JOURNAL OF INFORMATION SYSTEMS
Vol. 28, No. 1
Spring 2014
pp. 243–268

American Accounting Association
DOI: 10.2308/isys-50692

A Survey of Privacy-Aware Supply Chain Collaboration: From Theory to Applications

Yuan Hong
University at Albany, SUNY

Jaideep Vaidya
Rutgers, The State University of New Jersey

Shengbin Wang
North Carolina A&T State University

ABSTRACT: In the contemporary information era, the ubiquitous collection of data from different parties frequently accommodates significant mutual benefits to the involved participants. However, data is a double-bladed sword. Inappropriate access or use of data by the recipients may pose serious privacy issues that explicitly harm the data owners. In the past decade, swiftly increasing privacy concerns arise in many business processes such as supply chain management. How to protect the private information of different participants in the supply chain has become a key multidisciplinary research problem in information systems, production and operations management, computer science, and mathematics. Specifically, in the real world, manufacturers, distributors, and retailers commonly collaborate with each other to cater to the demands of supplying and marketing. In their traditional cooperation, all the parties completely share their proprietary information so as to jointly optimize their operations (e.g., maximize their profit or minimize their cost). Now, they realize that completely sharing such information would bring considerable negative impact to themselves. For overcoming this, some recent research results begin to make the following ideal occasion possible—all the participants collaboratively solve a realistic problem without revealing any private proprietary information to each other.

In this paper, we primarily review the literature on the applications of privacy-preserving techniques to supply chain collaboration among multiple parties. We first identify various private proprietary information required in the supply chain collaboration, and discuss several potential privacy-preserving techniques. Then, we review the relevant research results from theory to applications. Since intensive collaboration in modern supply chains opens even more opportunities in both academia and industry, we finally outline the future research trend and the potential challenges in this promising area.

Keywords: privacy; supply chain management; optimization.

This work is supported in part by the National Science Foundation under Grant No. CNS-0746943.

Editor's note: Accepted by Miklos A. Vasarhelyi.

Published Online: December 2013

I. REAL-WORLD SUPPLY CHAIN COLLABORATION

With the rapid growth of computing, storing, and networking resources, different corporations frequently collaborate with each other in the current business world, especially in the supply chain (Baltzan 2011). The Council of Supply Chain Management Professionals defines it as below:

> Supply Chain Management encompasses the planning and management of all activities involved in sourcing and procurement, conversion, and all logistics management activities. Importantly, it also includes coordination and collaboration with channel partners, which can be suppliers, intermediaries, third party service providers, and customers. In essence, supply chain management integrates supply and demand management within and across companies. Supply Chain Management is an integrating function with primary responsibility for linking major business functions and business processes within and across companies into a cohesive and high-performing business model. It includes all of the logistics management activities noted above, as well as manufacturing operations, and it drives coordination of processes and activities with and across marketing, sales, product design, finance and information technology.

The supply chain consists of many entities, such as suppliers, manufacturers, retailers, and customers, which jointly optimize their operations in general. Then, optimization across different parties plays an essential and fundamental role in various functions of supply chain management. Essentially, to facilitate the decision making, collaborative optimization problems are ubiquitously formulated among different parties for seeking their global maximum profit or minimum cost in the supply chain. According to the nature of collaboration, two typical categories of supply chain collaboration can be discovered:

1. Vertical Collaboration: The collaborative parties act *distinct* roles in the supply chain. For example, one company is the supplier/manufacturer, whereas another company is the distributor. The latter company delivers goods for the former company. They jointly minimize the transportation cost (Vaidya 2009b).
2. Horizontal Collaboration: The collaborative parties have *identical* roles in the supply chain. For example, two companies are both suppliers/manufacturers. They collaborate with each other to reduce their transportation cost by sharing the trucks (Buss 2003; Bednarz 2012).

We now specifically introduce some motivating examples in three typical subareas of supply chain management—logistics, production, and scheduling.

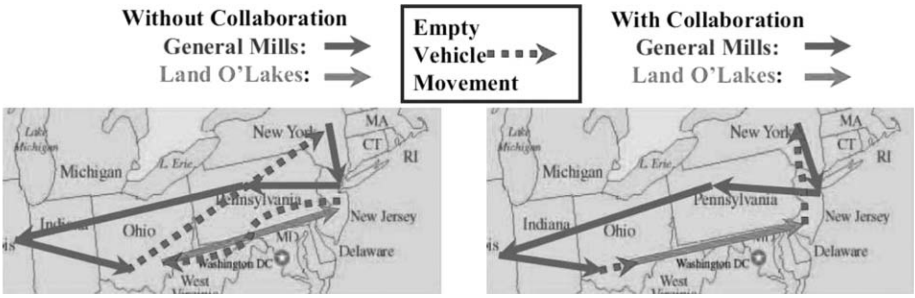
Examples of Supply Chain Collaboration

Collaborative Logistics

In the packaged goods industry, delivery vehicles are empty 25 percent of the time (Özener and Ergun 2008). Just four years ago, Land O'Lakes spent much of their time shuttling empty vehicles down slow-moving highways, wasting several million dollars annually. By using a web-based collaborative logistics service—Nistevo.com—to merge loads from different companies bound to the same destination, huge savings were realized.

For instance, General Mills and Land O'Lakes are two food-producing companies in the United States (Buss 2003). As shown in the left-hand side of Figure 1, General Mills plans to deliver pizza rolls from a warehouse in New York to New Jersey, Pennsylvania, Illinois, and Ohio, with an optimal route computed by itself. In the meanwhile, Land O'Lakes delivers its goods from Ohio to New Jersey. If they do not collaborate with each other, both companies' delivery routes include a considerable amount of "empty vehicle movement," which means that the vehicles are

FIGURE 1
Collaborative Logistics



empty, but traveling on the route. Alternatively, the collaborative logistics service enables Land O'Lakes and General Mills to share their vehicles and seek their global minimum transportation cost, where empty vehicle movement can be significantly reduced, as shown in the right-hand side of Figure 1. Then, the transportation costs—both fuel cost and drivers' working hours—can be saved via collaborative optimization. In fact, freight costs of Land O'Lakes were cut by 15 percent after employing collaborative logistics service, with an annual savings of $2 million (Turban, Rainer, and Potter 2005). This opens greater opportunities for us to pursue collaboration in logistics.

Note that the above example falls into horizontal collaboration, since both General Mills and Land O'Lakes are suppliers/manufacturers. Meanwhile, collaborative logistics belonging to vertical collaboration also occurs very often, since product delivery is widely outsourced to other shipping companies in the real world. A related example was described in Vaidya (2009b): a winemaking company grows grapes in various locations and then sends them to different wineries to produce wine; another company is responsible for the transportation.

Collaborative Production

Manufacturing planning and control facilitate decision makers to determine the optimal production activities with limited resources. For example, two factories manufacture two different products—bowls and mugs, respectively, with the same raw material, clay. The demand of clay and labor hours to produce every unit bowl and mug is given in Table 1, and each factory has a limited amount of clay and a limited number of laboring hours. Given the unit revenue of these two products, the factories should make the optimal production assignment that can maximize their revenue.

Considering the non-collaborative scenario, each factory can formulate a simple linear programming problem to obtain its maximum revenue, respectively, which are ten bowls ($400) and 15 mugs ($750) (Dantzig 1963). However, the resources are not well utilized if they derive the optimal solution individually, since Factory 1 does not have sufficient hours of labor compared to its possessed amount of raw materials. Instead, if they collaborate with each other by sharing the raw materials and/or hours of labor, they can formulate a global linear programming problem and obtain the optimal production assignment as 24 bowls and eight mugs ($1,360 in total, which is clearly greater than $400 + $750 = $1,150). Thus, supply chain collaboration can create more revenue for organizations by better utilizing scarce resources in production.

TABLE 1

Collaborative Production

Product	Labor (Hr./Unit)	Clay (Lb./Unit)	Revenue ($/Unit)
Bowl (Factory 1)	1	4	40
Mug (Factory 2)	2	3	50

Factory 1: Ten hours of labor and 60 pounds of clay every day.
Factory 2: 30 hours of labor and 60 pounds of clay every day.

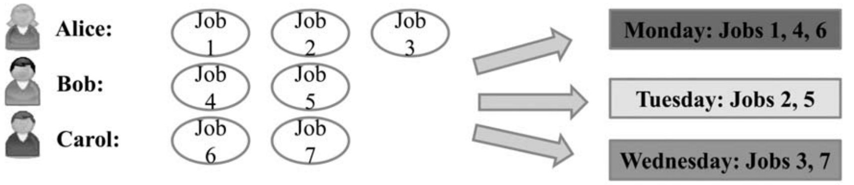
Similarly, the above horizontal collaboration in production occurs between two manufacturers. Indeed, vertical collaboration between two entities with different roles in manufacturing can be also possible. For example, if one manufacturer produces the product while the raw materials are procured or the inventory is controlled by another party.

Collaborative Scheduling

Scheduling covers a wide variety of real-life problems that are applied or extended from optimization models, particularly in supply chain management. A typical scheduling problem seeks the optimal solution of assigning some jobs or tasks to several time slots, while possibly subjecting to some constraints (Papadimitriou and Steiglitz 1982). Similar to transportation and production, many such problems also involve different parties to jointly optimize the scheduling process. For example, in Figure 2, Alice, Bob, and Carol have their own set of jobs, respectively, and they plan to assign their jobs into three time slots: *Monday, Tuesday,* and *Wednesday*. However, some jobs cannot be assigned into the same time slots, possibly because they share the same machine or need the same technician. Completing such a scheduling problem is to find an optimal job assignment solution without conflicts, where every conflicted pair of jobs are assigned into different time slots, respectively.

In the supply chain, collaborative scheduling could involve any two or more parties of suppliers, manufacturers, distributors, and retailers/customers. Horizontal, vertical, or mixed collaboration relies on specific problem settings. Also, the collaborative scheduling problem in

FIGURE 2
Collaborative Scheduling



Time Conflicts: some pair of jobs **cannot be assigned into the same time slot**
→ jobs share the same machine: **(1, 2), (1, 3), (2, 3), (4, 5), (6, 7)**
→ jobs need the same technician: **(2, 4), (5, 7), (3, 6)**

Figure 2 allows Alice, Bob, and Carol to share resources (i.e., machine or technician), benefiting all the parties.

