Synchronization Protocols for Reliable Communication in Fully Distributed Agent Systems

(Short Paper)

Hywel Dunn-Davies, Jim Cunningham
Department of Computing
Imperial College London
180 Queen's Gate
London
SW7 2AZ
{hdd2,rjc}@doc.ic.ac.uk

Shamima Paurobally
School of Informatics
University of Westminster
115 New Cavendish Street
London
W1W 6UW
s.paurobally@westminster.ac.uk

ABSTRACT

In order to prevent misunderstandings within groups of interacting agents, it is necessary to ensure that the agents' beliefs regarding the overall state of the interaction are consistent with each other at all times. In [7], Paurobally et al. proposed that these beliefs could be synchronized by adding a specialized protocol layer that incorporates protocols specifically designed to synchronize the agents' beliefs. Here we define the problem that such protocols would need to solve in the worst case, and prove it to be insoluble. We then consider the possibility of synchronizing the beliefs of groups of agents if it is assumed that the communication layer notifies the sender of a message whenever that message is not successfully delivered. Paurobally et al. proved that this assumption allows agents' beliefs to be synchronized in bilateral interactions. However, we prove that this assumption is insufficient to achieve belief synchronization in groups of three or more agents. Finally, we discuss the possibility of achieving adequate synchronization using probabilistic protocols.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed AI

General Terms

Reliability, Languages, Verification

Keywords

Agent, Protocol, Broadcasting, Synchronization

1. INTRODUCTION

Agent interaction protocols, or conversation policies, set out the kinds of message that are appropriate at any given point in an interaction, and are commonly used to expedite goal directed behavior in multi-agent systems [5]. In [7] and [8], Paurobally et al. emphasized the importance of consistency in the agents' beliefs regarding both the nature of the interaction protocol in use, and the protocol state, at any given time. They demonstrated that inconsistent beliefs may lead to misunderstandings between agents, compromising effective communication. This may have serious consequences. For example, two agents may be co-operating to sell an item to a customer, with one agent in charge of dispatching goods, and the other in charge of debiting the customer's account. If one of the agents believes that the transaction has been authorized and the other does not, the goods may be dispatched without the customer's account being debited, or vice versa. Any mechanism that could ensure that such misunderstandings could not occur would be extremely useful to researchers attempting to implement reliable multi-agent systems.

If it is assumed that communication channels are secure and reliable, in that all messages are delivered intact in a known period of time, and it is also assumed that agents have the capability to accurately keep track of the interaction state given sufficient information, synchronizing agents' beliefs regarding the state of the interaction is not difficult. However, co-ordinating agents' beliefs becomes considerably more challenging if agents may communicate over unreliable channels, as is the case in many real life applications (see, for example, [6, 9, 11]). Paurobally et al. [7, 8] have proposed that coordination of agents' beliefs could be ensured by means of synchronization protocols, situated in a layer between interaction protocols and network protocols, as shown in Figure 1.

Interaction Protocols
Speech-act like messages

Synchronisation Protocols
Protocols for message exchange and belief revision

Communication Protocols
Session, TCP/IP layers

Figure 1: Layers of protocols (taken from [7])

In the current paper we examine the extent to which the provision of such synchronization protocols could enable reliable communication in groups of agents communicating via

Cite as: Synchronization Protocols for Reliable Communication in Fully Distributed Agent Systems (Short Paper), H.R. Dunn-Davies, R.J. Cunningham and S. Paurobally, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 1539-1542.
Copyright © 2008, International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org). All rights reserved.

imperfect channels. The following section sets out the problem that an ideal synchronization protocol would solve, and includes an informal proof that this problem is insoluble. We then consider the effect of weakening this problem by assuming that the sender of a message is informed by the communication medium if the message is not successfully transmitted. Paurobally et al. [7, 8] have provided two protocols that solve this weakened synchronization problem for bilateral interactions, and have suggested that similar protocols may allow protocol synchronization to be achieved in multilateral interactions that involve more than two participants. Here we prove that this weakened synchronization protocol is also insoluble for groups of three or more agents. Finally, we discuss how agents' beliefs could be synchronized given these results, and consider the possibility of achieving adequate synchronization using probabilistic protocols.

