DOI 10.1007/s10845-011-0527-3

Mitigating the risk of information leakage in a two-level supply chain through optimal supplier selection

Da Yong Zhang · Xinlin Cao · Lingyu Wang · Yong Zeng

Received: 29 March 2010 / Accepted: 12 March 2011 / Published online: 31 March 2011

Abstract Information leakage in supply chains is drawing more and more attention in supply chain management. Unlike existing research, which usually focuses on the effect of information leakage on the supply chain's material and information flow, this paper aims to evaluate and mitigate the risk of information leakage. First, we formulate the problem of information leakage caused by inferences in a two-level supply chain where potential competition may exist between a supplier and the manufacturer. Second, we propose a method to mitigate the risk of such information leakage through optimal supplier selection. An example is used to demonstrate the problem and our proposed approach.

Keywords Information sharing · Information leakage · Inference · Risk mitigation · Optimal supplier selection

Introduction

Information leakage, which refers to the unintentional revelation of confidential information to an unauthorized party, is drawing more and more attention in supply chain management. Many authors have discussed the effect of information leakage on the supply chain's material and information flow (Lee and Whang 2000; Li 2002; Zhang 2002; Hoecht and Trott 2006; Anand and Goyal 2009). However, research has not been reported about how information leakage happens in supply chains and how to mitigate the risk of leakage if it has negative effects.

Supply chains, as a complex networks, are usually studied from different structural perspectives, such as dyadic, serial, divergent, convergent and network (Huang et al. 2003). Considering the structures of supply chains, information leakages may happen inside a supply chain partner, between the upstream and downstream partners in a dyadic, serial, divergent or convergent supply chain or between partners at the same level through the partners at an upper or lower level in a divergent or convergent supply chain.

In this paper, we first formulate the problem of information leakage caused by inferences in a two-level supply chain, within which potential competition may exist between a supplier and the manufacturer. We then propose a novel solution for mitigating the risk of such information leakage through optimal supplier selection by considering the constraints of product structure, supplier capability and cost. An example taken from process industry is used to illustrate how the proposed approaches work in practice.

The main contribution of the present paper is twofold. First, the formulation of information leakage in supply chain provides a better understanding of this pertinent issue. Second, by incorporating the information leakage issue into the supplier selection problem, we make it possible to address the former issue by adapting many well known solutions to the latter problem (as surveyed in Barnhart et al. 1998). Our present focus is to ensure the generality of algorithms while their complexity is not our current concern. Nonetheless, an analysis of the complexity of the proposed algorithms is conducted, based on which heuristic optimization methods can be developed.

The rest of this paper is organized as follows. Section "Related work" reviews the work in four related areas: risk management, information leakage, information leakage prevention and supplier selection. Section "The model" describes the two-level supply chain that is studied in this

D. Y. Zhang · X. Cao · L. Wang · Y. Zeng (✉)
Concordia Institute for Information Systems Engineering, Faculty of Engineering and Computer Science, Concordia University, Montreal, QC H3G 1M8, Canada
e-mail: zeng@ciise.concordia.ca

paper. Section “Supplier selection” devises an approach of supplier selection to mitigating the risk of information leakage caused by inferences in a two-level supply chain. Section “Example” presents an example by applying our approach to a product in the process industry. The last section concludes the paper and indicates future work.

Related work

Information leakage is one type of risk in a supply chain. Based on our previous research on the modeling and evaluating of information leakage in a supply chain (Zhang et al. 2011; Sun et al. 2010), this paper aims to mitigate the risk of leakage through optimal supplier selection. In this section, we review the existing research in four related areas in the context of (SCRM): risk management, information leakage, information leakage prevention, and supplier selection.

Risk management in supply chains

SCRM is a relatively new and growing research area. Some definitions of SCRM have been proposed by Giunipero and Eltantawy (2004) and Juttner (2005). Juttner et al distinguished four basic constructers of SCRM: supply chain risk sources, risk consequences, risk drivers, and mitigating strategies (Juttner et al. 2003). Based on those four constructers, four critical aspects of SCRM were identified: assessing the risk sources, identifying the risk concept, tracking the risk drivers, and mitigating risks. Recently, Neiger et al stated that the purposes of SCRM research is to develop “approaches for identification, assessment, analysis and treatment of areas of vulnerability and risk in supply chains” (Neiger et al. 2009).

Svensson considered the vulnerability as a result of the time- and functional dependencies between firms’ activities and resources in supply chains, based on which he proposed that the vulnerability may be measured and evaluated by four principal dimensions, namely service level, deviation, consequence and trend (Svensson 2000, 2002).

Juttner et al suggested that supply chain risk sources fell into three categories: environmental, network-related and organizational (Juttner et al. 2003). Mason-Jones and Towil and Juttner used a classification of five categories: environment, supply, demand, process and control (Mason-Jones and Towill 1998; Juttner 2005). Lockamy III and McCormack classified supply chain risk sources into three categories: operational, network and external (Lockamy and McCormack 2010).

Neiger et al proposed a methodology to identify process-based risks in supply chains based on the principles of Value-Focused Process Engineering (VFPE) (Neiger et al. 2009). Zsidisin et al. conducted seven case studies to analyze supply risk assessment techniques (Zsidisin et al. 2004). Lockamy III and McCormack presented a methodology for analyzing risks in supply networks to facilitate outsourcing decisions associated with revenue impact (Lockamy and McCormack 2010).

Juttner et al. adapted four generic risk mitigating strategies for single organizations to supply chains, namely avoidance, control, cooperation and flexibility (Juttner et al. 2003). Christopher and Lee suggested that improved confidence is one key element in any strategy to mitigate supply chain risk (Christopher and Lee 2004). Zsidisin and Smith conducted a case study of an aerospace supplier and found that early supplier involvement substantially reduced the likelihood of supply disruptions of the supplier (Zsidisin and Smith 2005). Khan et al addressed the importance of the impact of product design on supply chain risk based on an indepth longitudinal case study of a major UK retailer (Khan et al. 2008).

Current research in SCRM usually considers mainly risk sources that may cause supply chain disruptions, such as natural disasters, diseases, and political, social and economical emergencies and crises, while ignoring risk sources that affect supply chains in a less visible manner, like information sharing and information leakage, which are discussed in this paper.

Information leakage in supply chains

In a literature review of information sharing in supply chains, Lee and Whang showed that one manufacturer may leak confidential information to a competitor through the business practice of a common supplier (Lee and Whang 2000). In the meantime, information leakage may also occur when one supplier supports two competing manufactures (Anand and Goyal 2009). Hoecht and Trott discussed the case where a consultant working with multiple clients might use the best practice they acquired from one client to the advantages of other clients (Hoecht and Trott 2006).

Li’s research showed that the leakage effect might discourage retailers from sharing their demand information with the manufacturer (Li 2002). However, Zhang claimed that no information would be voluntarily shared between retailers and the manufacturer; The retailers were willing to share information completely and get side payment for the information sharing when their information was statistically less accurate or they benefited more from the effect of information leakage (Zhang 2002).

In supply chains, information leakage may occur when confidential information can be inferred from shared information due to the inherent engineering relationships between different pieces of information. Zhang et al examined the issue of information leakage caused by inferences and proposed a conceptual model of such information leakage in supply chains (Zhang et al. 2011). On the basis of their conceptual model, they devised a quantitative approach to evaluate the risk of information leakage caused by inferences when a given amount of information is shared in supply chains.

Information leakage prevention

Legal, organizational, social and technical methods are often used to prevent information leakage in supply chains. Existing technical methods can be roughly divided into four categories: access control, sanitization/suppression, generalization and Secure Multi-party Computation (SMC).

An excellent literature review of access control models for general collaborative systems has been conducted by Tolone et al. (2005). Consequently, in this section, we review only those access control models that are relevant for prevention of information leakage in supply chains.

Several access control models have been developed for information sharing and collaboration in supply chains. Leong et al proposed an access control model for a workspace-oriented distributed Product Data Management (PDM) system (Leong et al. 2003). Cera et al. (2004, 2006), and Kim et al. (2006) integrated multi-resolution geometry and Role Based Access Control (RBAC) model (Sandhu et al. 1996; Ferraiolo et al. 2007) to a collaborative 3D assembly design. S-RBDDAC (Wang et al. 2006) combines RBAC and cryptographic methods to protect intellectual properties in collaborative design. Trust is also considered in some access control models (Chen et al. 2008).

