Towards a Formal Analysis of Control Systems

Babak Sadighi Firozabadi[1] and **Leendert W.N. van der Torre**[2]

Electronic commerce became a buzzword when electronic networks gained popularity, in particular of course due to the internet. The security problem increases rapidly now that computer systems become more open. Most research on electronic commerce, and electronic data interchange in general, has focussed on technical instead of organizational problems, such as value added networks, trusted third parties, chip cards, electronic signatures, etc. Computer science has been pushing technology. However, these technical issues are not the bottleneck for acceptance of these new technologies. Success factors of introducing electronic commerce are solutions of the organizational problems, and the closely related legal issues.

Formal models of electronic commerce represent and reason about security policies [4], for example for fraud detection and prevention. Computer security can be based on cryptography, architectures or regulations. For example, to protect a file in a database from unauthorized access it can be encoded, or access can be restricted by special architectures or by regulations. Recent interest in security logics [2] shows how computer science can also contribute to the third category. In this paper we discuss a formal representation of control systems involving a number of independent agents, as they occur in e.g. business and trade procedures. The formal model is based on a division of the information into factual, epistemic and deontic knowledge, and the organisation of control systems into a hierarchy. The classification of these routines in different levels helps the user to audit procedures (for inter-organizational [3] and intra-organizational [1] procedures), analyze scenarios and design secure procedures [7]. We also give an example from international trade, and we discuss the use of the formal model.

1 Towards a formal model

We claim that a formal model of control systems should distinguish between three different types of knowledge in information systems, and that it should be based on a decomposition of the information system. The formal model can check whether a control system covers all potential fraud situations and thus can be used to design and verify such a system.

In Definition 1 a *devision* is made between what is factually the case (as observed by the agents), what agents believe to be the case (due to messages the agents receive), and what ought to be the case (as specified in a contractual agreement between agents). The agents can lie to each other, and as a result they can believe what is factually false. In particular, agents can lie about their performances, claiming that they fulfill obligations although they have not, and thus giving rise to fraud. Note that fraud is a legal term and thus can only be explained in terms of legal notions.

Epistemic logic, the logic of knowledge and belief [5], is used to formalize the consequences of actions on the beliefs of agents.

Deontic logic[3]**,** the logic of obligations, prohibitions and permissions [9], is used to formalize the legal consequences of actions of agents. In particular, the contractual commitments between the parties are formalized as contractual obligations.

Definition 1 (Information system) *An information system is a tuple $\langle A, E, O, B\rangle$ where A is a set of agents involved, E is the set of actions they can perform, O is the set of their obligations, and B is the set of their beliefs.*

In Definition 2 the *decomposition* of the system in several subsystems, called the core system and its control systems, lays bare the structure of the information system. The formal model not only explains *how* the system is built, but also – and this is crucial – *why* it is built in this way.

Definition 2 (Core and control systems) *A decomposition of an information system is a mapping of the elements of the information system to the set of integers. An element of the information system belongs to the core system if its value is zero, and it is part of the i-th control system if its value is $i > 0$.*

The two definitions are illustrated in the following section.

2 International trade example

Our example from international trade illustrates the distinction between core system and control systems. The core system is that part of the information system that would be there if the seller and buyer would be the same organisation. Hence, it is the part of the information system that remains when parties completely trust each other. Usually there is a core information system and a *hierarchy* of control systems, since each control system itself can be interpreted as an information system that has its own control system. An important property of control systems is that we can replace a set of control routines by another set without affecting the core system, which will remain the same. The example also illustrates the distinction between preventative (the metro in Paris) and detective (the French railway system) control systems. The use

[1] Dept. of Computing, Imperial College of Science, Technology and Medicine, 180 Queen's Gate, London SW7 2BZ, UK
[2] IRIT, porte 324, University Paul Sabatier, 118 Route de Narbonne, 31063 Toulouse, France, torre@irit.fr

ECAI 98. *13th European Conference on Artificial Intelligence*
Young Researcher Paper
Edited by Henri Prade
Published in 1998 by John Wiley & Sons, Ltd.

of deontic logic to model the legal consequences of trade procedures is well established, see e.g [8], and it was noticed in [6] that this use is restricted to detective control systems. Deontic logic is necessary to formalize detective control systems, if only to formalize the violations and associated sanctions [6]. However, most work in security logics and protocol analysis does not represent the legal consequences of actions and is therefore only useful for preventative control systems.

