Trust and Commitment in Dynamic Logic

Jan Broersen[1], Mehdi Dastani[2], Zhisheng Huang[1], and Leendert van der Torre[1]

[1] Faculty of Mathematics and Computer Science, Vrije Universiteit Amsterdam
{broersen,huang,torre}@cs.vu.nl
[2] Institute of Information and Computing Sciences, Utrecht University
mehdi@cs.uu.nl

Abstract. Trust and commitment have been identified as crucial concepts in electronic commerce applications. In this paper we are interested in the relation between these social concepts. We introduce a dynamic logic in which violations of stronger commitments result in a higher loss of trustworthiness than violations of weaker ones. We illustrate how the logic can be used to analyze some aspects of a well known example of trust within reason.

1 Introduction

In advanced applications of multi agent systems agents interact more frequently, deliberate more extensively, and in general act more autonomously. State of the art computer programs are capable of searching the web for the cheapest books, advising users on movies, negotiating bandwidth, participating in auctions, etc. Moreover, experiments with the contract net protocol have revealed that more flexible protocols based on levelled commitment lead to better global results, because agents can engage in several interactions simultaneously [13]. Researchers envision a continuation of this trend of increasing complexity of agent interactions and discuss washing machines negotiating the purchase of micro units of electricity with electricity companies [5]. One promising approach to build such complex agents introduces methods and concepts from the social sciences, such as organization, negotiation, commitment and trust [2, 4, 11].

A high level of trustworthiness is normally beneficial for the long term profits of agents, and a low level has a negative effect on them. However, whereas the short term profits are usually easy to calculate, these long term profits are much more difficult to quantify. This leads to a problem for an agent that has to balance its short term profits with its long term ones. The question is, how to balance the short term profit of violating a commitment with its cost in the long run due to the decrease in trustworthiness?

In order to formalize some of these concepts and reasoning mechanisms, we introduce a dynamic logic in which the violation of stronger commitments results in higher loss of trustworthiness than the violation of weaker ones. This logic describes an agent that proposes commitments, accepts proposals to engage in commitments, and violates commitments by performing actions other than the ones committed to.

This article is organized as follows. In Section 2 we motivate our work with an example of reasoning about trust. In Section 3 we introduce our dynamic logic. In Section 4 we show how to apply the logic to the motivating example. Finally, in Section 5 we discuss some formal properties of trust and commitment.

2 Trust and commitment in strategic decisions

The pennies pinching example is a problem discussed in philosophy that is also relevant for advanced agent-based computer applications. It is related to trust, but it has been discussed in the context of game theory, where it is known as a non-zero sum game. Hollis [8, 9] discusses the example and the related problem of backward induction as follows.

> A and B play a game where ten pennies are put on the table and each in turn takes one penny or two. If one is taken, then the turn passes. As soon as two are taken the game stops and any remaining pennies vanish. What will happen, if both players are rational? Offhand one might suppose that they emerge with five pennies each or with a six-four split – when the player with the odd-numbered turns take two at the end. But game theory seems to say not. Its apparent answer is that the opening player will take two pennies, thus killing the golden goose at the start and leaving both worse off. The immediate trouble is caused by what has become known as backward induction. The resulting pennies gained by each player are given by the bracketed numbers, with A's put first in each case. Looking ahead, B realizes that they will not reach (5,5), because A would settle for (6,4). A realizes that B would therefore settle for (4,5), which makes it rational for A to stop at (5,3). In that case, B would settle for (3,4); so A would therefore settle for (4,2), leading B to prefer (2,3); and so on. A thus takes two pennies at his first move and reason has obstructed the benefit of mankind.

Game-theory and its backward induction reasoning do not offer the intuitive solutions to the problem, because agents are assumed to be rational in the sense of economics and consequently game-theoretic solutions do not consider an implicit mutual understanding of a cooperation strategy [1]. Cooperation results in an increased personal benefit by seducing the other party in cooperation. The open question is how such 'super-rational' behavior can be explained.

