A free-format dialogue protocol for multi-party inquiry

Gerard Vreeswijk and **Joris Hulstijn**
ICS, Utrecht University,
P.O.Box 80.089, 3508 TB Utrecht
The Netherlands
`gv@cs.uu.nl`, `jorish@vs.uu.nl`

Abstract

In this paper we propose a formal account of multi-party inquiry. Inquiry is the dialogue game type in which participants try to get a common understanding of some open problem concerning an external state of affairs. We discuss some important issues for multi-party dialogue in general, and extend a simple account of inquiry, such that it accounts for the multi-party case.

1 Introduction

Formal dialogue is rapidly gaining status as a new paradigm for automated forms of information exchange. In this paper we consider the dialogue game of *inquiry*. According to Walton and Krabbe's typology of dialogue game types, inquiry is "a type of dialogue which strives to establish or 'prove' propositions in order to answer a question (solve a problem) in such a way that a stable and general agreement on the matter at issue results" (Walton and Krabbe, 1995, p. 72). Inquiry differs from persuasion dialogue in that it does not start from a conflict, but from an open problem. It differs from information exchange, in that parties have a common goal to reach agreement. For information exchange, the goal is mere dissemination of information.

In this paper we focus on multi-party inquiry: inquiry performed by a group of more than two cooperative agents. Dignum and Vreeswijk (2003) describe a simple protocol for multi-party inquiry. The protocol has a fixed turn-taking mechanism. Here we extend the protocol, looking in particular at different coordination and turn-taking mechanisms. Moreover, we believe that once we have solved the simpler case of inquiry, it becomes easier to extend the results to multi-party negotiation and persuasion dialogues.

The paper is structured as follows. First we give definitions for single party inquiry. Then we look at multi-party issues in general, such as open versus closed systems, roles and coordination. Section 3 contains an inquiry protocol adapted for the multi-party case.

2 Issues in multi-party dialogue

Multi-party dialogue can be conducted in various ways. To sketch the possibilities we briefly sketch the landscape. The following issues arise when considering dialogue games for more than two participants (Dignum and Vreeswijk, 2003).

Open vs. closed systems An issue that comes up right away, is who the participants are. In a closed system, all parties are present during the whole dialogue. Entry and exit to the dialogue is controlled, and therefore we can assume that each participant satisfies a basic set of assumptions. In an open system, any agent can join later or leave before the end of the dialogue. No assumptions can be made on for instance, common ground or use of vocabulary.

Roles A following issue is the role of each of the parties in the dialogue (Hulstijn, 2003). This can be looked upon from different perspectives. First, there are roles related to addressing. In a two-party dialogue there is always a speaker and an addressee. In a multi-party dialogue we can distinguish: speaker, addressee, auditor, overhearer and eavesdropper (see e.g. (Bell, 1984)). Second, there are roles constituted by the particular dialogue game type. Such roles define the expectations, preferences and dialogue game rules associated with a participant. For example, in a typical two-party persuasive dialogue there is a proponent and an opponent. However, for dialogues of inquiry or deliberation, the distinctions already get blurred. Third, there are roles that depend on the social organization of the interaction situation. A good example is that of a chairperson. Such roles determine turn taking, termination or entry and exit to the dialogue.

For each of the perspectives on roles one can choose whether roles are fixed once or can change during the dialogue. Again, we may need specific communicative acts or rituals to signal such changes.

Channel One may distinguish between synchronous and asynchronous communication channels. This choice has repercussions for addressing too. For example, in an asynchronous channel that stores messages for a long time, such as a newsgroup, we may expect many overhearers. By contrast, a synchronous medium, such as speech, is less suitable for one-to-many communication.

Coordination On a synchronous channel, only one party can speak at a time. Therefore one needs a turn taking mechanism. We could use a round-robin protocol, which is a generalization of the strict turn-taking for two parties. Otherwise one could have a chairman explicitly assign turns. On an asynchronous channel, in principle everybody may speak at the same time.

Termination Participants engage in a dialogue for some particular purpose. This purpose differs for each dialogue game.

