From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

Visualizing Agent Conversations: Using Enhanced Dooley Graphs for Agent Design and Analysis

H. Van Dyke Parunak

Industrial Technology Institute
PO Box 1485
Ann Arbor, MI 48106 USA
van@iti.org

Abstract

In the MAS/DAI community, most current work on speech acts focuses on formalizing individual utterances. The next stage will exploit the theory's power to explicate relationships within *conversations*, or groups of utterances. Computer scientists naturally seek to visualize these relationship in terms of graphs, focusing either on the identities of the individual agents involved or the states through which participating agents move. This paper introduces an alternative formalism, the Dooley Graph. It reviews the kinds of relations that can exist among individual communicative actions (including both speech acts and non-speech acts), shows the strengths and weaknesses of participant and state graphs, explains the derivation of Dooley Graphs, and suggests their value for designing agents and analyzing the behavior of communities of agents.

1. Introduction

Speech act theory [Cohen & Levesque 1988, 1995; Smith & Cohen 1995] has made major contributions to understanding the relationship between an agent's internal state and the utterances it exchanges with other agents. In the MAS/DAI community, most work on speech acts to date has focused on individual utterances. The next generation of speech act research, adumbrated in studies such as [Smith & Cohen 1995; Barbuceanu & Fox 1995], will exploit the theory's ability to explicate the relationships within *conversations*, or groups of utterances, providing a general and theoretical foundation to the various specific protocols that have been proposed from time to time (e.g., [Davis & Smith 1983; Kreifelts & von Martial 1991; Müller 1996]).

One naturally visualizes relationships among utterances in terms of graphs, whose nodes represent either the identities of the individual agents or the states through which these agents pass. Each of these straightforward approaches highlights one kind of information (the identity of the agents or their state, respectively), while leaving the other information almost completely hidden. An alternative formalism, the Enhanced Dooley Graph, combines both participant and state information. Based on a tool of descriptive linguistics, the Enhanced Dooley Graph is useful retrospectively in measuring the discourse behavior of a group of existing agents, and as a prospective aid in designing new agents.

Section 2 briefly reviews speech act theory. Section 3 distinguishes several ways in which successive communicative actions can be related. Section 4 describes classical graphical representations of these relations, while Section 5 describes the Dooley Graph formalism, and Section 6 describes how Dooley Graphs are useful in designing and analyzing agent-based systems. [1]

2. Introduction to Speech Act Theory

Speech act theory originates in the observation of [Austin 1962] that utterances are not simply propositions that are true or false, but attempts on the part of the speaker that succeed or fail. The agent community uses this notion widely as the basis for understanding how agents communicate with one another, and it is one inspiration of KQML [Finin et al. 1993], the Knowledge Query and Manipulation Language. A recent review of KQML [Cohen & Levesque 1995] emphasizes the need to organize the performatives in such a system in a class-like structure, as illustrated in Figure 1, a task that Phil Cohen and his colleagues are pursuing [Smith & Cohen 1995] (the labels in the figure are mine). For example (and very informally), an individual speech act is either an SOLICIT (an attempt to achieve mutual belief with the addressee[2] that the sender wants the addressee to do an act relative to the sender's wanting it done) or an ASSERT (an attempt to achieve mutual belief with the addressee that the sender believes the asserted statement). Each of these can be refined on the basis of the kind of action that the sender is

[1] A fuller version of this paper, entitled "Introduction to Speech Acts and Dooley Graphs," is available through http://www.iti.org/~van .

[2] I use this term to emphasize that we are speaking of the intended recipient, not eavesdroppers or accidental recipients.

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

Table 1: Sample Conversation

Seq	Sndr	Addrs	Utterance	Responds to	Replies to	Resolves	Completes
1.	A	B,C,D	REQUEST: Please send me 50 widgets at your catalog price by next Thursday.				
2.	B	C	QUESTION: Are you bidding on A's RFQ?	1			
3.	C	B	INFORM: Yes, I am.	2	2	2	
4.	B	A	REFUSE	3	1	1	
5.	C	A	PROPOSE (INFORM + REQUEST): How about 40 widgets at catalog price by next Friday?	1	1		
6.	A	C	REQUEST: Please send me 40 widgets at catalog price by next Friday.	5	5	5	
7.	C	A	COMMIT: I plan to send you 40 widgets at catalog price by next Friday.	6	6	6	
8.	D	A	COMMIT: I plan to send you 50 widgets at catalog price by next Thursday.	1	1	1	
9.	A	C	ASSERT: I've found a better supplier, and am not relying on your COMMIT.	7,8	7		
10.	C	A	REFUSE: I'm abandoning my COMMIT.	9	9		7
11.	D	A	SHIP: Here are your widgets. Please pay me.	1	1		8
12.	A	D	ASSERT + REQUEST: You're five short. Please send the difference.	11	11		
13.	D	A	SHIP: Here are five more widgets. Please pay me.	12	12	12	
14.	A	D	PAY	13	13	13	