In supply chains, companies may use heterogeneous CAD software packages to produce CAD data in collaborative assembly design (Shyamsundar and Gadh 2002; Chen et al. 2004; Kim et al 2004). Companies convert their CAD data in incompatible formats into neutral CAD data, such as in STEP format, to build the final assembly model of the product.

In the literature of privacy protection, Mun et al proposed a skeleton model based method, which represents essential data such as design specifications in an intuitive and explicit manner while it does not reveal data related to intellectual property contained in CAD models (Mun et al. 2009).

SMC protects confidential information by allowing users to perform joint computation on multiple datasets while not revealing information in these datasets (Yao 1986; Goldreich et al. 1987; Lindell and Pinkas 2002). Atallah et al introduced SMC into the area of preventing information leakage in supply chains (Atallah et al. 2003). They proposed several SMC protocols for supply-chain interactions, such as capacity allocation under various policies, and bidding and auctions under both discriminatory and nondiscriminatory pricing.

However, in supply chains, information usually has to be shared to facilitate collaborations between supply chain partners. In most cases, the shared information is valuable only when it is precise enough. Consequently, access control, sanitization, generalization and SMC are usually not very effective for preventing information leakage caused by inferences in supply chains.

Supplier selection

Supplier selection can be considered as a decision-making problem with many constraints such as cost, quality, risk and so on (Kubat and Yuce 2010). de Boer et al. positioned existing literature of supplier selection into a framework that has purchasing situations on one axis and phases in the supplier selection process on the other axis (de Boer et al. 2001). Aissaoui et al. focused on work that employed operations research and computational models for the final stage of the supplier selection process (Aissaoui et al. 2007). They proposed different classifications of decision models existing in the literature according to single or multiple sourcings, criteria, items, periods, objectives, etc. Among all the problems concerned with supplier selection, criteria and techniques for the selection are the most critical.

The study of criteria for supplier selection can be traced back to the 1960s. Dickson identified and ranked the importance of 23 vendor selection criteria based on a survey of purchasing agents and managers from the United States and Canada (Dickson 1966). According to Dickson's study, quality, delivery, performance history, warranties and claim policies, production facilities and capacity, price, technical capability and financial position are extremely or considerably important.

Consequently, some recent research has tried to organize supplier selection criteria hierarchically or in network. Kahraman et al gave four categories: supplier, production performance, service and cost (Kahraman et al. 2003). Huang and Keskar defined seven metric categories: reliability, responsiveness, flexibility, cost and financial, assets and infrastructure, safety and environment (Huang and Keskar 2007). Demirtas and UStun and Lee considered supplier selection criteria under the Benefits, Opportunities, Costs and Risks (BOCR) merits proposed by Saaty (Demirtas and Ustun 2008, 2009; Lee 2009; Saaty 2004).

Many decision-making techniques have been applied to supplier selection. A nonexhaustive list of these techniques includes Linear Weighting, Analytic Hierarchy Process (AHP), Analytic Network Process (ANP), Linear Programming, Mixed Integer Programming, Goal Programming,Multi-Objective Programming, Economical Order Quantity (EOQ), Total Cost of Ownership (TCO), Data Envelopment Analysis (DEA), Quality Fuction Development (QFD), Structure Matrix (Chen and Huang 2007), Cluster Analysis (Li et al. 2009), Case-Based Reasoning (CBR), Genetic Algorithm, Neural Network, Rough Set Theory and Fuzzy Set Theory (Carrera and Mayorga 2008; McCauley-Bell 1999). Many integrated techniques, combining more than one techniques in the list above, were also developed for supplier selection.

The model

Consider a two-level supply chain with one manufacturer and n suppliers. Denote the manufacturer with s_0, let $S_1 \equiv \{s_1, s_2, \ldots, s_n\}$ be a set of suppliers, and let $S = \{s_0\} \cup S_1$. The manufacturer s_0 produces a product, which consists of components. A component as an assembly may also consist of sub components. Each supplier $s_i \in S_1$ has the capabilities of making particular components.

In a two-level supply chain, there may be suppliers in S_1 who are potential competitors of the manufacturer s_0. Without loss of generality, we assume that supplier s_1 is a potential competitor. To facilitate collaboration, the manufacturer s_0 will share some non-confidential information with the supplier s_1. In the meantime, the manufacturer s_0 tries to conceal confidential information from the supplier s_1. However, since there are inherent engineering relationships among different pieces of information about the product, it is possible for supplier s_1 to infer confidential information from the shared non-confidential information and its knowledge of the product.

In this context, shared information and confidential information can be abstracted as "*parameters*". A parameter is an abstract information object that describes an attribute of a system. It may be a product design parameter or any other information object that can be described by a triplet (*name*, *actual value*, *working values*), in which *name* is an identifier of the parameter, *actual value* is the value that the parameter takes in the system and *working values* are the values that if the parameter takes, the system's performance becomes lower but still acceptable.

In the two-level supply chain, the manufacturer s_0 is the holder of confidential parameters and tries to prevent them from being revealed to supplier s_1; supplier s_1 is an inferrer who tries to acquire the working values of confidential parameters protected by the holder s_0.

Supplier s_1 may obtain its knowledge of s_0's confidential parameters through three sources: its initial knowledge, shared parameters and inferences. In this context, we can model knowledge of parameters as probability distributions. The manufacturer s_0 can estimate supplier s_1's knowledge obtained through inferences and evaluate the risk that its confidential information is leaked to supplier s_1 by the algorithms devised in our previous work (Zhang et al. 2011).

If there is only one confidential parameter, the scenario of information leakage caused by inferences in the two-level supply chain can be derived as follows (as shown in Fig. 1).

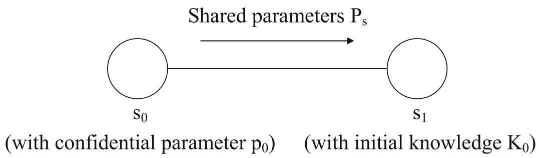

Shared parameters P_s
s_0
(with confidential parameter p_0)
s_1
(with initial knowledge K_0)

Fig. 1 Information sharing and information leakage in the two-level supply chain

(1) The manufacturer s_0 knows the actual value of a confidential parameter p_0. It tries to prevent the actual value of p_0 from revealing to supplier s_1 while it shares a set of parameters P_s with s_1 in some way.

(2) Supplier s_1 does not know the actual value of p_0. It tries to acquire a working value of p_0 by inferring on the basis of its initial knowledge K_0 and knowledge obtained through the sharing of parameters P_s.

(3) The manufacturer s_0 evaluates and mitigates the risk of information leakage caused by modeling and estimating supplier s_1's initial knowledge and knowledge obtained through inferences.

There may be many methods that the manufacturer s_0 can take to mitigate the risk of information leakage caused by inferences. In this paper, we will focus on supplier selection as one approach to mitigating such risk in supply chains.

Supplier selection

In this section, we model supplier selection as an optimization problem where the manufacturer s_0 tries to find an allocation from components to suppliers that has minimum cost while meeting the constraints of product structure, supplier capability and risk of information leakage. A generic process is provided to solve the optimization problem.

Essential component sets

The relations among the product, its components and relevant assembly tasks can be described in an extended product structure tree. There are two types of nodes in an extended product structure tree, component nodes and assembly task nodes. A component node represents a product or a component whereas an assembly task node, which is introduced into product structure tree for the purpose of simplifying issues relevant to assembly activities, represents the task of assembling its parent component (or an assembly). An edge, connecting two component nodes, represents a parent-child relationship between them. A parent component (or an assembly) consists of all its child components. An edge, connecting a component node and an assembly task node, indicates that the component is assembled by the assembly

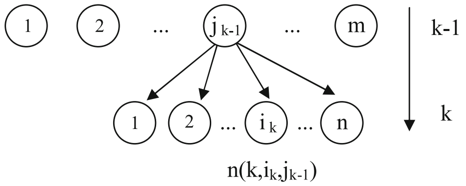


Fig. 2 A basic block of a product structure tree [*Source*: Adapted from Zeng and Gu (1999)]

task. According to Zeng and Gu (1999), a node of a product structure tree can be defined as $n(k, i_k, j_{k-1})$, if the node is at the i_kth 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 nodes together constitute a product structure tree recursively. Figure 2 shows a basic block of product structure tree. Figure 3 shows an extended product structure tree, which describes the relations among major components and assembly tasks of the product given in "Example".

