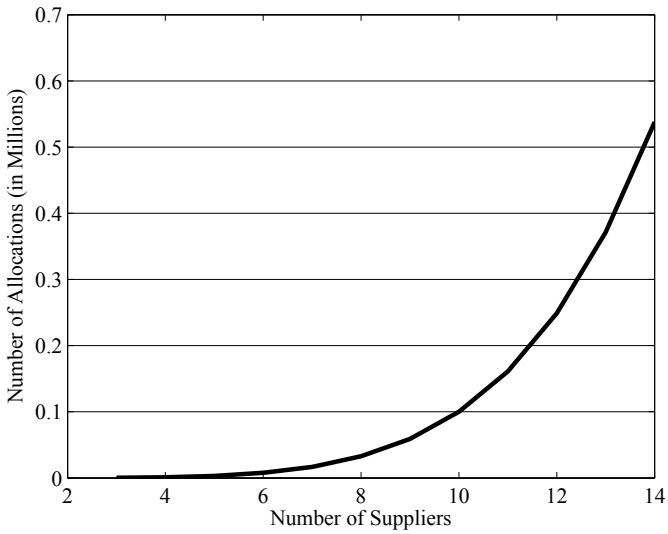

Figure 2: Increase in allocation

On the other hand, this exponential increase of the number of allocations in the number of suppliers also indicates the signif cant effect of eliminating suppliers from consideration. Therefore, we now narrow down our goal to f nd suppliers that can be removed from the process of supplier selection. To remove suppliers, we f x some criteria. Our primary goal is to f nd the set of suppliers which has the minimum risk of information leakage and minimum cost also. Therefore, we aim to f nd and f lter out those suppliers who have a greater risk of information leakage. In the coming sections, we will present different

unsafe assignment identif cation mechanisms considering three cases, namely duplication-based identif cation, set-based identif cation, and trust-based identif cation.

4.2 Duplication-based Identif cation

The aim in this f rst case of unsafe assignment identif cation is to avoid the unnecessary calculation of duplicates. Recall that, in existing supplier selection procedures, the allocations are generated as full combinations of component allocations to suppliers according to their capabilities. Such procedures, however, unavoidably introduce duplicates of supplier-component relationships, for example, two or more allocations in which the same supplier is allocated with exactly the same components. In such cases, if by risk assessment we have determined that an assignment is unsafe in one allocation, then we do not need to be repeat the risk assessment before we can know that this same assignment will also be unsafe in all other allocations. Based on such an intuition, the duplication-based rule of unsafe assignment identif cation is described as Rule 1 below.

Rule 1: If $s \longrightarrow n$ is unsafe in one allocation $F_a(n)$=s, then $A_{sn} = 0$ (that is, this assignment is unsafe) for $\forall F'_a$, $F'_a(n) = s$.

Example 6. *In Table 10, if by risk assessment we already know assignments* $s_2 \longrightarrow n_2$ *and* $s_3 \longrightarrow n_3$ *to be unsafe in* F_1*, then they will also be unsafe for allocations* F_2-F_{24}.

Table 10 shows that, in practice, there could be many duplicates of the same component set, and some of them may indeed be assigned to the same supplier. Table 11 shows the same allocations, but with duplicate assignments omitted. In this table, we can see that only 22 unique assignments exist, but together with their duplicates there are totally 240 assignments in the table. Therefore, identifying and avoiding risk assessment on the duplicates of unsafe assignments would save signif cant computational efforts.

Table 11: Assignment (Duplication omitted)

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_1	n_1,n_4	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_2	—	—	—	—	—	—	n_{12},n_{42}	—	ϕ	—
A_3	—	—	—	—	ϕ	—	—	n_{41}	—	—
A_4	—	—	—	—	—	—	—	—	—	—
A_5	n_1	n_4	—	—	—	—	—	—	—	—
A_6	—	—	—	—	—	—	—	—	—	—
A_7	—	—	—	—	—	—	—	—	—	—
A_8	—	—	—	—	—	—	—	—	—	—
A_9	—	—	—	—	n_4,n_{41}	—	—	—	—	—
A_{10}	—	—	—	—	—	—	—	—	—	—
A_{11}	—	—	—	—	n_4	—	—	—	—	—
A_{12}	—	—	—	—	—	—	—	—	—	—
A_{13}	n_4	n_1	—	—	—	—	—	—	—	—
A_{14}	—	—	—	—	—	—	—	—	—	—
A_{15}	—	—	—	—	—	—	—	—	—	—
A_{16}	—	—	—	—	—	—	—	—	—	—
A_{17}	ϕ	n_1, n_4	—	—	—	—	—	—	—	—
A_{18}	—	—	—	—	—	—	—	—	—	—
A_{19}	—	—	—	—	—	—	—	—	—	—
A_{20}	—	—	—	—	—	—	—	—	—	—
A_{21}	—	—	—	—	—	—	—	—	—	—
A_{22}	—	—	—	—	—	—	—	—	—	—
A_{23}	—	—	—	—	—	—	—	—	—	—
A_{24}	—	—	—	—	—	—	—	—	—	—

4.3 Set-based Identif cation

The principle of the second case of unsafe assignments identif cation is to mark unsafe assignments in which the assigned component set of one allocation is not exactly the same as the other, but thnonetheless one is a subset or superset of the other. This case is slightly more complicated. If a set of components is unsafe to assign to any particular supplier, then by def nition, any superset of that component set will also be unsafe to assign to that supplier. For example, if a supplier s is unsafe to allocate a component set n_4, then s is also unsafe to allocate n_1, n_4 in all allocations. This is the main intuition behind the set-based identif cation. More formally, the principle is stated as Rule 2.

Rule 2: If $s \longrightarrow n$ is unsafe, then $A_{sn'} = 0$ for $\forall n' \supseteq n$, where $F_a(n') = s$.

Example 7. *According to Rule 2, in Table 8, if* $s \longrightarrow n_4$ *is unsafe in allocation* F_5*, we can identify* $s \longrightarrow n_1, n_4$ *is unsafe in allocations* F_{17}*-*F_{20}.

Table 12 shows all allocations where any superset of an existing unsafe set is omitted. Here we did not take ϕ into account because, assigning ϕ to a supplier practically means that the supplier is not assigned with any components. In that case, the risk cannot be greater than the threshold. Since our goal is to find out those component sets which will be unsafe to assign and the minimum unsafe set of component need to be calculated, we only consider those sets which have at least one component and omit their supersets.

In table 12, we can make two observations. Firstly, we can see that very few assignments are omitted through the set-based identification in constrast to duplication-based identification. However, this does not mean this second case of identification of unsafe assignments is insignificant, because in real-life scenarios, the total number of assignments will be much larger than that in those examples, and thus this case would play a significant role in improving the efficiency. Secondly, although we have omitted all supersets of existing set as discussed above, in real-life scenarios, a set can be identified as unsafe only if there exist at least one of its subsets which has already been identified as unsafe and a risk assessment procedure may not always guarantee such a right order between any two sets, which means not all supersets of unsafe assignments would be successfully identified.

4.4 Trust-based Identification

As we have mentioned, in our model, a trust level indicates how safe it is to share confidential information with a supplier, and if the trust level is high, then it may be "safe" to share more product information even if the risk of doing so is higher. Therefore, a supplier at a higher trust level is eligible to share more sensitive information since it would be assigned a higher risk threshold than other suppliers with lower trust levels. By Definition 6, for any two suppliers S_i and S_j, if supplier s_i is less trustworthy than s_j then $t_{ns_i} \leq t_{ns_j}$ will be true for any component n. This will lead to following trust-based unsafe assignments identification rule.

Table 12: Assignment (Superset of existing set omitted)

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_1	n_1,n_4	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_2	n_1,n_4	ϕ	n_2	n_3	n_{41}	n_{11}	—	ϕ	ϕ	n_{43}
A_3	n_1,n_4	ϕ	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_4	n_1,n_4	ϕ	n_2	n_3	ϕ	n_{11}	—	n_{41}	ϕ	n_{43}
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	—	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	—	n_{41}	ϕ	n_{43}
A_9	n_1	ϕ	n_2	n_3	—	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	n_1	ϕ	n_2	n_3	—	n_{11}	—	ϕ	ϕ	n_{43}
A_{11}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	n_1	ϕ	n_2	n_3	n_4	n_{11}	—	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	—	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	—	n_{41}	ϕ	n_{43}
A_{17}	ϕ	—	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{18}	ϕ	—	n_2	n_3	n_{41}	n_{11}	—	ϕ	ϕ	n_{43}
A_{19}	ϕ	—	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{20}	ϕ	—	n_2	n_3	ϕ	n_{11}	—	n_{41}	ϕ	n_{43}
A_{21}	ϕ	n_1	n_2	n_3	—	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	ϕ	n_1	n_2	n_3	—	n_{11}	—	ϕ	ϕ	n_{43}
A_{23}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	ϕ	n_1	n_2	n_3	n_4	n_{11}	—	n_{41}	ϕ	n_{43}

Rule 3: If $s \longrightarrow n$ is unsafe in one allocation, it is also unsafe to assign component n to supplier s' in any allocaiton if $T(s') < T(s)$. That is, if $s \longrightarrow n$ and $A_{sn} = 0$ hold, then $A_{sn'} = 0$ is true $\forall s', T(s') < T(s)$ where $F_a(n) = s'$.

Example 8. *In Table 10, if we suppose that* $s_0 \longrightarrow n_1$ *is unsafe in* A_1*, then, according to rule 3, it is also unsafe for* $s_1 \longrightarrow n_1$ *in allocation* A_{13} - A_{16} *, where* $T(s_1) \leq T(s_0)$.

Table 13 and table 14 show the example of trust-based identif cation according to rule 3. In table 13, in allocations A_1, supplier s_0 is assigned with component set n_1,n_4. Assuming that trust level of s_0 is greater than trust level of s_1 and assigning component set n_1,n_4 to supplier s_0 is unsafe, we can identify some unsafe assignments as shown in the table. Table 14shows which unsafe components set can be identif ed and omitted.

Table 13: Part of assignment (Discussion on rule 3)

	s_0	s_1
A_1	n_1,n_4	ϕ
$\vdots$	$\vdots$	$\vdots$
A_{17}	ϕ	n_1,n_4
A_{18}	ϕ	n_1,n_4
A_{19}	ϕ	n_1,n_4
A_{20}	ϕ	n_1,n_4
$\vdots$	$\vdots$	$\vdots$

Table 14: Part of assignment (Discussion on rule 3 continued)

	s_0	s_1
A_1	n_1,n_4	ϕ
$\vdots$	$\vdots$	$\vdots$
A_{17}	ϕ	—
A_{18}	ϕ	—
A_{19}	ϕ	—
A_{20}	ϕ	—
$\vdots$	$\vdots$	$\vdots$

If we assume that when one component is assigned to one supplier, this supplier will always hold all the information of that component as well as all its sub-components (note this assumption made in existing work [49] is not mandatory in our work), then we can have an additional rule for unsafe assignment identif cation. For example, in a supply chain network, suppose $T(s_2) \geq T(s_1)$, $n_1 \supseteq \{n_{11}, n_{12}\}$ (the latter represents two sub-components). If $n_1 \longrightarrow s_2$ and $n_{11} \longrightarrow s_1$, then supplier s_2 is eligible to get information of n_1, n_{11} and n_{12}, and s_1 will be eligible to get information of n_{11} only. In practice, if such an assumption holds, then we can have following rule.

Rule 4: If $s \longrightarrow n$ is unsafe in one allocation, it is also unsafe to assign supersets of n to supplier s' if $T(s') < T(s)$. Mathematically, if $s \longrightarrow n$ and $A_{sn} = 0$ both hold, then $A_{s'n'} = 0$ is true $\forall s', T(s') < T(s)$, $n' \supseteq n$ where $F_a(n') = s'$.

