Simulation of a distributed expert-based information provision mechanism

N J Belkin, R D Hennings* and T Seeger*

In earlier work, a model of the information provision mechanism (IPM) as a collective of cooperating entities, each expert in a particular function associated with information interaction and provision, was proposed. This was originally a descriptive and analytical model, but it was also suggested as a design model. In order to investigate its design potential, its general validity, and some architectural issues, five simulations of the model, using human beings as the functional experts, were run. The results of the simulations indicate that the general model is valid as a design tool, that the functional specifications are at least necessary, and that a blackboard communication structure with modified distributed control seems optimum for system implementation. They also indicate a partial ordering on the sequence of function processing, and suggest that messages within such a system need to be typed.

Keywords: simulation, information provision mechanism, expert systems

INTRODUCTION

Distributed expert systems

Since about 1975, there has been a growing trend in artificial intelligence (AI) towards an approach to system design called 'distributed AI'. The bases of the distributed AI approach are *modularization* in problem solving and *parallel processing*. Its assumption is that some problems can be usefully decomposed into logically independent elements, each of which can be treated as a more manageable subproblem of the whole. The method, then, is to devise processors which deal independently and in parallel with each of the subproblems, and somehow communicate their results to one another. The overall result of the joint processing is a response to the original problem. There are a number of theoretical advantages to this approach over the unified approach to system design, especially in highly complex or uncertain problem domains. We have discussed some general characteristics of this approach elsewhere[1], and a number of reviews, surveys and special publications on distributed AI are available (see, e.g., Chandrasekaran[2], Davis [3], Smith[4]). We therefore confine our remarks here to some specific problem areas in this approach to system design that we have investigated in our own work.

It is obvious that the communication and control components of such a distributed system are crucial. These issues have been intensively discussed (see, e.g., Schindler and Spaniol[5], Smith and Davis[6], Wesson *et al.*[7]), the major result being that choice of structures is heavily dependent upon the nature of the problem and of the individual processors or 'experts'. If one considers such systems as distributed networks the nodes of which are the experts and the links are communication paths, one can distinguish two more-or-less polar types of control structures.

One pole is the 'flat' non-hierarchical 'cooperating experts' or *type X* organization, as exemplified by the HEARSAY-II speech understanding system architecture (e.g., Erman and Lesser[8]). This form of organization is composed of

Department of Information Science, The City University, London EC1, UK
*Projekt INSTRAT, Free University Berlin, D1000 Berlin 46, Federal Republic of Germany
Received 16 April 1984

specialists with little or no hierarchical structure. Such organizations solve problems by sharing individual perspectives, which refine and ultimately integrate local interpretations into a unified group consensus. Subtasking, reporting requirements, and resource allocation decisions are generally not specified *a priori*. The organization forms behaviour and communication patterns dynamically in response to the environment and changes the patterns in a data-driven way[7].

In the 'message puzzle task' this structure has been denoted by the term *anarchic committee*, which reflects

the absence of overt governmental structure and the tendency for this organization to spawn many overlapping committees to perform specific tasks[7].

In direct contrast to the 'cooperating experts' paradigm is the very hierarchical *theory Y* or 'preceptual cone' organization from organization theory (e.g., Lawrence and Lorsch[9]) or AI work in pattern recognition (e.g., Uhr[10]).

Organizations of this class are assembled as strict hierarchies of abstraction levels, where at each level the individual elements receive reports from levels below them, integrate the reports according to their special skills and position in the hierarchy, and report upward abstracted versions of their results. The highest level of the network may repeatedly order its subordinates to adjust some previous reports in accordance with its own global perspectives, or it can report the overall interpretation it has formed[7].

In the message puzzle task mentioned above, this form of organization has become known as the *dynamic hierarchical cone*, which is a perceptual cone organization, modified to be more responsive to either a spatially unbalanced or rapidly changing dataflow.

Another way of stating the control distinction is as between control distributed among the experts and control concentrated in one or a few experts. This is the way we have interpreted the control issue in our empirical work.

The second issue, communication structure, can also be viewed as basically polar. In this case the poles are broadcast, or *blackboard* communication, versus direct agent-to-agent or *actor* communication. In the former, again exemplified by HEARSAY-II[8], all of the experts communicate with a global database, the 'blackboard', posting their messages there and 'reading' from it what they need to continue their own processing. In this case, the experts do not need to know about one another, and send basically one type of message, the 'hypothesis' about their subproblem.

