The Secret Prover

Teepe, Wouter

The Secret Prover: Proving Possession of Arbitrary Files While not Giving Them Away

Wouter Teepe

Artificial Intelligence, University of Groningen,
Grote Kruisstraat 2/1, 9712 TS Groningen, The Netherlands,
email: w.g.teepe@ai.rug.nl

Abstract

The Secret Prover is a Java application which allows a user (A) to prove to another user (B), that A possesses a file. If B also possesses this file B will get convinced, and if B does not possess this file B will gain no information on (the contents of) this file.

This is the first implementation of the protocols described in the paper "New Protocols for Proving Knowledge of Arbitrary Secrets While not Giving Them Away" [2], which is also discussed in this volume [3].

1 Introduction: Proving Secrets

In application domains where sensitive information plays an important role, such as police research, intelligence, finance and the medical domain, one may want to ask whether someone knows a specific fact. Because of the sensitivity of the information concerned, it is often undesirable for the specific fact itself to be told by way of posing the question. For example posing the question "Did you know that Geertje is pregnant?" will inform the asked person about a fact. If it is the aim to ask this very question without informing the asked person about the fact, we need a dedicated protocol for asking such questions. [2] Introduces six protocols which offer a solution to this problem.

The trust needed between the participants of the protocol is minimal: essentially, only the prover must truly want to prove knowledge of a fact to others who also know.

In this demonstration, we introduce the Secret Prover, a Java application implementing these protocols.

The kind of secrets that the Secret Prover can handle is secrets in the form of a file. A file can be considered as a sequence of bits, and knowledge of this sequence can be proven using the Secret Prover. No limitation exists on what kind of files can be used.[1]

[1] Note that in this scenario, the file name is irrelevant to the protocol.

2 The ANITA project

The research contributing to the protocols and this demonstation is the Administrative and Normative Information-Transaction Agents project, ANITA for short[1]. The ANITA project is funded by NWO/ToKeN2000. Its aim is to use multi-agent systems to provide methods for both complete and legitimate information exchange of sensitive information, such as in the Dutch police domain.

The Dutch police offers us a very interesting application area for our protocols. Police investigation teams typically want to keep their files secret, but *do* want to know whether other teams are investigating on the same persons or locations. If indeed multiple teams are investigating the same person, they would better co-operate, or at least make sure they do not hinder one another.

3 System requirments

The demonstation software is a Java application, which can be used on any computer with a currect Java installed. To run the protocol, two computers running this software are needed, and the computers need to be connected through the internet. One of the computers needs to allow "incoming connections", which means its firewall should not be set too paranoid. The protocol can also be run within just one single computer, but this may make understanding the protocol somewhat less easier. The demonstration will approximately take 25 minutes. The software can be found at `http://www.ai.rug.nl/~woutr/provingsecrets/`

4 Future application of the software

The protocols can be run in standalone applications such as this demonstation, but typically the protocols will be components of larger access control systems. In our forthcoming research, these protocols will be incorporated within the prototypes which will be developed in the ANITA project.

References

[1] The ANITA project, `http://www.rint.rechten.rug.nl/onderzoek/anita/anita.html`.

[2] W. Teepe. New protocols for proving knowledge of arbitrary secrets while not giving them away. In Sieuwert van Otterloo, Peter McBurney, Wiebe van der Hoek, and Michael Wooldridge, editors, *Proceedings of the Knowlegde and Games Workshop*, Liverpool, July 2004. available at `http://www.ai.rug.nl/~woutr/provingsecrets/`.

[3] Wouter Teepe. New protocols for proving knowledge of arbitrary secrets while not giving them away. same volume.

Demonstration of a Software System for Automated Multi-Attribute Negotiation

Catholijn Jonker[a] Lourens v.d. Meij[a] Valentin Robu[b] Jan Treur[a]

[a] Vrije Universiteit Amsterdam, Department of Artificial Intelligence, De Boolelaan 1081a, 1081 HV Amsterdam

[b] Center for Mathematics and Computer Science (CWI),

PO Box 94079, 1090 GB Amsterdam, The Netherlands

jonker@cs.vu.nl lourens@cs.vu.nl robu@cwi.nl treur@cs.vu.nl

1. Overview of the proposed system

Negotiations have been identified as a key form of interaction in multi-agent systems. Multi-attribute negotiations are of particular interest, since in such cases outcomes that bring utility gains for both parties are possible. Our fundamental research [1,2] aims at bridging the gap between negotiation theory and human negotiation practice and to construct answers to open challenges (e.g., how to handle incomplete preference information). Based on this theoretical foundation, a software environment was developed to enable better understanding and testing of the model (this was originally presented as [3]).