soliciting or the nature of the proposition that the sender is asserting, respectively, as suggested in the Figure. The exact structure of the class tree is a matter of ongoing research and is outside the scope of this paper.

In systems situated in the real world, physical actions as well as digital messages convey information between agents [Parunak 1987]. In the commercial domain, two such "physical messages" are shipment of product and payment. We use speech act theory to organize utterances of the first type, which we call "speech acts," and call the other kinds of utterances "non-speech acts." The relationships we define will include both kinds of acts.

3. Sequential Relations among Communicative Acts

The model of speech acts and repartee developed by [Longacre 1976] recognizes two kinds of relations among successive utterances: Reply and Resolution. I propose two others: Response and Completion. The discussion will show that these relations are of a rather different character than the issues of anaphora, scope, and quantification treated in theories such as Discourse Representation Theory (DRT) [Kamp 1981]. Table 1 analyzes a sample (highly unrealistic) conversation to illustrate these relations. In general, we assume that utterances are completely ordered (or at least orderable) in time, and the "Seq" column assigns each utterance a sequence number for reference. The next four sections amplify the last four columns.[3]

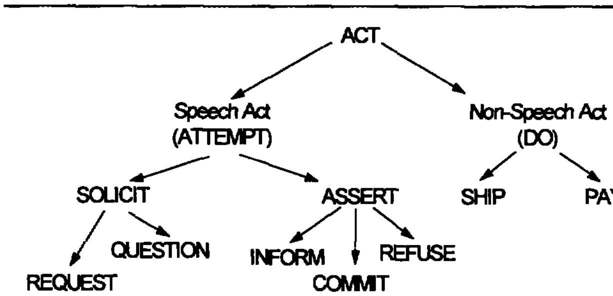

Figure 1: Base Types of Communicative Acts

Figure 2 shows which utterances in this example stand in the specified relation to which others, and illustrates how successive relations become increasingly selective and less connected.

3.1 Respond

Every utterance in a Conversation except for the first must Respond to another, otherwise there would be no conversation. Informally, my utterance Responds to another if that other utterance causes me to say what I say.

We begin with a definition that does not quite meet our need. Call it "Respond*". Utterance i Responds* to utterance j iff

1. the sender of i (S_i) previously received j and
2. the impact of j on S_i's mental state caused S_i to send i.

These conditions are necessary for a useful definition, but not sufficient. In the example, every utterance by B, C, and D follows utterance 1 and might be causally related to it, but many of these dependencies are mediated through

[3] For ease of discussion, I make the simplifying (and promising but theoretically undeveloped) assumption that both components of a compound performative (such as PROPOSE) can be analyzed as a single utterance.

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

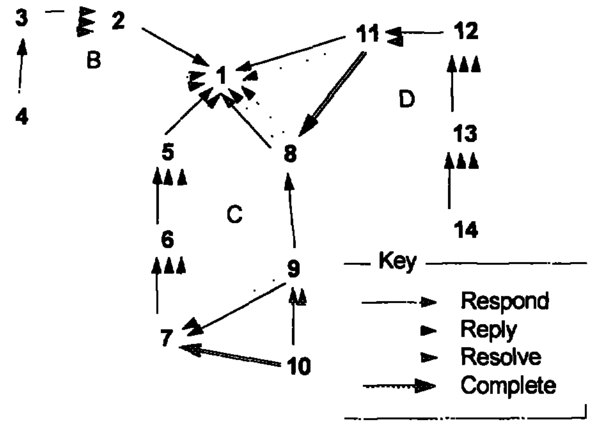

Figure 2: A Labeled Union of the Four Relations

intervening utterances, so graphing all of them would capture redundant information. Thus we need a third condition. Utterance i Responds to utterance j iff

1. the sender of i (S_i) previously received j and
2. the impact of j on S_i's mental state caused S_i to send I and
3. there exists no series of utterances $k_1 \ldots k_n$ such that k_1 Responds to j, i Responds to k_n, and $\forall m, < 1\ m \leq n$, k_m Responds to k_{m-1}.