In the actor communication structure, in contrast, each expert communicates directly with one or more other experts, as necessary. In this model, as shown for example by Hewitt[11], it is necessary for the experts to know about one another, what each needs and/or what each can provide, and it is possible, and perhaps desirable or even necessary to have different message types, such as question or request, and hypothesis.

These two issues, communication mode and control structure, are crucially important in specifying any distributed system architecture. A major goal of our empirical work then was to try to discover if some particular type and mode were more 'natural' or 'better' for the problem we are attacking and the experts we have defined than any other.

The MONSTRAT model

The problem with which we are concerned is that of the general information provision mechanism (IPM), that is, some intermediary mechanism in an information system that interacts with a person (the user) who has a problem that may require information for its management, in order to provide information (or some other response) appropriate to the user and the user's problem at that time. The problem that the IPM faces, then, is *understanding* or modelling the user and the user's situation in such a way that it can provide an appropriate response to the user. One should note especially that an appropriate response might be helping the user to understand her/his problem, rather than providing information for resolving it. In general, however, one might consider that the IPM has access to a knowledge resource that provides the basis for any response it makes to the user.

We have discussed the characteristics and theory of such an IPM elsewhere[1, 12, 13]. Here we mention only briefly what the IPM must minimally do in order to solve or manage its own problem, and describe how we have decomposed this problem into its constituent 'experts' or functions.

Our basic method in doing this has been analysis of human/human interaction; that is, of situations in which a human user interacts with a human advisor or with a human intermediary to some information system, for the purpose of acquiring information that will help her/him to deal with (treat, manage) some problem. Examples that we have analysed include student advisory interactions, rent advice situations, and interaction between academics (users) and intermediaries (searchers) for online bibliographic retrieval services (see, for instance, Belkin and Windel[14]). Our assumption has been that the 'best' such interactions will provide good functional models for human/machine interaction, and that 'bad' ones will demonstrate what functions are minimally necessary. We wish to stress, however, that given the complexity of interaction and level of knowledge necessary in the situation with which we are concerned, it is fully possible, indeed even likely, that some of the necessary functions can only be performed by humans.

Table 1 lists the functions we have identified as taking place in such interactions, and which we think are minimally necessary for successful interactions. That is, these are the subproblems which the IPM must resolve in order to achieve its goal of appropriate response. We interpret these as logically discrete components, which need to interact with one another in order to obtain the data necessary to perform their functions, but which compute in isolation, each attempting to resolve its own problem.

In brief, we can summarize our analysis by saying that, in the general information interaction, the IPM needs to have:

- an understanding of the state of the user in the problem solving process (PS),
- an idea about what kind of response or system capability is appropriate for this user and problem (PM),

Table 1. The functions of an information provision mechanism (After Belkin *et al.*[1])

Name of function	Description
Problem state (PS)	Determine position of user in problem treatment process, e.g., formulating problem, problem well specified
Problem mode (PM)	Determine appropriate mechanism capability, e.g., reference retrieval
User model (UM)	Generate description of user type, goals, beliefs, e.g., graduate student, thesis
Problem description (PD)	Generate description of problem type, topic, structure, environment
Dialogue mode (DM)	Determine appropriate dialogue type for situation, e.g., natural language, menu
Relevant world builder (RWB)	Choose and apply appropriate retrieval strategies to knowledge resource, e.g., best match, gap filling
Response generator (RG)	Determine propositional structure of response to user appropriate to situation
Input analyst (IA)	Convert input from user into structures usable by functional experts
Output generator (OG)	Convert propositional response to form appropriate to user and situation
Explanation (EX)	Describe mechanism operation, capabilities, etc., to user as appropriate

- a model of the user her/himself, including goals, intentions and experience (UM),
- a description of the problem the user is facing and the user's knowledge about it (PD),
- a hypothesis about what sort of dialogue mode is appropriate for this user and problem (DM).