Hollis considers in his book 'Trust within reason' [9] several possible explanations why an agent should take one penny instead of two. For example, taking one penny in the first move 'signals' to the other agent that the agent wants to cooperate (and it signals that the agent is not rational in the economic sense). Two concepts that play a major role in his book are trust and commitment (together with norm and obligation). One possible explanation is that taking one penny induces a commitment that the agent will take one penny again in his next move. If the other agent believes this commitment, then it has become rational for him to take one penny too. Another explanation is that taking one penny leads to a commitment of the other agent to take one penny too, maybe as a result of a social law. Moreover, other explanations are not only based on commitments, but also on the trust in the other party.

In this paper we do not want to sum up and classify all the analyses of the pennies pinching example discussed in the literature. We want to introduce a language in which some aspects of these analyses can be represented. In Section 4 we discuss these aspects as well as scenarios of pennies pinching with communication.

3 A dynamic logic of trust and commitment

Our logic formalizes a variety of examples such as the pennies pinching example as well as examples in electronic commerce. First it formalizes the discussed notions of trust and commitment. Second, it formalizes complex actions that enable the specification of protocols and communication acts. For example, the protocol of the pennies pinching game states that the only possible actions are to take one or two pennies at a time. Trust and commitments can be created by communication. Our logic therefore consists of a dynamic logic for actions and modal operators for trust and commitment.

The dynamic logic is an extension of standard propositional dynamic logic [6, 7] that contains operators $\cup$ for choice, $*$ for iteration and ; for sequence. The formula $\langle\alpha_i\rangle\varphi$ expresses that agent i is able to perform action α and by doing so it possibly reaches a state where φ holds. Our extension incorporates a concurrency operator $\cap$ and an action negation operator $-$. Concurrency is needed to synchronize processes or agents, and negation is needed to formalize obligations (for details see below and [3]).

In this dynamic logic we introduce a modality for commitment $C_{i,j}(\alpha \geq \beta)$, whose intended meaning is 'agent i, towards agent j, is more committed to perform α than to perform β', and we introduce a modality $T_{i,j}(\alpha \geq \beta)$, whose intended meaning is 'agent i trusts agent j more after the performance of α than after the performance of β'.

Definition 1. *Given a set G of agent identifiers, a set $\mathcal{A}$ of action symbols (that may be indexed by individual agents or sets of agents and that may include actions for speech acts), and a set $\mathcal{P}$ of proposition symbols. The well-formed formula $\varphi, \psi, \ldots$ are defined through the following BNF with $i, j \in G$, $a \in \mathcal{A}$ and $p \in \mathcal{P}$.*

$$\varphi, \psi, \ldots ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid \langle\alpha\rangle\varphi \mid C_{i,j}(\alpha \geq \beta) \mid T_{i,j}(\alpha \geq \beta)$$
$$\alpha, \beta, \ldots ::= a \mid any \mid -\alpha \mid \alpha \cup \beta \mid \alpha \cap \beta \mid \alpha;\beta \mid \alpha^*$$

For formulas φ we use the usual abbreviations $\vee, \rightarrow, \top$ and $\bot$.

The semantics is defined using modal action structures, supplemented with orderings $\succeq^C_{i,j}$ and $\succeq^T_{i,j}$ that interpret respectively $C_{i,j}(\alpha \geq \beta)$ and $T_{i,j}(\alpha \geq \beta)$. $\succeq^C_{i,j}$ orders levels of commitment of agent i with respect to agent j, and $\succeq^T_{i,j}$ is a reflexive and transitive ordering over S that orders levels of trust of agent i in agent j.

Definition 2. *Let $\mathcal{A}$ be a set of action symbols, G a set of of agent identifiers, and $\mathcal{P}$ a set of proposition symbols. A structure is a tuple $\mathcal{S} = (S, R, \pi, \succeq^C, \succeq^T)$, where S is a nonempty set of possible states, R defines for each action $a \in \mathcal{A}$ and agent $i \in G$ an accessibility relation over S, π is a valuation function $\pi : \mathcal{P} \rightarrow 2^S$ that interprets propositions $p \in \mathcal{P}$, and $\succeq^C$ and $\succeq^T$ each return for every pair of agents $i, j \in G$ a reflexive and transitive ordering over S.*

The semantics for the comparative commitment operator $C_{i,j}(\alpha \geq \beta)$ is, that an agent is more committed to choose α than to choose β if and only if the best possible outcome can be reached by α and the worst possible outcome can be reached by β. The semantics of the operator $T_{i,j}(\alpha \geq \beta)$ has a similar definition.