3 Inquiry

This section describes a simple dialogue game of inquiry. The game has two participants: *expert* and *nature*. The expert has knowledge about some particular aspect of the world, called a topic. For example, an agent may be an expert on the topic of financial information, or on real estate. The expert may do observations to extend its knowledge and combine bits of knowledge to reason with it. Inquiry can be seen as an information exchange with nature. When the expert carries out an experiment, this corresponds to a query; the observation provides the response. Although we believe the world itself is consistent, observations may be conflicting.

The knowledge of an agent is modeled by a knowledge base KB. We say that the agent knows φ whenever KB $\vdash_L \varphi$ for some suitable base logic L, with consequence $Cn()$. In this paper we will use propositional logic as the base logic, but obviously this can be extended with more expressive logics or knowledge representation formalisms, such as description logic.

Definition 1 (Dialogue state) *A dialogue state is a tuple $DS = \langle KB, Q, H, S\rangle$, where KB is the knowledge base, Q is a prioritized queue with queries that the agent is interested in, H is a sequence of moves representing the dialogue history, and S is a set with queries that have been made, but remain unresolved.*

We represent the fact that some expert e makes a query to nature whether φ holds by an expression query(e, φ). As a response, nature either allows the observation observation(e, φ) or observation$(e, \neg\varphi)$. In case the query must remain unresolved, there is no observation.

The dialogue state of an agent changes if it poses a query or if it does an observation. Thus, the meaning of a move, such as a query or observation, is the change it makes to the dialogue states that are kept by each agent. Accordingly, we define two transition functions that map dialogue states to new dialogue states.

Definition 2 (Inquiry) *If $DS_e = \langle KB, Q, H, S\rangle$ is a dialogue state, we define a query and observation as in Fig. 3.*

Thus, if e decides to pose a query, what effectively happens is this. First, e pops the next query from the repository of queries where it is interested in, namely the queue Q. According to the priority mechanism in Q, the formula φ may be considered as the most urgent query for e. To register that e has posed φ, the expert e appends "query(e, φ)" to its personal dialogue history H.

$$\text{query}(e,\varphi)(DS_e) = \langle\ \text{KB},$$
$$\text{pop}(\text{query}(e,\varphi), Q),$$
$$H + \text{query}(e,\varphi)), S \cup \{\varphi\}\ \rangle$$

$$\text{observation}(e,\varphi)(DS_e) =$$
if $\text{KB} \cup \{\varphi\} \nvdash \bot$ *then*
$\langle\ Cn(\text{KB} \cup \{\varphi\}), Q,$
$H +\ \text{observation}(e,\varphi), S \setminus \{\varphi\}\ \rangle$
else
$\langle\ \text{KB}, Q, H +\ \text{observation}(e,\varphi), S\ \rangle$
end

Figure 1: Definition of query and observation.

Finally, to remember that φ is asked but not answered (yet), φ is added to the list of posed but unanswered queries S.

We now explain the second equation. If e observes φ, there are two possibilities: the observation φ is consistent with what e knows, or it is not. If φ is consistent with KB, then φ is "epistemically adopted,", i.e., φ is added to KB. To register that φ has been observed, e appends "observation(e,φ)" to its personal dialogue history H. Finally, φ is crossed off as an unresolved query. If φ is inconsistent with KB, i.e. if φ contradicts e's knowledge, then e should ideally revise its knowledge, for example according to the AGM paradigm on belief revision (Gärdenfors, 1988). Since belief revision is another issue that falls beyond the scope of this paper, we have chosen for the semantically crude (but we believe technically adequate) solution that in case of inconsistencies the observation is ignored and the query remains unresolved.

The initial state of an agent DS^0 is $\langle KB^0, Q^0, \langle\rangle, \emptyset\rangle$, such that which $KB^0 \nvdash_L Q$, and $\langle\rangle$ is the empty sequence. The desired end state of the dialogue DS' is $\langle KB', \langle\rangle, H', \emptyset\rangle$. This means that all queries in Q^0 have been resolved.