In this paper, an extended product structure tree is denoted as T; all nodes of T are denoted as N_T; the root of T is denoted as $r(T)$. First, we define two functions that will be used in Definition 1: nodes of a subtree $N(n)$ and leaves of a subtree $LN(n)$.

(1) $N(n) = N_{T'}$, where $n \in N_T$, T' is a subtree of T and $r(T') = n$;
(2) $LN(n) = \{n' \mid n' \in N(n)$ and n' is a leaf node$\}$.

Definition 1 (*Essential Component Set* (*ECS*)) T is a product structure tree and $r(T) = n_0$, $\forall N \subseteq N_T$, N is called an essential component set of T, if it satisfies:

(1) $\forall n_i, n_j \in N, i \neq j, LN(n_i) \cap LN(n_j) = \emptyset$;
(2) $\bigcup_{n_i \in N} LN(n_i) = LN(n_0)$;

Obviously, a product structure tree T has at least one ECS. The ECSs of the product structure subtree shown in Fig. 3 include $\{n_0\}$, $\{n_1, n_2, n_3, n_4, n_5\}$, $\{n_1, n_6, n_7, n_8, n_3, n_4, n_5\}$, $\{n_1, n_2, n_3, n_4, n_9, n_{10}, n_{11}, n_{12}\}$ and $\{n_1, n_6, n_7, n_8, n_3, n_4, n_9, n_{10}, n_{11}, n_{12}\}$.

Allocations

We use supplier capability function F_{sc} to describe a supplier's capabilities to supply components and component supplier function F_{cs} to describe components' possible suppliers.

(1) For a product structure tree T, $\forall s \in S$, $F_{sc}(s) = \{n \mid n \in N_T$ and s can supply $n\}$. Table 1 lists all suppliers for the regeneration system of the natural gas dryer and Table 2 gives a F_{sc} function.
(2) For a product structure tree T, $\forall n \in N_T$, $F_{cs}(n) = \{s \mid s \in S$ and $n \in F_{sc}(s)\}$. Table 3 gives a F_{cs} function.

For the sake of simplicity, we assume that if a component is allocated to a supplier, all its child components are also allocated to the same supplier.

Definition 2 (*Allocation*) T is a product structure tree, N_T is the set of all nodes of T, $N \subset N_T$, S is a set of suppliers. A mapping $F_a : N \rightarrow S$ is called an allocation, if it satisfies:

(1) N is an ECS of T;
(2) if $F_a(n) = s$, then $n \in F_{sc}(s)$;

It is not necessarily true that there is such a F_a in all cases; but with an additional condition "$\exists N$, N is an ECS and $N \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$", it can be easily proven that F_a exists.

Lemma 1 (Sufficient condition for existence of allocations) *If N is an ECS of T and $N \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$, then there exists at least one allocation $F_a : N \rightarrow S$.*

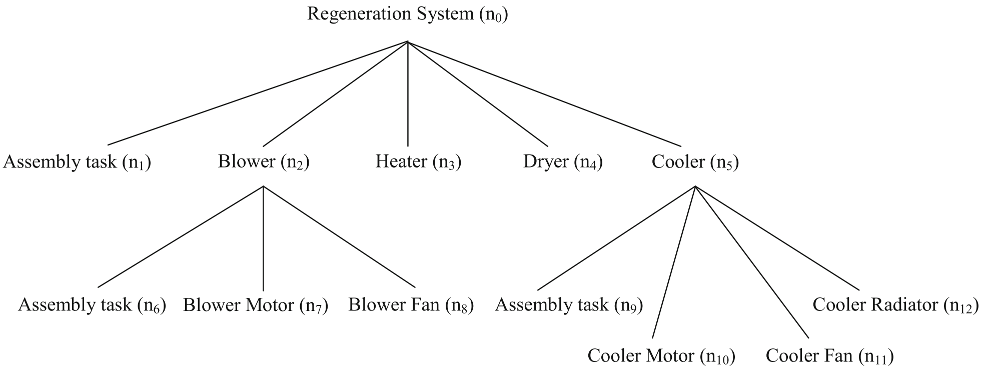


Fig. 3 An extended product structure tree

Table 1 Suppliers of the natural gas dryer

Supplier	Description
s_0	Manufacturer
s_1	Competitor
s_2	Blower supplier
s_3	Heater supplier
s_4	Cooler supplier

Table 2 Supplier capability function

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

Table 3 Component supplier function

Component n	$F_{cs}(n)$
n_1	s_0
n_2	s_1, s_2
n_3	s_1, s_3
n_4	s_0
n_5	s_1, s_4

Table 4 Allocations

n	F_a^1	F_a^2	F_a^3	F_a^4	F_a^5	F_a^6	F_a^7	F_a^8
n_1	s_0	s_0	s_0	s_0	s_0	s_0	s_0	s_0
n_2	s_1	s_1	s_1	s_1	s_2	s_2	s_2	s_2
n_3	s_1	s_1	s_3	s_3	s_1	s_1	s_3	s_3
n_4	s_0	s_0	s_0	s_0	s_0	s_0	s_0	s_0
n_5	s_1	s_4	s_1	s_4	s_1	s_4	s_1	s_4

Proof First, we construct a function $F : N \rightarrow S$. For each $n \in N$, since $N \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$, $n \in \bigcup_{s_i \in S} F_{sc}(s_i)$; so $\exists s_j \in S, n \in F_{sc}(s_j)$; let $F(n) = s_j$.

Then we prove that F is an allocation. (1) N is an ECS of T; (2) $\forall n \in N$ *and* $s \in S$, if $F(n) = s$, according to the construction of F, we have $n \in F_{sc}(s)$. F satisfies conditions in Definition 2, so it is an allocation. □

If F_a exists, then we say that F_{sc} is sufficient. The F_{sc} given in Table 2 is sufficient. Table 4 lists all possible allocations.

Based on Definition 2, we can prove that the conditions given in Lemma 1 are also the necessary for the existence of allocations.

Lemma 2 (Necessary conditions for existence of allocations) *If there exists an allocation $F_a : N \rightarrow S$, then N is an ECS of T and $N \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$.*

Proof First, if $F_a : N \rightarrow S$ is an allocation, N is an ECS of T.

Second, if $F_a : N \rightarrow S$ is an allocation, $\forall n \in N, \exists s \in S, n = F_{sc}(s)$.

Finally, since $F_{sc}(s) \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$, $\forall n \in N, n \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$.

Hence, $N \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$. □

Theorem 1 (Necessary and sufficient conditions for existence of allocations) *The necessary and sufficient conditions for existing an allocation $F_a : N \rightarrow S$ are N is an ECS of T and $N \subseteq \bigcup_{s_i \in S} F_{sc}(s_i)$.*

This theorem holds on the basis of Lemmas 1 and 2.

The optimization problem

Intuitively, in selecting suppliers, one needs to look at the component supplier function F_{cs} (see Table 3 for examples). If there is only one supplier for a component, then that supplier must be selected. When multiple suppliers may supply the same component, a decision must be made. For instance, the capability of s_1 is overlapped with s_2 because both of them can provide the component n_2 as is shown in Table 3. In this case, either s_1 or s_2 could be selected for component n_2. In the mean time, according to Definition 2, if a component is allocated to a supplier, then all its child components in the product structure tree must be allocated to the same supplier. For instance, the component node n_2 overlaps with n_6, n_7, n_8 as is shown in Fig. 3. We call the first case supplier capability overlapping whereas we call the second product structure tree overlapping. In short, the supplier selection problem can be taken as a problem looking for the optimal allocation of these two overlappings.

Figure 4 shows an example of product structure tree overlapping. The component set marked with the solid line overlaps the one marked with the dotted line.