The considered type of negotiation follows an alternating-offers protocol; a bid has the form of values assigned to a number of attributes. If the negotiation is about a car, for example, the relevant attributes considered are CD player, Extra Speakers, Airco, Tow Hedge, Price, and a bid consists of an indication of which CD player is meant, which extra speakers, airco and tow hedge, and what the price of the offer is. The proposed demonstration is based on this domain, and was originally developed in collaboration with Dutch Telecom KPN. However, the negotiation model presented in [1] and [2] is a generic one and instantiations in other domains are possible. In both cases, the DESIRE software environment (developed at Vrije Universiteit, Amsterdam) was

used to design and (automatically) implement the agents. The system supports 3 types of negotiation (all of which can be shown during the demonstration): human vs. human negotiation, human vs. software agent and software agent vs. software agent.

2. Purpose of the Demonstration

There are several aims that we wish to achieve in our demonstration. The first aim is to show how incomplete preference information can be used to increase the efficiency of the joint exploration of the utility space. The method used to achieve this is to compare the traces produced by two negotiations: a perfectly closed negotiation with no guessing and one where some profile info (in the form of one or several preference weights) and/or guessing is used (see [1] for a description).

The second important aim is to show how humans can use such a system to negotiate both against other humans or software agents. This is significant, because it gives us the possibility to analyze the behaviour of humans in complex negotiations over multiple attributes and in the presence of uncertain information. This may hold important clues for the design of future automated trading mechanisms.

Finally, the system can also be used as a training tool for introducing human negotiators into the complexities of multi-attribute utility theory (described in the classical work by Howard Raiffa and others). In this educational capacity, our software may be useful both to students, as well as professionals outside the academic field.

References

[1] Jonker, C., Robu, V. – "Automated Multi-Attribute Negotiation with Efficient Use of Incomplete Preference Information", accepted as full paper at the *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*, New York, 2004.

[2] Jonker, C.M., Treur, J., "An Agent Architecture for Multi-Attribute Negotiation". In: B. Nebel (ed.), *Proc. of the 17th International Joint Conference on AI, IJCAI'01*, 2001, pp.1195 - 1201.

[3] Jonker, C., van der Meij, L., Robu, V., Treur, J. – "Demonstration of a Software System for Automated Multi-Attribute Negotiation", accepted demonstration at the *Third International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2004)*, New York, 2004.

Visualization of a Market-Based Model for Logistics Management in Transportation

P.J. ‘t Hoen G. Redekar V. Robu J.A. La Poutré

Centre for Mathematics and Computer Science (CWI),

PO Box 94079, 1090 GB Amsterdam, The Netherlands

hoen@cwi.nl robu@cwi.nl hlp@cwi.nl

1. Overview of the transportation model

A recent development in multi-agent systems research is their application in the logistics of the transportation sector. Transportation is a challenging application area where, due to strong competition, profit margins are typically low. Furthermore, the current practice of centralized solutions is a bottleneck and does not support the flexibility required for incidence management or exploiting new and profitable opportunities. The multi-agent system paradigm can overcome these challenges and offer new opportunities for profit. In our research, this is achieved by developing robust, distributed, market mechanisms.

In recent research ([1, 2]), we have proposed a model with online, decentralized auctions, where agents bid for cargo to increase profits by exploiting new transportation opportunities that appear in the course of a day. In this context, we studied the effect of bidding strategies that are novel for such a large scale settings (e.g. allowing *decommitment* of previously won loads in favour of new, more profitable opportunities). The fundamental research on which this model is based and results from performed simulations are presented in [2]. Based on this fundamental model, a software tool was built to allow us to visualize the simulations, in the form of a Java applet. The demonstration paper for this software tool was originally presented as [1]. Due to space constraints, we cannot describe all the details of our simulation here, and the interested reader is asked to consult [1] for details.

2. The visualization applet

Our visualization is comprised of several panels. The main panel presents the structure of the world (i.e. the grid or the road network graph). Two side panels are used to display information about the current truck or depot selected and general information about the world. The visualization can run in two modes: static (in which the user can manually browse through the turns in a day) and dynamic, in which the appearance of new loads and movements of trucks are shown dynamically evolving during the course of a day.

The most relevant elements to visualize in such a simulation are the routes the trucks take during the day, since this can give an idea of the planning involved. There are 2 types of routes that may be visualized:

- *Actual routes* taken by the trucks (here the routes taken by individual trucks or by trucks owned by different companies may be highlighted).

- *Planned routes*. Viewing the evolution of the planned paths, as new loads appear at different time points, gives an insight of the complexity of planning algorithms used. The planned routes for each truck may change dynamically during the day, as plans are continuously expanded to cover the pick-up/delivery of loads newly won in the online auctions.

The objectives we pursued in building our visualization are to:

- Present all information on a single graphical interface
- Provide the user with the ability to easily navigate through the simulation, with complete information and intermediate results.
- The information given should be palatable: it can be understood without delving in the underlying complex semantics of the model.

A Power Point presentation (in the form of a story board), which contains screen shoots of this software is available at [3].