Informally, this third condition requires that i be the first utterance that satisfies the first two conditions.

The "Responds to" column in Table 1 indicates the sequence number of the utterance to which an utterance Responds. Other utterances may intervene between an utterance and its response. The Table does not contain an example, but it is possible for a single utterance to respond to multiple preceding utterances.

"Respond" is the most complete relation, in the sense that every utterance except the first responds to some other utterance. Thus it generates a completely connected graph among the utterances.

3.2 Reply

Reply builds on the Respond* relation. Utterance i Replies to utterance j iff i is the most recent Responder* to j that is also the addressee of j. Just as utterance 1 does not Respond, it also does not Reply, but in addition utterance 2 does not Reply. Thus Reply may yield a forest rather than a tree. While every utterance in Figure 2 either Replies or is Replied to, some conversations have no Replies (for example, a chain of rumors passed from one person to another and never returning to a previous participant).

3.3 Resolve

Utterances 4, 5, and 8 all Reply to utterance 1, but some Replies are more "expected" than others. When A issues a REQUEST, A is asserting control of the conversation, and a cooperative partner will either COMMIT to the request (as in utterance 8) or REFUSE it (as in utterance 4). The fact that participant B refuses the request, while emphasizing that A does not control B in an absolute sense, does show that B agrees to conduct the conversation on the grounds that A has proposed. C's reply, in utterance 5, is qualitatively different. C neither COMMITs nor REFUSEs, but takes control of the conversation by making a REQUEST of A. In general, a participant who utters a SOLICIT is not only communicating an aspect of its own mental state, but proposing rules of engagement for the next segment of the conversation, and a Reply can either accept those rules of engagement or ignore them. Following [Longacre 1976], a Reply that accepts the proposed rules is said to Resolve the utterance that proposed them.

Longacre recognizes three initiating performatives (Question, Proposal, and Remark), each with an appropriate resolving performative (Answer, Response, and Evaluation, respectively). The formal approach outlined above suggests an analysis at a finer level of granularity,[4] but the basic intuition is sound: performatives descended from SOLICIT expect a Responding ACT (speech or otherwise) from the addressee of the initial SOLICIT, and are in some sense incomplete until that Response has been made. So we can say that an appropriate INFORM Resolves a QUESTION, and an appropriate REFUSE, COMMIT, or even the immediate performance of the requested ACT Resolves a REQUEST.

The "Resolves" column in Table 1 contains a number iff that utterance Resolves a previous one, and the number identifies the Resolved utterance. A single utterance can have several Resolutions, if it has several addressees.

One can imagine a relation a little more constraining than Resolve (call it "Assent"), which implies that the addressee not only agrees to play the game by the sender's rules, but also gives the "desired" Response. In reply to a REQUEST, a COMMIT to do the requested act is presumably more "desired" than a REFUSE. However, in the case of a QUESTION, it is more difficult to define Assent simply on the basis of the history of the conversation, and one use of this kind of analysis is in describing the behavior of communities of agents for which we do not have access to agents' internal states. Even in the case of a REQUEST, a speaker who sends out

[4] For example, Remark, Answer, and Evaluation are all instances of INFORM. Answer and Evaluation presume some initiating utterance, while Remark does not, and the system can be simplified by parsing Remark as a composition of INFORM with Question and Evaluation as the Answer to that Question.

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

multiple REQUESTs to potential bidders hardly desires all of them to COMMIT. For now, I simply note the potential for an Assent relation, but do not pursue it further.

Figure 2 shows that "Resolve" continues the trend toward selectivity and partial coverage. Now six utterances in addition to the initial one do not appear in the domain of the relation (2, 5, 9, 10, 11, 12), and of these, three do not participate in the relation in any way (9, 10, 11).