The two dialogue actions can be uttered at any dialogue state, in any order. This results in a protocol that is extremely simple and rather liberal compared to other mechanized dialogue games. It is even an issue if we might speak of a true protocol here. The idea is that, in a running dialogue, experts pose questions at will, and "fish" for answers when and where appropriate, for example if time allows. In particular, there is no turn taking and observations may "come in" at any time. If the latter is put in agent-oriented terminology we may say that nature is not obliged to respond.

4 Multi-party inquiry

In addition to the protocol proposed in section 3, we make the following assumptions.

(i) A fixed number of equivalent participants engage in an inquiry dialogue.

(ii) There are no specific roles for the agents, although they may be in productive mode, or consumptive mode (see below).

(iii) Agents communicate through a central medium, called the forum, the function of which may be compared to the function of an internet newsgroup. Messages are public. They are not addressed to specific agents.

(iv) Agents act (listen, reason, and speak) in turn, for a fixed number of rounds.

(v) There is no criterion for termination., compare point(iv).

The following properties are not typical multi-party issues, but also determine the course of a dialogue.

(a) Participants are cooperative. This means that they are sincere, i.e. do not lie about their beliefs. All agents acknowledge and process all applicable messages. Moreover we assume that all agents have ample time to reason, and all agents have the opportunity to post all the messages desired.

(b) Agents have reasoning capacities. In particular, they do not ask what they already know or can infer. Before asking, an agent tries to infer the desired item itself.

(c) The facilitation of information is dialectic: claims are justified with other claims or denied with reasons that support a contradiction. Agents accept claims if and only if they can be resolved to information that they believe to be true, either on the basis of observation, or derived from acquired information.

(d) Regression to previous messages is always possible. Agents are allowed to question or justify prior claims. Thus, an immediate response is not required.

(e) For simplicities sake the agents have a shared ontology. One consequence of this assumption is that propositions (internal representations of claims) conveyed through messages do not have to be renamed.

5 Architecture

In this section we will describe the architecture that lies at the basis of our implementation.

We suppose that agents belong to a discussion group $G = \langle A, F \rangle$, where A is a (finite) set of agents A, and F is a newsgroup-like data structure called a *forum*. A forum is a sequence of entries $F = \langle m_1, ..., m_n \rangle$, where m_n is the last entry published. An entry m_i is a pair consisting of a query and a sequence of observations that count as responses:

$$m_i = \langle \text{ query}(i, \varphi), \langle \text{observation}(r_1, \psi_1), \ldots, \\ \text{observation}(r_1, \psi_1) \rangle \ \rangle$$

Thus, entries behave like topics or threads as found in newsgroups or mailing lists.

The internal structure of a participant $k \in A$ contains (at least) the dialogue state of section 3, along with a bookmark i to remember the first unread entry, and bookmarks (i, j) per entry for the first unread response to that query: $DS_k = \langle \text{KB}_k, Q_k, H_k, S_k, i, \langle (1, j), \ldots, (n, j) \rangle \rangle$.

Agents run concurrently, and have access to a forum that is shared by all agents. The forum is a passive asynchronous channel, but is responsible for the administration of messages. The idea is that instead of making observations, the participants will now first query the forum.

Each agent may be in *consumptive mode* or in *productive mode*. In the consumptive mode an agent takes actions that are supposed to deal with the accumulation of new knowledge: reading from the forum or posting new queries to it. This can be expressed by read$(k, F, \text{obs}(e, \varphi))$ and post$(k, F, \text{query}(k, \varphi))$. In the productive mode an agent disseminates knowledge. In our case, answering questions of other agents: post$(k, F, \text{query}(k, \varphi), \text{obs}(k, \psi))$.

6 Experiments

The multi-party inquiry set up discussed above is rather simplified. With respect to all the multi-party issues discussed in section 2, it always takes a simple solution. In order to allow experiments with different set-ups, to test if the resulting dialogues that are generated make any sense, we have made an implementation of the dialogue architecture in Ruby. This allows us to run dialogue generation experiments. The purpose of the implementation is to test different dialogue game parameter settings.