An allocation F_a can be described with a binary matrix A. If there are m components or tasks and n suppliers, an allocation matrix $A = [a_{ij}]_{m \times n}$ can be constructed as following.

$$a_{ij} = \begin{cases} 1 & \text{if } F_a \text{ allocates component or task } i \text{ to supplier } j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

If a set of components or tasks are allocated to a supplier, a set of relevant information has to be shared with that supplier. Therefore, given a private parameter and a supplier, we can calculate the risk that the private parameter may be leaked to the supplier under a specific allocation. For an allocation, all

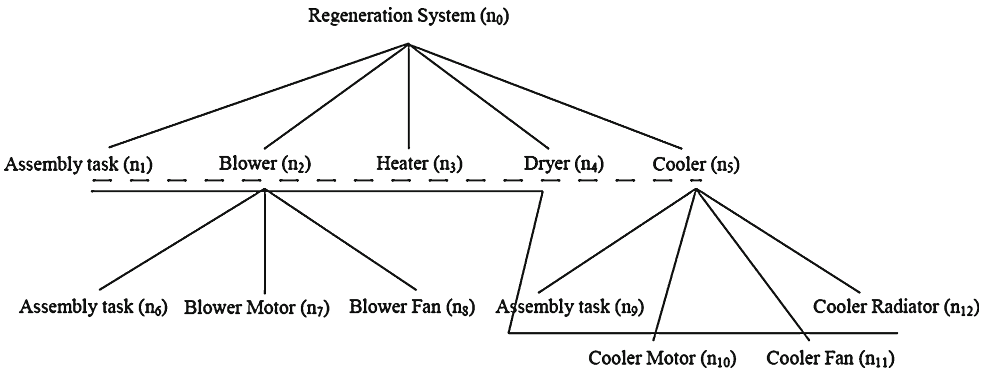


Fig. 4 An example of product structure tree overlapping

the risks that private parameters are leaked to suppliers form a matrix R. If there are p private parameters and n suppliers, matrix R will have p rows and n columns. The risk matrix can be denoted as $R = [r_{kj}]_{p\times n}$, where r_{kj} is the risk that the k^{th} private parameter is leaked to supplier j.

Given a private parameter k and a supplier j, we can define a threshold t_{kj}. If the risk that private parameter k is leaked to supplier j is lower than t_{kj}, then we consider the parameter sharing "safe"; otherwise, we consider it "unsafe". The thresholds for all combinations of private parameters and suppliers form a risk threshold matrix $R_T = [t_{kj}]_{p\times n}$. for which $R < R_T$ if and only if $\forall\ k, j, r_{kj} < t_{kj}$. To mitigate the risk of information leakage, an allocation A should be found that satisfies the constraint $R < R_T$.

If component or task i is allocated to supplier j, the total cost is c_{ij}. For all components or tasks and suppliers, the costs form a cost matrix $C = [c_{ij}]_{m\times n}$. From the perspective of cost, an allocation A should be found that incurs the minimal cost, which can be described as $min \sum_{i,\ j} c_{ij} \times a_{ij}$.

On the basis of the discussion above, the optimization problem of supplier selection can be described as finding an optimal allocation A that satisfies

$$min \sum_{i,j} c_{ij} \times a_{ij} \tag{2}$$

$$s.t.\ R < R_T. \tag{3}$$

The overall framework proceeds as follows (Fig. 5):

1. Find allocations according to the product structure tree and supplier capabilities.
2. Given the private parameter, we can compute the risk matrices, which represents the risk of information leakage caused by inferences, for each allocation.
3. Once we get the allocations and their corresponding risk matrices, we can select the optimal allocation based upon the objective function and the constraint shown in Eqs. 2 and 3.

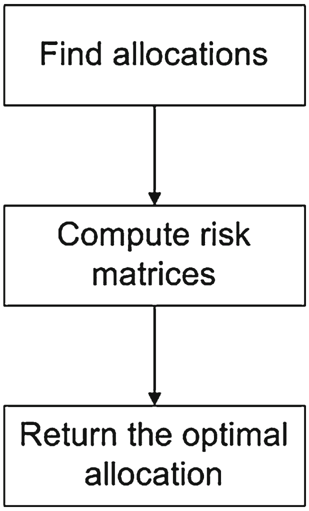


Fig. 5 Framework overview

In the end, the proposed framework can provide the manufacturer the optimal allocation where both the cost and the risk factors are taken into account.

Algorithm 1 can be used to solve the optimization problem given in Eqs. 2 and 3.

Algorithm 1 Supplier selection

```
Input: The supplier capability function $F_{sc}$; the component supplier
    function $F_{cs}$; product structure tree $T$; the logical dependency graph
    $G$ and parameter set $P_s$; cost matrix C; risk threshold matrix $R_T$;
Outlput: The optimal allocation matrix;
Step 1: Find allocations;
Step 2: Calculate risk matrix;
Step 3: Find the optimal allocation.
```

In this algorithm, we firstly need to find allocations based on product structure tree and supplier capabilities. Secondly, we calculate the risk of information leakage caused by inferences with the risk evaluation algorithm for each allocation (Zhang et al. 2011). Finally, we find one or more allocations from components and tasks to suppliers that satisfy the constraints of product structure and supplier capabilities with low risks of information leakage caused by inferences and minimum operational cost.

The detailed procedures for Step 2 are given in Zhang et al. (2011). The following present the algorithms for Step 1 and 3, respectively.

(1) Step 1: Find allocations
Algorithm 2 aims to find all possible allocations for allocating components and tasks to suppliers while considering the product structure and supplier capabilities. The input parameters are the product structure tree, supplier capability function and component supplier function. The output is a set of allocations. This algorithm calls on a subroutine to find all ECS's and then generates the allocations based on F_{cs}. In Algorithm 2, at line 1, Algorithm 3 is called to generate an ECS set by using $\bigcup_{s_i \in S} F_{sc}(s_i)$ as an input parameter. At line 3, we construct allocations by using component supplier function F_{cs}. Take the F_{cs} in Table 3 as an example, we can construct allocations shown in Table 4, which is ECS $\{n_1, n_2, n_3, n_4, n_5\}$.

Algorithm 2 Step 1: Find allocations

```
Input: Product structure tree $T$; the supplier capability function $F_{sc}$;
  the component supplier function $F_{cs}$.
Outlput: All allocations $\widetilde{F_a}$;
1: $\widetilde{N} \Leftarrow FindECS(T, \bigcup_{s \in S} F_{sc}(s))$
2: for all ECS $N \in \widetilde{N}$ do
3:   Generate allocations $\{F_a\}$ using $N$ and $F_{cs}$;
4:   $\widetilde{F_a} = \widetilde{F_a} \cup \{F_a\}$;
5: end for
```

Algorithm 3, which is a recursive algorithm, can be used to obtain a set of ECS's and every ECS is a subset of the input component set. Algorithm 3 takes $\bigcup_{s_i \in S} F_{sc}(s_i)$ as the input parameter and it can find every ECS that is a subset of supplier capabilities. There are two input parameters: the product structure tree and a component set. The following is an explanation of Algorithm 3.

- At line 1, $flag$ is used to indicate the ending condition of the recursion.
- Line 3 extracts a subtree C of M that takes M_i as its root.
- Line 4–8 indicates that if both a node and its children are all in the component set M, we need to find the ECS recursively.
- In lines 11–14, before we add the component set into ECS $\widetilde{N}$, we need to check if it can meet the definition of ECS. We need to check if it already exists in ECS $\widetilde{N}$ since this recursive algorithm might produce the duplicate results.

Algorithm 3 Find ECS

```
Input: Product structure tree $T$; component set $M$;
Outlput: All ECS $\widetilde{N}$ (The initial value is ø);
1: $flag = true$
2: for all $M_i \in M$ do
3:   $C = \{C_i \mid C_i$ is a child of $M_i$ in $T\}$
4:   if $C \subseteq M$ then
5:     $flag = false$
6:     Find ECS(T, $\{M_i\} \bigcup (M \backslash C)$)
7:     Find ECS(T, $C \bigcup (M \backslash \{M_i\})$)
8:   end if
9: end for
10: if $flag$ then
11:   if ($M$ is an ECS) and ($M \nsubseteq \widetilde{N}$) then
12:     $\widetilde{N} = \widetilde{N} \cup M$
13:   end if
14: end if
```

(2) Step 2: Calculate risk matrix
For an allocation, a private parameter and a supplier, the risk evaluation algorithm put forward in Zhang et al. (2011) can be employed to calculate the risk of information leakage caused by inferences. By comparing risk matrices with the risk threshold matrix R_T, allocations that are "safe" can be found by considering the risk of information leakage caused by inferences.