Definition 3. *The meaning of well-formed formulas in a state s of a structure $\mathcal{S}$ is given by:*

$$
\begin{array}{lll}
R_{\alpha\cap\beta} & = & R_\alpha \cap R_\beta \\
R_{\alpha\cup\beta} & = & R_\alpha \cup R_\beta \\
R_{-\alpha} & = & R_{any} \setminus R_\alpha \\
R_{\alpha;\beta} & = & R_\alpha \circ R_\beta = \{(s,s'') \mid (s,s') \in R_\alpha \textit{ and } (s',s'') \in R_\beta\} \\
R_{\alpha^*} & = & (R_\alpha)^* = Id \cup R_\alpha \cup R_\alpha \circ R_\alpha \cup \ldots \textit{ with } Id = \{(s,s) \mid s \in S\} \\
R_{any} & = & (R_a \cup R_b \cup R_c \ldots)^* \textit{ with } \{a,b,c,\ldots\} = \mathcal{A} \\
\mathcal{S}, s \models P & \textit{iff} & s \in \pi(P) \\
\mathcal{S}, s \models \neg\varphi & \textit{iff} & \textit{not } \mathcal{S}, s \models \varphi \\
\mathcal{S}, s \models \varphi \wedge \psi & \textit{iff} & \mathcal{S}, s \models \varphi \textit{ and } \mathcal{S}, s \models \psi \\
\mathcal{S}, s \models \langle\alpha\rangle\varphi & \textit{iff} & \exists s' \textit{ such that } (s,s') \in R_\alpha \textit{ and } \mathcal{S}, s' \models \varphi \\
\mathcal{S}, s \models C_{i,j}(\alpha \geq \beta) & \textit{iff} & \textit{for all } (s,s') \in R_{\alpha\cup\beta}, \textit{ there is} \\
 & & (1) \textit{ a } (s,s'') \in R_\beta \textit{ such that } s' \succeq^C_{ij} s'' \\
 & & (2) \textit{ a } (s,s''') \in R_\alpha \textit{ such that } s''' \succeq^C_{ij} s' \\
\mathcal{S}, s \models T_{i,j}(\alpha \geq \beta) & \textit{iff} & \textit{for all } (s,s') \in R_{\alpha\cup\beta}, \textit{ there is} \\
 & & (1) \textit{ a } (s,s'') \in R_\beta \textit{ such that } s' \succeq^T_{i,j} s'' \\
 & & (2) \textit{ a } (s,s''') \in R_\alpha \textit{ such that } s''' \succeq^T_{i,j} s'
\end{array}
$$

Validity of a formula on a structure and general validity are defined as usual.

For some applications the additional constraint may be added that commitments are restricted to the agent's own actions. However, for some mechanisms such as delegation it may be useful to express that an agent is committed to the actions of other agents. For example, a boss in an organization may be committed to actions of his employees.

In this logic, several other operators are available as syntactic definitions. First we define:

$$
\begin{array}{ll}
[\alpha]\varphi & =_{def} \neg\langle\alpha\rangle\neg\varphi \\
C_{i,j}(\alpha > \beta) & =_{def} C_{i,j}(\alpha \geq \beta) \wedge \neg C_{i,j}(\beta \geq \alpha) \\
C_{i,j}(\alpha = \beta) & =_{def} C_{i,j}(\alpha \geq \beta) \wedge C_{i,j}(\beta \geq \alpha)
\end{array}
$$

Traditional deontic notions can be defined in terms of the operator $C_{i,j}(\alpha \geq \beta)$. This expresses that an agent that has made commitments has put himself in a normative position with obligations, permissions and prohibitions. First we define intention $I_i(\alpha \geq \beta)$ as self-commitment (as in agent oriented programming [14]). Second we define obligation $O_{i,j}(\alpha)$ as the commitment to perform α rather than its complement $-\alpha$. The strict version of the commitment operator for obligation is justified by the observation that an obligation for α cannot be complied to in any way by performing an action that possibly brings us to a state not reachable by α. Prohibition is defined in terms of the obligation operator as the obligation to do $-\alpha$. Permission is defined as the negation of prohibition.