Example 9. *Again, if we suppose that $s_0 \longrightarrow n_1$ is unsafe in A_1, then, according to rule 4, it is also unsafe for $s_1 \longrightarrow (n_1, n_4)$ in allocation A_{17} to A_{20} , where $T(s_1) \leq T(s_0)$.*

Table 15 and table 16 shows the example of trust-based identif cation according to rule 4. In table 15, in allocations A_5, supplier s_0 is assigned with component set n_1. Assuming that trust level of s_0 is greater than trust level of s_1 and assigning component set n_1 to supplier s_0 is unsafe, then we can identify and omit some unsafe assignments as shown in Table 16.

Table 15: Part of assignment (Discussion on rule 4)

	s_0	s_1
A_5	n_1	n_4
⋮	⋮	⋮
A_{17}	ϕ	n_1,n_4
A_{18}	ϕ	n_1,n_4
A_{19}	ϕ	n_1,n_4
A_{20}	ϕ	n_1,n_4
⋮	⋮	⋮

Table 16: Part of assignment (Discussion on rule 4 continued)

	s_0	s_1
A_5	n_1	n_4
⋮	⋮	⋮
A_{17}	ϕ	—
A_{18}	ϕ	—
A_{19}	ϕ	—
A_{20}	ϕ	—
⋮	⋮	⋮

4.5 Identif cation Mechanism

Based on the above four rules, we now propose a mechanism for unsafe assignment identif cation. The mechanism of unsafe assignment identif cation is based on four steps, as shown in Table 17. Based on above discussions, we apply rule 1 where the supplier and component set are both identical for two assignments. In that case, we can avoid calculating the second assignment and reuse the decision from the f rst assignment. When the supplier

is the same between two assignments but the component set is different, we apply rule 2 to avoid supersets of unsafe component sets. If the supplier is different but the component set is the same in two different allocations, then, rule 3 can be applied to avoid repeating risk assessment for lower trust suppliers. Finally, if both the supplier and component set in two assignment are different, then rule 4 can be applied. Of course, when applying these rules, the relationship of the component set and suppliers' trust level also need to be checked.

Table 17: Mechanism of unsafe assignment identif cation

Supplier	Component	Identif cation of unsafe assignment
same	same	rule 1
same	different	rule 2
different	same	rule 3
different	different	rule 4

The rules can be used to identify unsafe assignments in any order. However, based on the complexity and the number of assignment identif cation, we found that the order mentioned here is more eff cient than others. We now present an algorithm to instantiate the aforementioned mechanism for identifying unsafe assignments.

Algorithm 1 Algorithm of unsafe assignment identif cation

Require: Allocation of suppliers;
Ensure: safe/unsafe assignments;

```
1:  for All allocations do
2:    if allocation i has no tag then
3:      for all assignments do
4:        if assignment(i, j) has no tag then
5:          Calculate risk of assignment(i, j)
6:        end if
7:        if tag of assignment(i, j) is unsafe then
8:          Set the tag of allocation i to unsafe;
9:          Algorithm 2: Find match in other allocations for the same supplier
10:         Algorithm 3: Find match in other allocations for different suppliers
11:       end if
12:     end for
13:   end if
14: end for
```

Discussion on Algorithm 1

In this algorithm we check all the allocations and all assignments in an allocation and add a tag to them about their safety status. If an assignment is already tagged, than it will be skipped. Otherwise the risk of the assignment is calculated and then other allocation are checked against the rules mentioned above. In this step, there can be two cases, which are handled with two procedures detailed below. Firstly, all allocations are checked for the same supplier's assignments only, or all allocations are checks for those suppliers' assignments where the suppliers' trust level are related or comparable to each other. For example, if we find an assignment with supplier s and component set N and find that it has no tag, then we calculate the risk of this assignment. Then, according to the risk, if the assignment is unsafe then we check all other allocations for supplier s's assignments only, and if in any other allocations supplier s is assigned with the same component set N or any superset of N, then we put an unsafe tag on those assignments. Then we check if there is any other supplier whose trust level is related to supplier s. If there exist such suppliers then we do the test for those suppliers too.

Algorithm 2 Algorithm of finding match for the same supplier

Require: Allocation of suppliers;
Ensure: safe/unsafe assignments;

```
1:  for k=i+1 to k  imax do
2:    if allocation k has no tag AND assignment (k,j)has no tag then
3:      if tag of assignment(i,j)) unsafe AND (assignment (k,j) is equal or a superset of
        assignment(i,j) then
4:        Set tag of assignment(k,j) unsafe AND Set tag (allocation k) unsafe
5:      else
6:        if tag of assignment(i,j) = safe AND (assignment (k,j) is equal OR subset of
          assignment(i,j) then
7:          Set tag of assignment(k,j) to safe
8:        end if
9:      end if
10:   end if
11: end for
```

Discussion on Algorithm 2

We employed Algorithm 2 to find matching assignments with the same supplier in other allocations. To do that, first we check whether the assignment we calculated in algorithm

1 is safe. If the assignment is unsafe, then we check all allocations starting from the next allocation until the last allocation. For each allocation, we check the suppliers for whom the risk has been calculated in algorithm 1. If the component set in the assignment calculated in algorithm 1 and the component set in the current assignment is the same, or its superset, than we add an unsafe tag to the current assignment. If the assignment is safe, then we check all allocations starting from the next allocation until the last allocation. For each allocation, we check the suppliers for whom the risk has been calculated in algorithm 1. If the component set in the assignment calculated in algorithm 1 and the component set in current assignment is the same, or its subset, than we add a safe tag to the current assignment. In this way, we will f nd each matching assignment with the same supplier in other allocations. After that we use Algorithm 3 for the next step.

Algorithm 3 Algorithm of f nding match for different suppliers

Require: Allocation of suppliers;
Ensure: safe/unsafe assignments;

```
1:  Check trust level of i;
2:  if trust level of i has related set of nodes N then
3:     for n=0 to n  size of N do
4:        for k=0 to k  imax do
5:           if allocation k has no tag AND assignment (k,j)has no tag then
6:              if trust level of N[n] ≤trust level of i AND ((assignment(k,n) is equal or
                superset of assignment (i,j))) AND if tag of assignment(i,j) = unsafe then
7:                 Set tag of assignment(k,n)= unsafe
8:                 Set tag of allocation k= unsafe
9:              else
10:                if (trust level of N[n] ≥trust level of i) AND ((assignment(k,n) EQUALS
                   OR subset of assignment (i,j))) AND if (tag(assignment(i,j) = safe) then
11:                   Set tag of assignment(k,n) to safe
12:                end if
13:             end if
14:          end if
15:       end for
16:    end for
17: end if
```

Discussion on Algorithm 3

In this algorithm, we find matching assignments for a different supplier with comparable trust levels in other allocations. To do that, first we check the trust level of the suppliers for whom the risk has been calculated in algorithm 1. If that supplier's trust level is comparable with any other supplier than we store those suppliers in a temporary array. And for all suppliers in this list, we do the following. In the next step, we check whether the assignment calculated in algorithm 1 was safe or unsafe. Depending on the result, we do one of the following. If the assignment is unsafe, the trust level of that supplier from algorithm 1 is greater or equal, and the current supplier and the current allocation has no tag, then we add an unsafe tag to the current allocation. Otherwise, if the assignment is safe, the trust level of that supplier from algorithm 1 is lower or equal than the current supplier, and the current allocation has no tag, then we add a safe tag to the current allocation.

These algorithms help to identify the safety of assignments with less calculations through applying the identification rules to identify unsafe assignments without conducting risk assessment. Whenever an unsafe assignment is found, the corresponding allocation is marked as unsafe. In this way, we also avoid checking the other assignments in that allocation. This identification mechanism will be further discussed in the next chapter when we introduce the elimination approaches.

Complexity of Algorithms

The complexity of Algorithm 1 is O(n*m) where n is the number of allocations and m is the number of assignments. The number of assignments in an allocation is always the equal to the number of suppliers. Usually the number of suppliers is not a big number and hence the complexity of this algorithm mostly depends on the number of allocations generated. Algorithm 2 and Algorithm 3 are called in Algorithm 1. The complexity of Algorithm 2 in worst case is $O(n)$ where n is the number of allocation. The complexity of Algorithm 3 is O(s*n) where in worst case s is equal to the number of suppliers and in worst case n is equal to the number of allocations.

Chapter 5

Unsafe Assignment Elimination

In the previous chapter, we have proposed a mechanism for identifying unsafe assignments to avoid unnecessary risk assessment. However, in this chapter, we will show that this approach can be further improved. For this purpose, we f rst take a closer look at the existing approach to selecting optimal allocations in order to better understand its limitations. Then, we propose a series of improved approaches to supplier selection and analyze their advantages and drawbacks. Each approach takes the input information about suppliers and components, and output the allocation in which all the assignments of components to suppliers are safe, and the cost is minimized. The key difference lies in the detailed way that allocations are generated, evaluated, and selected.

5.1 Existing Approach [49]

In the existing approach [49], the supplier capability table and product structure tree are taken as inputs. In the next step, all possible allocations are generated. To identify the unsafe allocations in this approach, after generating all allocations the risk of information leakage is calculated on every allocation. That is, the complicated task of performing risk assessment must be performed for all assignments in all allocations. After that, the calculated risk will be compared with a risk threshold to decide whether each particular assignment is safe. Then, unsafe allocations were removed and safe allocations remain.

Finally, the cost of all safe allocations is calculated incorporating given cost information, and the minimum cost allocation will be selected as the optimum allocation.

The overall process of this approach can be divided into four main steps: collecting input, generating allocations, f ltering allocations, and outputting results, as shown in Figure 3. The input consists of the suppliers' capability and product structure tree. Generating allocation step generates allocations based on information from the input step. Filtering allocation step consists of identifying unsafe allocations, comparing with the risk threshold, eliminating unsafe allocations, and incorporating cost information. The output step consists of comparing the cost and selecting the optimum allocation which represents the optimum distribution of components among suppliers.

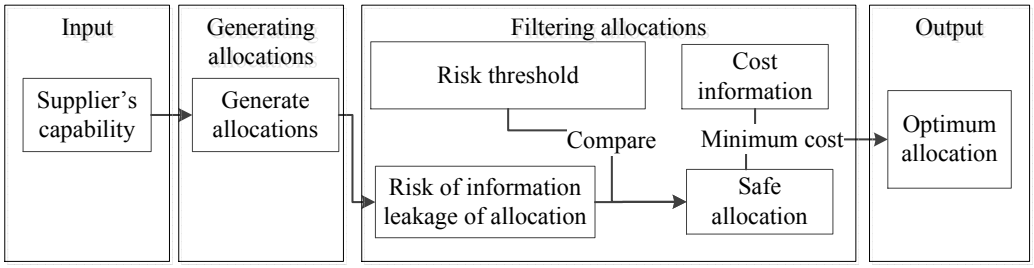

Figure 3: The existing approach [49]

As we have already shown in Section 4.1, the main drawback of this approach is that risk assessment must be performed on all possible allocations, which is usually prohibitive in practice since the number of such allocations increases exponentially as the inputs (the number of components, suppliers and their capabilities) increase.

In our proposed approaches we have shown how to get the set of safe allocation. This is a critical part of f ltering allocation to avoid large number of calculations. However, there is one more f ltering step which is required to get the optimum allocation. To get the optimum allocation, the costs of safe allocations are compared and the minimum cost allocation is selected as optimum allocation. The process of cost f ltering can be done by sorting the cost of all safe allocation from low to high and select the lowest one.

