Integrating the Communicative Plans of Multiple, Independent Agents*

Terrence Harvey† Keith Decker†

Owen Rambow†‡

† Department of Computer Science
University of Delaware
Newark, DE 19716
‡CoGenTex, Inc.
{harvey, decker, rambow}@cis.udel.edu

April 13, 1999

1 Introduction

In some agent architectures, multiple, diverse agents will have varied information that needs to be communicated to a human user. We argue that if any topics are shared by different agents' communications, then an integrated communicative plan is preferable to having multiple agents communicate separately with the user.[1] This preference results from two premises. First, domain expert agents know best how to communicate knowledge of their domain to a user; thus it makes more sense to have each agent develop its own communicative plan, and then integrate the plans, than to try to integrate raw data from multiple agents into a single communicative plan. Second, increased coherence can result from proper integration of communicative plans.

The first premise arises from the recognition that communication styles, and therefore plan construction, vary across domains (Kittredge et al., 1991). The most efficient place to put knowledge on how to communicate about a specific domain is in the agent that is a specialist in that domain (e.g. weather). This allows simultaneous design and maintenance of the domain's subject matter and communication expertise, and also facilitates having an open system, since knowledge about communicating in a domain is added with every new domain agent.

Key to the second premise is the ability to integrate communicative plans to enhance coherence. Unmanaged dependencies and interactions between the communications of independent agents can result in the overall message of the system appearing incoherent. To avoid this possibility, agent architectures need to recognize and manage dependencies and interactions between the communications of independent agents. Some success has already been demonstrated in integrating text plans to increase coherence (Harvey and Carberry, 1998).

We are developing an Agent-Based Dynamic Presentation Assembly Framework. In this framework, agents of different kinds have each developed plans to communicate specialized information (e.g. weather, travel plans) to a user. The goal of the framework is to create a single presentation that coherently combines the plans of the specialized agents.

*Harvey and Decker are supported by National Science Foundation grant NSF-9812764.

[1] Our work focuses on the importance of communicative plan integration when presenting to humans, but many of our conclusions are appropriate for inter-agent communication as well.

2 Integrating Communicative Plans

The key agent in the framework will be the high level presentation agent (PA) that performs domain-independent integration of communicative plans. While the approach of the PA is domain-independent, some kinds of integration will require inter-agent communication and the specialized knowledge of the domain agents to determine when plans are related in a manner that should affect the presentation. This requires the development of an inter-agent language designed to enable communication about presentation interdependency: a meta-communication language. This language will make it possible for the system to be open to the addition of agents from any domain.

Consider an information system that has many domain expert agents, including a Weather agent and a Travel agent. The system as a whole is presenting diverse information about a country. The Weather agent has a plan to present information which highlights a hurricane in Zone Y. The Travel agent has a plan to present information about buses all over the country, including Bus B moving into Zone Y.

Under these circumstances, a *coherent* presentation would emphasize the proximity of Bus B and Zone Y, and the fact that a hurricane was in the area. To do otherwise would be to disregard the relation of content, and a properly integrated structure must reflect the relations of content.

This example points out two major hurdles to development of a real-time Presentation Agent. First, domain-specific knowledge is necessary to capture two kinds of information: that there is an important relationship between buses and hurricanes, and that a presentation must therefore emphasize information about buses in zones with hurricanes. Second, the knowledge discovery to enable this example takes exponential time. These two issues will be discussed in Section 3.

2.1 Intention in Communicative Plans

A strength of the Presentation Assembly Framework is that the communicative plans of the domain agents include not only data and the relationships between data, but the reasons the domain agent had for communicating the data. It is the incorporation of communicative intention into the plan structure that allows the PA to perform meaningful integration or superimpose clarifying structure on sets of lower level communicative plans. This idea is taken from Rhetorical Structure Theory (RST) (Mann and Thompson, 1987); the idea was adapted to text planning by (Hovy, 1991) and (Moore and Paris, 1990). RST posits that a coherent text plan consists of segments related to one another by rhetorical relations such as MOTIVATION or BACKGROUND.

2.1.1 Intention in Text Plan Integration

RTPI (Harvey and Carberry, 1998) demonstrates that domain-independent strategies can be used to integrate multiple, independently generated text plans. Each text plan presented to *RTPI* is a tree structure in which individual nodes are related by RST-style relations. The top-level communicative goal for each text plan is expressed as an intended effect on the user's mental state (Moore, 1995), such as `(GOAL USER (DO USER ACTION27))`. The explicit representation of communicative intention present in the text plans allows *RTPI* to enhance coherence by integrating related trees.