(3) Step 3: Find the optimal allocation
Using allocations, risk matrices and the cost matrix, we can enumerate all the allocations and compute the cost according to Eqs. 2 and 3. The complexity of enumeration is $O(n)$ (n is the number of allocations). Generally speaking, an enumeration algorithm is feasible to find the optimal solution. Still, genetic algorithm can also be used in finding a good solution in practical applications. Each allocation can be encoded as binary chromosome while Eqs. 2 and 3 can be used as the cost function.

The complexity of algorithms

The complexity of Algorithm 1 is determined by the complexity of its three steps: find allocation, calculate risk matrix and find the optimal allocation. As was indicated in Zhang et al. (2011), the complexity of Step 2 is $O(n^2)$ (n is the number of parameters). As was discussed in "The optimization problem", the complexity of Step 3 is $O(n)$.

The complexity of Algorithm 2 depends on two factors: the product structure tree and supplier capabilities. For example, Table 5 shows the worst case of supplier capabilities where every component can be provided by all of the suppliers. In the case of Table 5, Algorithm 2 can produce the most allocations for each ECS. In this case, we can assign

Table 5 Full component supplier function

Component n	$F_{cs}(n)$
n_1	s_0, s_1, s_2, s_3, s_4
n_2	s_0, s_1, s_2, s_3, s_4
n_3	s_0, s_1, s_2, s_3, s_4
n_4	s_0, s_1, s_2, s_3, s_4
n_5	s_0, s_1, s_2, s_3, s_4
n_6	s_0, s_1, s_2, s_3, s_4
n_7	s_0, s_1, s_2, s_3, s_4
n_8	s_0, s_1, s_2, s_3, s_4
n_9	s_0, s_1, s_2, s_3, s_4
n_{10}	s_0, s_1, s_2, s_3, s_4
n_{11}	s_0, s_1, s_2, s_3, s_4
n_{12}	s_0, s_1, s_2, s_3, s_4

Table 6 An example of full ECS (1)

Supplier s	$F_{sc}(s)$
s_0	$n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}$
s_1	$\emptyset$
s_2	$\emptyset$
s_3	$\emptyset$
s_4	$\emptyset$

each component node in an allocation to n different suppliers while n is the number of suppliers. Algorithm 3 is called in Algorithm 2 to get the set of ECS. As we can see from the pseudocode, the complexity of Algorithm 2 is $\|\widetilde{N}\| \times \|E\|_{aver} \times K(\|\widetilde{N}\|$ is the number of ECS; $\|E\|_{aver}$ is the average size of ECS; K is the number of suppliers).

The complexity of Algorithm 3 is determined by supplier capabilities and the product structure tree.

Firstly, the ECS is computed using Algorithm 3. The size of the input component set can affect the complexity of Algorithm 3. The maximum size of the input component set is the number of component nodes in the product structure tree. Equation 4 shows the case of full supplier capabilities.

$$\bigcup_{s_i \in S} F_{sc}(s_i) = N_T, \tag{4}$$

which means that every component node in a product structure tree can be provided by at least one supplier. Given the product structure subtree shown in Fig. 3, Tables 6, 7, and 8 show examples of Eq. 4 where every component node in the product structure tree can be made by at least one supplier. In this case, Algorithm 3 can produce the following ECS: $\{n_0\}$, $\{n_1, n_2, n_3, n_4, n_5\}$, $\{n_1, n_6, n_7, n_8, n_3, n_4, n_5\}$, $\{n_1, n_2, n_3, n_4, n_9, n_{10}, n_{11}, n_{12}\}$ and $\{n_1, n_6, n_7, n_8, n_3, n_4, n_9, n_{10}, n_{11}, n_{12}\}$. In the case of Table 2, Algorithm 3 can find out only one ECS: $\{n_1, n_2, n_3, n_4, n_5\}$.

Table 7 An example of full ECS (2)

Supplier s	$F_{sc}(s)$
s_0	n_0, n_1, n_2, n_3
s_1	n_4, n_5, n_6
s_2	n_7, n_8, n_9
s_3	n_{10}, n_{11}
s_4	n_{12}

Table 8 An example of full ECS (3)

Supplier s	$F_{sc}(s)$
s_0	$n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}$
s_1	$n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}$
s_2	$n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}$
s_3	$n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}$
s_4	$n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}$

Secondly, the complexity of the product structure tree itself can also determine the complexity of Algorithm 3. We can use the full k-ary tree, which is a tree in which every node other than the leaves has k children, as the worst case. Suppose that the depth of the full k-ary tree is)h, $C(h)$ is the number of ECS's for the product structure tree T with the depth of h.

$$C(h) = C(h-1)^k + 1 \tag{5}$$

Note 1 (The maximum number for ECS of a full k-ary tree) If n_0 is the root of T, n_0 has k child nodes, $\{n_1, n_2, \ldots, n_k\}$, and $\{n_0\}$ is an ECS of T, then the $C(h)$ of each node is the union of ECS of its subtree; i.e., $C(h) = \bigcup_{1 \leq i \leq k} C(h-1)$. Therefore, for a k-ary tree with the depth of h, $C(h) = C(h-1)^k + 1$.

For a two-level supply chain, $h = 3$. From Eq. 5, the upper bound of ECS for a k-ary tree is $2^k + 1$. Figure 6 shows how the number of ECS increases with k.

In the worst case, the maximum number of ECS is $C(3) = 2^k + 1$, i.e., the complexity of ECS is $O(2^n)$ in the worst case. Figure 6 reaches the worst case when:

(1) The product structure tree is a k-ary full tree;
(2) Supplier capabilities can cover every node in the k-ary full tree (Eq. 4).

Given a k-ary full tree with the depth of 3, it is easy to know that the total number of nodes is $1 + k + k^2$. It can be seen from Fig. 6 that the $C(h)$ would have a huge jump if $k > 28$.

Also we know that the total number of nodes in a full 28-ary tree is 813. Thus, Algorithm 3 can deal with a small to

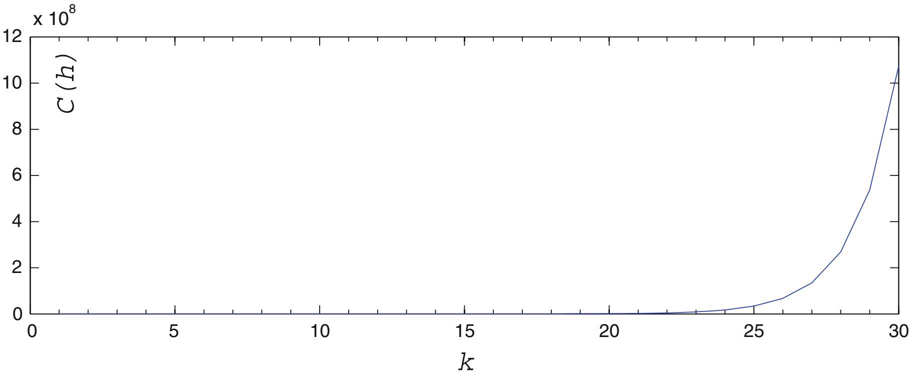


Fig. 6 The maximum number of ECS ($h = 3$)

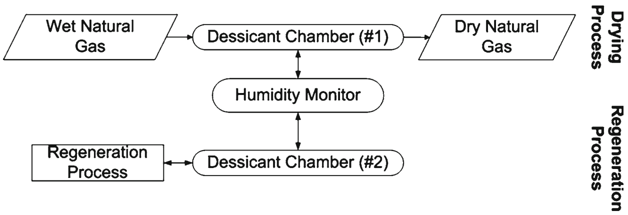


Fig. 7 Product introduction

Table 9 Allocation matrices

$F_a^1 \begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&1&0&0&0\\1&0&0&0&0\\0&1&0&0&0\end{pmatrix}$	$F_a^2 \begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&1&0&0&0\\1&0&0&0&0\\0&0&0&0&1\end{pmatrix}$	$F_a^3 \begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&0&1&0\\1&0&0&0&0\\0&1&0&0&0\end{pmatrix}$	$F_a^4 \begin{pmatrix}1&0&0&0&0\\0&1&0&0&0\\0&0&0&1&0\\1&0&0&0&0\\0&0&0&0&1\end{pmatrix}$
$F_a^5 \begin{pmatrix}1&0&0&0&0\\0&0&1&0&0\\0&1&0&0&0\\1&0&0&0&0\\0&1&0&0&0\end{pmatrix}$	$F_a^6 \begin{pmatrix}1&0&0&0&0\\0&0&1&0&0\\0&1&0&0&0\\1&0&0&0&0\\0&0&0&0&1\end{pmatrix}$	$F_a^7 \begin{pmatrix}1&0&0&0&0\\0&0&1&0&0\\0&0&0&1&0\\1&0&0&0&0\\0&1&0&0&0\end{pmatrix}$	$F_a^8 \begin{pmatrix}1&0&0&0&0\\0&0&1&0&0\\0&0&0&1&0\\1&0&0&0&0\\0&0&0&0&1\end{pmatrix}$