Example 1 (Trade) *Consider the information system* $\langle A, E, O, B \rangle$ *of a seller of jeans in Taiwan, who sells a shipment to a buyer in Rotterdam, where the jeans are shipped by a third party. In principle, the only actions needed are (1) giving the goods to the mover (2) the shipment of the goods by the mover, (3) handing over the goods to the buyer and (4) paying the seller (by the buyer), and (5) paying the mover (by the seller). This core system, which explains the aim of the business, can be modeled by actions (*E*) and obligations (*O*). However, actual trading systems are much larger, because big risks and low levels of trust between trading agents make them adopt control routines. In particular, they exchange a number of documents to confirm, check and coordinate each other's performances they cannot observe themselves. The documents induce* beliefs*, and the trading process is therefore based on beliefs about the other's performances. We call the set of routines (e.g. the exchange of certain documents) introduced for ensuring the performances in a core system the level-1 control system. These routines may involve additional parties such as banks which are trusted by the trading parties (and therefore called trusted third parties).*

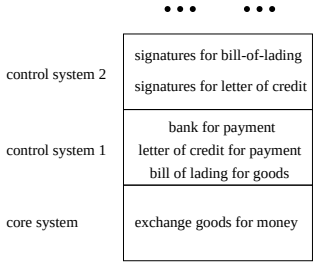

Figure 1. Control layers of a trade procedure

Now consider the decomposition of the information system in Figure 1. If the buyer receives a specific document from the mover (level-1 control), then he may believe that the seller has delivered the goods to the mover (core system) and on this belief he starts the payment procedure to the seller (core system). Otherwise, if the buyer is not completely sure about the validity of the document, he may have other control routines to validate the document, like checking the stamps or the signature on the document. The latter routines are adopted by the parties to create beliefs about the truth of the statements in the first level of control and not directly about the performances in the core system. We therefore call them second level control routines, and it is obvious that in principle we can construct an infinite number of control levels.

For practical reasons we always stop the regression of control routines at a certain level. The risks are negligible, because of confidence in the artifacts, or the agents trust each other at this level.

3 Closing remarks

In this paper we proposed a multi-layered formal model of information systems, that consists of a core system and several layers of control systems. The policies in these layers can be expressed by obligations and actions of the involved parties. The formal model can be used for designing and analysing detective and preventative control systems. Existing security logics can only formalize preventative control systems, because they do not formalize the legal consequences of actions (no obligations and therefore no violations).

The decomposition method described in this paper can be operationalized as follows. First, find the obliged actions in the system which belong to the core system (e.g. assume both parties are the same agent). This set of action has to describe the aim of the business. Second, find the control routines which are introduced to check those actions that belong to the core system. This set of control routines is the first level control system. Follow the same procedure for the other control levels until there is a reason to stop, such as trust between agents or a low level of risk. This way of structuring the system may be called sceptical, because we assume that the agents are opportunistic and may lie, unless they are completely trustworthy.

The main practical problem of developing an information system along these lines is how to interconnect the different control systems. For example, how is in Figure 1 the core system for the transportation of goods of the buyer and seller connected to the level-1 control system for the payment via the bank. The design of these interfaces can be based on the functional relationships between the levels, i.e. on the reasons why the control level was created. A control activity is introduced at a certain level in order to induce a belief about the fulfillment of some activities in the level below. So the set of activities in level 2 just controls the fulfillment (validity) of the activities in level 1, and so on.

REFERENCES

[1] R.W.H. Bons, R.M. Lee, R.W. Wagenaar, and C.D. Wrigley, 'Modeling inter-organizational trade procedures using documentary Petri nets', in *Proceedings of the Hawaii International Conference on System Sciences (HICSS'95)*, (1995).

[2] Michael Burrows, Matrin Abadi, and Roger Needham, 'A logic of authentication', Technical Report SRC Resaerch Report 39, Digital Equipment Corporation, (1989).

[3] K.-T. Chen and R.M. Lee, 'Schematic evaluation of internal accounting control systems', Technical Report RM-1992-0801, EURIDIS, Erasmus University Rotterdam, (1992).

[4] B. Christianson, ed. *Security protocols : 5th international workshop*, volume LNCS 1361, Paris, France, 1997. Springer.

[5] R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi, *Reasoning About Knowledge*, MIT press, 1995.

[6] B.S. Firozabadi and Y.H. Tan, 'Formal models of fraud', in *Proceedings of the Fourth International Workshop on Deontic Logic in Computer Science (*Δ*EON'98)*, pp. 371–385, (1998).

[7] P. Ramos and J.L. Fiadeiro, 'A deontic logic for diagnosis of organisational process design', in *Proceedings of the Fourth International Workshop on Deontic Logic in Computer Science (*Δ*EON'98)*, pp. 353–369, (1998).

[8] W. Thoen and Y.H. Tan, 'The dynamics of transferrable obligations', in *Proceedings of The Language/Action Perspective Int. Workshop on Communication Modelling*, (1996).

[9] L.W.N. van der Torre, *Reasoning About Obligations: Defeasibility in Preference-based Deontic Logic*, Ph.D. dissertation, Erasmus University Rotterdam, 1997.