To understand the importance of the explicit representation of intention, consider these two independently generated critiques whose text plans were inputs to *RTPI*:

* *Performing local visual exploration of all abdominal wounds is preferred over doing a peritoneal lavage for ruling out a suspicious abdominal wall injury.*

* *Please remember to check for laparotomy scars before you do a peritoneal lavage.*

It is important to note that these two critiques do not logically conflict at either the content level or the intentional level. We can demonstrate this by looking at a simplified representation of the discourse intention behind each critique:

```
(DISSUADE user lavage :reason (preferred LVE lavage))
```

and

```
(KNOW user (in-order check-scars lavage))
```

If the second critique had an intention to PERSUADE the user to perform a lavage, then the two critiques would conflict logically, since PERSUADE and DISSUADE are opposites. Since the second critique has no such intention, the conflict is not direct conflict, but only *apparent* conflict.

One factor that makes them appear to conflict is the inference, drawn by the reader, that since the second critique presents the preconditions for a lavage, it presumes that a lavage will be performed. One attempt to resolve this incoherence is to simply reword the module's template for the second critique, so that this particular inference can be prevented by making the critique a conditional:

> * *Performing local visual exploration of all abdominal wounds is preferred over doing a peritoneal lavage for ruling out a suspicious abdominal wall injury.*
>
> * *If you do a peritoneal lavage, please remember to first check for laparotomy scars.*

Now the second critique can no longer be interpreted to suggest an assumption that a lavage *will* be performed. However, simply presenting the new second text with the first critique does *not* make them appear fully coherent. What is missing is some indication from the system about how to relate the possibility of a lavage being performed to the system's recommendation to avoid a lavage.

The reason that this example appears incoherent can be found in RST, which holds that when a structure of relations meets certain criteria, the text will be coherent. According to Mann and Thompson, "Perturbing text to prevent the (implicit or explicit) expression of one of its relational propositions causes the text to become incoherent." In this example, a coherent text is not being perturbed to prevent a relation's expression, but rather insufficient (implicit or explicit) clues were provided to the reader to allow them to *recognize* the relations, which effectively prevents expression of the relation. Only explicit representation of the relations in the structure would enable a text planner to ensure that the relations were either implicitly or explicitly communicated. Thus a corollary to Mann and Thompson's statement might be "Combining related text plans without identifying and representing the relational propositions that hold between them results in incoherent text."

So in fact, the incoherence results from two conditions: 1) the two critiques *are* related to one another; 2) the relationship is not made clear, nor is it easily inferable. Because the two critiques are independently generated and realized as text, this relationship *cannot* be made explicit in the text plans or realized in the final text for the critiques. The rule in the text plan integrator that matches this situation inserts a CONCESSION relation between the critiques to reflect that while the system recommends one course of action, it *concedes* that the physician may choose a different course[2]. Consider the following text, where the relationship is finally made explicit:

> *Performing local visual exploration of all abdominal wounds is preferred over doing a peritoneal lavage for ruling out a suspicious abdominal wall injury. However, if you do a peritoneal lavage, then remember to first check for laparotomy scars.*

[2]We use the CONCESSION relation in a slightly different way than Mann and Thompson. The choice of who concedes to whom, and under what circumstances, is dependent upon the relative levels of knowledge, expertise, and responsibility of the participants(Harvey and Carberry, 1998).

2.1.2 Intention in Presentation Assembly

Similarly, the Presentation Assembly Framework's domain agent communicative plans will contain RST-style relations and possess top-level communicative goals. However, while *RTPI* operated in a single, stable domain, the PA must handle communicative plans from a broad range of domains, and must consider the possible addition of agents from new domains.

Consider integrating the communicative plans produced by a Planning Agent and an Inventory Agent. Suppose, for example, that the PA is presented with communicative plans for the following messages:

Planner-01: *Use the Phillips screwdriver. It is in the first drawer.*
Inventory-03: *The Phillips screwdriver is in the second drawer.*

In this example, two agents are presenting conflicting information. The likelihood of this event in a widely distributed multi-agent system is quite high. Despite research into automating the process of reducing data inconsistency (Adali and Subrahmanian, 1994), such conflict will always exist to some extent. Likely sources of data inconsistency include breakdowns or restrictions in communications, varying periodicity of updates, misinterpretation of soft data (e.g. satellite photos), and variations between local and non-local knowledge bases. When a PA encounters data conflict, it cannot wait for corrections that may never arrive; instead it must notify the participating agents of the inconsistency and proceed with the presentation, taking the conflict into account.