$F_a = \{F_a^1, F_a^2, F_a^3, F_a^4, F_a^5, F_a^6, F_a^7, F_a^8\}$

Table 10 The relation between components and shared design parameters

Component	Shared parameters
Blower	Blower power, Delta P
Blower fan	Blower power, Delta P
Blower motor	Blower power
Cooler	Cooler fan efficiency, cooler radiator heat transfer rate, cooler motor efficiency, total cooler fan pressure drop, cooler power
Cooler fan	Cooler fan efficiency, total cooler fan pressure drop
Cooler motor	Cooler motor efficiency, cooler power
Cooler radiator	Cooler radiator heat transfer rate
Heater	Heater power, heater transfer efficiency

medium size product structure tree in the worst case. Fortunately, most real industry cases do not reach the worst case because most real cases cannot often meet Eq. 4; hence Algorithm 3 can work for most real cases.

Table 11 Components and relevant parameters

Component	Relevant parameters
n_2	Blower power, calculated blower power, Delta P, regeneration flow rate, mass flow rate
n_3	Heater power, calculated heater power, heater transfer efficiency, heater outlet temp, regeneration flow rate, mass flow rate
n_5	Cooler inlet temp, cooler fan efficiency, calculated air quantity, cooler radiator heat transfer rate, calculated cooler power, Cooler motor efficiency, total cooler fan pressure drop, cooler power, regeneration flow rate, mass flow rate

Considering both Algorithms 2 and 3 together, the whole complexity of allocation is subject to the complexity of product structure tree and supplier capabilities.

To reduce the complexity of the Algorithms 2 and 3, rules can be developed for a specific application to reduce the size of $\bigcup_{s_i \in S} F_{sc}(s_i)$ according to specific products and supply chains.

Example

In this section, we present the example of the regeneration system of a natural gas dryer (Li and Geng 2008). A natural gas dryer is a device to remove water from compressed natural gas. As is shown in Fig. 7, a dual tower natural gas dryer has two chambers. Natural gas is dried by the desiccant in one chamber while the desiccant in another chamber is being regenerated.

The regeneration system consists of four major components: blower, heater, dryer and cooler. The regeneration system uses natural gas as the regeneration gas. First, the blower is used to increase the pressure at the outlet of the blower to force the regeneration gas to flow toward the heater. The heater blower heats the regeneration gas to a high

Table 12 Allocations, suppliers and the probabilities of information leakage caused by inferences

	s_1 (%)	s_2 (%)	s_3 (%)	s_4 (%)
F_a^1	100	1.82	2.07	2.77
F_a^2	9.56	1.82	2.07	3.17
F_a^3	100	1.82	2.14	2.77
F_a^4	5.14	1.82	2.14	3.17
F_a^5	3.04	1.18	2.07	2.77
F_a^6	3.32	1.18	2.07	3.17
F_a^7	2.73	1.18	2.14	2.77
F_a^8	2.93	1.18	2.14	3.17

Table 13 Suppliers and risk thresholds

	s_1 (%)	s_2 (%)	s_3 (%)	s_4 (%)
Threshold	5	10	10	10

Table 14 Components, suppliers and costs

	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

temperature. When hot regeneration gas passes through the dryer, it removes moisture from the desiccant. The cooler separates the moisture from regeneration gas by condensation.

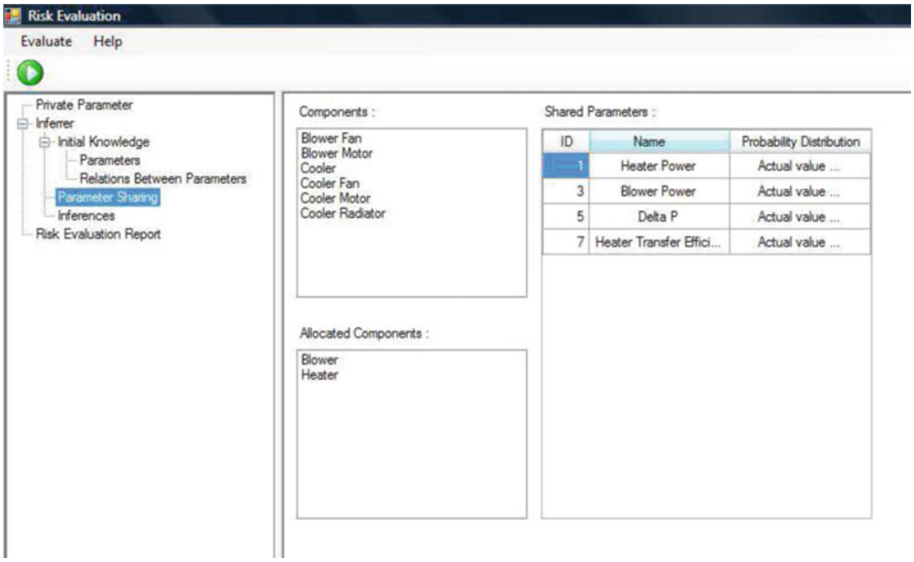
The design of the regeneration system is crucial to the efficiency of the natural gas dryer. Hence, the manufacturer wants to prevent the design parameters of the regeneration system, including pressures, temperatures and flow rates, from being revealed to its (potential) competitors.

We carry out the supplier selection according to the algorithms introduced in "Supplier selection".

(1) Step 1: Find allocations
Some parts of the example have already been introduced in "Supplier selection", like the product structure (Fig. 3), components, suppliers (Table 1), supplier capabilities (Table 2) and allocations (Table 4). To find the allocations, we call Algorithm 2 whereas Fig. 3 and Table 2 are the input parameters for which the output is allocation matrices (Table 9).

(2) Step 2: Calculate risk matrices
In this example, the manufacturer s_0 is the holder whereas the supplier and potential competitor s_1 is the inferrer. The design parameter $DryerOutletTemp$ is the private parameter p_0. What design parameters the manufacturer shares with a supplier usually depends on what components the supplier supplies. Table 10 gives the relation between components and shared design parameters.
Suppliers s_1, s_2, s_3 and s_4 may have a different initial knowledge of the parameters. Corresponding to the supplier capabilities given in Tables 2 and 3, we assign continuous uniform distributions, which range from 0.7 × $actual\ value$ to 1.3 × $actual\ value$, to the initial knowledge of parameters relevant to components that a supplier has the capability to supply. In the meantime, continuous uniform distributions, ranging from 0.4 × $actual\ value$ to 1.6 × $actual\ value$, are assigned to the other parameters in this example. Table 11 gives components and their relevant parameters. The probabilities of information leakage caused by inferences are computed by using the algorithm that we introduced in another paper (Zhang et al. 2011). We call that algorithm with Table 11, supplier information, and the logical dependency graph built in that paper as the input parameters.
The output of Step 2 is the probability of information leakage. Table 12 gives the probability of information leakage of the private parameter $DryerOutletTemp$ caused by inferences for each combination of allocation and supplier. The results in Table 12 are obtained when the parameters are shared with their actual values, and the working values of the private parameter p_0 are within the range from 0.99 × $actual\ value$ to 1.01 × $actual\ value$.