5.2 Brute Force Approach

In the previous chapter, we have proposed four rules and a mechanism for identifying unsafe assignments and avoiding the risk assessment on those assignments. Now we describe the process of employing this identif cation mechanism for saving computational efforts. Since it still assumes all possible allocations have already been generated, we call this the brute force approach. This approach will modify the f lter allocation step mentioned in the existing approach described in Section 5.1. In this new approach, at the time of calculating the risk of information leakage, we do not calculate it for all possible assignments and allocations. Instead, we apply the identif cation mechanism, such that many unsafe assignments can be eliminated without performing risk assessment. The block diagram of this approach is shown in Figure 4. The process is described in algorithm and the way it works is explained below through an example.

Algorithm 4 Brute force approach

Require: supplier's capability, product structure tree;
Ensure: Safe allocations;
1: Generate allocations
2: Find unsafes
3: Remove unsafe allocation
4: Find optimum allocation

Complexity of Algorithm 4

The complexity of this algorithm is determined by several steps: generate allocations, f nd unsafes and f nd optimum allocation. As we indicated in [49] the complexity of generating allocations depends on the number of ECS, the average size of ECS and the number of suppliers. The complexity of f nding unsafe assignments is described previously in this thesis. The complexity of f nding optimum allocation is O(n) as was indicated in [49].

Example 10. *The input is suppliers capability shown earlier in Table 8. Similar to the previous approach, we will still need to generate all possible allocations, which is shown in Table 9. Also, Table 10 shows the allocations from suppliers' point of view (that is,*

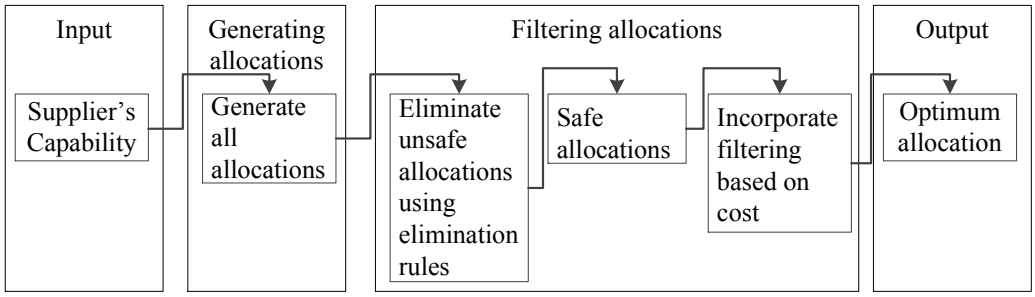

Figure 4: The brute force approach

assignments). This second view (assignments) will be particularly useful here, because it shows exactly which component set is given to each supplier in each allocation. Once we have obtained the table of assignments, the next step is for risk assessment. We do not calculate the risk of information leakage of all allocations at this point. Instead, we select the f rst allocation and calculate the risk of its assignments one at a time. For simplicity purpose, here we will only calculate risk for the allocation shown in table 18.

Table 18: An example allocation

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_1	n_1,n_4	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}

After calculating the risk of this f rst allocation, assume that we obtain the result shown in Table 19.

Table 19: Risk information about the assignments in allocation mentioned in table 18

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_1	*unsafe*	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}

Next, we can apply the duplicate identif cation rule, which is introduced in the previous chapter, to Table 18 and Table 19 in order to obtain the risk of information leakage of some other assignments that appear in table 10. In doing so, we add a tag to each assignment about the risk information of that allocation. After this is done, we will have Table 20. That is, we have identif ed three other unsafe assignments without performing any additional risk assessment.

Table 20: Duplicate-based Identif cation of unsafe assignments

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_1	*UNSAFE*	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_2	*UNSAFE*	ϕ	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_3	*UNSAFE*	ϕ	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_4	*UNSAFE*	ϕ	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_9	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{11}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{17}	ϕ	n_1.n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{18}	ϕ	n_1,n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{19}	ϕ	n_1,n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{20}	ϕ	n_1,n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{21}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{23}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

Next, since an allocation is unsafe it includes at least one unsafe assignment, we check which allocations have at least one unsafe assignment and delete such allocations so we only keep those allocations which consist of only safe assignments. After removing the unsafe allocations we have table 21.

After this, we apply the trust level and subset identif cation rules introduced in the previous chapter. For this example, suppose that, supplier s_0 has a higher trust level than supplier s_1. Using the rules mentioned earlier, we can f nd more unsafe assignments, as shown in Table 22.

Again, the allocations having one or more unsafe assignments will be deleted and thus we have Table 23.

Table 21: After eliminating the unsafe allocations

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_9	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{11}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{17}	ϕ	n_1, n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{18}	ϕ	n_1,n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{19}	ϕ	n_1,n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{20}	ϕ	n_1,n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{21}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{23}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

We need to repeat above steps until no more unsafe allocations can be removed. For the next iteration, we will scan the f rst allocation from Table 23, which is shown in table 24.

Suppose, the result of risk assessment in the 2nd iteration is shown in Table 25. In this iteration, we will have to perform two calculations only, because the information about the risk of many allocations is already known from the calculations of the previous iteration.

Based on the calculations, we apply the identif cation rules to Table 23 and we update the risk information about corresponding assignments. The result is shown in Table 26.

In this iteration, we cannot f nd any unsafe allocation. We go to the next iteration. In this next iteration, we again do risk calculation on one allocation, followed by duplicate-based identif cation and trust level and subset-based identif cation. After these are repeated several iterations, we will be left with only safe allocations, and the process terminates at that point. Suppose at the end of the process, we get all the safe allocations represented in

Table 22: Trust level and subset identif cation

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_9	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{11}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{17}	ϕ	*UNSAFE*	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{18}	ϕ	*UNSAFE*	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{19}	ϕ	*UNSAFE*	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{20}	ϕ	*UNSAFE*	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{21}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{23}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

Table 27. The intermediate steps are omitted for simplicity.

We can observe that, by applying this brute force approach, we only need to calculate the risk of information leakage for 15 allocations, and by we can save the efforts of calculating the risk of other 225 assignments using our identif cation mechanism. That is, we only need about 6.25% of computational effort compared to the existing approach mentioned in the previous section. Cleary, our method is much more eff cient.

However, the main drawback of this brute force approach is that, we will still need to generate all the possible allocations, which means exponential complexity, and we will also need to go through a large number of assignments in each iteration in order to check whether those are safe. In the next sections, we will improve this brute force approach by employing the identif cation mechanism inside the allocation generation process in order to generate comparatively less allocations.

Table 23: After eliminating the unsafe allocations

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_9	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	n_1	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{11}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	n_1	ϕ	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{21}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	ϕ	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{23}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	ϕ	n_1	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

Table 24: First allocation from table 23

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}

5.3 Partially Proactive Approach

In the partially proactive approach, we will modify the generating allocation step mentioned previously in section 5.2 to make the process more eff cient. In this approach, we precalculate some safety information to identify and consequently avoid generating certain unsafe assignments in order to avoid generating corresponding unsafe allocations. In this way, we will generate fewer allocations compared to the brute force approach. Also, the f ltering process that has been described in the previous section for the brute force approach will have a smaller amount of data to work with and as a result it will take less time. Hence, the

Table 25: Risk calculation result:2nd iteration

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	*SAFE*	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}

Table 26: Duplicate-based identif cation in the 2nd iteration

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	*SAFE*	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	*SAFE*	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	*SAFE*	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	*SAFE*	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_9	*SAFE*	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{10}	*SAFE*	ϕ	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{11}	*SAFE*	ϕ	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{12}	*SAFE*	ϕ	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{21}	*SAFE*	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{22}	*SAFE*	n_1	n_2	n_3	n_4,n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{23}	*SAFE*	n_1	n_2	n_3	n_4	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{24}	*SAFE*	n_1	n_2	n_3	n_4	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

partially proactive approach will be more time eff cient compared to brute force approach.

In this new approach, a supplier capability set, for example, Table 28, will f rst be generated from the supplier capability table. Then, one or more allocations are generated using the supplier capability function. The purpose of generating these are to identify unsafe allocations from the very beginning in order to avoid generating such unsafe allocations. When we identify unsafe allocations based on the allocations generated so far we remove them

Table 27: Output: All safe allocations

	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9
A_5	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_6	n_1	n_4	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_7	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_8	n_1	n_4	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}
A_{13}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12}	ϕ	n_{42}	n_{43}
A_{14}	n_4	n_1	n_2	n_3	n_{41}	n_{11}	n_{12},n_{42}	ϕ	ϕ	n_{43}
A_{15}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12}	n_{41}	n_{42}	n_{43}
A_{16}	n_4	n_1	n_2	n_3	ϕ	n_{11}	n_{12},n_{42}	n_{41}	ϕ	n_{43}

immediately from the supplier capability set. And then we use the updated supplier capability set instead of the original supplier capability function to generate allocations. In this way, we could avoid generating many unsafe allocations. Filtering allocations under this approach will be the same as discussed in the brute force approach. The partially proactive approach is presented with a block diagram in Figure 5. The process is described formally with Algorithm 5.

Algorithm 5 Brute force approach

Require: supplier's capability, product structure tree;
Ensure: Safe allocations;
1: Generate suppliers total capability
2: Generate few allocations
3: Find unsafe assignments
4: Update suppliers capability
5: Generate allocations
6: Find unsafe assignments
7: Remove unsafe allocation
8: Find optimum allocation

Complexity of Algorithm 5

The complexity of Algorithm 5 is determined by several steps: generate suppliers total capability, generate few allocations, f nd unsafe allocations, f nd unsafe assignments, generating rest of the allocations, f nding unsafe allocations and f nding optimum allocations. The complexity of most of the steps are discussed before. The complexity of step 1 is $O(2^n)$. The complexity of step 2 and 3 will depend on how much allocations we are generating in this step. The worst case complexity is discussed before. As only few allocations will be generated in this algorithm, so, the effect can be ignored. Other steps are discuss in this thesis.

