A Reactive-Deliberative Model of Dialogue Agency

David R. Traum
TECFA, FPSE
Université de Genève
9, Route De Drize, BAT D
CH-1227 Carouge, Switzerland
David.Traum@tecfa.unige.ch

1 Overview

For autonomous agents that operate in a realm of heterogeneous agents (including human agents), an agent communication language should have many of the features of natural language dialogue. The language should allow flexible turn-taking and initiative, reactions to the utterances of others, and flexible mechanisms of establishing mutual understanding and repairing problems when they occur. This kind of ability also puts some constraints on the type of agency theory. Such an agent must have (at least implicitly) functionally equivalent mechanisms to many of the attitudes of human agents.

In this paper I describe a theory of social agency including attitudes and operations that have been used to model plan execution, and repair and provide a formal foundation for certain kinds of speech acts. A simplification of this agency theory has been implemented as the dialogue manager of the TRAINS-93 NL conversation system. While the TRAINS system as a whole converses with a human user in Natural Language, the dialogue manager's I/O is in terms of a generalized form of speech acts (the *conversation acts* introduced in [Traum and Hinkelman, 1992]. Thus, any formal language which could be mapped to this set of speech acts would work as well.

2 Aspects of Individual Agency

The approach to agency is informally sketched below, with the agent's action types in italic, and mental attitudes in bold.[1] The formal details are presented in [Traum and Allen, 1994b, Traum, 1994].

[1]The approach presented here is based on the BDI model of Bratman *et al.* [1988].

We start with **beliefs** and **desires**. From the desires (and with reference to beliefs and other goals and intentions), the agent will *deliberate* and choose a set of **goals**: conditions that the agent will try to achieve. *Planning* or *means-ends reasoning* will lead to the formation or selection of a **plan recipe** designed to meet the goals. We view recipes as consisting of a set of actions coupled with a set of constraints relating various properties of these actions (e.g. constraints on relative timing or objects and locations of actions, preconditions or effects represented as events or states which must hold over times related to the times of actions). The agent then can *adopt* or *commit to* plans. **Plans** are treated as individuals, the structure of which can be modified through time. At any given moment in time, a particular plan will correspond to a particular plan recipe. This plan adoption will lead to **commitment** to achieve (or maintain) the constraints of the plan and **intention** to perform the actions in the plan.

When an agent has committed to a plan, she can *try* to perform an action in that plan. She can also *observe* the situation she finds herself in, perhaps revising her beliefs and desires. She can also *replan*, revising the plans she is executing to correspond to different plan recipes, and thus changing her commitments and intentions. We distinguish two types of plan revision: *plan elaboration*, in which additional actions (e.g., decompositions of non-primitive actions) or constraints are added to a plan, and *plan repair*, in which some of the actions or constraints are removed. We can also distinguish the action of *plan repair* from *changing plans*. In both cases, the intentions and commitments of an agent change, but in the former case, the agent is changing the contents of a plan which she continues to execute, whereas in the latter, the agent drops all intentions and adopts a new set.

From the point at which a plan is adopted to the point at which the intentions and commitments are dropped, we say that the agent is *executing* the plan. When an agent believes that an action in the plan has been performed by some execution (physical performance by the agent), we say that plan action is **instantiated**. An agent, **Agt**, is said to be executing plan **P** iff:

1. Agt intends to perform all uninstantiated actions in plan P
2. Agt committed to the occurrence of all constraints in P
3. Agt believes instantiated actions have been realized by executions
4. Agt committed to realization of instantiated actions by executions

We can distinguish at least three distinct notions of the culmination of plan:

Action Completion – all actions in the plan have been performed.

Successful Completion – all actions in the plan have been performed and all of the constraints have been met as well.

Goal Satisfaction – the goals which motivated adoption of the plan have been achieved.