This information will be gained through interaction with the user, which will require analysis of the user's part of the dialogue so that it can be used by the other functions (IA). The resulting model can then be used to specify what aspects of the knowledge resource or database might be relevant to the user at this time (RWB), and from this potentially relevant 'world' a response particular to the specific situation can be generated (RG). This response needs to be put into the appropriate dialogue mode, that is, an output to the user is generated (OG). Finally, it may be necessary to explain the IPM's operation and competence to the user (EX). These functions we consider as necessary subproblems of the overall IPM problem. Processors aimed at solving these subproblems thus constitute the 'expert' components of our distributed expert model of the IPM — The MONSTRAT model.

Again, we have argued elsewhere[1] that it makes sense to treat such a system from a distributed AI point of view, the experts being functionally defined. The argument is basically that the overall problem is too complex for unitary system design, that it decomposes naturally into discrete expert functions, and that the interaction among functions is too complex for other than parallel processing.

This all sounds good from an abstract, analytical point of view. However, if one wants to use such a model actually to design and build a machine-based IPM, then some validation of the basic premises and of the model is necessary, as is some idea of how it might actually be constructed, that is, of an appropriate system architecture. We decided to carry out this validation and testing exercise by means of *simulation* of a MONSTRAT system, using human beings as the functional experts. The simulation consisted of assigning each (with some exceptions) of the MONSTRAT functions to a different person, inputting to the system thus constructed a question or request for information, based on a well-specified problem situation, and allowing the experts to compute their functions, pass messages to one another and communicate with the user (via the OG) until the system's response satisfied the user, or the user lost patience. The remainder of this paper reports on the methods of the simulation, its results and their implications.

METHODS

Goals of the simulation

The MONSTRAT model as described above implies a wide variety of problems that could be tested in a simulation with human experts. Our main interest concentrated upon three major issues.

- Whether the intellectual architecture of the MONSTRAT model is valid. At the most gross level, this means whether the simulated system were actually able to generate satisfactory answers to queries put to it or not. Despite the fact that the role-taking mechanisms of humans in simulations of this kind are very difficult to control and to measure, the general process of message-generation and distribution could give an overall picture of the completeness of the predefined functions associated with the experts taken together. In this sense, it is not of great importance that every single expert fulfils its functions — as predefined for the simulation — in a very strict way, but that all the experts together accomplish the ultimate goal of the information provision process, regardless of the performance of single experts. Finally, we wished to discover whether the single message type, hypothesis, is sufficient for such a system.
- Whether communications mode would affect the interactions among experts sufficiently to affect system performance, and if so, whether any particular communication structure seems best. We chose to investigate the relative performance of two modes which seemed to offer the greatest contrast: blackboard communication versus actor communication (as described above).
- Whether control structure would affect the interactions among experts sufficiently to affect system performance, and if so, whether any particular control structure seems best. We chose to investigate the relative performance of two control structures: distributed control, in which each expert decides for itself when to do processing and what to do, and to whom to send messages (in the actor communication mode); and, centralized control, in which some expert is designated to oversee the processing as a whole, controlling

message passing and processing activity of the other experts. These structures we designated as 'uncontrolled' and 'controlled', respectively.

Variables

In order to investigate the first point we needed to see whether any answer at all was produced by the system, and then to judge whether it was a reasonable answer. For this issue, we needed to define no specific variable, but we did need an idea of what constitutes a reasonable answer. We describe how we approached this problem in the discussion of queries put to the system. To come to some idea of the sufficiency of our functional description, we analysed the messages passed by the experts for content and function, and to investigate message types, we did the same. We also wanted at this level to get some idea of communication patterns among experts, to see if some are more 'central' or 'important' than others, if hierarchical or serial structures are present, whether societies of experts evolve, whether some experts are overloaded, and so on. This we did in two ways, first by drawing graphs of the communication patterns, and second by quantitative measures on the individual experts' activities. The measures chosen were the ratio of the single expert's messages that were used by other experts to the total of all messages used; and, the ratio of the single expert's messages used to the total sent by that expert. These measures, with some qualifications, discussed in the results, we interpreted as reflecting, respectively, 'importance' and 'efficiency' of each expert.