Why Supply Chain Collaboration Needs Private Proprietary Information

In supply chain collaboration, the global goal of the involved parties is to optimize their objective function (e.g., the global cost/profit). To achieve this, a great deal of their proprietary information is indispensable for formulating the constraints based on the current limited resources, and deriving the optimal solution with the satisfaction of all the constraints. More specifically, almost every optimization function in supply chain collaboration needs the participants (suppliers/manufacturers, distributors, or retailers) to model the problem with their own local proprietary information, and eventually solve it. For instance, in the collaborative logistics shown in Figure 1, General Mills and Land O'Lakes need the following proprietary information to minimize their transportation cost:

- Their own food production amount at different locations
- Their own number and capacity of the trucks at different locations
- Their own demand of delivery at different locations
- If their delivery routes are not identical, e.g., in Figure 1, each company's delivery route

The local proprietary information, indeed, contains a considerable amount of commercial secrets, e.g., how many pizza rolls should be delivered from New York to New Jersey for General Mills. Meanwhile, those cooperative companies (especially in horizontal collaboration such as Land O'Lakes and General Mills) are competitors on the market. Thus, it is nearly impossible for those companies to completely share their private proprietary information with each other.

Innumerable such situations exist, where appropriate safe and secure use of proprietary information leads to immense financial and social benefits. For example, consider the case of Ford and Firestone. In 2001, numerous accidents due to tread separation were reported. Initially, both companies blamed each other. It turned out that it was only Ford Explorers with Firestone tires from the Decatur, Illinois, plant in specific situations that had these problems. If found out earlier, much loss could have been avoided. While both companies individually collect a lot of pertinent testing data, this was not shared due to commercial concerns. Moreover, Walmart, Target, and Costco ship millions of dollars of goods over the seas every month. These feed into their local ground transportation network. The cost of sending half-empty ships is prohibitive, but the individual corporations have serious problems with disclosing freight information. If it were possible to determine which trucks should make their way to which ports to be loaded onto certain ships, i.e., solve the classic transportation problem, without knowing the individual constraints, the savings would be enormous. In all of these cases, complete sharing of data (*viz*. the private proprietary information) would lead to invaluable savings/benefits. However, since unrestricted information sharing is a competitive impossibility or requires great trust, better solutions must be found.

This paper summarizes the work that has made such possible without the release of private proprietary information in supply chain collaboration, and surveys an emerging research direction in supply chain management and information systems. The remainder of this paper is organized as follows. In Section II, we discuss the potential techniques to protect private proprietary information in supply chain collaboration. Then, in Section III, we review the relevant theoretical work, mainly including the privacy-preserving collaborative optimization techniques (*note that the optimal operations and decisions in supply chain collaboration are commonly derived from this*). Beyond the theoretical work, we discuss their corresponding applications in supply chain collaboration and survey some other specific applications in Section IV. Finally, in Section V, we analyze the future research trend in this multidisciplinary area, and discuss the potential technical and practical challenges that researchers may encounter in securing supply chain management.

II. APPROACHES TO PRESERVING PRIVACY IN COLLABORATION

In order to protect private proprietary information in supply chain collaboration, all the corporations can simply employ a trusted third party to solve their collaborative optimization problem (Hong 2013). However, in most cases, a trusted third party may not exist in practice; then they should develop alternative privacy-preserving techniques. We now present some potential effective approaches and discuss their applicability to supply chain collaboration.

Trusted Third Party

Many commercial companies currently employ a trusted third party to solve their collaborative problems in the supply chain by completely sharing their local private information with the third party beforehand. For instance, IBM Sterling Transportation Management System, which was formerly Nistevo.com (Turban et al. 2005; Özener and Ergun 2008), provides collaborative logistics services for suppliers/manufacturers and retailers in the United States, e.g., General Mills and Land O'Lakes. Although a trusted third party accommodates an easy-to-use solution in reality, it is not always the best choice for commercial companies due to several inherent drawbacks.

First, a trusted third party requires a high degree of trust in reality, where all participants involved in the collaboration should trust the external third party that solves the collaborative optimization problem. In the case that one party does not trust the third party, the collaboration might be stuck. At this time, every participant's benefits might be immensely hurt due to the possibly terminated collaboration.

Second, with the trusted third party, although the local private proprietary information is not directly shared among collaborative corporations, such information is disclosed to the external third party. For instance, as competitors, General Mills and Land O'Lakes do not know the proprietary information from each other (e.g., the production amount and delivery route/amount), yet the problem-solver Nistevo.com knows everything from both companies since it acts the coordinator for such supply chain collaboration. For this reason, some corporations might be reluctant to reveal their local private information, especially their confidential commercial secrets, to any external party, although sharing them with the third party with signed agreements would not harm them directly. Also, outsourcing the decision making to an external party seems not sufficiently safe from the security perspective of each corporation. All the collaborative corporations have to bear the risk of misappropriation of their intellectual property.

Third, from the economic perspective of every collaborative corporation, the trusted third party usually charges an extremely expensive bill for their services, e.g., Land O'Lakes pays $250,000 for the collaborative logistics service to Nistevo.com every year (Turban et al. 2005; Özener and Ergun 2008). To some extent, finding the optimal solution without a trusted third party can avoid such a big expenditure.

With the above concerns, it is extremely worth exploring alternative approaches that can securely optimize the performance in supply chain collaboration for different parties without establishing a trusted third party. This is also the aim of most of the surveyed work in this paper.

Techniques Without a Trusted Third Party

If a trusted third party cannot be established beforehand for a particular collaboration in the supply chain, all the parties have to securely optimize their collaborative operation by themselves. In this case, past fundamental research results (e.g., Secure Multiparty Computation; Yao 1986) and similar problems addressed in other well-studied areas (e.g., privacy-preserving data analysis; Agrawal and Srikant 2000; Evfimievski, Srikant, Agrawal, and Gehrke 2002; Vaidya and Clifton 2003; Lin, Clifton, and Zhu 2005) have inspired two possible categories of approaches to secure such process, detailed as below.

Secure Multiparty Computation (SMC)

The field of Secure Multiparty Computation (SMC) addresses exactly this problem. The celebrated results (Yao 1986; Goldreich, Micali, and Wigderson 1987; Ben-Or, Goldwasser, and Wigderson 1998) in this area show that any function can be securely computed in a distributed manner efficiently (e.g., in polynomial time with respect to the size of the circuit required to compute the function). However, this can lead to very inefficient solutions for complex functions or for large input sizes. More efficient solutions are necessary for some practical problems with complex functions or large-scale inputs in supply chain collaboration.

Some existing work built secure communication protocols for all the involved parties in supply chain collaboration based on the extension of SMC theory (Li and Atallah 2006; Hong, Vaidya, Lu, and Shafiq 2011b; Vaidya 2009b). The major advantage of SMC-based technique is that the communication protocol among all the involved parties can be provable secure (based on cryptography). In other words, we can formally prove and conclude that "Nothing can be revealed to the others" while seeking the optimal solutions in supply chain collaboration. To achieve this, most of the SMC-based protocol is somewhat inefficient to solve some complex problems in supply chain collaboration, e.g., finding the optimal scheduling without violating the local and global constraints held by different parties. Notice that this category of approaches ensures very rigorous security and strong privacy guarantee, then it is widely adopted in the development of privacy-preserving techniques.

Secure Transformation

Besides the SMC-based techniques, obfuscating the private data/information has been identified as another common privacy-preserving technique in the privacy research community. While applying to supply chain collaboration, such category of approaches is considered as secure transformation (e.g., Vaidya 2009a). As the name implies, the main idea of implementing such approach is: the data/information owner transforms its data/information into a randomized format and then shares it with other parties. Then, they analyze the randomized data/information or solve a new, transformed problem formulated with the randomized data/information. Note that the proposed approach can be concluded to be effective if, and only if, it can show that nothing about other parties' private information can be derived from the randomized data/information. Finally, every party can reconstruct their jointly computed result without sharing their private proprietary information.

For instance, in Vaidya (2009a), every party mathematically formulates its share of the optimization problem with its local proprietary information, and imposes private transformation (which is known to itself only) to its share of the mathematical formulation. Thus, the linear programming problem is transformed to a randomized format that reveals minor information (no proprietary information leakage can be identified or inferred), even though the transformed data/information has been revealed to external parties. Finally, each party can reconstruct its share of the decision in the collaborative optimal operation without knowing other parties' local private information.

Although the secure transformation-based approaches cannot always be considered as "Provably Secure" (uncertain privacy risk may still exist), maintaining good privacy protection and dramatically enhancing the efficiency could be a big plus in developing privacy-preserving applications in supply chain collaboration. Many existing works belong to this category, as well.

III. REVIEW OF PUBLISHED THEORETICAL WORK

Privacy is an important problem in many diverse areas. Several research communities have independently developed solutions that contribute to privacy protection. Much of this work can be leveraged, if appropriately combined. In this section, we review the related theoretical work for privacy-aware supply chain collaboration. Since SMC is an effective and strong theoretical foundation for developing privacy-preserving approaches in different contexts, including supply chain collaboration, we first present the literature development of SMC in the subsection Secure Multiparty Computation (note that another category of privacy-preserving technique—transformation-based technique—is problem-oriented and highly diverse; very limited summary of the theoretical work can be given). In addition, many optimization models are important theory used to tailor specific methods for solving the practical problems in supply chain collaboration. As different parties collaborate to optimize their operations, collaborative optimization models with private inputs are the essence. We then detail how to securely solve various collaborative optimization problems (using both SMC and Secure Transformation-based approaches) in the second subsection, Privacy-Preserving Collaborative Optimization: Theoretical Models of Privacy-Aware Supply Chain Collaboration.