3.4 Complete

Participants alternate as one moves from a SOLICIT to the Resolving ACT: if one SOLICITs, it is up to the other to ACT. Participants can be left expecting an ACT in another way that does not depend on alternation. If one participant COMMITs, we expect a subsequent ACT by the same participant. The relation between the COMMIT and the subsequent ACT is not Resolution, since the issue is not maintaining control of the conversation. Rather, the ACT "completes" the COMMIT. Since a REQUEST can be resolved either by the requested ACT or by a COMMIT to perform the act at some future time, we can view a COMMIT as a surrogate for the final ACT, and see the ACT as a Completion of the COMMIT utterance. Other performatives that deal with future activity might also benefit from the notion of completion.

Some COMMITs do not result in their promised ACT. Whether through duplicity of the committer or by some act of God, the commitment may be abandoned. Thus a REQUEST that would have led directly to a REFUSE, had the supplier been omniscient (or honest), may instead lead first to a COMMIT, followed by the REFUSE. There is a certain symmetry here, in that a REQUEST may lead directly to the requested ACT or to a REFUSE; or the ACT or REFUSE may follow later, after an intermediate COMMIT. The symmetry is far from perfect. The COMMIT leads us to expect the ACT rather than the REFUSE, and there will likely be sanctions for the subsequent REFUSE. However, the very fact that the ACT is not yet delivered means that the REFUSE is still possible and thus not completely unexpected, and we consider that REFUSE as well as ACT can Complete a COMMIT.

The "Completes" column in Table 1 indicates which utterances Complete which ones. Resolution and Completion are mutually exclusive. A given utterance might Resolve another, or complete another, or do neither, but it cannot do both. Complete is the sparsest of the four relations.

4. Participant and State Graphs

There are at least three straightforward ways to draw pictures of conversations, based on the kinds of inter-utterance relations we have defined: utterance graphs, participant graphs, and state graphs.

A direct graph of the relations among utterances (like Figure 2) gives some sense of the structure of the conversation, but does not make the role of the participants explicit.

In a participant graph, each node is a each participant, and utterances are edges between sender and addressee. Figure 3 is the participant graph for the example. It clarifies which participants talk to which ones, something that was only incompletely represented in Figure 2, but de-emphasizes the distinct roles of the various messages in the overall conversation.

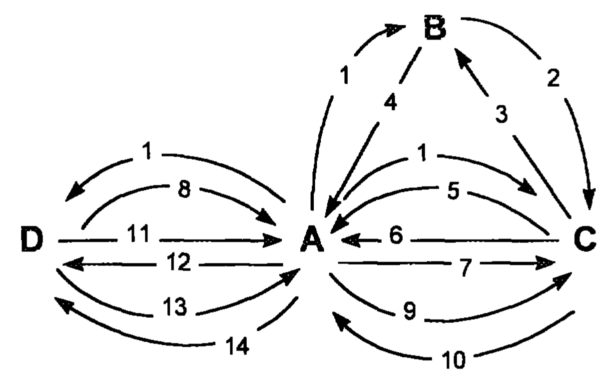

Figure 3: A Participant Graph

State machines have a long history of usage for defining network protocols, and have been adapted to speech act theory by [Barbuceanu & Fox 1995; Smith & Cohen 1995] and others. For example, the latter explicate relations among utterances using the state machine proposed by [Winograd & Flores 1988]. Figure 4 shows the structure of the Winograd/Flores model in terms of the performatives in Figure 1, between participants A and B. The arcs represent speech acts. Dark circles are absorbing (end) states.

A state model clarifies the various states through which a discourse may move, but it obscures the identity of the speaker at any point, and loses the evolution of a particular conversation in time. For example, Figure 4 indicates that a discourse can include a number of counteroffers between A and B, but does not tell us how many such counteroffers actually took place in a given interaction. A state model shows what utterances COULD be related sequentially to which other ones, while the participant and utterance graphs show which ones actually ARE so related.

Thus a conversation can be explored from different perspectives, each highlighting some information and suppressing other information). The representation proposed in this paper combines state and participant information in a way that mediates among these simpler representations, giving a richer picture that is more useful for some purposes.