(3) Step 3: Find the optimal allocation
Table 13 gives the risk thresholds used in this example. Since Supplier s_1 is a potential competitor, we assign a lower threshold 5% than then threshold assigned to other suppliers.
By comparing probabilities in Table 12 and thresholds in Table 13, it is easy to conclude that allocations F_a^5, F_a^6, F_a^7 and F_a^8 are "safe", considering the risk of information leakage caused by inferences.
For each allocation in Table 4, we can calculate the costs on component n_2, n_3 and n_5. The total costs are 6, 7, 7, 8, 7, 8, 8 and 9, respectively, when the cost for each combination of component and supplier is assigned as given in Table 14.
Because $\{n_6, n_7, n_8, n_9, n_{10}, n_{11}, n_{12}\}$ cannot be provided by any suppliers according to the Table 3, they are not included in the matrices. By using Tables 9, 12, 13, and 14 as the input parameters, we can get the optimal allocation through an enumeration method or a genetic algorithm. In the example, the output of Step 3 is that F_a^5 is the optimal solution. Figure 8 shows the how the components are assigned to the supplier; Fig. 9 shows the risk value of supplier 1 in the allocation 2.



Components :

Blower Fan
Blower Motor
Cooler
Cooler Fan
Cooler Motor
Cooler Radiator

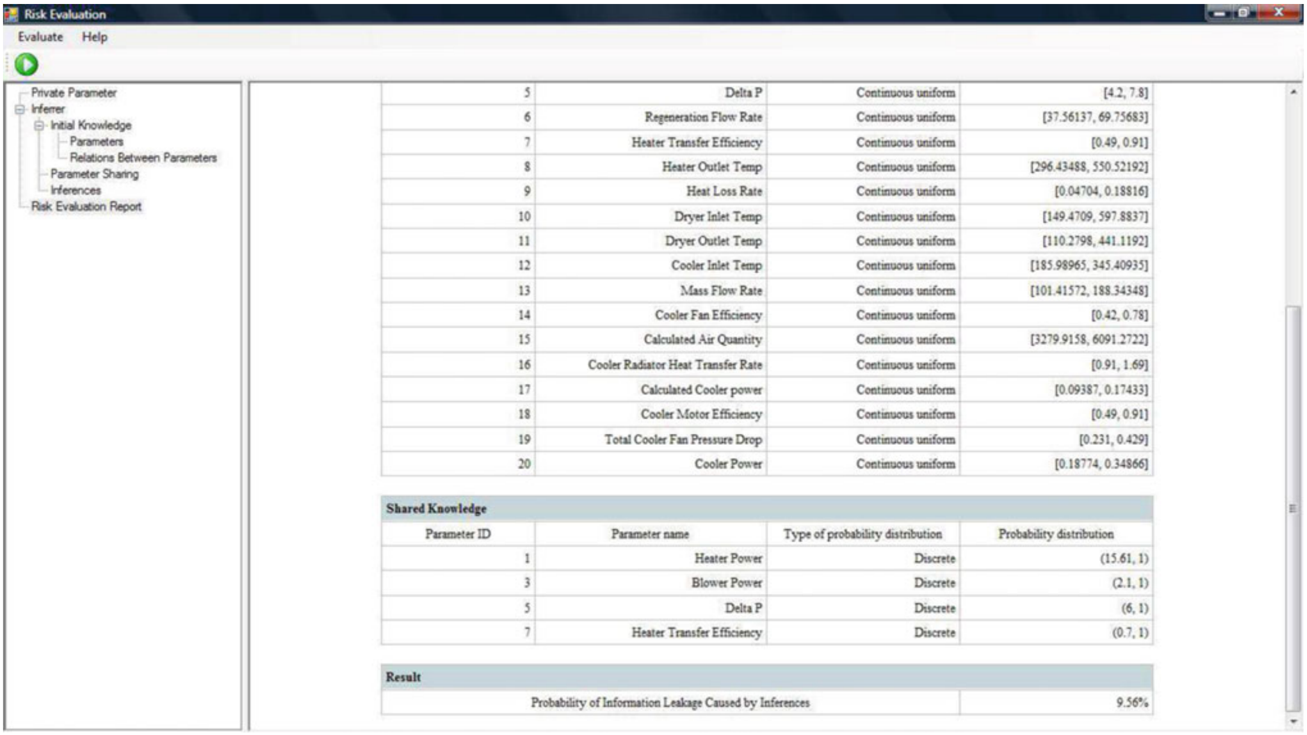
Allocated Components :

Blower
Heater

Shared Parameters :

ID	Name	Probability Distribution
1	Heater Power	Actual value ...
3	Blower Power	Actual value ...
5	Delta P	Actual value ...
7	Heater Transfer Effici...	Actual value ...

Fig. 8 A screenshot of parameters sharing



5	Delta P	Continuous uniform	[4.2, 7.8]
6	Regeneration Flow Rate	Continuous uniform	[37.56137, 69.75683]
7	Heater Transfer Efficiency	Continuous uniform	[0.49, 0.91]
8	Heater Outlet Temp	Continuous uniform	[296.43488, 550.52192]
9	Heat Loss Rate	Continuous uniform	[0.04704, 0.18816]
10	Dryer Inlet Temp	Continuous uniform	[149.4709, 597.8837]
11	Dryer Outlet Temp	Continuous uniform	[110.2798, 441.1192]
12	Cooler Inlet Temp	Continuous uniform	[185.98965, 345.40935]
13	Mass Flow Rate	Continuous uniform	[101.41572, 188.34348]
14	Cooler Fan Efficiency	Continuous uniform	[0.42, 0.78]
15	Calculated Air Quantity	Continuous uniform	[3279.9158, 6091.2722]
16	Cooler Radiator Heat Transfer Rate	Continuous uniform	[0.91, 1.69]
17	Calculated Cooler power	Continuous uniform	[0.09387, 0.17433]
18	Cooler Motor Efficiency	Continuous uniform	[0.49, 0.91]
19	Total Cooler Fan Pressure Drop	Continuous uniform	[0.231, 0.429]
20	Cooler Power	Continuous uniform	[0.18774, 0.34866]

Shared Knowledge

Parameter ID	Parameter name	Type of probability distribution	Probability distribution
1	Heater Power	Discrete	(15.61, 1)
3	Blower Power	Discrete	(2.1, 1)
5	Delta P	Discrete	(6, 1)
7	Heater Transfer Efficiency	Discrete	(0.7, 1)

Result

Probability of Information Leakage Caused by Inferences	9.56%

Fig. 9 A screenshot of the prototype

The prototype program consists of the following modules: (1) calculating partitions and allocations; (2) generating Logical Dependency Graphes, which generates Logical Dependency Graphes from product parameters and equations among them; (3) assigning probability distributions to product parameters, thereby allowing users to assign probability distributions to product parameters. The prototype supports three types of probability distributions now, namely discrete distributions, continuous uniform distributions and normal distributions; (4) calculating the risks of information leakage for each allocation.

Conclusion

In this paper, we have formulated the problem of information leakage caused by inferences in a two-level supply chain, within which potential competition may exist between a supplier and the manufacturer. On the basis of our previous work on modeling and evaluating information leakage caused by inferences in supply chains, we have here discussed how to mitigate the risk caused by inference based on the risk evaluation model by using supplier selection for such a two-level supply chain. The problem is modeled as an optimization problem, for which a generic solving process is presented. The necessary and sufficient conditions for the existence of allocations were also proposed in this paper. A practical example based on a product in the process industry has been used to demonstrate our proposed method.

Currently, we are applying our approach to pylon/engine design supported by five collaborating aerospace companies. Besides the optimal supplier selection, we will consider a greater number of possible approaches so as to mitigate the risk caused by inference based on the risk evaluation model. We also expect to extend our supplier-selection-based risk mitigation method from the two-level supply chain model herein discussed to include supply chains with other structures.

Acknowledgments The research reported in this paper is partially supported by NSERC through a CRD project (PJ 350114-06). We are grateful to the financial support from NSERC, CRIAQ, Pratt & Whitney Canada Corp., Bombardier Inc., CMC Electronics Inc., and Rolls-Royce Canada Limited.

References

Aissaoui, N., Haouari, M., & Hassini, E. (2007). Supplier selection and order lot sizing modeling: A review. *Computers & Operations Research, 34*(12), 3516–3540.

Anand, K. S., & Goyal, M. (2009). Strategic information management under leakage in a supply chain. *Management Science, 55*(3), 438–452.

Atallah, M. J., Elmongui, H. G., Deshpande, V., & Schwarz, L. B. (2003). Secure supply-chain protocols. In: *Proceedings of IEEE international conference on E-commerce 2003* (pp. 293–302).

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W. P., & Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations Research, 46*(3), 316–329. doi:10.1287/opre.46.3.316.