$$
\begin{array}{ll}
I_i(\alpha \geq \beta) =_{def} C_{i,i}(\alpha \geq \beta) & O_{i,j}(\alpha) =_{def} C_{i,j}(\alpha > -\alpha) \\
F_{i,j}(\alpha) =_{def} O_{i,j}(-\alpha) & P_{i,j}(\alpha) =_{def} \neg F_{i,j}(\alpha)
\end{array}
$$

A further discussion of these deontic notions and a comparison with alternative definitions is beyond the scope of this paper.

4 The pennies pinching example in dynamic logic

In this section, we illustrate the dynamic logic by formalizing aspects of the pennies pinching example. In all examples we accept the following relation between trust and commitment, which denotes that violations of stronger commitments result in a higher loss of trustworthiness than violations of weaker ones.

$$C_{i,j}(\alpha > \beta) \rightarrow T_{j,i}(\alpha > \beta)$$

We first consider the example without communication. The set of agents is $G = \{1, 2\}$ and the set of atomic actions $A = \{take_i(1), take_i(2) \mid i \in G\}$, where $take_i(n)$ denotes that the agent i takes n pennies. The following formula denotes that taking one penny induces a commitment to take one penny later on.

$$[take_1(1); take_2(1)]C_{1,2}(take_1(1) > take_1(2))$$

The formula expresses that taking one penny is interpreted as a signal that the agent 1 will take one penny again on his next turn. When this formula holds, it is rational for agent 2 to take one penny.

The following formula denotes that taking one penny induces a commitment for the other agent to take one penny on the next move.

$$[take_1(1)]C_{2,1}(take_2(1) > take_2(2))$$

The formula denotes the implications of a social law, which states that you have to return favours. It is like giving a present to someone's birthday, thereby giving the person the obligation to return a present for your birthday.

More complex examples involve besides the commitment operator also the trust operator. For example, the following formula denotes that taking one penny increases the trust.

$$T_{i,j}((\alpha; take_j(1)) > \alpha).$$

The following formulas illustrate how commitment and trust may interact. The first formula expresses that each agent intends to increase the trust (=long term benefit). The second formula expresses that any commitment to itself is also a commitment to the other agent (a very strong cooperation rule).

$$T_{i,j}(\beta > \alpha) \rightarrow I_j(\beta > \alpha).$$

$$C_{j,j}(\beta > \alpha) \leftrightarrow C_{j,i}(\beta > \alpha).$$

From these two rules, together with the definitions and the general rule, we can deduce:

$$C_{i,j}(take_i(1) > take_i(2)) \leftrightarrow T_{j,i}(take_i(1) > take_i(2))$$

In this scenario, each agent is assumed to act to increase its long term benefit, i.e. act to increase the trust of other agents. Note that the commitment of i to j to take one penny increases the trust of j in i and vice versa. Therefore, each agent would not want to take two pennies since this will decrease its long term benefit.

We now consider the extension of the set of primitive actions with the communication actions or speech acts $\text{propose}_{i,j}(\alpha \text{ for } \beta)$ and $\text{accept}_{i,j}(\alpha \text{ for } \beta)$, that denote a proposal of agent i to agent j to perform α in return for β, and the acceptance of the proposal from agent j by agent i. For example, an agent may propose to the other agent to take one, if the other agent will take one afterwards too:

$$\text{propose}_{1,2}(take_1(1) \text{ for } take_2(1))$$

Moreover, the agent may propose that the other agent will take one, and that he in return will take one instead of two.

$$\text{propose}_{1,2}(take_2(1) \text{ for } take_1(1))$$

The following formula expresses that a propose followed by an accept action creates a commitment for both agents. To make the formula fit the page, we abbreviate propose by p, accept by a, and $take$ by t.