We opted for an implementation in Ruby because it is a pure object-oriented scripting language with an intuitive syntax, suited for prototyping. Fig. 2 shows the data structures of three agents, viz. Mr. Priestley, a prominent English chemist and a strong proponent of the phlogiston theory of combustion, Mr. Lavoisier, the founding father of the oxygen theory of combustion, and you, the reader, who supposedly wants to know more of combustion theory and queries the experts Lavoisier and Priestley. Other queries can be posed as well, mostly with the same effects. The resulting dialogue is displayed in Table. 1.

During our experiments, we noticed that all discussion terminate. This can be understood as follows. As a finite number of queries may be linked to a finite number of answers. Moreover, agents keep an account of which queries they have answered, so that eventually termination is ensured. We also observed that agents will reach a conclusion on accessible facts within a reasonable amount of turns. This can be explained by the fact that explanations (i.e., explanatory rules) cannot be chained infinitely. A a consequence each justification has a stopping place, so that agents will either accept facts or abandon search on explained statements with a bounded number of dialogue moves.

7 Related Research

Although it is arguably one of the simpler types of dialogue, inquiry has received less attention than negotiation or persuasion. An exception is the work by McBurney and Parsons (2001) on scientific investigation. Our purpose is very similar to

1.	Reader:	Gentlemen, how is it to be explained that in combustion, heat and light are given off?
2.	Lavoisier:	Dear Reader. You asked why in combustion, heat and light are given off. Well, that is because pure air contains oxygen, pure air contains matter of fire and heat, and in combustion, oxygen from the air combines with the burning body.
3.	Priestley:	Dear Reader. You asked why in combustion, heat and light are given off. Well, that is because combustible bodies contain phlogiston, combustible bodies contain matter of heat, and in combustion, phlogiston is given off.
4.	Reader:	Let me think. Do I know that pure air contains oxygen? ..
5.	Reader:	.. no.
6.	Reader:	Sorry, I it is not clear to me why pure air contains oxygen. Can you explain this a bit more?
7.	Reader:	Let me think. Do I know that combustible bodies contain phlogiston? ..
8.	Reader:	.. no.
9.	Reader:	Sorry, I it is not clear to me why combustible bodies contain phlogiston. Can you explain this a bit more?
10.	Reader:	Gentlemen, how is it to be explained that pure air contains oxygen?
11.	Reader:	Gentlemen, how is it to be explained that combustible bodies contain phlogiston?
12.	Lavoisier:	Well, one of the hypotheses of my theory is that pure air contains oxygen.
13.	Priestley:	Well, one of the hypotheses of my theory is that combustible bodies contain phlogiston.
14.	Reader:	Ok, thanks Lavoisier.
15.	Reader:	Ok, thanks Priestley.

Table 1: Resulting dialogue.

theirs. They describe a risk agora, as they call it, that allows the storage of multiple arguments for and against some claim. However, they do not treat multi-party issues explicitly. The agora is an asynchronous channel; no coordination rules are given.

There is also a correspondence to the Newscast protocol (Voulgaris et al., 2003). This is a kind of ‘gossiping’ protocol that can be used to disseminate information in distributed systems. A difference is that the newscast protocol can only pass on information. No mechanism exists to specify queries. The Newscast protocol is complementary to our work, in the sense that it may provide an implementation of the forum in distributed systems.

8 Conclusion

In this paper we proposed a simple protocol of inquiry among experts and between experts and nature. We discussed several issues that are relevant to multi-party dialogue in general: open versus closed systems, roles, type of channel, coordination and termination. We then make some choices regarding these issues, for the game of multi-party inquiry. Under some assumptions, we can show such games will terminate. However, many assumptions remain unwarranted. Therefore we hope this first attempt will stimulate more research into multi-party issues.

References

A. Bell. 1984. Language style as audience design. *Language in Society*, 13:145–204.

F. Dignum and G. Vreeswijk. 2003. Towards a testbed for multi-party dialogues. In *International workshop on Agent Communication*.