2. THE STRONG BELIEF SYNCHRONIZATION PROBLEM

2.1 Problem Description

Consider a group of autonomous agents that can communicate with each other solely via message exchange. It is assumed that a communication channel exists between each agent and every other agent. It is also assumed that messages are never corrupted, duplicated, created or reordered by the communication channels, and that messages are always delivered within a period of time that is common knowledge in the group of agents. However, the channels are taken to be unreliable in that they can lose messages, subject to a fairness assumption which states that if infinitely many messages are sent, infinitely many are received, meaning that it is always the case that a message will eventually get through[1]. All of the agents in the group are assumed to be following the same set of interaction and synchronization protocols, and to start the interaction in the same protocol state. The protocols are taken to have been agreed on in advance, and to be specified in a complete and consistent manner, so that the precise nature of the protocols being followed, and the common start state, are also common knowledge within the group. The agents are also assumed to have access to a common clock (or to synchronized internal clocks), and are assumed to take the same amount of time to perform the internal act of belief update. Furthermore, it is assumed that the agents themselves are reliable, in that they never 'crash' or stop responding for any other reason.

The belief synchronization problem is that of designing a protocol to ensure that, despite the possibility of message loss, agents always update their internal representation of the protocol state in a synchronous manner as a result of messages communicated by other agents. An acceptable solution to this problem would meet both safety and liveness conditions, as well as a simple validity condition:

Safety condition: at any given point in the interaction any agent's belief regarding the protocol state is the same as that of all other agents in the group.

Liveness condition: it is possible that the interaction will progress to a valid state other than the commonly known initial state in a finite length of time.

Validity condition: if no agent ever receives a message, no agent ever updates its state[2].

2.2 Proof of Insolubility

Here we give a short inductive proof that the strong synchronization problem is insoluble. Induction is done on the number of rounds of messages sent, where a complete round comprises one message sent to each agent other than the sender, and a partial round comprises one message sent to one or more (but not all) of these agents. This is done to simulate broadcast communication, where messages may be sent to multiple agents in one round[3].

Inductive hypothesis:

Regardless of the number of rounds of messages transmitted, no agent can either safely update its belief regarding the protocol state, or safely commit to belief update at some time in the future.

Base case:

No messages have been sent. Given the properties of the communication medium, we can be sure that at this stage no messages have been received. We can, therefore, conclude from the validity condition that the induction hypothesis holds.

Inductive case:

We assume that at time t some finite number n ≥ 0 of rounds of messages has been sent, each of which specifies that the receiving agent should update its belief about the protocol state at some time x a finite distance in the future[4]. We also assume that at time t the inductive hypothesis holds. We then assume that some agent sends another round of messages at time t' between time t and time x, taking the number of rounds of messages sent to n+1. Regardless of whether the agent sends a partial round of messages or a complete round, that agent cannot commit to belief update until acknowledgements are received from every recipient, since there is a chance that one or more of the messages it sends will be lost. Because of this the agents receiving messages cannot commit to belief update either, even if all of the messages get through, as this would lead to a situation where the receiver of a message was committed to update and the sender was not. Such a situation could, given a worst-case scenario where every message sent between time t' and time x was lost, lead to one agent updating its belief alone, violating the safety condition. From this we can conclude that the inductive hypothesis holds when n+1 rounds of messages have been sent.

The strong belief synchronization problem described here can be thought of as a generalization of the co-ordinated

---

[1] This allows us to ignore the extreme case whereby the communication medium loses every message (synchronization via message transmission is clearly impossible in situations where no message transmission ever occurs).

[2] This rules out trivial algorithms that instruct the agents to update their representation of the protocol state at a given time regardless of whether any messages are exchanged.

[3] It should be noted that this proof also addresses the case where messages are sent one by one, since the sending of one message constitutes a partial round.

[4] We assume that x and t could be the same, reflecting the possibility that the message reads 'update now'.

attack problem [3, 4, 10], which has also been shown to be insoluble by numerous authors (e.g. [1, 3, 4]).

Protocol synchronization is also impossible to achieve in situations where message receipt is guaranteed but the time it takes to deliver a message is not known. This is because in order to commit to belief update the agent sending a message must be sure that the message will arrive before time x, and this cannot be assumed if the messages are subject to unknown delays.

3. THE WEAKER BELIEF SYNCHRONIZATION PROBLEM

Paurobally et al. [7, 8] have argued that since 100% packet loss can usually be detected, it is not unreasonable to assume that the communication channels inform the sender of any failure of message delivery. Given that this is the case it is possible to consider a weaker (but still plausible) variant of the strong belief synchronization problem that incorporates two additional assumptions: Firstly, should message loss occur, it is assumed that the agent that sent the message will always be notified of this within a known period of time; and secondly, it is assumed that the presence and properties of the notification mechanism are common knowledge within the group of agents.