Example 11. *The input supplier capability is given as in Table 8. Using this table, the supplier capability set is generated as shown in Table 28. One allocation is generated as in table 18 on which risk assessment is performed and the calculated risk is shown in Table 19. Here we can see that assigning component set* n_1,n_4 *to supplier* s_0 *has a*

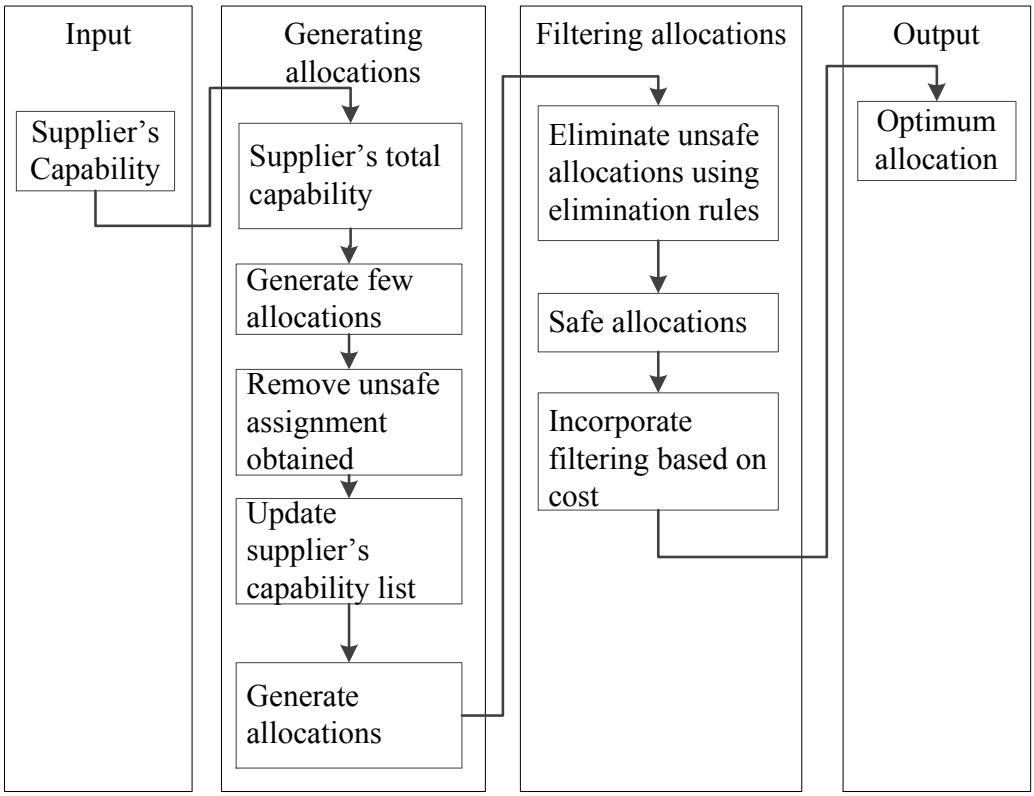

Figure 5: Partially proactive approach

high risk of information leakage. So, component set n_1,n_4 *will be removed from supplier* s_0*'s capability set. Again, by applying the identif cation rules, component set* n_1,n_4 *is removed from supplier* s_1*'s capability set as* s_1 *has a lower trust level than supplier* s_0*. The updated table is given in Table 29. After that, this updated suppliers capability set is used to generate allocations. This time the allocation table generated has only 16 allocations as shown in Table 26, whereas the allocation table generated using the previous brute force approach contained 24 allocations. Then, we f lter allocations as described in previous approach and obtain all safe allocations.*

The advantage of this method over the previous brute force approach is that, in the generating allocation process, fewer allocations are generated. As a result, the overall running time will be less than the previous approach. The above example shows that, interleaving risk assessment and the generating allocation process resulted in generating as low as 60% allocations compared to before. It also showed that removing unsafe assignments

Table 28: Suppliers capability set

supplier	*Capability set*			
s_0	$\{n_1\}$	$\{n_4\}$	$\{n_1, n_4\}$	$\{\phi\}$
s_1	$\{n_1\}$	$\{n_4\}$	$\{n_1, n_4\}$	$\{\phi\}$
s_2	$\{n_2\}$	$\{\phi\}$		
s_3	$\{n_3\}$	$\{\phi\}$		
s_4	$\{n_4\}$	$\{n_{41}\}$	$\{n_4, n_{41}\}$	$\{\phi\}$
s_5	$\{n_{11}\}$	$\{\phi\}$		
s_6	$\{n_{12}\}$	$\{n_{42}\}$	$\{n_{12}, n_{42}\}$	$\{\phi\}$
s_7	$\{n_{41}\}$	$\{\phi\}$		
s_8	$\{n_{42}\}$	$\{\phi\}$		
s_9	$\{n_{43}\}$	$\{\phi\}$		

Table 29: Updated capability set

supplier	*Capability set*			
s_0	$\{n_1\}$	$\{n_4\}$		$\{\phi\}$
s_1	$\{n_1\}$	$\{n_4\}$		$\{\phi\}$
s_2	$\{n_2\}$	$\{\phi\}$		
s_3	$\{n_3\}$	$\{\phi\}$		
s_4	$\{n_4\}$	$\{n_{41}\}$	$\{n_4, n_{41}\}$	$\{\phi\}$
s_5	$\{n_{11}\}$	$\{\phi\}$		
s_6	$\{n_{12}\}$	$\{n_{42}\}$	$\{n_{12}, n_{42}\}$	$\{\phi\}$
s_7	$\{n_{41}\}$	$\{\phi\}$		
s_8	$\{n_{42}\}$	$\{\phi\}$		
s_9	$\{n_{43}\}$	$\{\phi\}$		

while generating allocations is efficient in terms of the amount of calculations required to filter unsafe allocations as well. In the brute force approach, we went through 240 assignments to apply rules and assess their risk (for some of them we identify and eliminate using identification rules). But in the current partially proactive approach, we only have to deal with 160 assignments. In the brute force approach, a cartesian product over the supplier capability set was used to generate all allocations. In the partial proactive approach, a recursive algorithm can be used to generate allocations from suppliers' capability set. It can be shown that generating allocations using this second approach is more efficient. This idea motivated us to design an Proactive Approach where all risk information is precalculated during generating allocations and we will thus only generate safe allocations. Proactive

Approach is discussed next.

5.4 Proactive Approach

After examining the advantages and disadvantages of previous two proposed approaches, we are motivated to proposed our final approach. This proactive approach generates only safe allocations and thus reduce the computational complexity significantly. In fact, in previous two proposed approaches, we generated possible allocations and, for each allocation, we find which component sets are assigned to a particular supplier. Using that information, the risk of information leakage is calculated. Instead, if we concentrate on the suppliers capability set (Table 28) and assignment table (Table 10), we can see that all the assignments in Table 10 are from table 28 even though they are repeated several times. The reason is that Table 28 includes all the sets of components that can be assigned to each supplier (for example, if supplier s_1 is capable of supplying component n_1 and component n_4, the supplier capability set of s_1 is n_1, n_4, n_1,n_4, and ϕ), so it would be sufficient to calculate all the assignments and components sets simply based on Table 28 and use only safe sets to generate safe allocations.

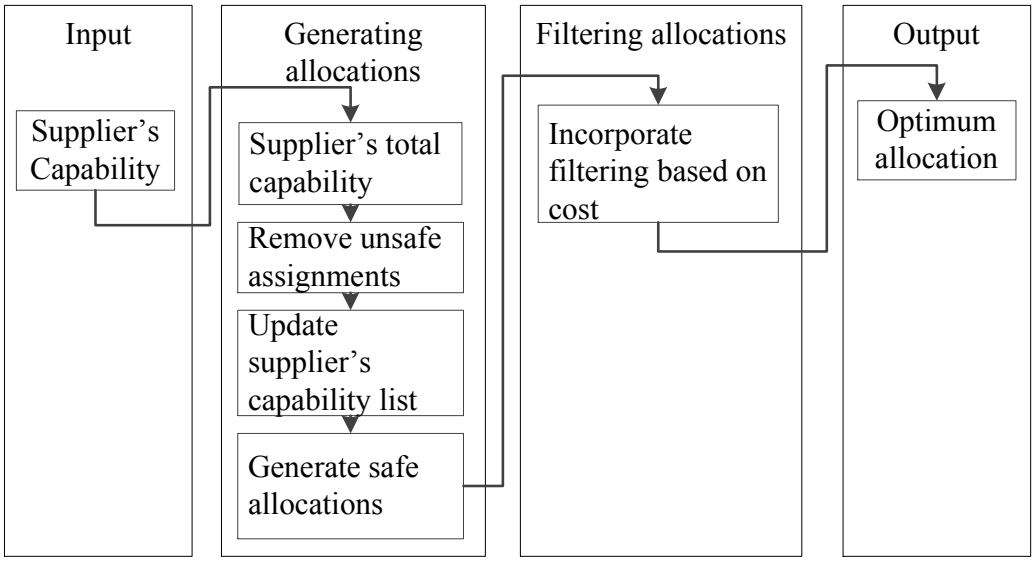

Figure 6: Proactive approach

The process of generating safe allocations only using safe supplier capability set is described as follows. Unlike the previous approaches, this new approach does not generate all allocations first and then delete the known unsafe ones, rather it generates suppliers total capability table first. Then for each supplier it checks the risk of information leakage for each set of components in the total capability table. Then it matches the information calculated with other suppliers using the identification rules. If the supplier's trust level is related to another supplier, then it checks whether there is any match for any set of component in that supplier's total capability list. If a match is found, it adds a tag about safety. If there is any unsafe assignments, then it deletes that assignment to ensure that when we generate alloaction table, there will be no unsafe assignments. After this, the method uses this list to generate allocations. Using a backtrack algorithm it recursively selects all suppliers and tries to assign components to suppliers. The process of generating suppliers safe capability is formally described in Algorithm 6. Algorithm 7 describes the proactive approach.

Algorithm 6 Generate suppliers safe capability table

Require: supplier's total capability, Trust level;
Ensure: Suppliers safe capability table;

```
1:  for each supplier do
2:      for each component set do
3:          calculate risk of information leakage
4:          if any unsafe assignment found then
5:              for this supplier and other suppliers where trust level is related with the trust
                level of this supplier do
6:                  check assignments and mark for elimination
7:                  eliminate all unsafe assignments found
8:              end for
9:          end if
10:     end for
11: end for
```

Algorithm 6 takes supplier's total capability as input and for each supplier explores all the component sets that can be assigned to that particular supplier. For each component set the risk of information leakage is calculated. Note that component set that are assigned to a supplier are refered as assignments. When an unsafe component set (assignment)

is found it is marked as unsafe for the current supplier and also for all those supplier's where the trust levels of supplier is equal or lower than the current supplier. The marking also includes all super sets of the current component set. At the end, all unsafe marked assignments are deleted. And the output is supplier's safe capability table. The complexity of this algorithm is O(max(m,n) where m is the number of suppliers and n is maximum number of component set for worst case. Note that, the number of component set is large than the number of suppliers in real cases. Therefore the complexity becomes O(n).

Algorithm 7 describes formally the procedure of proactive approach. In this algorithm supplier's safe capability is taken as input then it generates only safe allocations. the steps are depicted with f gures with an example. Then the optimum allocation is selected based on the cost information. The complexity of Algorithm 7 is determined by two steps: generating safe allocations and f nding optimum allocation. The worst case complexity of generating safe allocation from supplier's safe capability is O(number of component set for supplier 1 * number of component set for supplier 2 * .... * num of component set for supplier n). Complexity of f nding optimum allocation is O(n) as described in the paper [49].

Algorithm 7 Proactive approach

Require: supplier's safe capability;
Ensure: Safe allocations;
1: Generate safe allocations
2: f nd optimum allocation

Complexity of Algorithm 6 and Algorithm 7

The complexity of Algorithm 6 is $O(n^2 * m)$ in worst case where n is the number of suppliers and m is the number of components. The complexity of Algorithm 7 in the worst case is O(number of components supplier s_1 can supply * number of components supplier s_2 can supply* ...* number of components supplier s_n can supply).

Example 12. *The proactive approach is illustrated through an example shown in Figure 7 through 19. The safe capability table is shown in Table 30. Note that when the same*

component is assigned to two different suppliers there might be a conf ict, as shown in Figure 8.

For this example, the supplier's safe capability (Table 30) is taken as input. In suppliers safe capability we have suppliers and their corresponding components sets which are safe to assign to that particular supplier. The target is to construct safe allocation using the safe capability list. To do this suppliers are selected one after another and safe component set is assigned to them. To make a successful allocation all component is ECS should have been assigned. At f rst supplier s_0 is selected. The corresponding safe capability list is accessed and the f rst component set is assigned to supplier s_0 for this allocation as shown in Figure 7. The next supplier s_1 is selected and component set n_1 is assigned. At this point, there is a conf ict as shown in Figure 8 step 2 because same component (component n_1) is assigned to two different suppliers. So, the next safe component set n_4 is selected for supplier s_1. As there is no conf ict this time, next supplier is selected. For supplier s_2 component set n_2 is selected as shown in Figure 10. Figure 11 through 14 shows that for supplier s_3 component set n_3 is selected, component n_{41} is selected for supplier s_4 , Supplier s_5 is assigned with component n_{11} and supplier s_6 is assigned with component n_{12}. Figure 15 shows another conf ict as component n_{41} was assigned to supplier s_7. Component n_{41} is already assigned to supplier s_4 for this allocation. The next safe component in list for supplier s_7 is ϕ. That means supplier s_7 is not assigned with any components for this allocation. Next in Figure 17 and 18, supplier s_8 and s_9 is assigned with component n_{42} and component n_{43} respectively. Figure 19 shows the complete allocation where all suppliers are assigned with safe component sets. Next the process is repeated to generate remaining safe allocations.