D. Traum et al. 2002. Embodied agents for multi-party dialogue in immersive virtual worlds. In *Proc. of the First Int. Joint Conf. on Autonomous Agents and Multi-Agent Systems*, pages 766–773. ACM Press, July.

P. Gärdenfors. 1988. *Knowledge in Flux: Modelling the Dynamics of Epistemic states*. Bradford Books.

J. Hulstijn. 2003. Roles in dialogue. In I. Kruijff-Korbayova, editor, *Proceedings of the 7th Workshop on Formal Semantics and Pragmatics of Dialogue (Diabrck)*, pages 43–50. Universitt des Saarlandes.

P. McBurney and S. Parsons. 2001. Representing epistemic uncertainty by means of dialectical argumentation. *Annals of Mathematics and Artificial Intelligence*, 32(1):125–169.

S. Voulgaris, M. Jelasity, and M. van Steen. 2003. A robust and scalable peer-to-peer gossiping protocol. In *Proceedings 2nd International Workshop on Agents and Peer-to-Peer Computing (AP2PC 2003)*.

D. N. Walton and E.C.W. Krabbe. 1995. *Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning*. State University of New York Press, Albany, NY.

```
#!/sw/bin/ruby

translation_table = {

    ’E1’ => ’in combustion, heat and light are given off’,
    ’E2’ => ’inflammability is transmittable from one body to another’,
    ’E3’ => ’combustion only occurs in the presence of pure air’,
    ’E4’ => ’increase in weight of a [.. snip ..] weight of air absorbed’,
    ’E5’ => ’metals undergo calcination’,
    ’E6’ => ’in calcination, bodies increase weight’,
    ’E7’ => ’in calcination, volume of air diminishes’,
    ’E8’ => ’in reduction, effervescence appears’,
   ’OH1’ => ’pure air contains oxygen’,
   ’OH2’ => ’pure air contains matter of fire and heat’,
   ’OH3’ => ’in combustion, oxygen from the air combines with the burning body’,
   ’OH4’ => ’oxygen has weight’,
   ’OH5’ => ’in calcination, metals add oxygen to become calxes’,
   ’OH6’ => ’in reduction, oxygen is given off’,
   ’PH1’ => ’combustible bodies contain phlogiston’,
   ’PH2’ => ’combustible bodies contain matter of heat’,
   ’PH3’ => ’in combustion, phlogiston is given off’,
   ’PH4’ => ’phlogiston can pass from one body to another’,
   ’PH5’ => ’metals contain phlogiston’,
   ’PH6’ => ’in calcination, phlogiston is given off’

}
```

```
Agent.new(
   ’name’      => ’Priestley’,
   ’questions’ => {},
   ’knowledge’ => {
      ’E1’ => [ %w(PH1 PH2 PH3) ],
      ’E2’ => [ %w(PH1 PH3 PH4) ],
      ’E5’ => [ %w(PH5 PH6) ],
      ’PH1’ => TRUE,  ’PH2’ => TRUE,
      ’PH3’ => TRUE,  ’PH4’ => TRUE,
      ’PH5’ => TRUE,  ’PH6’ => TRUE
   }
)
```

```
Agent.new(
   ’name’      => ’Lavoisier’,
   ’questions’ => {},
   ’knowledge’ => {
      ’E1’ => [ %w(OH1 OH2 OH3) ],
      ’E3’ => [ %w(OH1 OH3) ],
      ’E4’ => [ %w(OH1 OH3 OH4) ],
      ’E5’ => [ %w(OH1 OH5) ],
      ’E6’ => [ %w(OH1 OH4 OH5) ],
      ’E7’ => [ %w(OH1 OH5) ],
      ’E8’ => [ %w(OH1 OH6) ],
      ’OH1’ => TRUE,  ’OH2’ => TRUE,
      ’OH3’ => TRUE,  ’OH4’ => TRUE,
      ’OH5’ => TRUE,  ’OH6’ => TRUE
   }
)
```

```
Agent.new(
   ’name’      => ’Reader’,
   ’questions’ => {
      ’E1’ => TRUE
   },
   ’knowledge’ => {}
)
```

Figure 2: Translation table, followed by data structures for three agents.