Upon recognizing the conflict between ((INFORM (LOC PHILLIPS-SCREWDRIVER DRAWER-1)) and ((INFORM (LOC PHILLIPS-SCREWDRIVER DRAWER-2)), one option would be to simply eliminate the two branches of the plan containing the conflict. While less informative than the original message, the result *"Use the Phillips screwdriver."* is arguably accurate. However, this strategy fails in the following case, where current local information about a Zone conflicts with a Planning Agent communication:

Planner-07: *Take Road 36 N through Zone A because that is the safest route.*

```
(PERSUADE USER (DO ''Take Road 36...''))
   (MOTIVATION ''Take Road 36...'' ''safest route'')
               ...
```

Local-03: *Zone A is unsafe due to enemy activity.*

```
(BELIEVE USER ''route through Zone A unsafe'')
   (EVIDENCE ''enemy activity in Zone A'')
```

In this case, removing the parts of the communicative plans with conflicting information results in the persistence of a possibly dangerous instruction from Planner-07.

Thus it is important for the Presentation Agent to recognize the *communicative intentions* behind the planner's mention of (SAFETY ZONE-A HIGH) and (SAFETY ZONE-A LOW). When this conflict is observed, the intentional structure surrounding the conflicting items in the plan will be matched against domain-independent rules. The rule that fires in this case will encode the knowledge that when the communicative plan's intent is to *motivate* Act01, and that the motivating item is in conflict, then (PERSUADE USER (DO ACT01)) should not appear in its original form. The PA does not have sufficient information or expertise to decide what information is more "correct"; instead, the conflict should be made explicit. One possible resolution would be

> *Planner-07 recommends taking the road through Zone A because it is the safest route, but Local-03 indicates that Zone A is unsafe due to enemy activities. Planner-07 has been asked for an update.*

By making the conflict explicit, the presentation agent gives the user all of the information available, while the manner of the presentation points out the need to compare both pieces of information and the sources.

While making conflict explicit does not solve the problem for the user, calling attention to the problem is an important task. When the statements of the above example are next to one another, the need for some kind of resolution may be clear to the user; but if the user is exposed to a barrage of information from thirty or forty sources, calling attention to such conflict becomes a critical function. Furthermore, explicitly noting the conflict may spare the user the effort of double-checking the conflicting communications.

Incorporating communicative intention into the communicative plans can be quite simple. If a simple domain agent always communicates its information in the same format, a plan template can be designed by hand to match the format, and that plan is usable by the PA. However, to achieve the full capabilities of the system we have in mind, the domain agents must be able to flexibly present a variety of communicative plans, and reason about the intentions within those plans.

While the examples presented here all use text plans for simplicity, there is no reason that a communicative plan cannot be realized using other methods of presentation, such as graphs or maps. The use of other modes does not alter the need for careful plan integration. For example, while a map overlay can easily show the proximity of a bus and a hurricane, choosing which maps to overlay in a dynamic presentation environment requires the full integration process.

3 Method

Our Presentation Assembly Framework uses a real-time open agent system developed using DECAF (Distributed, Environment-Centered Agent Framework). DECAF is a toolkit which allows a well-defined software engineering approach to building multi-agent systems. The toolkit provides a stable platform to design, rapidly develop, and execute intelligent agents to achieve solutions in complex software systems. DECAF provides the necessary architectural services of a large-grained intelligent agent (Decker and Sycara, 1997): communication, planning, scheduling, execution monitoring, coordination, and eventually learning and self-diagnosis.

The presentation agent (PA) will take as input the individual presentation plans provided by a set of domain expert agents, and will develop a "rough draft" of an integrated presentation plan. Domain-independent rules will recursively recognize potential improvements in the presentation plan. The rules will specify how to communicate with the domain expert agents to further identify and resolve communicative plan interactions. The results of the communications will be used to determine whether to fire the rule and how to instantiate the variables of the rule. An "anytime" algorithm (Garvey and Lesser, 1994) will be used, so that the most coherent plan developed so far can be delivered at any point in the calculations. While attempting optimization within this algorithm is clearly exponential, it is reasonable to expect that given a corpus of agent communications, future work with machine learning could be used to predict the most productive (effective) communications, thus reducing the communication burden.