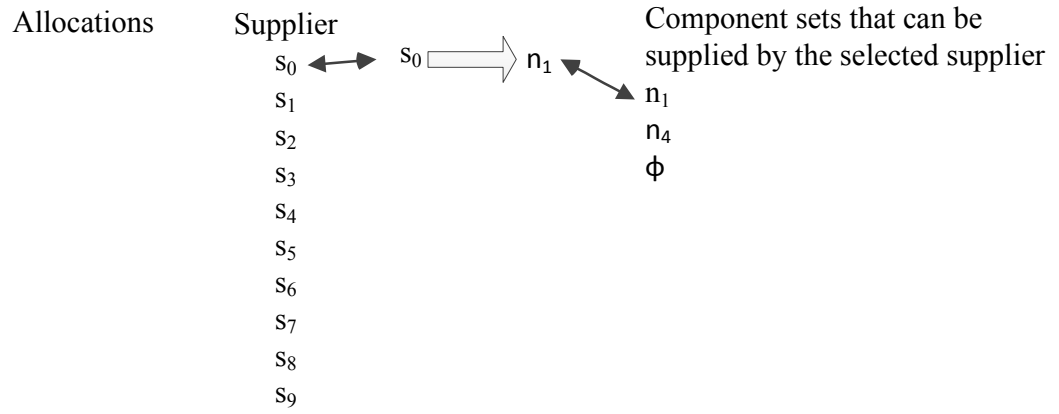

Figure 7: Step one

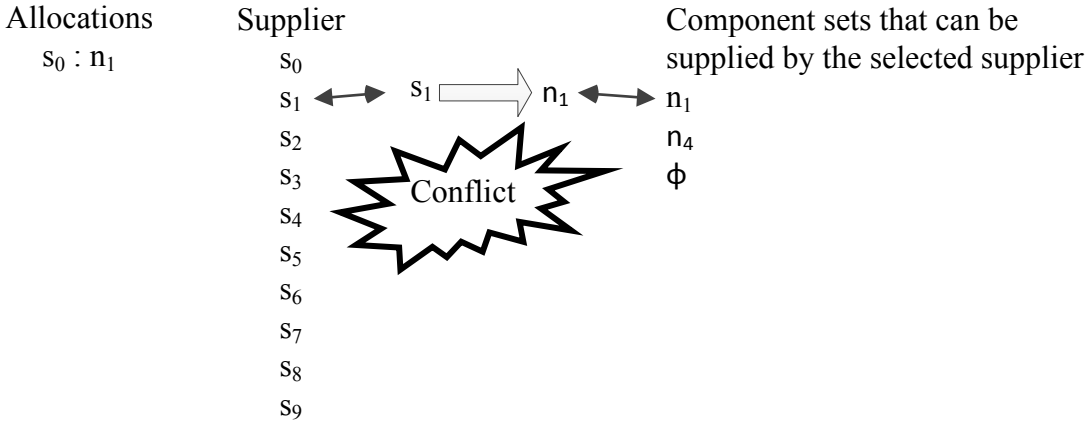

Figure 8: Step two

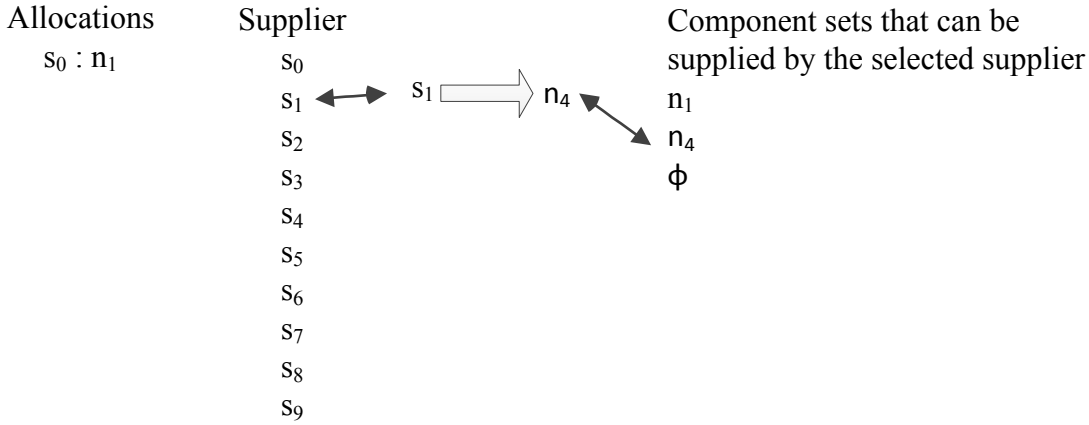

Figure 9: Step three

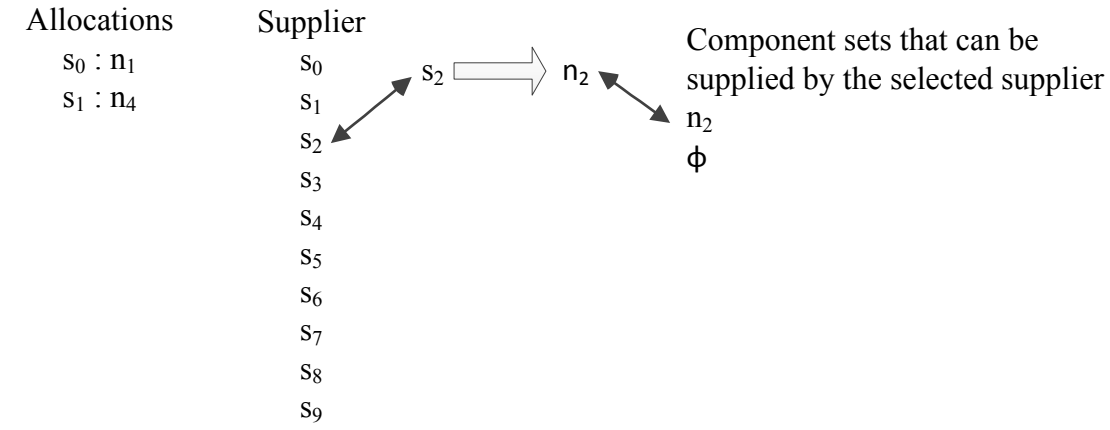

Figure 10: Step four

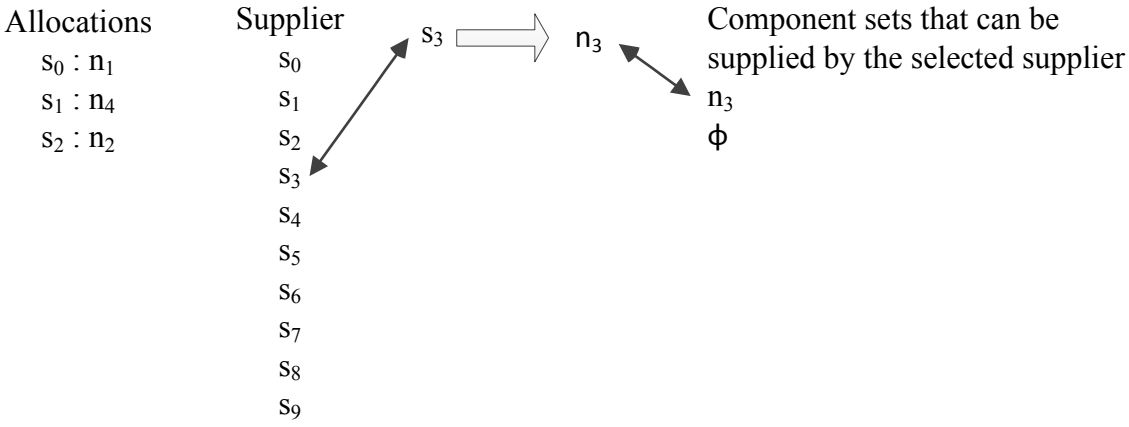

Figure 11: Step f ve

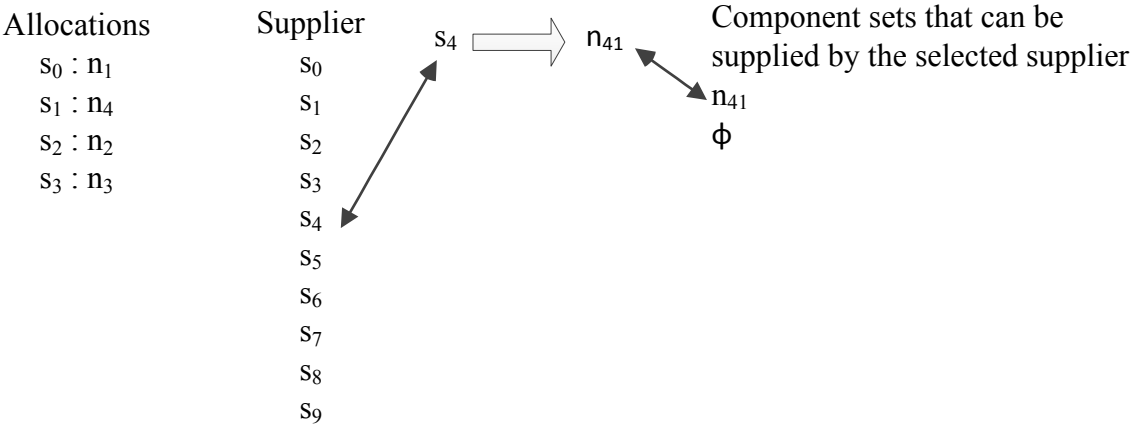

Figure 12: Step six

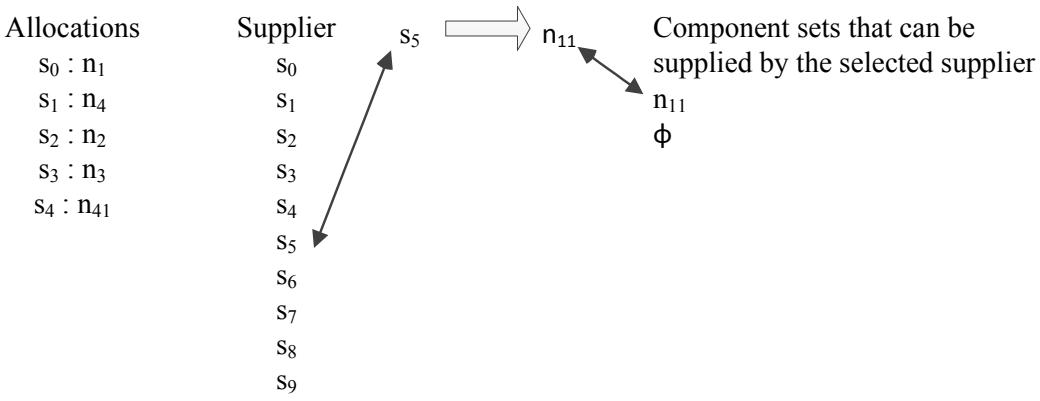

Figure 13: Step seven

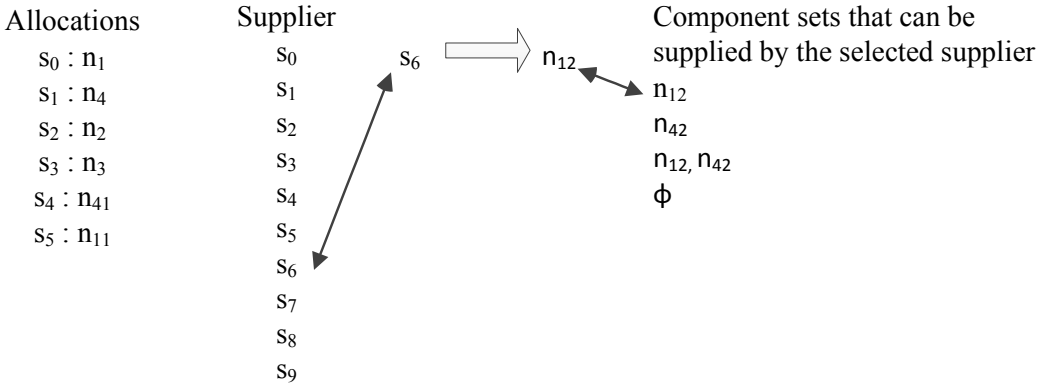

Figure 14: Step eight

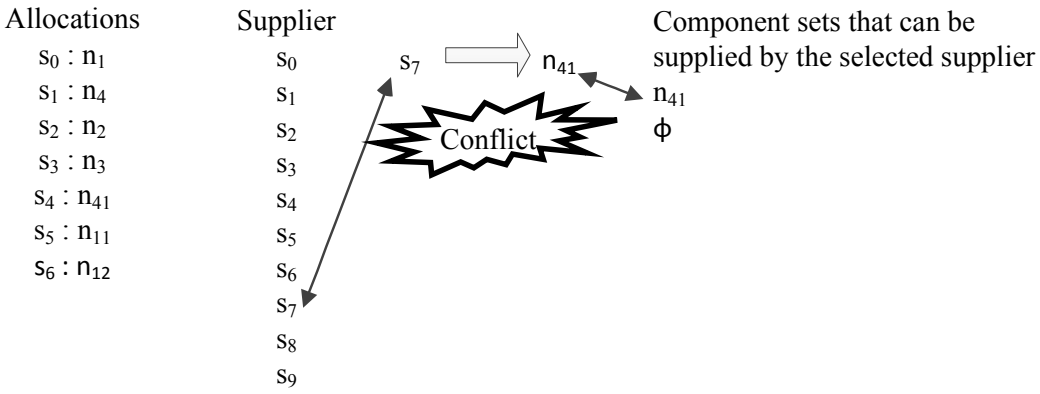

Figure 15: Step nine

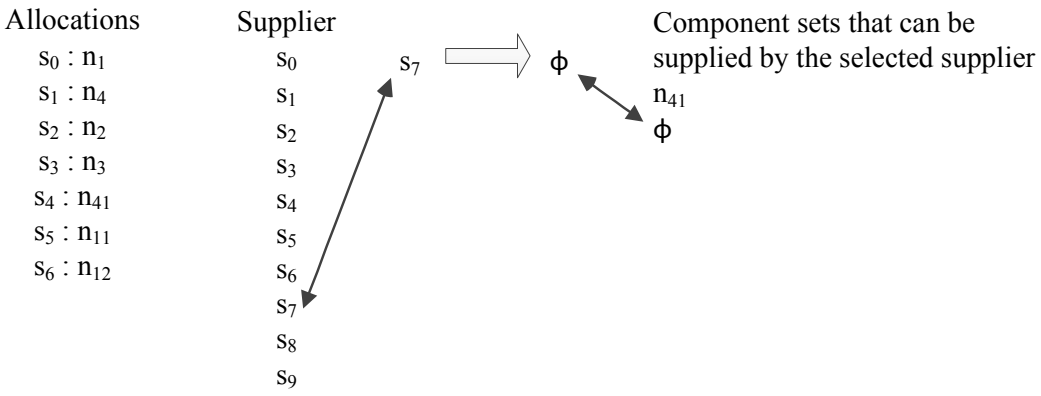

Figure 16: Step ten

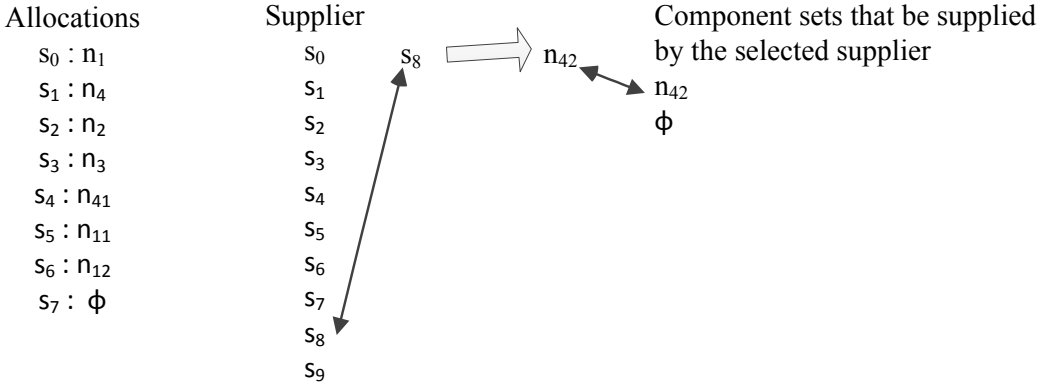

Figure 17: Step eleven

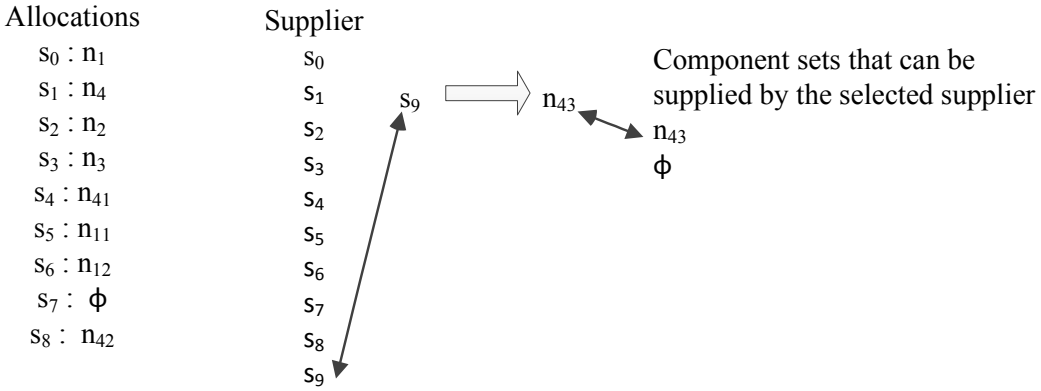

Figure 18: Step twelve

Allocations

$s_0 : n_1$

$s_1 : n_4$

$s_2 : n_2$

$s_3 : n_3$

$s_4 : n_{41}$

$s_5 : n_{11}$

$s_6 : n_{12}$

$s_7 : \phi$

$s_8 : n_{42}$

$s_9 : n_{43}$

Figure 19: Step thirteen

Table 30: Suppliers safe capability set

supplier	Suppliers' safe capability			
s_0	$\{n_1\}$	$\{n_4\}$	$\{n_1, n_4\}$	$\{\phi\}$
s_1	$\{n_1\}$	$\{n_4\}$	$\{n_1, n_4\}$	$\{\phi\}$
s_2	$\{n_2\}$	$\{\phi\}$		
s_3	$\{n_3\}$	$\{\phi\}$		
s_4		$\{n_{41}\}$		$\{\phi\}$
s_5	$\{n_{11}\}$	$\{\phi\}$		
s_6	$\{n_{12}\}$	$\{n_{42}\}$	$\{n_{12}, n_{42}\}$	$\{\phi\}$
s_7	$\{n_{41}\}$	$\{\phi\}$		
s_8	$\{n_{42}\}$	$\{\phi\}$		
s_9	$\{n_{43}\}$	$\{\phi\}$		

Chapter 6

Simulations

In this chapter, we evalute the performance of the proposed approaches to eff cient supplier selection through simulations. Recall that, for a given allocation, our approaches attempt to determine whether it is unsafe by applying these identif cation rules. If none of those rules applies then