Secure Multiparty Computation (SMC)

The setting of SMC encompasses tasks as simple as coin-tossing and broadcast, and as complex as electronic voting, electronic auctions, electronic cash schemes, contract signing, anonymous transactions, and private information-retrieval schemes. The key idea behind SMC is that a computation is secure if at the end of the computation, no party learns anything except its own input and the results (and anything that can be inferred from these two pieces). The gold standard is that of a trusted third party that performs the entire computation. Thus, the key is to achieve the same results without having a trusted third party. While some communication is obviously required in order to perform the computation, it is necessary to stick to messages that are useful, yet do not reveal anything new. How is this possible? The answer lies in non-determinism. By allowing non-determinism in the exact values sent in the intermediate communication (e.g., encrypt with a randomly chosen key), and proving that a party using its own input and the result can generate a predicted intermediate computation that is as likely as the actual values is sufficient to show that no new information is revealed.

Secure computation has a very rich history. Yao (1986) first postulated the two-party comparison problem (Yao's Millionaire Protocol) and developed a provably secure solution. Goldreich et al. (1987) generalized this to multiparty computation and proved that there exists a secure solution for any functionality. The approach used is as follows: the function to be computed is first represented as a combinatorial circuit, and then the parties run a short protocol for every gate in the circuit. Every participant gets random shares of the input and output wires for every gate. This approach, although appealing in its generality and simplicity, means that the number of rounds of the protocol grow with the size of the circuit. This grows with the size of the input. This is highly inefficient for large-scale inputs or complicated circuits, as in the supply chain optimization problems. Although this proves secure solutions exist, achieving efficient secure solutions for a wide variety of supply chain collaboration problems is still open.

There has been significant theoretical work in this area. Both Yao (1986) and Goldreich et al. (1987) assumed polynomial time bounded passive adversaries. In a line of work initiated by Ben-Or et al. (1998), the computational restrictions on the adversary were removed, but users were assumed to be able to communicate in pairs in perfect secrecy. Ben-Or et al. (1998) assumed passive adversaries (they are honest to follow the protocol, but curious to derive private information from each other), while Chaum, Crepeau, and Damgard (1988) extended this to active adversaries (they may cheat and corrupt the protocol). Ostrovsky and Yung (1991) introduced the notion of mobile adversaries, where the corrupt users may change from round to round. Finally, the coercing adversary who can force users to choose their inputs in a way he favors was introduced in the context of electronic elections by Benaloh and Tuinstra (1994), and generalized to arbitrary multiparty computation by Canetti and Gennaro (1996). Much effort has been devoted to developing crisp definitions of security (Mielikainen 2004; Goldwasser and Levin 1991; Ben-Or 1993). However, due to efficiency reasons, it is completely infeasible to directly apply the theoretical work from SMC to form secure protocols for supply chain optimization.

SMC does make two key contributions to protecting privacy (every party's local proprietary information) in supply chain collaboration: first, accommodating methods for securely computing functions with private inputs, like the building blocks of the complex problem in the supply chain; second, definitions and proof techniques for private and secure computations in a distributed environment, since the SMC theory could also be an effective privacy leakage quantifier (Goldreich 2004).

Privacy-Preserving Collaborative Optimization: Theoretical Models of Privacy-Aware Supply Chain Collaboration

Optimization problems occur in all walks of real life, particularly in the context of supply chain management. There is work in collaborative optimization that aims to achieve a global objective using only local information. This falls in the general area of distributed decision making with incomplete information. This line of research has been investigated in a worst-case setting (with no communication between the distributed parties) by Papadimitriou and Yannakakis (1991, 1993). They first explored the problem facing a set of decision makers who must select values for the variables of a linear program when only parts of the matrix are available to them, and prove lower bounds on the optimality of distributed algorithms having no communication. Awerbuch and Azar (1994) proposed a distributed flow control algorithm with a global objective that gives a logarithmic approximation ratio and runs in a polylogarithmic number of rounds. Bartal, Byers, and Raz's (2004) distributed algorithm obtains a better approximation while using the same number of rounds of local communication.

Collaborative/Distributed constraint satisfaction was formalized by Yokoo, Durfee, Ishida, and Kuwabar (1992) to solve naturally distributed constraint satisfaction problems. These problems are divided between parties, who then have to communicate among themselves to solve them. To address distributed/collaborative optimization, complete algorithms like OptAPO and ADOPT have been recently introduced. ADOPT (Modi, Shen, Tambe, and Yokoo 2003) is a backtracking-based bound propagation mechanism. It operates completely decentralized and asynchronously. The downside is that it may require a very large number of messages, thus producing big communication overheads. OptAPO (Mailler and Lesser 2004) centralizes parts of the problem; it is unknown *a priori* how much needs to be centralized where, and privacy is an issue. Collaborative/Distributed local search methods like DSA/DBA (Kirkpatrick, Gelatt, and Vecchi 1983; Zhang and Wittenburg 2003) for optimization, and DBA for satisfaction (Yokoo and Hirayama 1995) started with a random assignment, and then gradually improved it. Sometimes they produced good results with a small effort. However, they offered no guarantees on the quality of the solution, which can be arbitrarily far from the optimum. Termination is only clear for satisfaction problems, and only if a solution was found.

DPOP (Petcu and Faltings 2005e) is a dynamic programming-based algorithm that generates a linear number of messages. However, in case the problems have highly induced width, the messages generated in the high-width areas of the problem become too large. There have been proposed a number of variations of this algorithm that address this problem and other issues, offering various tradeoffs (see Petcu and Faltings 2005a, 2005b, 2005c, 2005d, 2005e, 2005f, 2006). Petcu and Faltings (2005a) propose an approximated version of this algorithm, which allows the desired tradeoff between solution quality and computational complexity. This makes it suitable for very large, distributed problems, where the propagations may take a long time to complete.

However, in general, much of the work in collaborative optimization has concentrated on reducing communication costs and has paid little or no attention to security constraints. Thus, some of the summaries may reveal significant information. In particular, the rigor of security proofs has not been applied much in this area. There is some work in secure optimization. Silaghi and Rajeshirke (2004) showed that a secure combinatorial problem solver must necessarily pick the result randomly among optimal solutions to be really secure. Silaghi and Mitra (2004) proposed arithmetic circuits for solving constraint optimization problems that are exponential in the number of variables for any constraint graph. A significantly more efficient optimization protocol specialized on generalized Vickrey auctions and based on dynamic programming was proposed by Suzuki and Yokoo (2003), although it is not completely secure under the framework in Silaghi and Rajeshirke (2004). Yokoo, Suzuki, and Hirayama (2002) also proposed a scheme using public key encryption for secure collaborative/distributed constraint satisfaction. Silaghi, Faltings, and Petcu (2006) showed how to construct an arithmetic circuit with the complexity properties of DFS-based variable elimination, and that finds a random optimal solution for any constraint optimization problem. However, much of this work is still based on generic solutions and not quite ready for practical use. Even so, some of this work can definitely be leveraged to advance the state of the art by building general transformations or privacy-preserving variants of well-known methods.

More recently, there has been significant interest in the area of privacy-preserving collaborative optimization, which gives perfect solutions to protect private information for their corresponding applications in supply chain collaboration. We now introduce some of the relevant work here, and then discuss some of their applications in Section IV. Essentially, we can classify such work into two groups: (1) linear programming, and (2) other optimization problems, since linear programming is the most fundamental problem in optimization and has many applications in supply chain management.

On one hand, we describe the work in privacy-preserving linear programming in terms of different problems solved based on either secure transformation or SMC.

1. Du (2001) and Vaidya (2009a) assume a two-party collaborative linear programming problem, where one party holds the objective function while the other party holds the constraints. They transform the linear programming problem by letting two parties jointly post-multiply the same monomial matrix to both the constraints matrix and the objective function. Finally, they can obtain the optimal solution of the original problem by pre-multiplying the same monomial matrix to the optimal solution of the transformed problems. Bednarz, Bean, and Roughan (2009) and Bednarz (2012) pointed out a potential attack to the above transformation approach. To correct the flaw in Vaidya (2009a), Hong and Vaidya revised the transformation and extended the work to the multiparty scenario in Hong and Vaidya (2013b). For the same linear programming problem, Vaidya (2009b) proposed a secure revised simplex approach using SMC to protect each party's private information in the collaboration.

2. Li and Atallah (2006) addressed the collaborative linear programming problem between two parties where the objective function and constraints can be arbitrarily partitioned, and proposed a secure simplex method for such problem using cryptographic tools (SMC).
3. For a multiparty case (two or more participants in the collaborative linear programming problem), Mangasarian (2012, 2011) presented two transformation approaches for horizontally partitioned linear programs, and vertically partitioned linear programs, respectively. Different from "Horizontal Supply Chain Collaboration," horizontally partitioned linear program means every constraint belongs to only one party. Also different from "Vertical Collaboration," vertical partition means that every constraint is co-held by different parties. Then, W. Li, H. Li, and Deng (2013) extended the transformation approach (Mangasarian 2012) for horizontally partitioned linear programs with equality constraints to inequality constraints. However, Hong and Vaidya (2013a) have identified a potential inference attack to Mangasarian's (2012) and Li et al.'s (2013) transformation-based approach recently, and revised the transformation with significantly enhanced security guarantee.
4. Apart from the horizontally or vertically partitioned linear programming problem in a multiparty scenario, Hong, Vaidya, and Lu (2011a, 2012a) proposed approaches to securely solve the arbitrarily partitioned collaborative linear programming problems in both semi-honest and malicious adversarial models. Dreier and Kerschbaum (2011) also proposed a secure transformation approach for a complex data partition scenario, and illustrated the effectiveness of their approach. Catrina and de Hoogh (2010) presented a solution to solve collaborative/distributed linear programs based on secret sharing. The protocols utilized a variant of the simplex algorithm and secure computation with fixed-point rational numbers, optimized for such application.

On the other hand, many other collaborative optimization problems can be the theoretical model for many applications of supply chain collaboration. Sakuma and Kobayashi (2007) proposed a genetic algorithm for securely solving a two-party distributed traveling salesman problem (TSP). They consider the case that one party holds the cost vector while the other party holds the tour vector. The TSP that is completely partitioned among multiple parties has been discussed, but not solved, in Sakuma and Kobayashi (2007). Hong (2013) generalized the scenario of Sakuma and Kobayashi (2007): the TSP can be jointly formulated by more than two parties to further reduce the transportation cost. Hong (2013) also solved such collaborative TSP with SMC-based meta-heuristics. Meanwhile, Hong et al. (2011b) addressed the privacy concern in a collaborative graph coloring problem that is partitioned among two or more parties. It is the theoretical model of the collaborative scheduling in Section I. An SMC-based solver is also given for the above problem to securely complete the scheduling without sharing the proprietary information.