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

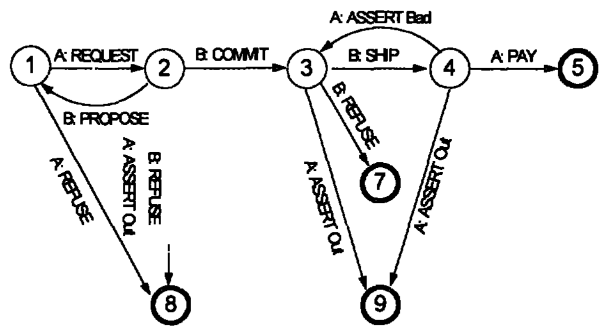

Figure 4: Winograd/Flores Model with Formalized Performatives

5. Introduction to Dooley Graphs

Each node of a Dooley Graph [Dooley 1976] of a discourse represents a "character," a participant at a distinguished stage of a discourse, where "distinguished" is defined based on the notions of Resolution and Completion. Utterances that resolve or complete one another tend to form tightly-connected components of the graph, while those that take off in new directions spawn new components. This section exhibits the basic Dooley Graph for the example, explains how it is computed, and proposes an enhancement.

5.1 An Example

Figure 5 shows the Dooley Graph for our example conversation. Several components invite discussion. I label components by sets of characters, and include by reference all edges among those characters. The bundle of utterances between two characters is highly predictable, usually following a regular protocol, and we call it an *interchange*.

The conversation originates in $\{A_1, B_1, C_1, D_1\}$ as A broadcasts a REQUEST to its trading partners B, C, and D. B and D respond as expected, resolving A's REQUEST with a REFUSE and a COMMIT, respectively, so their responses remain within the original component. Because D's original SHIP (utterance 11) completes its COMMIT (utterance 8), the SHIP is also part of this component.

C does not accept the terms of the discussion. Its PROPOSE (utterance 5) does not Resolve A's original REQUEST. Thus it spawns a new component, $\{C_1, A_2\}$, in which C and A agree on new terms, leading to a COMMIT by C at utterance 7.

The interchange between B and C before B's decision not to bid (utterances 2 and 3) generates a separate component of the graph $\{B_2, C_3\}$. This component is separate because none of its utterances Replies to or Completes any of those in the main component. We can integrate it with the rest of the graph by using information from the "Responds" relation, but there are tradeoffs, discussed below.

The sequence numbers show that D's COMMIT (utterance 8) arrives after C's (utterance 7). Because D's COMMIT matches A's original REQUEST while C's does not, A cancels the arrangement with C in utterance 9. This utterance does not Resolve any utterances in $\{C_1, A_2\}$, and so initiates a new component, $\{A_2, C_2\}$, a topological reflection of the discontinuity that such a withdrawal represents in the overall conversation.

A's conversation with D also spawns a new component, $\{A_1, D_2\}$, when A finds D's initial SHIP (utterance 11) deficient. Again, the topology of the graph captures the discontinuity in the conversation.

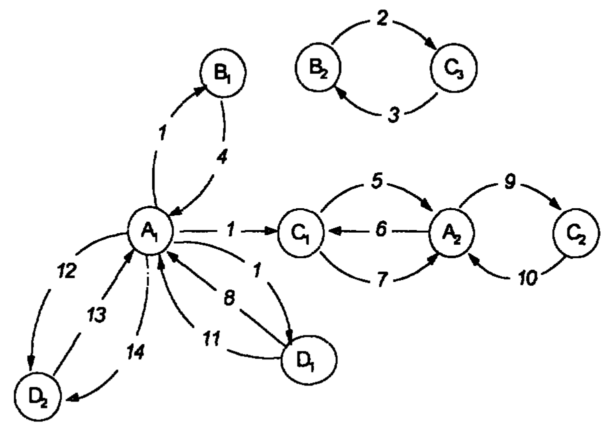

Figure 5: Dooley Graph for Table 1

5.2 The Basic Dooley Graph

Formally, a Dooley graph is generated by a 4-tuple <E, P, M, A>, where

E = { 1, 2, ..., n} is a set of counting numbers indexing the chronologically ordered utterances in the conversation (if the utterances form only a partial order, indices are assigned arbitrarily to incomparable utterances);

$P = \{p_1, p_2, ..., p_m\}$ is the set of participants in the conversation;

$A = \{<p_i, p_j, k> : <p_i, k> \in S \ \& \ <k, p_j> \in R\}$ is a set of ordered triples, defined with the aid of two sets of ordered pairs over E and P: the Sender set $S = \{<p_i, k> :$ participant p_i sends utterance k}, and the Addressee set $R = \{<k, p_j> :$ participant p_j receives utterance k}. (The notation 'R' reflects Dooley's use of the word "recipient" rather than "Addressee.") There is no assumption that each utterance has only one sender and one addressee. However, we do not plot promiscuous eavesdropping. Each triple in A will become an arc in the Dooley graph.