Note that Successful Completion and Goal Satisfaction are somewhat independent. It may be the case that the goals are met before the plan is completed (e.g., imagine a plan to open an elevator door by pushing a button: the goal is to get the elevator door open, but suppose it opens by itself when someone else gets out). Depending on the plan adoption procedures, a plan might also be successfully completed without having satisfied the goals. A plan is an entity independent from the goals for which it was adopted, and the same plan could be used to try to achieve many different sets of goals. If a goal is not a constraint of the plan, it may be the case that the plan is successfully executed but the goals are not met. In a distributed control scenario, a plan executor may not have access to actual goals and may simply adopt particular plans under orders of a superior.

A plan executor will monitor the success of the plans it is executing, and engage in a repair if the plan is not successful. But the plan executor will also need to monitor goals: plan repair might also be warranted if the situation changes. This could include eliminating unnecessary actions if the goals are met though events external to the plan itself.

3 Social Agency

In order to model multi-agent interactions, such as natural language conversation, a model of agency must be extended in several directions which impact rationality.

Pragmatic theories of several aspects of utterance meaning (e.g., implicature and indirect speech acts) crucially take into account an assumption of speaker rationality in order to infer what was really meant in a given situation (beyond the compositional semantic meaning of the utterance itself).

A social agent model must contain *social* attitudes in addition to the individual-level attitudes (e.g., belief, intention, etc). Mutual belief is one commonly assumed social attitude. In [Traum, 1994], I develop a computational account of the process (called *grounding* [Clark and Schaefer, 1989]), by which natural language conversants construct mutual belief in conversation. Contrary to most previous AI accounts of speech acts, which have assumed that mutual belief results merely from the performance of an utterance within a shared situation, the study of conversation shows that there is quite a rich system for coordinating understanding. This system includes the production of feedback including both acknowledgments and repairs. This grounding process affects agents in several ways. First, much coordination of action and accomplishment of individual goals will necessitate grounding in order to be fulfilled. The grounding itself, will also provide the motivation for adoption of additional beliefs and intentions given a model of the other agent as rational.

For social situations including natural language conversation, we claim that *Obligations* are necessary, e.g., for modeling the effects of some speech acts, such as requests [Traum and Allen, 1994a]. Obligations represent what an agent *should* do, according to some set of norms. The notion of obligation has been studied for many centuries, and its formal aspects are examined using Deontic Logic (e.g., [von Wright, 1951, McCarty, 1986]. Generally, obligation is defined in terms of a modal operator often called *permissible*. An action is *obligatory* if it is not permissible not to do it. An action is *forbidden* if it is not permissible. An informal semantics of the operator can be given by positing a set of rules of behavior R. An action is obligatory if its occurrence logically follows from R, and forbidden if its non-occurrence logically follows from R. An action that might occur or not-occur according to R is neither obligatory nor forbidden.

Just because an action is obligatory with respect to a set of rules R does not mean that the agent will perform the action. So we do not adopt the model suggested by [Shoham and Tennenholtz, 1992] in which agents' behavior cannot violate the defined social laws. If an obligation is not satisfied, then this means that one of the rules must have been broken. We assume that agents generally plan their actions to violate as few rules as possible, and so obligated actions will usually occur. But when they directly conflict with the agent's personal goals, the agent may choose to violate them. Obligations are quite different from and can not be reduced to intentions and goals. In particular, an agent may be obliged to do an action that is contrary to his goals (for example, consider a child who has to apologize for hitting her younger brother).

In [Traum and Allen, 1994a] we argued that obligations play an important role in accounting for many of the interactions in dialog. Obligations do not replace the plan-based model (e.g., [Cohen and Perrault, 1979, Allen and Perrault, 1980]) but augment it. The resulting model more readily accounts for discourse behavior in adversarial situations and other situations where it is implausible that the agents adopt each others goals. The obligations encode learned social norms, and guide each agent's behavior without the need for intention recognition or the use of shared plans at the discourse level. While such complex intention recognition may be required in some complex interactions, it is not needed to handle the typical interactions of everyday discourse. Furthermore, there is no requirement for mutually-agreed upon rules that create obligations. Clearly, the more two agents agree on the rules, the smoother the interaction becomes, and some rules are clearly virtually universal. But each agent has its own set of individual rules, and we do not need to appeal to shared knowledge to account for local discourse behavior.