In summary, Table 2 shows some representative work on collaborative optimization. Whether privacy protection has been integrated into the published work and how strong it is have been identified in the table—from the theoretical perspective. Note that "Minor" privacy protection means that some private information can be protected with the technique, but there is no privacy analysis or specific privacy-preserving technique proposed; "No" privacy protection means there is no privacy consideration; "Strong" privacy protection means that all the private information can be protected; and "Risky" means that the proposed technique can be attacked in certain scenarios. We can observe that privacy-preserving techniques have been developed for most of the optimization models or research problems in the table (only excepting Dynamic Programming), and some privacy-preserving linear programming techniques are still vulnerable.

TABLE 2

Summary of Collaborative Optimization Research (Theory)

Collaborative Optimization	Privacy Protection	Selected Research Work (Or Studied Problem)
Linear Programming	Minor	(Papadimitriou and Yannakakis 1991)
	Minor	(Papadimitriou and Yannakakis 1993)
	Strong	(Du 2001; Vaidya 2009a)
	Strong	(Bednarz et al. 2009; Bednarz 2012)
	Risky	(Vaidya 2009b)
	Strong	(Li and Atallah 2006)
	Risky	(Mangasarian 2012)
	Strong	(Mangasarian 2011)
	Risky	(Li et al. 2013)
	Strong	(Hong and Vaidya 2013a)
	Strong	(Dreier and Kerschbaum 2011)
	Strong	(Catrina and de Hoogh 2010)
	Strong	(Hong et al. 2011a, 2012a)
Resource Allocation	No	(Awerbuch and Azar 1994)
	No	(Bartal et al. 2004)
	Strong	(Hong et al. 2011b)
Constraints Satisfaction	No	(Yokoo et al. 1992)
	Minor	(Modi et al. 2003)
	No	(Mailler and Lesser 2004)
	Minor	(Yokoo and Hirayama 1995)
	Strong	(Silaghi and Rajeshirke 2004)
	Strong	(Silaghi and Mitra 2004)
	Strong	(Yokoo et al. 2002)
	Strong	(Silaghi et al. 2006)
Dynamic Programming	No	(Petcu and Faltings 2005a–2005f)
Vickrey Auction	Strong	(Suzuki and Yokoo 2003)
Traveling Salesman	Strong	(Sakuma and Kobayashi 2007)
	Strong	(Hong 2013)
Graph Coloring	Strong	(Hong et al. 2011b)

IV. REVIEW OF APPLICATIONS IN PRIVACY-AWARE SUPPLY CHAIN COLLABORATION

Previously, research on supply chain management focused on the centralized case: a standalone decision is made by optimizing a single objective function (e.g., maximum profit or minimum cost) for decision makers. Today, with the influence of marketing, optimization for the supply chain becomes distributed by involving more parties and optimizing a joint objective function or multiple local objective functions. In this section, we first illustrate the applications of the addressed theoretical models—privacy-preserving collaborative optimization in supply chain collaboration—and then review some published work on specific collaborative problems in supply chain management. An overview of the representative privacy-aware applications is given in Table 3. We now address the applications below.

TABLE 3

Overview of Privacy-Aware Supply Chain Collaboration (Applications)

Privacy-Aware Applications	Selected Work and Techniques
Load Swap by Distributors	(Clifton et al. 2008), SMC
Capacity Allocation	(Atallah et al. 2003), SMC
E-auctions	
Price Competing by Multiple Retailers and One Supplier	(Li and Zhang 2008), SMC
Influence of Confidentiality on Prices and Profits	(Daughety and Reinganum 2007), SMC
Procurement Negotiation	(Deshpande et al. 2011), SMC
Transportation Assignment	(Du 2001; Vaidya 2009a), Transformation
Production Assignment	(Bednarz et al. 2009), Transformation
Flight Crew Assignment/Rostering	(Bednarz 2012), Transformation
(Linear Programming)	(Vaidya 2009b), SMC
	(Li and Atallah 2006), SMC
	(Mangasarian 2012), Transformation
	(Mangasarian 2011), Transformation
	(Li et al. 2013), Transformation
	(Hong and Vaidya 2013a), Transformation
	(Dreier and Kerschbaum 2011), Transformation
	(Catrina and de Hoogh 2010), Secret Sharing/SMC
	(Hong et al. 2011a), Transformation
	(Hong et al. 2012a), Transformation
Location	(Yokoo et al. 2002), SMC
Scheduling	(Silaghi and Mitra 2004), SMC
Car Sequencing	(Silaghi and Rajeshirke 2004), Transformation
Cutting Stock	(Silaghi et al. 2006), SMC
Vehicle Routing	(Hong et al. 2011b), SMC
Rostering	
(Constraint Satisfaction)	
Vickrey Auction	(Suzuki and Yokoo 2003), SMC
Transportation	(Sakuma and Kobayashi 2007), SMC
Production, etc.	(Hong 2013), SMC
(Traveling Salesman Problem)	
Resource Allocation	(Hong et al. 2011b), SMC
Scheduling, etc.	
(Graph Coloring)	

Linear Programming

Linear programming can be used to model many supply chain problems. In transportation, given the supply amount and demand amount of all the locations, the transportation cost can be minimized by finding the optimal delivery assignment with linear programming. In manufacturing, given the limited raw materials, the profit of all the products can be maximized by finding the optimal production assignment with linear programming. In the airline industry, airline companies can make the optimal assignment on crews to their flight by minimizing the cost while satisfying several constraints (e.g., each flight is covered, each pilot can only fly a certain amount each day). In many supply chain collaborations, linear programming problems are formulated with private proprietary information from different suppliers, distributors, and retailers. The existing work has successfully protected the private information among all the parties.

More specifically, Hong et al. (2011a, 2012a) enable General Mills and Land O'Lakes to find the optimal delivery assignment in collaborative logistics without sharing their private delivery information, and also accommodate secure channels for different manufacturers to obtain the optimal production assignment without sharing their private production information. The constraints of linear programming problem in Mangasarian (2012) and Li et al. (2013) have been horizontally partitioned; then the proposed approach can be utilized to secure the production process in which each factory privately holds a different kind of raw material, or secure the task-machine scheduling process in which every machine is held by one party. The vertically partitioned linear programming problem in Mangasarian (2011) could secure the transportation in which all companies share their trucks to bound the shipping. In addition, the work of Vaidya (2009a, 2009b) and Du (2001) protects the private proprietary information in the following two-party scenario: one party is the manufacturer/supplier while the other party is the distributor/shipping company—the manufacturer/supplier wishes to obtain the minimum transportation cost from the distributor; however, the manufacturer/supplier is not willing to share their supply and demand information for different locations. In summary, since the privacy issues in many variants of the collaborative linear programming have been well studied in literature, the private information of all the involved participants can be protected in most of the linear programming-based supply chain collaboration.

Traveling Salesman Problem (TSP)

TSP is a computationally difficult problem in combinatorial optimization (Papadimitriou and Steiglitz 1982). Given a set of cities and the distances between every pair of cities, the optimization solver finds the shortest route to visit all the cities once and back to the original city. Clearly, the overall distance is minimized in such problem. This optimization model has many important applications, such as logistics, planning, and microchip production, where the distance and city can represent different objects in different scenarios. More generally, the problem can be formulated as: given the cost between every pair of cities (replacing the distance), minimizing the total cost of the traveling route to visit all the cities.

An important application of TSP can be also found in transportation. To deliver goods to a given set of cities, the overall transportation cost could be minimized by solving a TSP. Sakuma and Kobayashi (2007) investigated the privacy concern in a two-party collaborative TSP in a supply chain: one party is the supplier while the other party is the distributor (shipping companies). Before signing the contract, the supplier would like to know the possible minimum cost, but they are not willing to share the set of cities to the distribution. The proposed approach in Sakuma and Kobayashi (2007) makes this possible. More recently, Hong (2013) extended it to a more general scenario in the supply chain, by eliminating the constraint of only one distributor. A new secure solver for the generalized scenario is also given in Hong (2013). Similarly, the private information of all the involved participants in TSP-based supply chain collaboration can also be protected.

Constraint Satisfaction Problem (CSP)

Given a finite domain for some variables, a constraints satisfaction problem seeks the variable values from the domain such that all the constraints can be satisfied (e.g., the decision version of the graph coloring problem [Hong et al. 2011b]). Constraint satisfaction problem can be utilized to solve a wide variety of practical problems in operations research and supply chain management, such as location, scheduling, car sequencing, cutting stock, vehicle routing, and rostering (Brailsford, Potts, and Smith 1999), detailed as below:

- Location: locating the facilities (e.g., warehouses) to supply the demand of customers
- Scheduling: see Figure 2
- Car sequencing: sequence the cars on the assembly line after the basic model has been manufactured (adding different options, e.g., air-conditioning, ABS brakes), where no workstation capacity (constraint) is exceeded (violated)
- Cutting stock: in production, minimizing waste during cutting the material to smaller pieces according to the requirements
- Vehicle routing: in transportation, distribution, and logistics, seeking to service customers with a number of vehicles from a single or multiple depots within a time window
- Rostering: allocating staff to duties by scheduling and rostering

The security and privacy issues in many variants of constraints satisfaction problem have been theoretically tackled, as shown in Table 2. Therefore, if any (collaborative) application of constraint satisfaction problem calls for the privacy preservation among different parties, the published work in Tables 2 and 3 could do so.

Graph Coloring

Graph coloring covers a set of "coloring"-related optimization problems over the graph, such as vertex coloring, edge coloring, and total coloring. Note that other coloring problems can be transformed into a vertex coloring version (Papadimitriou and Steiglitz 1982). Thus, the prior work (Hong et al. 2011b) focused on vertex coloring problem. Specifically, given a graph, we assign colors to all the vertices where every pair of adjacent vertices cannot be assigned with the same color, and find the minimum number of colors. A typical application of this optimization model is the scheduling for supplier, distributor, or retailers with time conflicts of different jobs. If the jobs should be done by different parties (e.g., the collaborative scheduling in Figure 2), their time conflicts and the assigned time slots would be the private proprietary information in the collaboration. Hong et al. (2011b) presented a secure communication protocol that can help different parties in the supply chain complete the scheduling without revealing any private information.