M is a relation from S∪R to S∪R, which generates the vertices of the graph (the characters) and indicates which arcs (utterances) are linked at which characters. We require that whenever two ordered pairs are M-related, their P

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

coordinates must be the same. That is, M may relate two utterances if either

1. the two are sent by the same participant,
2. the two are addressed to the same participant,
3. the addressee of the first sends the second, or
4. the addressee of the second sent the first.

M *may* relate two utterances under these conditions. Whether it actually does or not depends on the particular discourse theory embedded in M. M defines the characters. Requirements 1-4 ensure that each character corresponds to a single participant, while the restrictions imposed by the discourse theory distinguish the various characters played by a single participant. Within the bounds of this restriction, we can experiment with the actual definition of M to achieve the right balance of state and participant information.

Dooley originally suggested the condition that $<i,p>M<p,j>$ iff any of three conditions is met:

a) j Replies to i,

b) i Replies to and Resolves j,

c) i Replies to j and is the last utterance in the conversation.[5]

To capture the information in the Completes relations between utterances, I add a fourth option:

d) i Completes k and k Replies to and Resolves j.

Thus Completion becomes a way for utterances to inherit Resolution from one another. This definition does not capture any information from Responds. The next section discusses why Responds is needed, shows why it difficult to capture it in the M relation, and suggests a solution.

We move from $<E, P, M, A>$ to a graph by the following steps.

Define an equivalence relation N over $S \cup R$ by first copying M into N, then closing N under symmetry, reflexivity, and transitivity. (If aNb, add bNa, aNa, and bNb. If aNb and bNc, add aNc. Repeat until there are no further additions to N.)

N induces a partition $V = (S \cup R)/N$ of $S \cup R$. The elements of V are the characters. The P coordinates of all the members of any one element of V are the same (though there may be several vertices representing a given participant), and are labeled by their P coordinate and an appropriate index. The arcs of the graph are the triples in A. For $<p_i, p_j, k> \in A$, there exist unique members v_1, v_2 of V that contain $<p_i, k>$ and $<p_j, k>$, respectively, and an arc labeled k is drawn from v_1 to v_2. For this M, a participant changes character when control of the conversation changes, that is, when another participant Replies to a SOLICIT with other than the ACT expected by the SOLICIT, or otherwise initiates a new thread. Thus a conversation in which every participant knows its place and speaks only when spoken to appears as a graph with only one character per participant, while a free-for-all spawns long chains as participants change character in attempts to gain control of the conversation.

5.3 Enhancing Graphs with "Responds"

The definition of M given above can yield graphs with multiple components. This disconnection captures important information about the flow of the conversation, but also loses some information. In the example, the interchange between B and C takes place because B has been invited to bid, and this causal connection is lost in the graph. A similar structure would arise if B successfully invited C to become a subcontractor, and then dealt with A on the basis of this subcontracting relation, for then one would have product flowing from C to B and then to A, and payment flowing in the reverse direction without any corresponding links on the graph joining the A-B component with the B-C component.

The "Responds" relation captures just this sort of information. If we add Responds to the M relation, all Dooley Graphs will be connected. However, doing so sometimes merges different characters of the same participant. Extensive experimentation shows that it is futile to try to capture the Responds information within the Dooley Graph. We can, however, capture it by extending the graph with unlabeled edges that connect different characters of the same participant. To emphasize that these edges are not utterances, we represent them with a different kind of line. Because they are unlabeled, we can use a double-headed arrow as shorthand for one unlabeled edge in each direction. I propose the following rule:

If $<i,p> \in R$, $<p,j> \in S$, j Responds to i, j does not Reply to i, and $<i,p>$ and $<p,j>$ are in different nodes not already linked according to this rule, add an unlabeled edge from the node that includes $<i,p>$ to the node that includes $<p,j>$.

Figure 6 shows the results of applying this rule to the example.

Thus extended, the graphs capture all the participant/state information that they did before, without forcing the merger of nodes that should remain separate. The unlabeled edges show where disconnected components relate to the rest of the graph, while their distinctive type permits us to count the separate components in the conversation. In the example (Figure 6), the new rule not only relates $\{B_2, C_3\}$ to the main graph, but also highlights the relation between utterance 9 and A_1, which includes the results of D's bid. The new formalism is less compact

[5] This final category is needed to ensure that final but nonresolving utterances have a node on which to terminate.