Another common social attitude is that of *Joint intention* [Cohen and Levesque, 1991] or *shared plan* [Grosz and Sidner, 1990]. In [Traum, 1994], I also develop a similar notion, that of agents executing a multi-agent plan. This is an extension of the notion of executing a plan described above. A group of agents $\{A_i\}$ is executing a multi-agent plan **MP** iff:

1. Each A_i is executing a single-agent plan $\mathbf{MP}_i$, which has as its actions A_i's actions from $\mathbf{MP}$, and actions by other agents are constraints (which A_i will be committed to the occurrence of).

2. Each A_i is obliged to the other agents to perform her own actions as part of the multi-agent plan.

This formulation is also used to formally model the grounding process mentioned above.

4 An Example Implemented Social Agent

The dialogue manager of the TRAINS-93 system [Allen *et al.*, 1995] is implemented as a reactive deliberative agent. A rich model of the mental and conversational state (including private, nested, and mutual beliefs; private, proposed and shared plans; conversational goals, intentions, and obligations) is maintained and updated as the conversation progresses. This includes adopting new beliefs and other attitudes as a result of the language interpretation process, as well as through the system's own reasoning and deliberation.

In designing an agent to control the behavior of the dialogue manager, we choose a *reactive* approach in which the system will not deliberate and add new intentions until after it has performed the actions which are already intended. As shown above, though, new obligations will need to be addressed before performing intended actions. The agent's deliberative behavior could thus be characterized in an abstract sense as:

loop
  perceive world and update beliefs
  if system has obligations
  then address obligations
  else if system has performable intentions
  then perform actions
  else deliberate on goals

When deciding what to do next, the agent first considers obligations and decides how to update the intentional structure (add new goals or intentions) based on these obligations. Obligations might also lead directly to immediate action. If there are no obligations, then the agent will consider its intentions and perform any actions which it can to satisfy these intentions. If there are no performable intentions, then the system will deliberate on its overall goals and perhaps adopt some new intentions (which can then be performed on the next iteration).

For the discourse actor, special consideration must be given to the extra constraints that participation in a conversation imposes. This includes some weak general obligations (such as acknowledging utterances by others and not interrupting) as well as some extra goals coming from the domain setting to maintain a shared view of the world and the domain plans which are to be executed. We prioritize the sources for the deliberations of the actor as follows:

1. Discourse Obligations (e.g., answer a question, repair a misinterpretation)
2. Weak Obligation: Don't interrupt user's turn
3. Intended Speech Acts
4. Weak Obl: Grounding (coordinate mutual beliefs)
5. Discourse Goals: Domain Plan Negotiation
6. High-level Discourse Goals

The implemented actor serializes consideration of these sources into the algorithm in Figure 1.

```
while conversation is not finished
    if system has obligations
    then address obligations
    else if system has turn
    then if system has intended conversation acts
         then call generator to produce NL utterances
         else if some material is ungrounded
         then address grounding situation
         else if some proposal is not accepted
         then consider proposals
         else if high-level goals are unsatisfied
         then address goals
         else release turn
              or attempt to end conversation
    else if no one has turn
    then take turn
    else if long pause
    then take turn
```

Figure 1: Discourse Actor Algorithm

The updating of the conversational state due to perceived conversation acts or actions of other modules of the system progresses asynchronously with the operation of the discourse actor. Whenever the discourse actor is active, it will first decide on which task to attempt, according to the priorities given in Figure 1, and then work on that task. After completing a particular task, it will then run through the loop again, searching for the next task, although by then the context may have changed due to, e.g., the observance of a new utterance from the user. The actor is always running and decides at each iteration whether to speak or not (according to turn-taking conventions); the system does not need to wait until a user utterance is observed to invoke the actor, and need not respond to user utterances in an utterance by utterance fashion.