Other Specific Applications in the Supply Chain

Many other specific applications in the supply chain have privacy concerns. For example, Clifton et al. (2008) focused on a problem faced by independent distributors that have separate pick-up and delivery tasks and wish to swap tasks for enhancing efficiency. They proposed a privacy-preserving technique to limit the revealed information while identifying those swaps. Beyond the privacy protection in such supply chain collaboration, they also proved that all the participants are incentive-compatible—following the protocol and providing correct input. Atallah, Deshpande, Elmongui, and Schwarz (2003) presented secure protocols for two types of supply chain interaction: capacity allocation and e-auctions. For capacity allocation, they consider a single supplier and N retailers: the supplier has a constant marginal production cost, but limited capacity; the retailers operate in non-competing retail markets. In the allocation, the supplier does not know the demand information of the retailers; otherwise, the supplier will try to capture all the retailers' profits. Besides this, all the retailers are enforced to play honestly in the protocol (note that prior to Atallah et al.'s [2003] work, Deshpande and Schwarz [2005] designed an incentive-compatible mechanism to guarantee honesty while revealing the demand information). For e-auctions, they consider two broad models: (1) all buyers/bidders get the same unit price from the supplier, and (2) different buyers can get different prices from the supplier depending on their demand. The proposed protocol can also guarantee honesty and collusion prevention beyond privacy protection.

In some supply chain management-related areas such as marketing, information sharing occurs very often between retailers and suppliers/manufacturers. Li and Zhang (2008) protected the private information in a decentralized supply chain, where one manufacturer supplies to multiple retailers competing in price. The retailers have their own demand information (which might be disclosed to the manufacturer), while the manufacturer sets a wholesale price based on the received information. If the information is not kept confidential, other parties might be benefited from the market. They presented three information-sharing scenarios: (1) disclosing information to the manufacturer and all the retailers, (2) disclosing information to the manufacturer and all participating retailers, and (3) disclosing information to the manufacturer only. Then, they solved a three-stage game for each scenario, compared the payoffs, and investigated the impact of the privacy protection. Daughety and Reinganum (2007) studied the influence of confidentiality on prices and profits through modeling non-cooperative signaling by two firms that compete over customers. They assume that consumers have private concern about their own preferences for the products, while the product quality is the firm's private information. Deshpande, Schwarz, Atallah, Blanton, and Frikken (2011) developed and tested a privacy-preserving business process to negotiate the procurement of component parts to be used by an electronic manufacturing service (EMS) in the manufacture/assembly of the branded products of an original equipment manufacturer (OEM). The private information from both OEMs and EMSs (individual component prices) are protected—no participant can reconstruct the private inputs from other participants in the business process.

V. FUTURE RESEARCH TREND AND CHALLENGE DISCUSSION

The importance of privacy preservation in supply chain management has attracted wide attention in academia. One popular topic studying information sharing in the supply chain with protected privacy during the last decade is the impact of information asymmetry in supply chain contract. The recent work on it can be found in Lim (2001), Cachon and Zhang (2006), Li and Debo (2009), Ren, Cohen, Ho, and Terwiesch (2010), Kayis, Erhun, and Plambeck (2013), Zhang, Zhou, and Liu (2010), and Kim and Netessine (2013). It has been widely accepted that disclosing private information, such as retailers' demand to channel partners, can help accommodate more benefits for the whole supply chain and, at the same time, can also bring in more risks. Simple mathematical models based on game theory are the primary tools used to analyze the supply chain efficiency and effectiveness under different ways of sharing private information. Nevertheless, when the number of channel partners becomes bigger and the supply chain network becomes more complex, these models can hardly be established and analyzed. In this case, empirical studies and simulation techniques will be promising substitutes as the analytical tools.

Meanwhile, different industries have their own business characteristics that must be considered when the supply chain contract is investigated. For example, Miller and Tucker (2009) study the patient data sharing among hospitals where healthcare supply chain features are presented. Similar work in other applications areas is limited. Thus, as indicated in Gal-Or and Ghose (2005), research with specific applications in various industries needs to be explored.

Although optimization models have been applied in security and privacy issues of the supply chain for many decades, most do not consider specific quantitative measures of security risks (Smith, Watson, Baker, and Pokorski 2007). Very few results can be found in literature where security is quantified in supply chain applications (Meixell and Norbis 2012). Notice that quantitatively measuring security/privacy has been studied in many areas with different private information (e.g., tabular data [Sweeney 2002], statistical data [Dwork 2008], search data [Hong, He, Vaidya, Adam, and Atluri 2009; Hong, Vaidya, Lu, and Wu 2012b], and social data [He, Vaidya, Shafiq, Adam, and Atluri 2009]). Whenever an information security breach happens in a supply chain, it will result in propagation along the supply chain (Bandyopadhyay, Jacob, and

Raghunathan 2010). Appropriate computation for security risk propagation gives firms a comprehensive picture of their potential losses and, therefore, helps avoid bad decisions. Thus, a good risk measure would greatly benefit the protection of private proprietary information in the future supply chain collaboration.

Among existing research providing information security mechanisms, protocols, and algorithms between practical supply chain partners, most of the work focused on either vertical collaboration (supplier or manufacturer to retailer or customer) (Atallah, Deshpande, Frikken, and Schwarz 2004; Li and Zhang 2008; Chen, Wang, Liu, and Singh 2011; Zhang et al. 2010), or horizontal collaboration (supplier to supplier or retailer to retailer) (Clifton et al. 2008; Miller and Tucker 2009). However, privacy protection issues in a full-collaboration relationship incorporating both vertical and horizontal collaborations are seldom studied. Note that Hong et al. (2011a, 2012a) studied a mixed data/information partition model for securely solving collaborative linear programming. However, such a model still belongs to the horizontal collaboration in the supply chain. Moreover, the collaborative graph coloring model in Hong et al. (2011b) can be applied to any kind of collaboration. Such work presented a theoretical model based on privacy-preserving graph coloring, and practical applications with potential real-world constraints in supply chain collaboration have not been discussed yet. Also, the collaboration in the TSP problem addressed in Hong (2013) involves one supplier/manufacturer and several distributors, which has initialized the full collaboration recently. A representative discussion accounting for both collaborations can be also found in Meixell and Norbis (2012). When information is intended to be shared along two dimensions in a supply chain, existing results cannot be directly applied. Therefore, there is a need for the design of new information-sharing mechanisms and algorithms with privacy protection in such complicated scenarios.

Zeng, Wang, Deng, Cao, and Khundker (2012) summarized the literature related to supply chain collaboration and security. Besides the security issues studied in such work, potential privacy concerns need to be addressed and tackled in various supply chain collaborations in the future. Also, among all the results in their paper, most have been focused on tackling theories, models, and algorithms. There is a significant lack of work with an emphasis on real-world case studies and applications. In Chen et al. (2011), the authors use real data from the Australian bulk material supply chains industry to evaluate the collaborations among different parties in a coal chain. However, similar work is also difficult to find in this area. Some of the results imply the feasibility of hypotheses testing about vertical and horizontal information sharing, demand, and inventory sharing, and supply chain effectiveness. We should expect more real-world applications to emerge in the future to verify the analytical results obtained so far.

Technical and Practical Challenges

We now address the potential challenges that we may encounter following this line of research. As the theoretical foundations of privacy-aware supply chain collaboration, privacy-preserving collaborative optimization has its unique features that pose several specific and practical challenges, in addition to the general challenges underlying privacy-preserving computation:

- *Utility Metrics for Secure Transformation*. In the context of collaborative optimization, we can exclusively identify two groups of secure transformation-based techniques for privacy preservation: (1) transformation does not alter the original optimal solution (Vaidya 2009a; Du 2001; Hong et al. 2012a; Mangasarian 2012), and (2) the quality of the optimal solution may vary (but the utility of the output can be retained to some extent [Silaghi and Rajeshirke 2004]). For the former one, no utility metric is required to evaluate the output quality since the original optimal solution can be either derived directly (the solution of the transformed problem is equal to the original one [Mangasarian 2012]) or reconstructed with an inverse transformation (Vaidya 2009a; Du 2001; Hong et al. 2012a). For the latter one, we need an effective utility metric that measures how useful the new output becomes, since the problem might be transformed to another format per conventional manners (Agrawal and Srikant 2000; Sweeney 2002). One possibility is to use degree of difference—i.e., the amount of difference between the original and the transformed data in the problem formulation. However, this may not accurately reflect the utility of the data in optimization/supply chain management. For example, consider two transformations—one in which all of the constraints are slightly changed, and another where certain constraints are significantly changed. While the total difference might be greater in the second transformation, those changes may be redundant and the optimal solution may remain the same, while in the first case, the optimal solution may be changed. The problem is that the difference that matters is with the optimal solution (i.e., in general, according to a specific metric/goal), which is typically unknown before solving the problem. Instead, sensitivity analysis can help figure out how well the transformation works by gauging the degree of tolerance (Hong et al. 2012b). In itself, the tolerance level of different data sets to transformation may be different.