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

mathematically than the original one, but still unambiguous and formally executable, and much more useful.

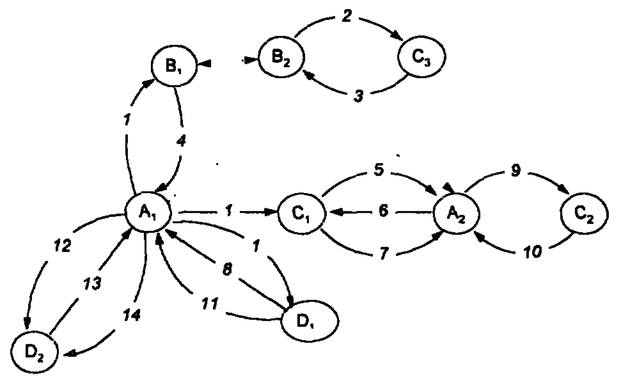

Figure 6: Full Example (Enhanced)

6. Uses of Enhanced Dooley Graphs

Enhanced Dooley Graphs are useful both in analyzing the performance of existing communities of agents, and in designing new agents. Because the graphs capture information about agent interaction in a way that makes sense to human designers and system operators, they can be applied not only to agents that reason explicitly about speech acts, but also to systems in which human designers but not agents reason explicitly about speech acts.

6.1 Analysis

Many details of the behavior of agent-based systems cannot be predicted analytically in advance, but must be observed in simulation. An Enhanced Dooley Graph of the interactions that emerge from a community of agents can provide the basis for a number of quantitative measures that are relevant to the adaptability of the community. Such measures are an important tool in enabling communities of commercial agents to thrive in the current business environment of continual and unexpected change [Nagel & Dove 1991; Goldman et al. 1995].

For example, separated components in the basic Dooley Graph (without adding the edges from the Responds relation) represent sidebar conversations that participants undertake in response to the state of the overall conversation but not in reply to specific utterances. The more willing and able participants are to enter into such sidebar conversations, and the wider the range of potential partners with which they can initiate them, the more agile the community is. Components of roughly the same size reflect more concurrency in resolving complex issues and more parity among participants, while one very large component and several smaller ones reflects centralization and domination by one player. Each component has an initiator, a participant who produces the first utterance in that component. If this initiator is always the same participant, we may conclude that this participant dominates the conversation, resulting in a less agile structure than one in which many participants can initiate conversations.

Similarly useful measures can be derived from the lengths of various paths in the graph (such as reply cycles, a series of utterances that returns to the starting character) and the degree (number of incident utterances or adjacent characters) of a specified character.

These measures are only a few examples of how the Dooley Graph (both basic and enhanced) induces a topology from an existing body of discourse, a topology whose metrics reflect meaningful aspects of the dynamics of that discourse.

6.2 Design

Although Dooley Graphs are generated from an interchange among existing agents, these existing agents can be designers role-playing an agent-based system, and the resulting graphs have proven useful in revealing patterns that can guide subsequent implementation. In particular, the interchanges in an Enhanced Dooley Graph help identify smaller reusable computational modules from which agents can be composed.

This modular approach is exemplified in Ferber's Bric architecture [Ferber 95], and is proving fruitful in the AARIA project for agent-based scheduling [Parunak et al. 1996], in which these modules are called "actions," and in the architecture underlying the NCMS Shop-Floor Agents project [Parunak 1996], in which they are called "contexts." The definition of these modules is a critical design decision. They should large enough that they represent a significant reduction in effort when they are reused, but not so large that they embed custom functionality that limits their reusability.

The interchange between a given pair of characters is usually fairly predictable, and the same set of utterance types is often repeated several places in a conversation. For example, the interchanges between A_1 and its potential suppliers B_1, C_1, and D_1 are all prefixes of the same protocol, a customer-initiated contract net, while the interchange between C_1 and A_2, a supplier-initiated contract net, follows the same pattern with an additional utterance at the beginning. Because of their stereotyped nature, interchanges are excellent candidates for agent modules. We sketch out sample Enhanced Dooley Graphs for a new system using a combination of brainstorming and role-playing, then identify repeated interchange types, and define a module for each end of each interchange. The interactions among the modules in a single participant are in turn defined by those interactions that intersect in a single participant. Thus the common structure made visible by the Enhanced Dooley Graph is proving a useful design aid for constructing modular agents.