References

[1] ‘t Hoen, P.J., Redekar, G., Robu, V., La Poutré, J. A. “Simulation and Visualisation of a Market-Based Model for Logistics Management in Transportation”, *Proceedings of AAMAS 2004*, ACM Press, pp. 1218-1219

[2] Hoen, P.J. t’, La Poutré, J. A. “A Decommitment Strategy in a Competitive Multi-Agent Transportation Setting”, Agent-Mediated Electronic Commerce, 2003. Also published as LNCS 3048, Springer-Verlag, 2004.

[3] http://homepages.cwi.nl/~robu/AAMAS_presentation.ppt

Demonstration of a Multi-Agent Simulation Model of Trust in Supply Chains

Sebastiaan Meijer Tim Verwaart

Social Sciences Group, Wageningen UR

Abstract

The trust and tracing game - played by humans - is a research tool designed for study of trust in commodity supply chains. Preliminary observations from the game suggest that rational choice is to some extent dominated by player's personal preferences. Multi-agent simulation systems are being developed for comparison between behavioural models and experimental results. A prototype will be demonstrated and directions for future development will be presented. The purpose of this demonstration is to get in touch with AI researchers working in related areas.

1. Description of the Trust and Tracing Game

The trust and tracing game[1] is a research tool designed for study of actor behaviour in commodity supply chains. The focus of study is on trust in the stated quality of the commodities. The game is played by a group of persons that play the roles of producers, middle-men, retailers, or consumers. In the initial state, consumers, retailers and middle-men are supplied with (artificial) money. Producers are supplied with envelopes of different colours representing lots of different product types A, B, and C. Each lot is of either low or high quality, represented by a ticket covered in the envelope. Each combination of product type and quality grade has a different consumer satisfaction value (Table 1). Producers are informed about the quality of each lot, but other players may not open the envelopes. Buyers must either trust sellers or involve a tracing agency, at the cost of a tracing fee. If the tracing agency finds an untruthful quality statement, the seller will be punished with a fine and public disgrace. So a seller may put some money and her reputation at stake by deceiving buyers or by trusting her supplier. A bottle of wine is the reward for the winners in each player category. In the consumers category the player having gained the most satisfaction points is the winner; in other categories profit is the criterion (Fig 1).

Table 1: Consumer satisfaction value for product types A, B, and C, by quality.

Quality grade	A	B	C
Low	1	2	3
High	2	6	12

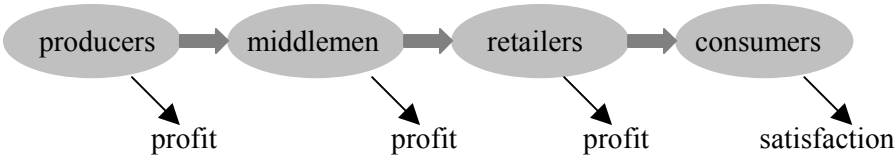

Figure 1: Commodity flow and performance criteria.

2. Multi-Agent Simulation Model

A pay-off matrix has been formulated, assuming rational choice[1]. However, preliminary game observations suggest that rational choice is influenced by personal preferences for taking risk, being honest, reputation, and interpersonal relations. People do not only maximize profit. They also follow rules of appropriateness. Multi-agent systems are being developed for study of these phenomena. We demonstrate a prototype using the Swarm simulation environment[2]. The purpose of this prototype is to assess feasibility of using multi-agent models for research of behaviour in supply chains.

The prototype comprises player agents for the roles of producers, middlemen, retailers, and consumers, and a tracing agent that may on request assess the quality of a product and impose a fine in case of deception. The trading agents are composed of processes for:

- *Needs determination*: determine if an agent intends to buy or sell; consumers always buy; middlemen and retailers buy if stocks is below threshold; producers, middlemen, and retailers sell any product in stock.
- *Trade partner selection*: agents intending to sell advertise; buyers contact sellers with trust-based preference; sellers may refuse (busy or distrusting).
- *Cheating decision*: the seller randomly decides whether to cheat or not, weighted with an "honesty" parameter and trust in the trade partner.
- *Price negotiation*: based on agent's belief about reasonable price; agents will terminate negotiations that do not satisfactorily proceed.
- *Trust-or-trace decision*: after successful transaction buyer decides whether to request a trace or not, based on its trust in the trade partner.
- *Trust maintenance*: agents maintain trust for any other agent they have done business with, based on the outcome of negotiations and traces.
- *Price belief maintenance*: adjust price belief based on negotiation results.

With this prototype the feasibility of multi-agent systems for supply chain research has been demonstrated. Currently we are developing more extensive models and exploring other agent platforms.

---

[1] Meijer, S. and G.J. Hofstede "The Trust and Tracing game" In: *Proc. 7th Int. workshop on experiential learning.* IFIP WG 5.7 SIG conference, May 2003, Aalborg, Denmark.
[2] http://wiki.swarm.org/wiki/Main_Page