- *Information Leakage beyond SMC*. Exterior knowledge needs to be effectively modeled to decide whether a particular computation may lead to privacy breach. The theory of SMC is only concerned with the correct and secure evaluation of a function or the composition of numerous functions, not about what exterior knowledge exists and how it affects the privacy/security. However, this has to be accounted for real and practical use. For the problems in supply chain collaboration, exterior knowledge could itself be modeled in the form of constraints on the problem. One possibility is to write the constraints on the private data and then run a constraint satisfaction algorithm to find feasible answers. By modeling exterior knowledge, also, in the form of constraints, we can judge the difference in availability of feasible solutions and, thus, determine the degree of effect the exterior knowledge has. Similarly, result analysis is increasingly crucial to practical secure computation (private information might be inferred from the input and output of many collaborative optimization problems [Hong and Vaidya 2013a]). Even if the function is securely computed, there is still a problem that the results themselves may leak information. Kantarcioglu, Jin, and Clifton (2004) showed one approach to quantifying this.
- *Incentives to Dishonest Play*. For any multiparty computation, besides learning information in a semi-honest adversarial model (Goldreich 2004), all the participants may play dishonestly for gaining additional payoff. The research then goes to the intersection of cryptography and game theory. Many such works exist in literature; for example, Fischer and Wright (1993) provided an application of game theoretic techniques to the analysis of a class of multiparty cryptographic protocols for secret bit exchange. Dodis, Halevi, and Rabin (2000) provided a cryptographic protocol to the correlated element selection problem. Teague (2004) extended this protocol to work also for non-uniform distribution. Other work that addresses the same problem without help from a third party includes Barany (1992), Ben-Porath (1998), Gerardi (2004), Lepinski, Micali, Peikert, and Shelat (2004), and Atallah, Blanton, Frikken, and Li (2006). Matsuura (2003) provided a survey of the emerging interdisciplinary area between information security and economics. Specifically, in supply chain collaboration, researchers need to pay more attention to this malicious payoff-gaining incentive of all the participants, since most of the participants are competitors in horizontal collaboration (e.g., General Mills and Land O'Lakes). Clifton et al. (2008) explored secure solutions to the transportation load-swapping problem, and showed how such a protocol can be incentive-compatible, thus protecting from malicious adversaries. Hong et al.'s (2012a) secure protocol provides incentive compatibility for collaborative linear programming problems where constraints are arbitrarily partitioned. Li and Zhang

(2008) studied a three-stage game in the pricing problem of a decentralized supply chain scenario between one supplier/manufacturer and multiple retailers/customers. The game theoretical approach needs to be explored further in almost every supply chain collaboration.

- *Ensuring Correct Inputs*. Another related issue is how to ensure that participants stick to their correct inputs for the protocol. Mechanism design in game theory can help with this, as well as ensuring that if one cheats, either they are caught or else they suffer. Clifton et al. (2008) showed a representative example where all parties are bound to submit their correct inputs, as well as stick to the protocol due to their own incentives. However, in many general problems in supply chain collaboration, this may be difficult and challenging.
- *Privacy Leakage in Decision Implementation*. Supply chain collaboration does not usually terminate when the optimal solution/decision is jointly computed. They have to eventually complete the best operation with the decision of the optimal solution. Occasionally, different parties share resources (e.g., sharing delivery trucks in collaborative logistics [Turban et al. 2005], sharing raw materials in collaborative production [Hong et al. 2012a]). Is it possible that additional exterior knowledge can be modeled or learned from each other in such processes? How to quantify such privacy leakage? These are new challenges we have to tackle beyond the original goal of privacy preservation—ensuring the collaborative computation does not leak any private information in addition to the received messages in the SMC protocol.
- *New Security Definitions for Difficult Problems*. One of the biggest successes of SMC is that it has provided a solid, quantifiable mathematical way of proving the security of an algorithm. One can have confidence in the security of an algorithm after it has been proven secure in the SMC framework. However, the computational security definitions in SMC depend on the fact that problems can be solved in polynomial time (i.e., the solutions are "efficient"). What happens when we are considering problems that are exponential in the worst case? A new set of security definitions must be formulated to define and quantify security in this case. Many optimization problems in supply chain collaboration are good candidates for this, since many solutions for nonlinear programming and integer programming are exponential in the worst case. Tackling such issues can also advance the state-of-the-art in the security and privacy community.
- *Iteration*. Iterative algorithm is a significant obstacle to efficient, secure computation (Goldreich 2004). In supply chain management, many optimization models typically require iteration by nature. The effect of iteration on security is enormous. It is possible to write a secure algorithm such that the results of all iterations are also kept secret, but this is certain to lead to severe problems with efficiency. Instead, each iteration could probably be independently made secure with good efficiency. The question then is what to do with the results computed from numerous functions. Can they be protected in some form? What do they reveal? What is the threshold for security—how many iterations are to be allowed before a security violation occurs? Indeed, is it possible to define the parameters for a secure violation? The issue of composability of secure protocols, as well as parallelizability, has been well studied (Lindell, Lysyanskaya, and Rabin 2002; Canetti, Lindell, Ostrovsky, and Sahai 2002; Canetti, Kushilevitz, and Lindell 2006; Lindell 2003a, 2003b; Kushilevitz, Lindell, and Rabin 2006). But the issue of iteration needs to be similarly addressed in securing the process of collaborative optimization/supply chain collaboration.
- *Integral Values versus Precision and Definitions*. A key problem is the fact that provable security in cryptography requires that any algorithm only operates over numbers in a field. Thus, all numbers are required to be integral. This can cause havoc with an optimization algorithm in the supply chain, since many numbers are expected to be real. Indeed, many algorithms are notoriously sensitive to the quality of the data (slightly altering the data may

even result in infeasibility of the optimization problem). Simply adding a few bits of precision and converting to integers is not likely to be sufficient to fix the problems. Indeed, this might require that the current definitions of security themselves be relaxed a little to allow efficient handling of real versus integer issues, as well as to handle the issue of malicious adversaries. Protecting against malicious adversaries currently requires great effort and causes severe efficiency constraints. However, given the subtle nature of security, the implications of any change must be carefully studied.

VI. CONCLUSION

With increasing globalization of the market, nowadays, enterprises increasingly explore their reliable supply chains to compete with each other. Information technologies have featured various aspects of supply chain management with their rapid expansion. In order to gain extra benefits or further cost cut, every corporation's supply chain spreads extensively to multiparty collaboration in the current networked market. Along with the recent successful research results on information sharing in supply chain collaboration, the privacy concerns of cooperative participants in this scenario have also drawn great interest. Corporations' private proprietary information gets widely involved in real-world supply chain collaboration. To prevent competitors from obtaining the private proprietary information, researchers and practitioners have made considerable efforts in this emerging area.

This paper reviews and summarizes the literature on privacy-aware supply chain collaboration. Specifically, we first investigate the privacy issues in some real-life collaborative supply chain problems, and discuss why private proprietary information plays a pivotal role in supply chain collaboration. To protect privacy in the collaboration, involved participants can either employ a trusted third party or some privacy-preserving techniques (if no trusted third party can be found). To the best of our knowledge, almost all the research work focuses on the privacy-preserving techniques, including Secure Multiparty Computation (SMC) and Secure Transformation.

Since most supply chain management problems explore the optimal solutions while satisfying the constraints under different settings, many optimization models are regarded as the theoretical foundations of practical supply chain problems with great insights. We then organize and present the literature from collaborative optimization (theory) to specific applications. More specifically, Table 2 presents the state-of-the-art of collaborative optimization. A number of typical optimization problems have their corresponding collaborative format with limited information disclosure in literature (linear programming, constraints satisfaction problem, traveling salesman problem, graph coloring, etc.). In the meanwhile, some of the earlier work only paid limited attention to the privacy issues and the proposed approach can only provide minor protection of privacy—information is partially shared (also shown in Table 2). Consequently, we summarize the privacy-aware applications in supply chain collaboration in Table 3. The applications of the typical optimization problems have been discussed, together with some specific applications in the supply chain without sharing private information.

At the moment, privacy protection has been treated as a very important problem when the researchers come up with any new problem with possible information exchanging and sharing. Therefore, in the future, more techniques applicable to new problems in supply chain collaboration are highly desirable. In addition, we also envision the potential research directions in privacy-aware supply chain collaboration by exploring the practical collaborative scenarios and the current information-sharing scheme with insufficient privacy consideration. Finally, we discuss the specific technical and practical challenges we may encounter in tackling the privacy issues of supply chain collaboration. Although those challenges may establish obstacles for securely solving new problems in supply chain collaboration, they could indeed facilitate researchers to advance the state-of-the-art of this literature, since challenges typically indicate breakthrough points for the uncertain fields.

REFERENCES

Agrawal, R., and R. Srikant. 2000. Privacy-preserving data mining. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, 439–450, Dallas, TX. Available at: http://dl.acm.org/citation.cfm?doid=342009.335438

Atallah, M. J., V. Deshpande, H. G. Elmongui, and L. B. Schwarz. 2003. Secure supply-chain protocols. In *Proceedings of the 2003 IEEE International Conference on E-Commerce*. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.158.4346&rep=rep1&type=pdf

Atallah, M. J., V. Deshpande, E. K. B. Frikken, and L. B. Schwarz. 2004. Secure supply-chain collaboration. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.131.5111

Atallah, M. J., M. Blanton, E. K. B. Frikken, and J. Li. 2006. Efficient correlated action selection. In *Proceedings of Financial Cryptography*, 296–310. Available at: https://www.cs.purdue.edu/homes/mja/sscc/documents/fc06-slides.pdf

Awerbuch, B., and Y. Azar. 1994. Local optimization of global objectives: Competitive distributed deadlock resolution and resource allocation. In *Proceedings of IEEE Symposium on Foundations of Computer Science*, 240–249. Available at: http://www.cs.tau.ac.il/~azar/conc.pdf

Baltzan, P. 2011. *Business Driven Information Systems*. Boston, MA: McGraw-Hill/Irwin.

Bandyopadhyay, T., V. S. Jacob, and S. Raghunathan. 2010. Information security in networked supply chains: Impact of network vulnerability and supply chain integration on incentives to invest. *Information Technology and Management* 11 (1): 7–23.

Barany, I. 1992. Fair distribution protocols or how the players replace fortune. *Mathematics of Operations Research* 17 (2): 327–340.

Bartal, Y., J. W. Byers, and D. Raz. 2004. Fast, distributed approximation algorithms for positive linear programming with applications to flow control. *SIAM Journal on Computing* 33 (6): 1261–1279.

Bednarz, A., N. Bean, and M. Roughan. 2009. Hiccups on the road to privacy-preserving linear programming. In *Proceedings of the 8th ACM Workshop on Privacy in the Electronic Society*, 117–120. Available at: http://www.maths.adelaide.edu.au/matthew.roughan/Papers/p117-bednarz.pdf

Bednarz, A. 2012. *Methods for Two-Party Privacy-Preserving Linear Programming*. Ph.D. thesis, University of Adelaide.