we will have to calculate the risk of assignments in order to f nd whether the assignments are safe, and f nally only when all assignments in an allocation are safe can the allocation be considered safe. Therefore, We will focus on evaluating the effectiveness of different approaches in employing the identif cation rules of unsafe assignments.

Our algorithm was tested with 5 suppliers. We vary the number of components from 10 to 20 and the number of allocations from 100 to 100000. For simulation we used java as the programming language. The experiment was conducted in a PC with 3.4 GHz core i7 CPU and 16GB memory and each experiment was conducted 100 times. To implement the supplier capability random generator functions were use. Arraylist was used to store supplier, components, assignment and allocations.

The number of generated allocations will determine the amount of efforts for either performing risk assessment or applying identif cation rules on allocations. Figure 20 shows how many allocations will be generated under different approaches. The f gure shows the actual number of allocations generated under the three approaches for different number of allocations. Clearly, the brute force approach generates the maximum number of allocations, while the partial proactive approach generates comparatively fewer allocations,

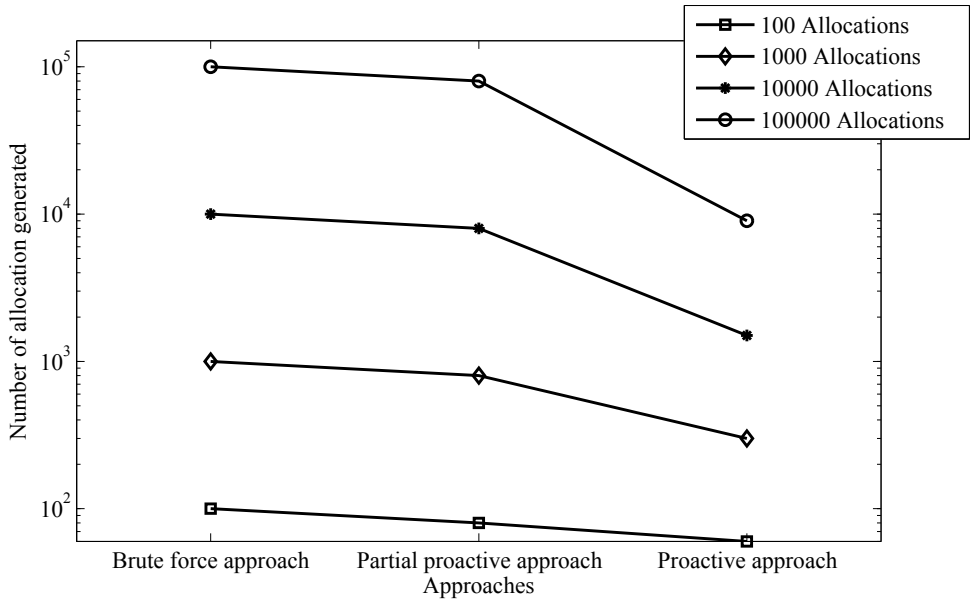

Figure 20: Number of allocations generated under different approaches

and the proactive approach generates signif cantly fewer allocations than both approaches discussed above. Those results show clear evidences that the proactive approach can signif cantly improve the performance (up to an order of magnititude in some cases).