$$[\text{p}_{1,2}(t_1(1) \text{ for } t_2(1)); \text{a}_{2,1}(t_1(1) \text{ for } t_2(1))](C_{1,2}(t_1(1) > t_1(2)) \wedge [t_1(1)]C_{2,1}(t_2(1) > t_2(2)))$$

Finally, properties of the protocol can be specified in the logic. Due to space limitations we are brief. The first formula says that first agent i and then agent j in turn take one or two pennies, and that no other actions are involved. The second formula further constrains the allowed actions by stipulating that the decision to take one penny each time cannot be repeated more than five times. The third formula gives the additional constraint that at any stage, after taking two pennies no more actions can be performed, which means that the game has stopped.

$$[-(((take_i(1) \cup take_i(2)); (take_j(1) \cup take_j(2)))^*)]\bot$$

$$[(take_i(1); take_j(1))^5; (take_i(1) \cup take_i(2))]\bot$$

$$[(take_i(2) \cup take_j(2)); (take_i(1) \cup take_j(1) \cup take_i(2) \cup take_j(2))]\bot$$

Other properties may require further extensions of the logic. The following formulas say that taking one penny increases the number of pennies an agent possess with one, and that taking two pennies increases the number of pennies an agent possesses with two. To formalize these formulas we either have to introduce a first order language, in which we can quantify over the variable k in the formulas, or we may read the formulas as representing the finite set of formulas that we get by instantiating k with the finite set of values relevant for the example.

$$((Possess_i = k) \rightarrow [take_i(1)](Possess_i = k + 1))$$

$$((Possess_i = k) \rightarrow [take_i(2)](Possess_i = k + 2))$$

Further possible extensions of the logic are a decision model, for example based on the fact that agents have as a goal to maximize the value of $Possess_i$.

5 Some properties of the operators

In this section we mention some properties of the logic to give the reader a feeling of it. A full account of the logic is beyond the scope of this paper.

To distinguish between both cases we will use the symbols $\models$ and $\not\models$. With the transitivity and reflexivity of the orderings $\succeq^C_{i,j}$ and $\succeq^T_{i,j}$ correspond the following properties:

$$\models C_{i,j}(\alpha \geq \beta) \wedge C_{i,j}(\beta \geq \gamma) \rightarrow C_{i,j}(\alpha \geq \gamma) \qquad \models C_{i,j}(\alpha \geq \alpha)$$
$$\models T_{i,j}(\alpha \geq \beta) \wedge T_{i,j}(\beta \geq \gamma) \rightarrow T_{i,j}(\alpha \geq \gamma) \qquad \models T_{i,j}(\alpha \geq \alpha)$$

We now show that we avoid the counter-intuitive properties of the normal preference logics [12, 15], like disjunction expansion: if getting an apple is better than getting an orange, then getting an apple or losing million dollars is better than getting an orange. The construction with best and worst choices guarantees to the following requirements:

$$\not\models C_{i,j}((\alpha \cup \beta) \geq \gamma) \rightarrow C_{i,j}(\alpha \geq \gamma)$$
$$\not\models C_{i,j}(\alpha \geq \gamma) \rightarrow C_{i,j}((\alpha \cup \beta) \geq \gamma)$$
$$\not\models C_{i,j}((\alpha \cap \beta) \geq \gamma) \rightarrow C_{i,j}(\alpha \geq \gamma)$$
$$\not\models C_{i,j}(\alpha \geq \gamma) \rightarrow C_{i,j}((\alpha \cap \beta) \geq \gamma)$$

Within the set of actions an agent can be obliged to perform, it may distinguish certain levels concerning the relative commitment to perform actions. The following properties show how the logic behaves with respect to these situations:

$$\models O_{i,j}(\alpha \cup \beta) \wedge C_{i,j}(\alpha \geq \beta) \rightarrow O_{i,j}(\alpha) \quad \models F_{i,j}(\alpha \cup \beta) \wedge C_{i,j}(\alpha \geq \beta) \rightarrow F_{i,j}(\alpha)$$
$$\not\models O_{i,j}(\alpha \cup \beta) \wedge C_{i,j}(\alpha \geq \beta) \rightarrow O_{i,j}(\beta) \quad \not\models F_{i,j}(\alpha \cup \beta) \wedge C_{i,j}(\alpha \geq \beta) \rightarrow F_{i,j}(\beta)$$