Ben-Or, M., R. Canetti, and O. Goldreich. 1993. Asynchronous secure computation. In *Proceedings of Twenty-Fifth Annual ACM Symposium on Theory of Computing*, 52–61. Available at: http://dl.acm.org/citation.cfm?id=167088.167109

Ben-Or, M., S. Goldwasser, and A. Wigderson. 1998. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, 1–10. Available at: http://www.researchgate.net/publication/221590203_Completeness_Theorems_for_Non-Cryptographic_Fault-Tolerant_Distributed_Computation_%28Extended_Abstract%29

Ben-Porath, E. 1998. Correlation without mediation: Expanding the set of equilibrium outcomes by "cheap" pre-play procedures. *Journal of Economic Theory* 80 (1): 108–122.

Benaloh, J., and D. Tuinstra. 1994. Receipt-free secret-ballot elections (extended abstract). In *STOC '94: Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing*, 544–553. Available at: http://www.researchgate.net/publication/221591588_Receipt-free_secret-ballot_elections_%28extended_abstract%29

Brailsford, S. C., C. N. Potts, and B. M. Smith. 1999. Constraint satisfaction problems: Algorithms and applications. *European Journal of Operational Research* 119 (3): 557–581.

Buss, D. 2003. Case study: Land O'Lakes and collaborative logistics. Available at: http://www.cioinsight.com/c/a/Case-Studies/Case-Study-Land-OLakes-and-Collaborative-Logistics/

Cachon, G. P., and F. Zhang. 2006. Procuring fast delivery: Sole sourcing with information asymmetry. *Management Science* 52 (6): 881–896.

Canetti, R., and R. Gennaro. 1996. Incoercible multiparty computation. In *FOCS '96: Proceedings of the 37th Annual Symposium on Foundations of Computer Science*, 504. Available at: http://www.researchgate.net/publication/3671942_Incoercible_multiparty_computation

Canetti, R., Y. Lindell, R. Ostrovsky, and A. Sahai. 2002. Universally composable two-party and multi-party secure computation. In *STOC '02: Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, 494–503. Available at: http://www.cs.ucla.edu/~rafail/PUBLIC/57.pdf

Canetti, R., E. Kushilevitz, and Y. Lindell. 2006. On the limitations of universally composable two-party computation without set-up assumptions. *Journal of Cryptology* 19 (2): 135–167. An extended abstract appeared in Eurocrypt 2003. New York, NY: Springer-Verlag (LNCS 2656), 68–86.

Catrina, O., and S. de Hoogh. 2010. Secure multiparty linear programming using fixed-point arithmetic. In *Proceedings of the 15th European Symposium on Research in Computer Security*, 134–150. Available at: http://dl.acm.org/citation.cfm?id=1888893

Chaum, D., C. Crepeau, and I. Damgard. 1988. Multiparty unconditionally secure protocols. In *STOC '88: Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, 11–19. Available at: http://dl.acm.org/citation.cfm?id=62214

Chen, S., C. Wang, D. Liu, and G. Singh. 2011. Secure multi-party collaboration systems in supply chain management. In *Proceedings of the 1st International Conference on Logistics, Informatics and Service Science*, 105–107. Available at: http://www.researchgate.net/publication/221251905_Secure_Multi-party_Collaboration_Systems_in_Supply_Chain_Management

Clifton, C., A. Iyer, R. Cho, W. Jiang, M. Kantarcioglu, and J. Vaidya. 2008. An approach to identifying beneficial collaboration securely in decentralized logistics systems. *Manufacturing & Service Operations Management* 10 (1): 108–125.

Dantzig, G. 1963. *Linear Programming and Extensions*. Princeton, NJ: Princeton University Press.

Daughety, A., and J. Reinganum. 2007. Competition and confidentiality: Signaling quality in a duopoly when there is universal private information. *Games and Economic Behavior* 58 (1): 94–120.

Deshpande, V., and L. B. Schwarz. 2005. *Optimal Capacity Choice and Allocation in Decentralized Supply Chains*. Working paper, Purdue University.

Deshpande, V., L. Schwarz, M. Atallah, M. Blanton, and K. Frikken. 2011. Outsourcing manufacturing: Secure price-masking mechanisms for purchasing component parts. *Production and Operations Management* 20 (2): 165–180.

Dodis, Y., S. Halevi, and T. Rabin. 2000. A cryptographic solution to a game theoretic problem. In *CRYPTO '00: Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology*, 112–130. Available at: http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.106.3670

Dreier, J., and F. Kerschbaum. 2011. Practical privacy-preserving multi- party linear programming based on problem transformation. In *Proceedings of the Third IEEE International Conference on Privacy, Security, Risk and Trust*, 916–924. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6113241

Du, W. 2001. *A Study of Several Specific Secure Two-Party Computation Problems*. Ph.D. thesis. West Lafayette, IN: Purdue University.

Dwork, C. 2008. Differential privacy: A survey of results. *Theory and Applications of Models of Computation* 4978: 1–19.

Evfimievski, A., R. Srikant, R. Agrawal, and J. Gehrke. 2002. Privacy preserving mining of association rules. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 217–228. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.19.3747

Fischer, M., and R. Wright. 1993. An application of game theoretic techniques to cryptography. *Discrete Mathematics and Theoretical Computer Science* 13: 99–118.

Gal-Or, E., and A. Ghose. 2005. The economic incentives for sharing security information. *Information Systems Research* 16 (2): 186–208.

Gerardi, D. 2004. Unmediated communication in games with complete and incomplete information. *Journal of Economic Theory* 114 (1): 104–131.

Goldreich, O., S. Micali, and A. Wigderson. 1987. How to play any mental game or a completeness theorem for protocols with honest majority. In *Proceedings of the 19th ACM Symposium on Theory of Computing*, 218–229. Available at: http://www.bibsonomy.org/bibtex/211382b846bc9c7535d37daf703a9a00f/danfunky

Goldreich, O. 2004. Chapter 5: Encryption schemes. In *The Foundations of Cryptography, Volume 2*. Cambridge, MA: Cambridge University Press.

Goldwasser, S., and L. A. Levin. 1991. Fair computation of general functions in presence of immoral majority. In *CRYPTO '90: Proceedings of the 10th Annual International Cryptology Conference on Advances in Cryptology*, 77–93. New York, NY: Springer-Verlag. Available at: http://www.researchgate.net/publication/221355525_Fair_Computation_of_General_Functions_in_Presence_of_Immoral_Majority

He, X., J. Vaidya, B. Shafiq, N. R. Adam, and V. Atluri. 2009. Preserving privacy in social networks: A structure-aware approach. In *Proceedings of the 8th International Conference on Web Intelligence*, 647–654. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5284903

Hong, Y., X. He, J. Vaidya, N. R. Adam, and V. Atluri. 2009. Effective anonymization of query logs. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, 1465–1468. Available at: http://dl.acm.org/citation.cfm?doid=1645953.1646146

Hong, Y., J. Vaidya, and H. Lu. 2011a. Efficient distributed linear programming with limited disclosure. In *Proceedings of the 25th Annual IFIP WG 1.3 Conference on Data and Applications Security and Privacy*, 170–185. Available at: http://dl.acm.org/citation.cfm?id=2029917

Hong, Y., J. Vaidya, H. Lu, and B. Shafiq. 2011b. Privacy-preserving tabu search for distributed graph coloring. In *Proceedings of the Third IEEE International Conference on Privacy, Security, Risk and Trust*, 951–958. Available at: http://www.bibsonomy.org/bibtex/2aa159b42f383bea482fb0cb469eff030/dblp

Hong, Y., J. Vaidya, and H. Lu. 2012a. Secure and efficient distributed linear programming. *Journal of Computer Security* 20 (5): 583–634.

Hong, Y., J. Vaidya, H. Lu, and M. Wu. 2012b. Differentially private search log sanitization with optimal output utility. In *Proceedings of the 15th International Conference on Extending Database Technology*, 50–61. Available at: http://www.bibsonomy.org/bibtex/24e11f0441865a8c682d51f2ea307bc29/dblp

Hong, Y. 2013. *Privacy-Preserving Collaborative Optimization*. Ph.D. thesis, Rutgers, The State University of New Jersey.

Hong, Y., and J. Vaidya. 2013a. An inference-proof approach to privacy-preserving horizontally partitioned linear programs. *Optimization Letters* 8 (1): 267–277.

Hong, Y., and J. Vaidya. 2013b. Secure transformation for multiparty linear programming. *Rutgers Technical Report*. Available at: http://www.albany.edu/faculty/hong/pub/SecTran.pdf

Kantarcioglu, M., J. Jin, and C. Clifton. 2004. When do data mining results violate privacy? In *Proceedings of the 2004 ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 599–604. Available at: http://dl.acm.org/citation.cfm?doid=1014052.1014126

Kayis, E., F. Erhun, and E. L. Plambeck. 2013. Delegation vs. control of component procurement under asymmetric cost information and simple contracts. *Manufacturing and Service Operations Management* 15 (1): 45–56.

Kim, S. H., and S. Netessine. 2013. Collaborative cost reduction and component procurement under information asymmetry. *Management Science* 59 (1): 189–206.

Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi. 1983. Optimization by simulated annealing. *Science* 220 (4598): 671–680.

Kushilevitz, E., Y. Lindell, and T. Rabin. 2006. Information-theoretically secure protocols and security under composition. In *STOC '06: Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing*, 109–118. Available at: http://dl.acm.org/citation.cfm?id=1132532

Lepinski, M., S. Micali, C. Peikert, and A. Shelat. 2004. Completely fair SFE and coalition-safe cheap talk. In *PODC '04: Proceedings of the Twenty-Third Annual ACM Symposium on Principles of Distributed Computing*, 1–10. Available at: http://dl.acm.org/citation.cfm?id=1011769

Li, C., and L. G. Debo. 2009. Second sourcing vs. sole sourcing with capacity investment and asymmetric information. *Manufacturing & Service Operations Management* 11 (3): 448–470.

Li, J., and M. J. Atallah. 2006. Secure and private collaborative linear programming. In *Proceedings of the 2nd International Conference on Collaborative Computing: Networking, Applications and Worksharing*, 1–8. Available at: http://www.researchgate.net/publication/224704468_Secure_and_Private_Collaborative_Linear_Programming

Li, L., and H. Zhang. 2008. Confidentiality and information sharing in supply chain coordination. *Management Science* 54 (8): 1467–1481.