Figure 21 shows the percentage of allocations that will be generated under each approach. The brute force approach (and the previous approach [49]) has a 100% generation rate since both approaches must generate all possible allocations. The partial proactive approach generates around 80% allocations in this case, which is an improvement over the previous two approaches. On the other hand, we see that the proactive approach generates much less, about 10 to 60 percent of allocations. This shows that the percentage of generated allocations depends on how much information was precalculated before generating allocations for the f rst time (the proactive approach precalculates more information and hence the best performance). We can also notice that, while the improvement among the three approaches is almost linear for 100 allocations, it is more signif cant for larger numbers of allocations. We can thus conclude that, for larger inputs, the performance gain of the proactive approach will be more signif cant.

Figure 22 and 23 both show how much calculation is needed for risk assessment under

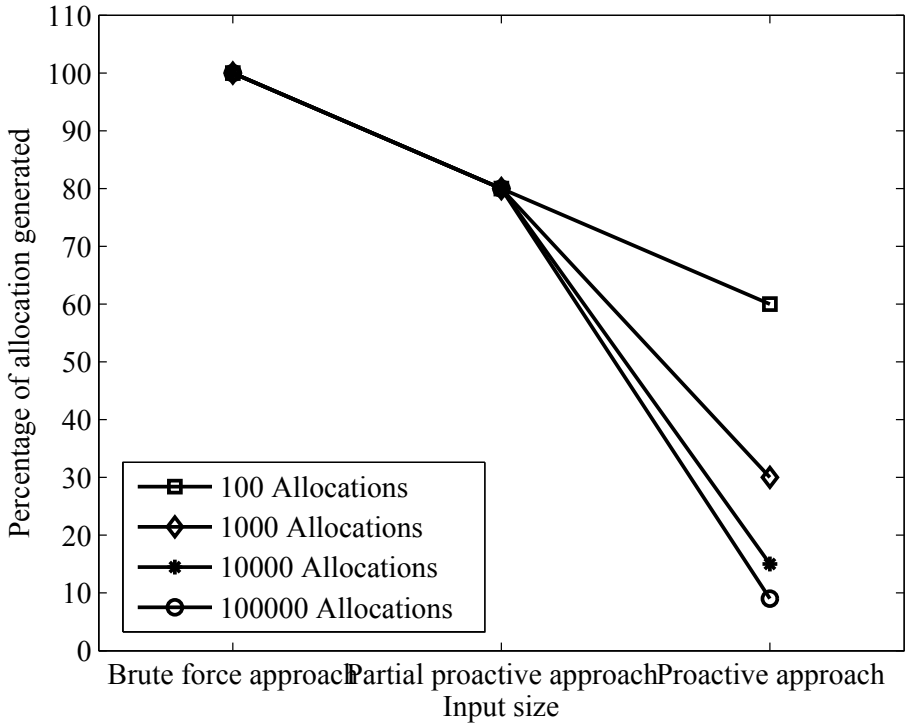

Figure 21: Percentage of allocations generated for different approaches and different input sizes

different approaches and input sizes. It shows that the brute force approach and the partial proactive approach need almost the same amount of efforts for risk assessment. As the number of allocations increases, the amount of required calculations increases almost linearly for the brute force and partial proactive approaches. But for the proactive approach, the rate of increase is much lower. Both figures also show that, for very small number of allocations, initially the proactive approach needs to calculate more than other two approaches do, but for larger numbers of allocations, the proactive approach is more efficient than the other two approaches.

Figure 24 shows the overall runtime of each of the three approaches. All the results are based on about 100,000 allocations. The result shows that, as we increase the number of components, the runtime will increase. However, for the same number of components, the run time of the brute force approach is more than other two approaches. Run time of the partial proactive approach is slightly less than the brute force approach, and run time

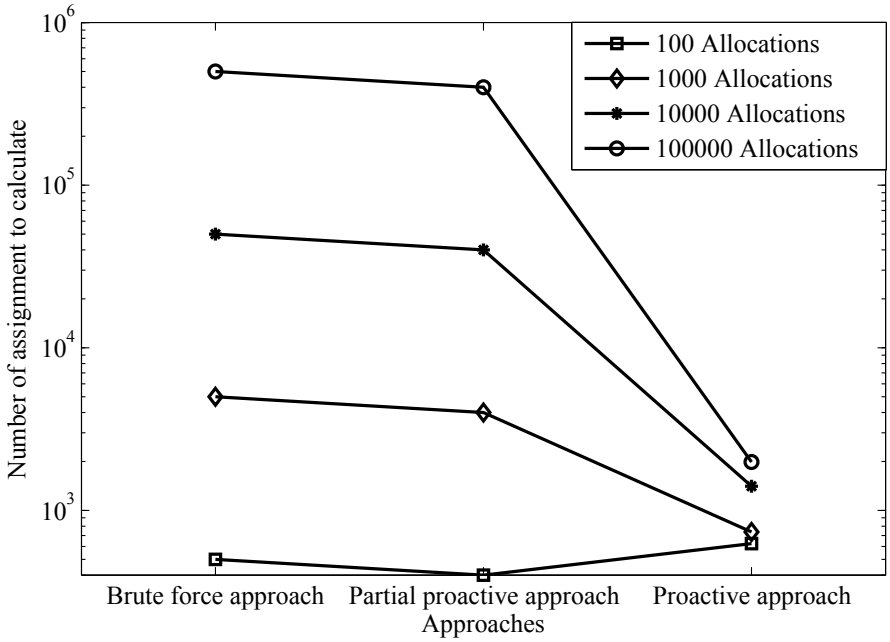

Figure 22: Number of assignments that require risk assessment

of the proactive approach is notably lower than the other two. We can also see that, for 20 components, the proactive approach will require about 100 seconds, which shows the approach to be eff cient enough for practical applications.

Figure 25 shows the run time of the three approaches for different numbers of allocations. The results are based on about 20 components. The f gure shows similar trends as the previous f gure. Both f gures shows that the run time of brute force approach and partial proactive approach is close to each other, whereas the proactive approach takes signif cantly less run time than the other two approaches.

Figure 26 shows the percentage of difference of run time for the brute force approach and the proactive approach. It shows that the proactive approach can save up to 90% of time compared to the brute force approach for large inputs. The percentage of time saving is very close for different numbers of components, whereas it varies from 60% to 90% depending on the number of allocations. From the above analysis, we can conclude that, the proactive approach is more eff cient, especially when there are a large number of allocations.

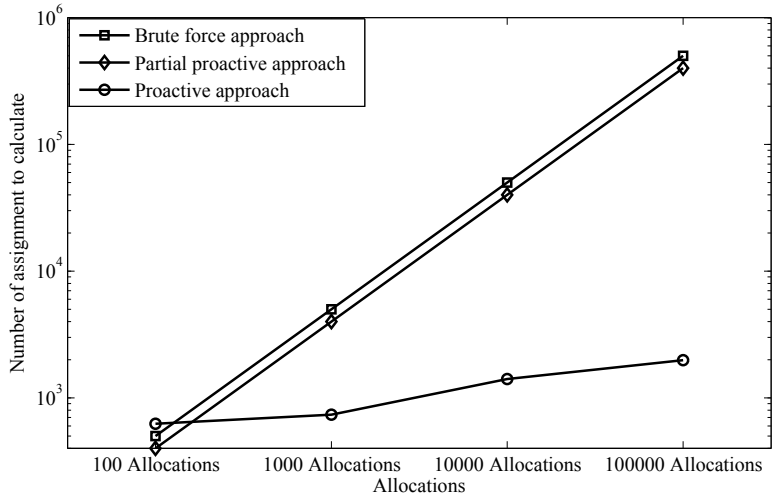

Figure 23: Number of assignments that require risk assessment

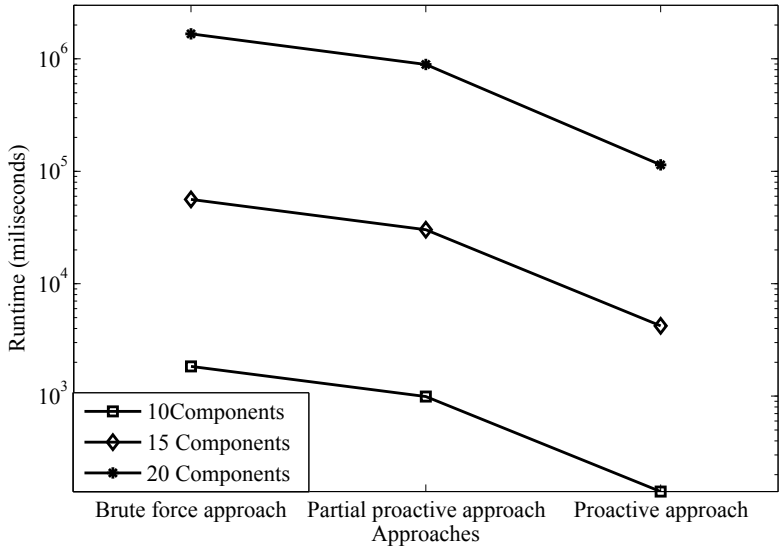

Figure 24: Runtime of different approaches for different numbers of components

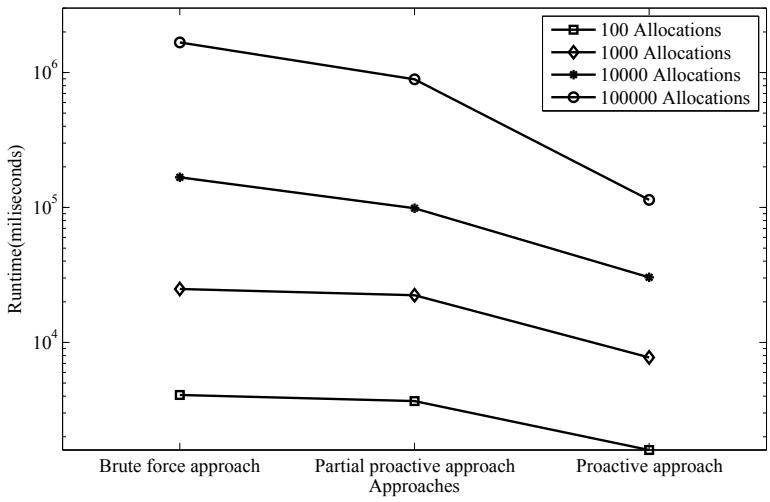

Figure 25: Runtime of different approaches for different numbers of allocations

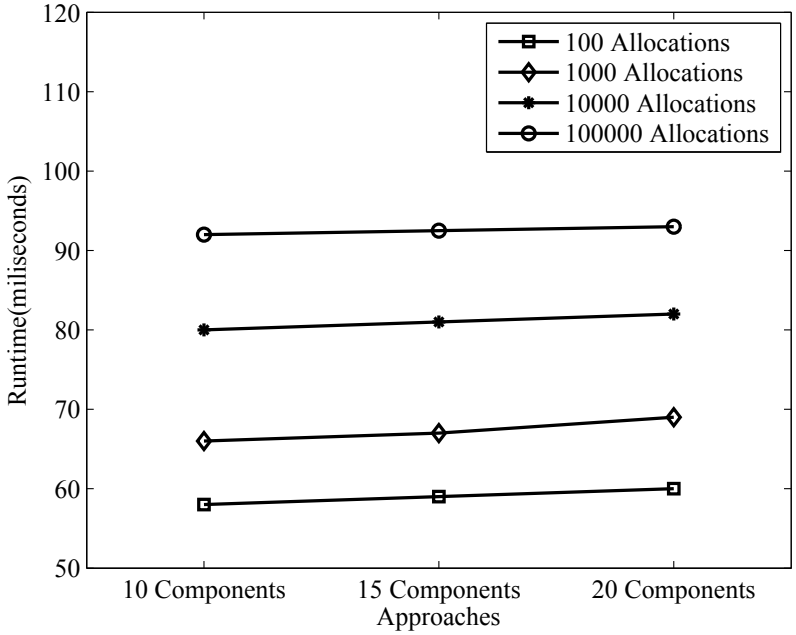

Figure 26: Runtime saving with the proactive approach compared to the brute force approach

Chapter 7

Conclusion and Future Work

7.1 Conclusion

Supplier selection is one of the most important stragtegies to reducing the risk of intellectual property leakage in outsourcing. This thesis aims to propose eff cient approaches of supplier selection in order to improve the practicality of existing approaches. In this context, we consider a multi-level trust model for capturing situations in which suppliers may be endorsed with different levels of trust. To address the scalability issue of the previous approach, we introduce duplication-based, trust-based, and set-based identif cation rules to identify unsafe allocations without risk assessment. Based on these three rules, different approaches are proposed not only to avoid unnecessary risk assessment, but also to reduce the number of allocations that must be generated. Results conf rm that our proposed proactive approach is more eff cient compared to the other methods reported in the literature and by us. Our proposed solution is applicable in any supply chain environment where there is a focal manufacturer, several suppliers with products or tasks that can be decomposed into parts. In fact, this is a general solution and can be modif ed or specialized to address more specif c problems.