From: Proceedings of the Second International Conference on Multiagent Systems. Copyright © 1996, AAAI (www.aaai.org). All rights reserved.

7. Acknowledgments

This work was supported in part by DARPA Contract F33615-95-C-5513 to Sirius-Beta, under a subcontract to the Industrial Technology Institute. The views expressed are those of the author, and do not necessarily reflect official positions of the Industrial Technology Institute, Sirius-Beta, DARPA, or the United States Government.

8. References

Austin, J.L. ed. 1962. *How to Do Things with Words.* J.O.Urmson, Oxford University Press.

Barbuceanu, M., and Fox, M.S. 1995. COOL: A Language for Describing Coordination in Multi Agent Systems. In Proceedings of the First International Conference on Multi-Agent Systems, 17-24. ICMAS-95.

Cohen, P, and Levesque, H. 1988. Rational Interaction as the Basis for Communication. SRI Technical Note 433.

Cohen, P. and Levesque, H. 1995. Communicative Actions for Artificial Agents. In Proceedings of the First International Conference on Multi-Agent Systems, 65-72. ICMAS-95.

Davis, R., and Smith, R.G. 1983. Negotiation as a Metaphor for Distributed Problem Solving. *Artificial Intelligence* 20:63-109.

Denning, P. 1992. Work Is a Closed-Loop Process. *American Scientist*, July-August

Dooley, R.A. 1976. Repartee as a Graph. Appendix B in Longacre 76: 348-58.

Ferber, J. 1995. *Les systèmes multi-agents: vers une intelligence collective.* Paris: InterEditions.

Goldman, S.L., Nagel, R.N., and Preiss, K. 1995. *Agile Competitors and Virtual Organizations: Strategies for Enriching the Customer.* Van Nostrand Reinhold.

Kamp, H. 1981. A Theory of Truth and Semantic Representation. In Groenendijk, J., Janssen, Th., and Stokhof, M. editors, *Formal Methods in the Study of Language. Pt. I.* Amsterdam: Mathematisch Centrum, 277-322.

Kreifelts, T., and vonMartial, F. 1991. A Negotiation Framework for Autonomous Agents. In Demazeau, Y., and Mueller, J.P., editors, *Decentralized A.I.—2.* Amsterdam: North-Holland, 71-88.

Longacre, R.E. 1976. An Anatomy of Speech Notions. Lisse: Peter de Ridder.

Mann,. W.C., and Thompson, S.A. 1987. Rhetorical Structure Theory: A Theory of Text Organization. ISI Reprint Series ISI/RS-87-190.

Müller, H.J. 1996. Negotiation Principles. In O'Hare, G.M.P., and Jennings, N.R. editors, *Foundations of Distributed Artificial Intelligence.* New York: John Wiley, 211-29.

Nagel, R.N., and Dove, R. 1991. 21st Century Manufacturing Enterprise Strategy. Edited by S.L. Goldman, S.L., and Preiss, K. Agility Forum Publication LU91-01.

Parunak, H.V.D. 1987. Manufacturing Experience with the Contract Net, in *Distributed Artificial Intelligence*, Huhns, M. Ed. London, England: Pitman, 285-310.

Parunak, H.V.D. 1996. Workshop Report: Implementing Manufacturing Agents. Report of the workshop sponsored by the Shop Floor Agents Project of the National Center for Manufacturing Sciences in conjunction with PAAM'96, Westminster Central Hall, London, UK, 25 April 1996. Available at http://www.iti.org/~van/paamncms.ps .

Parunak, V., Baker, A., and Clark, S., 1996. The AARIA Agent Architecture: An Example of Requirements-Driven Agent-Based System Design. Submitted to ICAA '97. Available at http://www.iti.org/~van/aaria.ps .

Smith, I.A., and Cohen, P.R. 1995. Toward a Semantics for a Speech Act Based Agent Communications Language. In Proceedings of CIKM '95 Workshop on Intelligent Information Agents. Available at http://www.cs.umbc.edu/~cikm/iia/ .

Winograd, T., and Flores, F. 1988. *Understanding Computers and Cognition.* Addison-Wesley.