Li, W., H. Li, and C. Deng. 2013. Privacy-preserving horizontally partitioned linear programs with inequality constraints. *Optimization Letters* 7 (1): 137–144.

Lim, W. S. 2001. Producer-supplier contracts with incomplete information. *Management Science* 47 (5): 709–715.

Lin, X., C. Clifton, and M. Zhu. 2005. Privacy preserving clustering with distributed EM mixture modeling. *Knowledge and Information Systems* 8 (1): 68–81.

Lindell, Y., A. Lysyanskaya, and T. Rabin. 2002. On the composition of authenticated byzantine agreement. In *STOC '02: Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, 514–523. Available at: http://dl.acm.org/citation.cfm?id=509982&CFID=403945105&CFTOKEN=15350398

Lindell, Y. 2003a. *Composition of Secure Multi-Party Protocols—A Comprehensive Study, Volume 2815 of Lecture Notes in Computer Science*. New York, NY: Springer-Verlag.

Lindell, Y. 2003b. General composition and universal composability in secure multiparty computation. In *Proceedings of the 44th Symposium on Foundations of Computer Science*, 394–403. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1238213

Mailler, R., and V. Lesser. 2004. Solving distributed constraint optimization problems using cooperative mediation. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 438–445. Available at: http://dl.acm.org/citation.cfm?id=1018777

Mangasarian, O. L. 2011. Privacy-preserving linear programming. *Optimization Letters* 5 (1): 165–172.

Mangasarian, O. L. 2012. Privacy-preserving horizontally partitioned linear programs. *Optimization Letters* 6 (3): 431–436.

Matsuura, K. 2003. Information security and economics in computer networks: An interdisciplinary survey and a proposal of integrated optimization of investment. In *Proceedings of the 9th International Conference of Computing in Economics and Finance*. Available at: http://depts.washington.edu/sce2003/Papers/48.pdf

Meixell, M. J., and M. Norbis. 2012. Integrating carrier selection with supplier selection decisions to improve supply chain security. *International Transactions in Operations Research* 19 (5): 711–732.

Mielikainen, T. 2004. Privacy problems with anonymized transaction databases. In *Discovery Science: 7th International Conference Proceedings, Lecture Notes in Computer Science, Volume* 3245, 219–229. New York, NY: Springer-Verlag.

Miller, A. R., and C. Tucker. 2009. Privacy protection and technology diffusion: The case of electronic medical records. *Management Science* 55 (7): 1077–1093.

Modi, P. J., W-M. Shen, M. Tambe, and M. Yokoo. 2003. An asynchronous complete method for distributed constraint optimization. In *AAMAS '03: Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems*, 161–168. Available at: http://teamcore.usc.edu/papers/2003/modi-aamas03.pdf

Ostrovsky, R., and M. Yung. 1991. How to withstand mobile virus attacks (extended abstract). In *PODC '91: Proceedings of the Tenth Annual ACM Symposium on Principles of Distributed Computing*, 51–59. Available at: http://dl.acm.org/citation.cfm?doid=112600.112605

Özener, Ö., and Ö. Ergun 2008. Allocating costs in a collaborative transportation procurement network. *Transportation Science* 42 (2): 146–165.

Papadimitriou, C. H., and K. Steiglitz. 1982. *Combinatorial Optimization: Algorithms and Complexity*. Upper Saddle River, NJ: Prentice-Hall.

Papadimitriou, C. H., and M. Yannakakis. 1991. On the value of information in distributed decision-making (extended abstract). In *PODC '91: Proceedings of the Tenth Annual ACM Symposium on Principles of Distributed Computing*, 61–64. Available at: http://dl.acm.org/citation.cfm?id=112606

Papadimitriou, C. H., and M. Yannakakis. 1993. Linear programming without the matrix. In *STOC '93: Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing*, 121–129. Available at: http://dl.acm.org/citation.cfm?id=167127

Petcu, A., and B. Faltings. 2005a. Approximations in distributed optimization. In *Proceedings of CP05—Workshop on Distributed and Speculative Constraint Processing* (*DSCP*). Available at: http://link.springer.com/chapter/10.1007%2F11564751_68

Petcu, A., and B. Faltings. 2005b. An efficient constraint optimization method for large multiagent systems. In *Proceedings of AAMAS05—LSMAS Workshop*. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.385.5116

Petcu, A., and B. Faltings. 2005c. Incentive compatible multiagent constraint optimization. In *Proceedings of WINE'05: Workshop on Internet and Network Economics*. Available at: http://citeseer.ist.psu.edu/viewdoc/summary;jsessionid=BBADCA8B49538E3E4BE6244CEDC970D8?doi=10.1.1.385.9181

Petcu, A., and B. Faltings. 2005d. Optimal solution stability in continuous time optimization. In *Proceedings of IJCAI05—Distributed Constraint Reasoning Workshop* (*DCR05*). Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.108.5650

Petcu, A., and B. Faltings. 2005e. A scalable method for multiagent constraint optimization. In *Proceedings of the 19th International Joint Conference on Artificial Intelligence* (*IJCAI-05*), 266–271. Available at: http://dl.acm.org/citation.cfm?id=1642336

Petcu, A., and B. Faltings. 2005f. Superstabilizing, fault-containing multiagent combinatorial optimization. In *Proceedings of the National Conference on Artificial Intelligence* (*AAAI-05*), 449–454. Available at: https://www.aaai.org/Papers/AAAI/2005/AAAI05-071.pdf

Petcu, A., and B. Faltings. 2006. Distributed generator maintenance scheduling. In *Proceedings of the First International ICSC Symposium on Artificial Intelligence in Energy Systems and Power* (*AIESP-06*). Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.386.870&rep=rep1&type=pdf

Ren, Z. J., M. A. Cohen, T. H. Ho, and C. Terwiesch. 2010. Information sharing in a long-term supply chain relationship: The role of customer review strategy. *Operations Research* 58 (1): 81–93.

Sakuma, J., and S. Kobayashi. 2007. A genetic algorithm for privacy preserving combinatorial optimization. In *Proceedings of 2007 Genetic and Evolutionary Computation Conference*, 1372–1379. Available at: http://dl.acm.org/citation.cfm?doid=1276958.1277214

Silaghi, M. C., and D. Mitra. 2004. Distributed constraint satisfaction and optimization with privacy enforcement. In *Proceedings of IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, 531–535. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=1343012

Silaghi, M. C., and V. Rajeshirke. 2004. The effect of policies for selecting the solution of a DisCSP on privacy loss. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 1396–1397. Available at: http://www.computer.org/csdl/proceedings/aamas/2004/2092/03/20921396.pdf

Silaghi, M. C., B. Faltings, and A. Petcu. 2006. Secure combinatorial optimization simulating DFS tree-based variable elimination. In *Proceedings of the 9th Symposium on Artificial Intelligence and Mathematics*. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.125.4234

Smith, G. E., K. J. Watson, W. H. Baker, and J. A. Pokorski II. 2007. A critical balance: Collaboration and security in the IT-enabled supply chain. *International Journal of Production Research* 45 (11): 2595–2613.

Suzuki, K., and M. Yokoo. 2003. Secure generalized Vickrey auction using homomorphic encryption. *Financial Cryptography* 2742: 239–249.

Sweeney, L. 2002. K-anonymity: A model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-Based Systems* 10 (5): 557–570.

Teague, V. 2004. Selecting correlated random actions. In *Proceedings of Financial Cryptography*, 181–195. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.11.1791

Turban, E., R. K. Rainer, and R. E. Potter. 2005. Chapter 2: Information technologies: Concepts and management. In *Introduction to Information Technology*. 3rd Edition. New York, NY: John Wiley and Sons.

Vaidya, J., and C. Clifton. 2003. Privacy-preserving k-means clustering over vertically partitioned data. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 206–215. Available at: http://dl.acm.org/citation.cfm?id=956776

Vaidya, J. 2009a. Privacy-preserving linear programming. In *Proceedings of the 24th Annual ACM Symposium on Applied Computing*, 2002–2007. Available at: http://dl.acm.org/citation.cfm?id=1529729&CFID=403945105&CFTOKEN=15350398

Vaidya, J. 2009b. A secure revised simplex algorithm for privacy-preserving linear programming. In *AINA '09: Proceedings of the 23rd IEEE International Conference on Advanced Information Networking and Applications*, 347–354. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5076220

Yao, A. C. 1986. How to generate and exchange secrets. In *Proceedings of the 27th IEEE Symposium on Foundations of Computer Science*, 162–167. Available at: http://dl.acm.org/citation.cfm?id=1382944

Yokoo, M., E. H. Durfee, T. Ishida, and K. Kuwabar. 1992. Distributed constraint satisfaction for formalizing distributed problem solving. In *In Proceedings of International Conference on Distributed Computing Systems*, 614–621. Available at: http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=235101

Yokoo, M., and K. Hirayama. 1995. Distributed breakout algorithm for solving distributed constraint satisfaction problems. In *Proceedings of the First International Conference on Multi-Agent Systems*, edited by Lesser, V. Cambridge, MA: MIT Press.

Yokoo, M., K. Suzuki, and K. Hirayama. 2002. Secure distributed constraint satisfaction: Reaching agreement without revealing private information. In *Proceedings of the Eighth International Conference on Principles and Practice of Constraint Programming* (*CP-2002*), 387–401. Available at: http://www.researchgate.net/publication/221633421_Secure_Distributed_Constraint_Satisfaction_Reaching_Agreement_without_Revealing_Private_Information

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

Zhang, W., and L. Wittenburg. 2003. Distributed breakout algorithm for distributed constraint optimization problems—DBArelax. In *Proceedings of the International Joint Conference on Autonomous Agents and Multi Agent Systems* (*AAMAS-03*), 1158–1159. Available at: http://dl.acm.org/citation.cfm?doid=860575.860844

Zhang, W., D. Zhou, and L. Liu. 2010. Procurement control against supply chain upstream risk and inefficiency. Available at: http://ssrn.com/abstract=1537826