Rules will be designed to resolve apparent conflict, manage conflict, and to exploit plan inter-relations to enhance coherence. They will do so by using representations of intention, importance, and domain content to recognize these kinds of interactions, communicate about the interactions, and resolve the interactions to create new, integrated presentation plans.

4 Scenario

Here is an example illustrating the operation of the Presentation Assembly Framework. Consider a situation where a multi-agent system is responsible for reporting, among other things, the locations of hundreds of enemy ships, including E-ship-04. A wide range of agent talents is required.

1. Agent presentations are submitted to the PA, including the following three: Planner-14 expects E-ship-04 at Loc-1; Satellite-03 reports E-ship-04 at Loc-2; and Weather-09 reports hurricane warnings for Loc-1.

2. PA detects conflict, and forms a rough draft:

   Satellite-03 reports E-ship-04 at Loc-2; however, Planner-14 reports E-ship-04 expected at Loc-1.

3. PA scans other agent presentations for information that may be related to the conflicting information, finds no information on E-ship-04 or Loc-2, but does find the hurricane info from the weather agent which is tagged as important (from a weather perspective). The plan is altered to create:

   Satellite-03 reports E-ship-04 at Loc-2; however, Planner-14 reports E-ship-04 expected at Loc-1. A hurricane warning has been issued for Loc-1.

4. Now the PA has extracted all possible information from the individual presentations. It examines the conflict at hand:

```
(conflict (Planner-14 (expected (at E-ship-04 date Loc-1)))
          (Satellite-03 (at E-ship-04 date Loc-2)))
```

   and based on knowledge of agent capacities, chooses to send the conflict report and the three relevant presentations (Planner-14, Satellite-03, and Weather-09) to the Planner-14 and an agent expert on ship movement.

5. In this case, the planner can't do anything useful with the conflict report. However, Ship-Movement-02 can reason about what might cause a ship to be in an unexpected location. It determines that possible causes include weather, fuel, and enemy activity. Based on the information in the supplied Weather-09 presentation, the Ship-Movement-02 agent reports to the PA and Planner-14 that a possible reason for the information conflict is an altered course due to hurricane warnings:

```
(reason
        :certainty possible
        :effects (conflict
                        (Planner-14 (expected (at E-ship-04 Loc-1)))
                        (Satellite-03 (at E-ship-04 Loc-2)))
        :cause (altered-course
                        :previous (course E-ship-04 (dest Loc-1))
                        :cause (Weather-09 Loc-1 hurricane)))
```

6. The PA now alters the plan to say:

   Satellite-03 reports E-ship-04 at Loc-2; however, Planner-14 reports E-ship-04 expected at Loc-1. Possible reason for disagreement is an altered ship course due to hurricane warnings in Loc-1.

7. Ship-Movement-02 directly queries other agents to investigate other possible reasons for a course change. If none are found, the strength of the `reason` will be upgraded to likely.

Note that with a fast network, it is very likely that the only message that would appear to the user would be the final, most informative wording. If for some reason the process were slowed, the PA can return its most current presentation at any stage (since the PA algorithm is anytime), and then update the screen if more information becomes available.

References

S. Adali and V.S. Subrahmanian. 1994. Amalgamating knowledge bases, III: Distributed mediators. *International Journal of Intelligent Cooperative Information Systems.*

Keith S. Decker and Katia Sycara. 1997. Intelligent adaptive information agents. *Intelligent Information Systems*, 9:239–260.

Alan Garvey and Victor Lesser. 1994. A survey of research in deliberative real-time artificial intelligence. *The Journal of Real-Time Systems*, 6.

Terry Harvey and Sandra Carberry. 1998. Integrating text plans for conciseness and coherence. In *Proceedings of the 17th International Conference on Computational Linguistics and the 36th Annual Meeting of the Association for Computational Linguistics*, pages 512–518, Montreal, Canada.

Eduard Hovy. 1991. Approaches to the planning of coherent text. In *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pages 153–198. Kluwer.

Richard Kittredge, Tanya Korelsky, and Owen Rambow. 1991. On the need for domain communication knowledge. *Computational Intelligence*, 7(4):305–314.

William C. Mann and Sandra A. Thompson. 1987. Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, ISI/USC, June.

Johanna Moore and Cecile Paris. 1990. Planning text for advisory dialogues. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 203–211, Vancouver, Canada.

Johanna D. Moore, 1995. *Participating in Explanatory Dialogues*, chapter 3. MIT Press.