For notifications to be helpful, it must be assumed that the notification mechanism is reliable, in that notifications are always received if a message is lost, and that they are never subjected to unknown delays. If the notification may be lost or delayed unpredictably, the sender of a message could never conclude that the message has been delivered successfully on the basis that no notification has been received, since it is then always possible that the message has not been delivered, but either the notification was lost or it has yet to arrive.

3.1 The Bilateral Case

In [7], Paurobally et al. presented two protocols to solve this weaker variant of the synchronization problem in groups of two agents, and provided an epistemic proof of the effectiveness of these protocols. Perhaps the simplest solution to the weaker belief synchronization problem for two agents would be the following:

One agent sends a message specifying that belief update should occur at some time x later than time $t+n$, where n is some period of time after which any message sent at time t will definitely have either arrived or triggered a notification of delivery failure. That agent waits until time $t + n$, and, if no notification has been received at that time, concludes that the message has been received and commits to belief update at time x. If a notification has been received the agent does not commit to belief update. The other agent commits to belief update at time x as soon as a message arrives, but does not commit if it does not receive a message.

We can see that this protocol will suffice by the following reasoning:

If we assume that a notification is received before time $t+n$, we can be sure that the sender will not commit to belief update as it has received notification that the message was not delivered, and the other agent will not commit either since it never receives the message. This means that the validity and safety conditions are both met.

If we assume that no notification is received, the agent that sends the message will commit to belief update at time x. Since in the case of message loss a notification would always be received before time $t + n$, the sender can conclude that the message was received. Furthermore, since x was set to be later than the known arrival time of the message, and the agent receiving the message always commits to belief update at time x when the message is received, we can conclude that both agents will update their beliefs simultaneously at time x. This means that the safety condition is met. The liveness condition is also met, since there is some execution of the protocol where both agents update their beliefs simultaneously. Since a message is received here, the validity condition holds vacuously.

3.2 The Multilateral Case

In [7], it was suggested that future research might lead to the design of synchronization protocols for multilateral interactions over unreliable channels, subject to the assumption that the sender of a message is always notified if message loss occurs. Unfortunately, we show that the weaker belief synchronization problem cannot be solved for groups of more than two agents, meaning that it is impossible to guarantee co-ordinated interactions in groups of three or more agents by means of synchronization protocols, even when notification is present. We do this by means of an inductive proof similar to the proof presented in section 2.2 for the strong belief synchronization problem. As before, induction is done on the number of rounds of messages sent. The inductive hypothesis and base case are exactly the same as in section 2.2. They are repeated here for convenience:

Inductive hypothesis:

Regardless of the number of rounds of messages transmitted, no agent can either safely update its belief regarding the protocol state, or safely commit to belief update at some time in the future.

Base case:

No messages have been sent. Given the properties of the communication medium, we can be sure that at this stage no messages have been received. We can, therefore, conclude from the validity condition that the induction hypothesis holds.

Inductive case:

As in section 2.2, we assume that at time t some finite number n ≥ 0 of rounds of messages has been sent, each of which contains the information that the receiving agent should update its belief about the protocol state at some time x a finite distance in the future. We also assume that at time t the inductive hypothesis holds. We then assume that some agent sends another round of messages at time t' between time t and time x, taking the number of rounds of messages sent to n+1. If this is a partial round, at least one agent will not receive a message. This means that no agent can commit to belief update at time x since not all agents will commit as a result of this round. If it is a complete round, the agents that do receive a message will have no way of knowing whether all of the other agents apart from the sender received the message. Because of this, once again, no agent will be able to commit to belief update at time x on the basis of this round, even if all of the messages are delivered successfully. Therefore, regardless of whether a complete or partial round of messages is sent, or of whether or not some messages are lost, the induction hypothesis will still hold when n+1 rounds of messages have been sent.

The above proof rests on the fact that common knowledge of whether a given message was received is only established between the sender and receiver of the message as a result of the notification. Any given agent cannot know whether other communications in which it was not directly involved were successful or not. Synchronization could be achieved in interactions involving three or more agents if every agent was reliably informed whenever a message was lost. However it seems reasonable to assume that a medium that could reliably deliver notifications of message delivery failure to every agent would be able to deliver the original messages reliably, meaning that such an assumption would be unlikely to be useful in practice.