The following properties hold for sequence of actions:

$$\models O_{i,j}(\alpha;\beta) \rightarrow \langle\alpha\rangle O_{i,j}(\beta) \quad \models F_{i,j}(\alpha;\beta) \rightarrow \langle\alpha\rangle F_{i,j}(\beta)$$

6 Concluding remarks

An autonomous agent can decide to violate its commitments and obligations. A rational autonomous agent has to balance short term effects like paying penalties with long term effects like loosing trustworthiness and reputation. Resource bounded agents cannot quantify this balance and therefore base their decisions on qualitative decision models. This paper shows how trust and commitment can be related to each other in such qualitative models. We illustrate how the logic can formalize various aspects of the pennies pinching example. We think that reasoning about trust and commitment is highly relevant for advanced agent applications for the following reason. Agents may imagine flexible and realistic negotiation protocols which, beside buying or selling, allow reservations with (or without) a deadline in such a way that retracting a reservation commitment implies less penalty than retracting a buy or a sell commitment. In general, more flexible protocols allow agents to achieve different levels of agreements at

different stages of negotiation and therefore allow agents to make what is called levelled commitments [13].

An agent's reputation is based on the degree of trust other agents have in his behavior, in particular the degree in which he fulfills his commitments. Whether agents trust other agents, and whether they use this trust in making decisions, depends on the application at hand. The extension of trustworthiness to a full agent profile is left for further research. Another interesting issue for further research is the resolution of conflicts between desires and obligations. Often an agent prefers a state which is forbidden; how to act? In our system, this becomes a trade-off between loss in utility versus loss of trustworthiness, i.e. between short term and long term effects. To resolve this kind of conflicts additional machinery has to be introduced in the logic, such as qualitative preferences between these two items, or quantitative measures. One proposal can be found in [10].

References

1. R. Auman. Rationality and bounded rationality. *Games and Economic behavior*, 21:2–14, 1986.
2. C. Basu, H. Hirsh, and W. Cohen. Recommendation as classification: Using social and content-based information in recommendation. In *Proceedings of the AAAI-98*, pages 714–720, 1998.
3. J. Broersen. Relativized action negation for dynamic logics. In *Advances in Modal Logic*, 2002.
4. A. Chavez, P. Maes, and Kasbah. An agent market-place for buying and selling goods. In *Proceedings of the PAAM'96*, pages 75–90. The Paractical Application Company Ltd, 1996.
5. M. Dastani, Z. Huang, and L. van der Torre. Dynamic desires. In S. Parsons, P. Gmytrasiewicz, and M. Wooldridge, editors, *Game Theory and Decision Theory in Agent-Based Computing*. Kluwer, to appear.
6. M.J. Fischer and R.E. Ladner. Propositional dynamic logic of regular programs. *Journal of Computer and System Sciences*, 18(2):194–211, September 1979.
7. D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, 2000.
8. M. Hollis. Penny pinching and backward induction. *Journal of Philosophy*, 88:473–488, 1991.
9. M. Hollis. *Trust within Reason*. Cambridge University Press, 1998.
10. N.R. Jennings and J.R. Campos. Towards a social level characterisation of socially responsible agents. In *IEEE proceedings on software engineering*, pages 144:11–25, 1997.
11. C. Jonker and J. Treur. Formal analysis of models for the dynamics of trust based on experiences. In *Proceedings of MAAMAW'99. LNAI 1647*, 1999.
12. N. Rescher. The logic of preference. In *Topics in Philosophical Logic*. D. Reidel Publishing Company, Dordrecht, Holland, 1967.
13. T. Sandholm and V. Lesser. Issues in automated negotiation and electronic commerce. In *Proceedings of the ICMAS'95*, 1995.
14. Y. Shoham. Agent oriented programming. *Artifi cial Intelligence*, 60:51–92, 1993.
15. G.H. von Wright. *The Logic of Preference*. Edinburgh University Press, 1963.