5 Summary

This paper has briefly sketched a theory of agency suitable for an agent in a multi-agent domain. For flexibility, an agent's communication language should include most (if not all) of the properties of natural language. Both reactive and deliberative aspects are important for an agent in an uncertain world. Reactivity is crucial to deal with changing circumstances in a timely manner, but deliberation is also important to be able to do complex tasks. These two components can be integrated, as in the TRAINS system, by having some of the "reactions" be to deliberate on a particular problem. This deliberation must also be constrained to more local decisions rather than solving a whole problem, in order to allow the system to react in a timely manner.

References

[Allen and Perrault, 1980] James F. Allen and C. Raymond Perrault. Analyzing intention in utterances. *Artificial Intelligence*, 15(3):143–178, 1980.

[Allen *et al.*, 1995] James. F. Allen, L. K. Schubert, G. Ferguson, P. Heeman, C. H. Hwang, T. Kato, M. Light, N. Martin, B. Miller, M. Poesio, and D. R. Traum. The TRAINS project: a case study in building a conversational planning agent. *Journal of Experimental and Theoretical Artificial Intelligence*, 7:7–48, 1995.

[Bratman *et al.*, 1988] Michael E. Bratman, David J. Israel, and Martha E. Pollack. Plans and resource-bounded practical reasoning. Technical Report TR425R, SRI International, September 1988. Appears in Computational Intelligence, Vol. 4, No. 4, 1988.

[Clark and Schaefer, 1989] Herbert H. Clark and Edward F. Schaefer. Contributing to discourse. *Cognitive Science*, 13:259–294, 1989. Also appears as Chapter 5 in [Clark, 1992].

[Clark, 1992] Herbert H. Clark. *Arenas of Language Use*. University of Chicago Press, 1992.

[Cohen and Levesque, 1991] Phillip R. Cohen and Hector J. Levesque. Confirmations and joint action. In *Proceedings IJCAI-91*, pages 951–957, 1991.

[Cohen and Perrault, 1979] Phillip R. Cohen and C. R. Perrault. Elements of a plan-based theory of speech acts. *Cognitive Science*, 3(3):177–212, 1979.

[Grosz and Sidner, 1990] Barbara J. Grosz and Candace L. Sidner. Plans for discourse. In P. R. Cohen, J. Morgan, and M. E. Pollack, editors, *Intentions in Communication*. MIT Press, 1990.

[McCarty, 1986] L. Thorne McCarty. Permissions and obligations: An informal introduction. Technical Report LRP-TR-19, Dept. of Computer Science, Rutgers University, 1986.

[Shoham and Tennenholtz, 1992] Yoav Shoham and Moshe Tennenholtz. On the synthesis of useful social laws for artificial agent societies. In *Proceedings AAAI-92*, pages 276–281, 1992.

[Traum and Allen, 1994a] David R. Traum and James F. Allen. Discourse obligations in dialogue processing. In *Proceedings of the 32th Annual Meeting of the Association for Computational Linguistics*, pages 1–8, 1994.

[Traum and Allen, 1994b] David R. Traum and James F. Allen. Towards a formal theory of repair in plan execution and plan recognition. In *Proceedings of the 13th Workshop of the UK Planning and Scheduling Special Interest Group*, September 1994.

[Traum and Hinkelman, 1992] David R. Traum and Elizabeth A. Hinkelman. Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*, 8(3):575–599, 1992. Special Issue on Non-literal language.

[Traum, 1994] David R. Traum. *A Computational Theory of Grounding in Natural Language Conversation*. PhD thesis, Department of Computer Science, University of Rochester, 1994. Also available as TR 545, Department of Computer Science, University of Rochester.

[von Wright, 1951] G. H. von Wright. Deontic logic. *Mind*, 60:1–15, 1951.