7.2 Future Research and Development

The work presented in this thesis provide considerable performance gain in secure supplier selection. Since the topic represents a relatively new research direction in risk management in supply chains, the proposed methods have several limitations that should be improved in future work. In this thesis, we have considered a simple trust model. In future work, we plan to incorporate different factors, including human knowledge, expertise, and reputation to devise better trust models and metrics in order to ref ne our trust-based methods. Moreover, in this thesis we consider the trust level of suppliers as a constant value, but trust levels may vary depending on component types, supplier capabilities, and information sensitivity, so more work are needed to specify trust levels in relation with these factors. Finally, we plan to further evaluate and improve the proposed approaches using real life case studies and data.

Bibliography

[1] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving data mining. *ACM Sigmod Record*, 29(2):439–450, 2000.

[2] Najla Aissaoui, Mohamed Haouari, and Elkaf Hassini. Supplier selection and order lot sizing modeling: A review. *Computers & operations research*, 34(12):3516–3540, 2007.

[3] Krishnan S Anand and Manu Goyal. Strategic information management under leakage in a supply chain. *Management Science*, 55(3):438–452, 2009.

[4] Mikhail J Atallah, Hicham G Elmongui, Vinayak Deshpande, and Leroy B Schwarz. Secure supply-chain protocols. In *E-Commerce, 2003. CEC 2003. IEEE International Conference on*, pages 293–302. IEEE, 2003.

[5] Elisa Bertino, Latifur R Khan, Ravi Sandhu, and Bhavani Thuraisingham. Secure knowledge management: conf dentiality, trust, and privacy. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 36(3):429–438, 2006.

[6] Christian Cachin. Eff cient private bidding and auctions with an oblivious third party. In *Proceedings of the 6th ACM conference on Computer and communications security*, pages 120–127. ACM, 1999.

[7] Tsung-Yi Chen, Yuh-Min Chen, and Hui-Chuan Chu. Developing a trust evaluation method between co-workers in virtual project team for enabling resource sharing and collaboration. *Computers in Industry*, 59(6):565–579, 2008.

[8] Luitzen De Boer, Eva Labro, and Pierangela Morlacchi. A review of methods supporting supplier selection. *European Journal of Purchasing & Supply Management*, 7(2):75–89, 2001.

[9] X. Deng, G. Huet, S. Tan, and C. Fortin. Product decomposition using design structure matrix for intellectual property protection in supply chain outsourcing. *Computers in Industry*, 63(6):632–641, 2012.

[10] Gary W Dickson. An analysis of vendor selection systems and decisions. *Journal of purchasing*, 2(1):5–17, 1966.

[11] Stanley E Fawcett, Stephen L Jones, and Amydee M Fawcett. Supply chain trust: the catalyst for collaborative innovation. *Business Horizons*, 55(2):163–178, 2012.

[12] Stanley E Fawcett, Gregory M Magnan, and Matthew W McCarter. A three-stage implementation model for supply chain collaboration. *Journal of Business Logistics*, 29(1):93–112, 2008.

[13] Joan Feigenbaum, Yuval Ishai, Tal Malkin, Kobbi Nissim, Martin J Strauss, and Rebecca N Wright. Secure multiparty computation of approximations. In *Automata, Languages and Programming*, pages 927–938. Springer, 2001.

[14] Pui Kuen Fong and Jens H Weber-Jahnke. Privacy preserving decision tree learning using unrealized data sets. *Knowledge and Data Engineering, IEEE Transactions on*, 24(2):353–364, 2012.

[15] Kim Giffn. The contribution of studies of source credibility to a theory of interpersonal trust in the communication process. *Psychological bulletin*, 68(2):104, 1967.

[16] Larry C Giunipero and Reham Aly Eltantawy. Securing the upstream supply chain: a risk management approach. *International Journal of Physical Distribution & Logistics Management*, 34(9):698–713, 2004.

[17] Robert B Handf eld and Christian Bechtel. The role of trust and relationship structure in improving supply chain responsiveness. *Industrial marketing management*, 31(4):367–382, 2002.

[18] William Ho, Xiaowei Xu, and Prasanta K Dey. Multi-criteria decision making approaches for supplier evaluation and selection: A literature review. *European Journal of Operational Research*, 202(1):16–24, 2010.

[19] George Q Huang, Jason SK Lau, and KL Mak. The impacts of sharing production information on supply chain dynamics: a review of the literature. *International Journal of Production Research*, 41(7):1483–1517, 2003.

[20] Uta Jüttner. Supply chain risk management: understanding the business requirements from a practitioner perspective. *International Journal of Logistics Management, The*, 16(1):120–141, 2005.

[21] Uta Jüttner, Helen Peck, and Martin Christopher. Supply chain risk management: outlining an agenda for future research. *International Journal of Logistics: Research and Applications*, 6(4):197–210, 2003.

[22] Murat Kantarcioglu, Chris Clifton, et al. Privacy-preserving distributed mining of association rules on horizontally partitioned data. *IEEE transactions on knowledge and data engineering*, 16(9):1026–1037, 2004.

[23] Cemalettin Kubat and Baris Yuce. A hybrid intelligent approach for supply chain management system. *Journal of Intelligent Manufacturing*, 23(4):1237–1244, 2012.

[24] Ik-Whan G Kwon and Taewon Suh. Factors affecting the level of trust and commitment in supply chain relationships. *Journal of Supply Chain Management*, 40(1):4–14, 2004.

[25] Ik-Whan G Kwon and Taewon Suh. Trust, commitment and relationships in supply chain management: a path analysis. *Supply Chain Management: An International Journal*, 10(1):26–33, 2005.

[26] Julien Laganier and PV-B Primet. Hipernet: a decentralized security infrastructure for large scale grid environments. In *Grid Computing, 2005. The 6th IEEE/ACM International Workshop on*, pages 8–pp. IEEE, 2005.

[27] Hau L Lee and Seungjin Whang. Information sharing in a supply chain. *International Journal of Manufacturing Technology and Management*, 1(1):79–93, 2000.

[28] KK Leong, KM Yu, and WB Lee. A security model for distributed product data management system. *Computers in Industry*, 50(2):179–193, 2003.

[29] H Li and Y Geng. Conf dential information protection for industry design. Technical report, Concordia Institute for Information Systems Engineering, Concordia University, Montreal, 2008.

[30] Lode Li. Information sharing in a supply chain with horizontal competition. *Management Science*, 48(9):1196–1212, 2002.

[31] Archie Lockamy III and Kevin McCormack. Analysing risks in supply networks to facilitate outsourcing decisions. *International Journal of Production Research*, 48(2):593–611, 2010.

[32] Durgesh Kumar Mishra and Manohar Chandwani. Arithmetic cryptography protocol for secure multi-party computation. In *SoutheastCon, 2007. Proceedings. IEEE*, pages 22–22. IEEE, 2007.

[33] Durgesh Kumar Mishra and Manohar Chandwani. A zero-hacking protocol for secure multiparty computation using multiple ttp. In *TENCON 2008-2008 IEEE Region 10 Conference*, pages 1–6. IEEE, 2008.

[34] Sylvia Osborn, Ravi Sandhu, and Qamar Munawer. Conf guring role-based access control to enforce mandatory and discretionary access control policies. *ACM Transactions on Information and System Security (TISSEC)*, 3(2):85–106, 2000.

[35] Ilan Oshri, Julia Kotlarsky, and Leslie P Willcocks. *The handbook of global outsourcing and offshoring*. Palgrave Macmillan, 2011.

[36] Rohit Pathak and Satyadhar Joshi. Secure multi-party computation using virtual parties for computation on encrypted data. In *Advances in Information Security and Assurance*, pages 412–421. Springer, 2009.

[37] Benny Pinkas. Fair secure two-party computation. In *Advances in CryptologyEurocrypt 2003*, pages 87–105. Springer, 2003.

[38] Shariq J Rizvi and Jayant R Haritsa. Maintaining data privacy in association rule mining. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 682–693. VLDB Endowment, 2002.

[39] Bidya S Sahay. Understanding trust in supply chain relationships. *Industrial Management & Data Systems*, 103(8):553–563, 2003.

[40] Paul H Schurr and Julie L Ozanne. Inf uences on exchange processes: buyers' preconceptions of a seller's trustworthiness and bargaining toughness. *Journal of Consumer Research*, pages 939–953, 1985.

[41] Latanya Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):571–588, 2002.

[42] Jaideep Vaidya and Chris Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 639–644. ACM, 2002.

[43] Von Welch, Frank Siebenlist, Ian Foster, John Bresnahan, Karl Czajkowski, Jarek Gawor, Carl Kesselman, Sam Meder, Laura Pearlman, and Steven Tuecke. Security for grid services. In *High Performance Distributed Computing, 2003. Proceedings. 12th IEEE International Symposium on*, pages 48–57. IEEE, 2003.

[44] Scott H Williams. Collaborative planning, forecasting, and replenishment. *Hospital materiel management quarterly*, 21(2):44–51, 1999.

[45] Li Xiong, Subramanyam Chitti, and Ling Liu. Preserving data privacy in outsourcing data aggregation services. *ACM Transactions on Internet Technology (TOIT)*, 7(3):17, 2007.

[46] Ting Yu, Marianne Winslett, and Kent E Seamons. Supporting structured credentials and sensitive policies through interoperable strategies for automated trust negotiation. *ACM Transactions on Information and System Security (TISSEC)*, 6(1):1–42, 2003.

[47] Y Zeng and P Gu. A science-based approach to product design theory part ii: formulation of design requirements and products. *Robotics and Computer-Integrated Manufacturing*, 15(4):341–352, 1999.

[48] Y. Zeng, L. Wang, X. Deng, X. Cao, and N. Khundker. Secure collaboration in global design and supply chain environment: problem analysis and literature review. *Computers in Industry*, 63(6):545–556, 2012.

[49] D.Y. Zhang, X. Cao, L. Wang, and Y. Zeng. Mitigating the risk of information leakage in a two-level supply chain through optimal supplier selection. *Journal of Intelligent Manufacturing*, 23(4):1351–1364, 2012.

[50] D.Y. Zhang, Y. Zeng, L. Wang, H. Li, and Y. Geng. Modeling and evaluating information leakage caused by inferences in supply chains. *Computers in Industry*, 62(3):351–363, 2011.

[51] Hongtao Zhang. Vertical information exchange in a supply chain with duopoly retailers. *Production and Operations Management*, 11(4):531–546, 2002.

[52] Huixia Zou and Tao Yu. The research on decision model of supply chain collaboration management. In *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM'08. 4th International Conference on*, pages 1–6. IEEE, 2008.