Carrera, D., & Mayorga, R. (2008). Supply chain management: A modular fuzzy inference system approach in supplier selection for new product development. *Journal of Intelligent Manufacturing, 19*(1), 1–12.

Cera, C. D., Braude, I., Kim, T., Han, J., & Regli, W. C. (2006). Hierarchical role-based viewing for multi-level information security in collaborative CAD. *Journal of Computing and Information Science in Engineering, 6*(1), 2–10.

Cera, C. D., Kim, T., Han, J., & Regli, W. C. (2004). Role-based viewing envelopes for information protection in collaborative modeling. *Computer-Aided Design, 36*(9), 873–886.

Chen, L., Song, Z., & Feng, L. (2004). Internet-enabled real-time collaborative assembly modeling via an e-assembly system: Status and promise. *Computer-Aided Design, 36*(9), 835–847.

Chen, S. J. G., & Huang, E. (2007). A systematic approach for supply chain improvement using design structure matrix. *Journal of Intelligent Manufacturing, 18*(2), 285–299.

Chen, T. Y., Chen, Y. M., & Chu, H. C. (2008). 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.

Christopher, M., & Lee, H. (2004). Mitigating supply chain risk through improved confidence. *International Journal of Physical Distribution & Logistics Management, 34*(5), 388–396.

de Boer, L., Labro, E., & Morlacchi, P. (2001). A review of methods supporting supplier selection. *European Journal of Purchasing & Supply Management, 7*(2), 75–89.

Demirtas, E. A., & Ustun, O. (2008). An integrated multiobjective decision making process for supplier selection and order allocation. *Omega, 36*(1), 76–90.

Demirtas, E. A., & Ustun, O. (2009). Analytic network process and multi-period goal programming integration in purchasing decisions. *Computers & Industrial Engineering, 56*(2), 677–690.

Dickson, G. W. (1966). An analysis of vendor selection systems and decisions. *Journal of Purchasing, 2*(1), 5–17.

Ferraiolo, D. F., Kuhn, R., & Sandhu, R. S. (2007). RBAC standard rationale: Comments on a critique of the ANSI standard on role based access control. *IEEE Security & Privacy, 5*(6), 51–53.

Giunipero, L. C., & Eltantawy, R. A. (2004). Securing the upstream supply chain: A risk management approach. *International Journal of Physical Distribution & Logistics Management, 34*(9), 698–713.

Goldreich, O., Micali, S., & Wigderson, A. (1987). How to play any mental game. In: *Proceedings of the 19th annual ACM conference on theory of computing* (pp. 218–229).

Hoecht, A., & Trott, P. (2006). Outsourcing, information leakage and the risk of losing technology-based competencies. *European Business Review, 18*(5), 395–412.

Huang, G. Q., Lau, J. S. K., & Mak, K. L. (2003). The impacts of sharing production information on supply chain dynamics: A review of the literature. *International Journal of Production Research, 41*(7), 1483–1517.

Huang, S. H., & Keskar, H. (2007). Comprehensive and configurable metrics for supplier selection. *International Journal of Production Economics, 105*(2), 510–523.

Juttner, U. (2005). Supply chain risk management: Understanding the business requirements from a practitioner perspective. *The International Journal of Logistics Management, 16*(1), 120–141.

Juttner, U., Peck, H., & Christopher, M. (2003). Supply chain risk management: Outlining an agenda for future research. *International Journal of Logistics: Research and Applications, 6*(4), 197–210.

Kahraman, C., Cebeci, U., & Ulukan, Z. (2003). Multi-criteria supplier selection using fuzzy AHP. *Logistics Information Management, 16*(6), 382–394.

Khan, O., Christopher, M., & Burnes, B. (2008). The impact of product design on supply chain risk: A case study. *International Journal of Physical Distribution & Logistics Management, 38*(5), 412–432.

Kim, K. Y., Wang, Y., Muogboh, O. S., & Nnaji, B. O. (2004). Design formalism for collaborative assembly design. *Computer-Aided Design, 36*(9), 849–871.

Kim, T., Cera, C. D., Regli, W. C., Choo, H., & Han, J. (2006). Multi-level modeling and access control for data sharing in collaborative design. *Advanced Engineering Informatics, 20*(1), 47–57.

Kubat, C., & Yuce, B. (2010). A hybrid intelligent approach for supply chain management system. *Journal of Intelligent Manufacturing* (in press).

Lee, A. H. I. (2009). A fuzzy supplier selection model with the consideration of benefits, opportunities, costs and risks. *Expert Systems with Applications, 36*(2, Part 2), 2879–2893.

Lee, H. L., & Whang, S. (2000). Information sharing in a supply chain. *International Journal of Manufacturing Technology and Management, 1*(1), 79–93.

Leong, K. K., Yu, K. M., & Lee, W. B. (2003). A security model for distributed product data management system. *Computers in Industry, 50*(2), 179–193.

Li, L. (2002). Information sharing in a supply chain with horizontal competition. *Management Science, 48*(9), 1196–1212.

Li, H., & Geng, Y. (2008). *Confidential information protection for industry design*. Technical report, Concordia Institute for Information Systems Engineering, Concordia University, Montreal.

Li, J., Xiong, N., Park, J., Liu, C., Ma, S., & Cho, S. (2009). Intelligent model design of cluster supply chain with horizontal cooperation. *Journal of Intelligent Manufacturing*, 1–15.

Lindell, Y., & Pinkas, B. (2002). Privacy preserving data mining. *Journal of Cryptology, 15*(3), 177–206.

Lockamy, A., III, & McCormack, K. (2010). Analysing risks in supply networks to facilitate outsouring decisions. *International Journal of Production Research, 48*(2), 593–611.

Mason-Jones, R., & Towill, D. R. (1998). Shrinking the supply chain uncertainty circle. *Control, 24*(7), 17–22.

McCauley-Bell, P. (1999). Intelligent agent characterization and uncertainty management with fuzzy set theory: A tool to support early supplier integration. *Journal of Intelligent Manufacturing, 10*(2), 135–147.

Mun, D., Hwang, J., & Han, S. (2009). Protection of intellectual property based on a skeleton model in product design collaboration. *Computer-Aided Design, 41*(9), 641–648.

Neiger, D., Rotaru, K., & Churilov, L. (2009). Supply chain risk identification with value-focused process engineering. *Journal of Operations Management, 27*(2), 154–168.

Saaty, T. L. (2004). Fundamentals of the analytic network process-multiple networks with benefits, opportunities, costs and risks. *Journal of Systems Science and Systems Engineering, 13*(3), 348–379.

Sandhu, R. S., Coyne, E. J., Feinstein, H. L., & Youman, C. E. (1996). Role-based access control models. *IEEE Computer, 29*(2), 38–47.

Shyamsundar, N., & Gadh, R. (2002). Collaborative virtual prototyping of product assemblies over the Internet. *Computer-Aided Design, 34*(10), 755–768.

Sun, X., Zeng, Y., & Liu, W. (2010). Formalization of design chain management using environment-based design (EBD) theory. *Journal of Intelligent Manufacturing* (accepted).

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

Svensson, G. (2002). A conceptual framework of vulnerability in firms' inbound and outbound logistics flows. *International Journal of Physical Distribution & Logistics Management, 32*(2), 110–134.

Tolone, W., Ahn, G. J., Pai, T., & Hong, S. P. (2005). Access control in collaborative systems. *ACM Computing Surveys, 37*(1), 29–41.

Wang, Y., Ajoku, P. N., Brustoloni, J. C., & Nnaji, B. O. (2006). Intellectual property protection in collaborative design through lean information modeling and sharing. *Journal of Computing and Information Science in Engineering, 6*(2), 149–159.

Yao, A. (1986). How to generate and exchange secrete. In: *Proceedings of the 27th annual symposium on foundations of computer science* (pp. 162–167).

Zeng, Y., & Gu, P. (1999). 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.

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

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

Zsidisin, G. A., Ellram, L. M., Carter, J. R., & Cavinato, J. L. (2004). An analysis of supply risk assessment techniques. *International Journal of Physical Distribution & Logistics Management, 34*(5), 397–413.

Zsidisin, G. A., & Smith, M. E. (2005). Managing supply risk with early supplier involvement: A case study and research propositions. *Journal of Supply Chain Management, 41*(4), 44–57.