4. DISCUSSION AND FURTHER WORK

In this paper we have analyzed the extent to which it is possible to synchronize agent communication over unreliable channels by means of synchronization protocols, as proposed in [7] and [8]. We have defined a strong synchronization problem, which agents would have to solve to successfully communicate over channels that may lose messages, and have proved it to be insoluble. We then defined a weaker variant of this problem to model situations where the communication medium informs the sender if message delivery fails. Paurobally et al. [7] proposed a number of protocols that solve this weaker variant of the synchronization problem for bilateral interactions, and suggested that similar protocols may be developed to enable synchronization in larger groups. However, in this paper we have given an informal proof that even this weaker variant of the synchronization problem to be insoluble for groups of three or more agents. We recognize the need for a more formal treatment of the problems discussed in this paper, and are currently investigating possible approaches to this.

In all of the scenarios mentioned so far have we have assumed that all of the agents are either working to a common clock, or have internal clocks that are perfectly synchronized with each other. If this assumption is dropped it is no longer possible to solve the weak belief synchronization problem even for bilateral interactions, unless the precise length of time that every message takes to be delivered is common knowledge within the group of agents. This is because a message instructing the receiving agent to update its internal representation of the protocol state at a certain time would refer to the internal clock of the agent sending the message, which would be inaccessible to the receiving agent. If the exact time taken to deliver every message is common knowledge within the group of agents, messages could be timestamped by the sender, and sender's local time could be inferred by receiver (by noting the time according to its local clock when the message was received, subtracting the period of time the message took to arrive, and then comparing that time to the timestamp on the message). We assume the existence of a common clock since the existence of synchronized clocks appears to be a more realistic assumption than predictable message transmission time. We have also assumed that the agents involved in the interaction are reliable, in that they never crash or stop responding. If we assume that it is possible that one or more agent might crash at any time, there is no way of ensuring that synchronized belief update will occur, even if the communication medium is perfectly reliable, since one or more agents may crash as other agents are updating their beliefs.

Both variants of the synchronization problem discussed here have stipulated that synchronization must be perfect, in that the probability of synchronization failure (i.e. a situation where some agents update their beliefs regarding the state of the interaction while others do not) must be zero. Some degree of protocol synchronization would be possible, even in the absence of any notification of failed message delivery, if this criterion were weakened so that it was merely necessary to reach an acceptably low probability of synchronization failure. A simple protocol by which this could be achieved would instruct each sending agent to send a large number of copies of each message to each receiving agent before updating its belief at an agreed time, since each additional message sent would reduce the probability that all of the messages were lost. More complex probabilistic protocols, designed to achieve an optimal tradeoff between safety and liveness, have been proposed for the co-ordinated attack problem in [2] and [10]. The application of these protocols to the problem of synchronizing of agent interactions remains an interesting avenue for future work.

5. REFERENCES

[1] E. A. Akkoyunlu, K. Ekanadham, and R. V. Huber. Some constraints and tradeoffs in the design of network communications. Proceedings of the fifth ACM symposium on Operating systems principles, 1975.

[2] G. Avoine, and S. Vaudenay. How to Safely Close a Discussion. Information Processing Letters, 102(4), 2007.

[3] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. Reasoning about Knowledge. MIT Press, 1995.

[4] J. Gray. Notes on Data Base Operating Systems. In Operating Systems, An Advanced Course. LNCS 60, 1978.

[5] M. Huget (Ed.): Communication in Multiagent Systems, Agent Communication Languages and Conversation Policies. LNCS 2650, 2003.

[6] R. Meshulam, A. Felner, and S. Kraus. Utility-based multi-agent system for performing repeated navigation tasks. 4th Int. Joint Conf. on Autonomous Agents and Multi-Agent Systems, (AAMAS 2005), 2005.

[7] S. Paurobally, J. Cunningham and N. R. Jennings. Ensuring consistency in the joint beliefs of interacting agents. 2nd Int. Joint Conf. on Autonomous Agents and Multi-Agent Systems, (AAMAS 2003), 2003.

[8] S. Paurobally. Rational Agents and the Processes and States of Negotiation. PhD Thesis, Imperial College, 2002.

[9] O. Shehory, K. Sycara, G. Sukthankar, and V. Mukherjee. Agent Aided Aircraft Maintenance. 3rd Int. Conf. on Autonomous Agents (Agents '99), 1999.

[10] G. Varghese, and N. Lynch. A tradeoff between safety and liveness for randomized coordinated attack, Information and Computation, 128(1), 1996.

[11] B. Yu, P. Scerri, K. Sycara , Y. Xu, and M. Lewis. Scalable and Reliable Data Delivery in Mobile Ad Hoc Sensor Networks. 5th Int. Joint Conf. on Autonomous Agents and Multi-Agent Systems, (AAMAS 2006), 2006.