The second and third points in the previous section are concerned with getting some idea of the 'best' communication and control structures for a MONSTRAT system. 'Best' is a problematic term, but very generally it could be thought of as those structures that allow, or lead to, the most efficient and effective communication among the experts, and the most efficient and effective response to the user. The conditions of the simulation itself rule out much detailed consideration of the relation between communication and control among the experts and MONSTRAT's responses to the user, so we are forced to consider 'best' from the point of view of internal communication, keeping in mind, of course, that it must be consistent with achieving appropriate responses to the user. For instance, a structure that resulted in no response to the user at all would probably not be a candidate for 'best' structure. Thus our basic variables were concerned with internal efficiency and effectiveness of communication.

In general, a configuration that reached its goal with fewer messages passed than another could be thought of as being the more *efficient* of the two. From a system design point of view, a configuration that results in a fairly even distribution of messages, over time and over experts, would probably also be considered more *efficient* than one in which these distributions are skewed. Similarly, a configuration in which the messages passed have a higher incidence of usage than another would be judged the more *effective*, as would one in which most messages were available when needed, rather than before or after. This last quality is difficult to judge quantitatively, but a structure in which usages were distributed relatively evenly in time, in relation to message passing activity, might be thought to reflect it. We operationalized these considerations by specifying the measures summarized in Table 2.

Table 2. Specification of variables measured ('measures') for each simulation round, and criteria for rank ordering of simulation rounds by measure

Measure number	Definition	Rank 1 value
1	Mean messages sent, per 10 minute interval	minimum
2	Variance of messages sent, per interval	minimum
3	Mean messages used, per interval	maximum
4	Variance of messages used, per interval	minimum
5	Total messages used/total messages sent	maximum

Conduct of the simulation

The simulation was conducted with single persons as the exerts, each expert being observed by another person. Five rounds of the simulation were performed, each with a different topic of query, and with a different combination of communication and control structures, over a period of two days in January, 1983. Each expert was given a detailed description of how s/he was to perform her/his role or function; each observer had the same description, plus instructions on particular sorts of behaviour that were to be noted or described.

Each pair of expert and observer was spatially separated from each other pair, that is, each pair was in a separate office. It was forbidden for experts to communicate with one another verbally, in those cases where they had to leave their offices, and the only communication allowed was via specially designed message forms that each expert filled out and distributed (via the observer) as necessary. Observers also had special forms for their comments, but these were not distributed. Experts and observers also had detailed instructions for their behaviour during the simulation, including specific rules for the formal conduct of message form completion and distribution.

The time of sending of the message, and the time of receipt of the information that was used in generating the message was noted on each message form. The form itself was divided into three major sections:

- input — identification of the messages which the expert used as data for generation of her/his message,
- treatment — specification of the operations the expert performed on the data in order to generate the message (according to the expert's functional specification),
- output — the message being sent. Each message of the type hypothesis was required to have a certainty level on a five point scale, ranging from positive to blind guess, associated with it.

The observers' forms were analogously structured, with space for observations and comments upon the expert's behaviour. All of the experts and observers received about two hours' instruction in role playing and behaviour for the simulation, prior to the simulation itself.

The simulation itself was managed by a team of three 'supervisors', one of whom was solely responsible for overseeing the general conduct of each round of the simulation; the others assisted in this responsibility, and also acted either as user/input analyst or output generator, and in the controlled simulations, as controlling function. These three were responsible for compiling the queries and associated problems that were put to the IPM: these were unknown to the experts beforehand.

Through the measures described above, we tried to exclude or control for at least the most obvious interfering factors that will affect the results of simulations of this kind. In brief, these measures were:

- control of experts by observers,
- control of the most obvious environmental influences by strict organization of the simulation,
- reliance on written messages only,
- exclusion of verbal communication among the simulation participants,
- strict role definitions in order to force concentration on a limited number of tasks,
- strict rules for behaviour and for the message forms, in order to allow a precise reconstruction of single events.

Thus, we hoped to have a simulation that would duplicate as closely as possible the key components of the MONSTRAT model, that is, independent, parallel computation of specific functions by specific experts, each expert's processing being based on messages from various of the other experts, as defined, the result of each expert's processing being a hypothesis about the solution to its own subproblem.

The experts and their roles

In the simulation we considered only seven of the ten functional experts listed in Table 1, but added, for the rounds that were controlled, one additional expert for the controlling function. On the grounds that the functions associated with direct interaction with the user (that is, those that converted input to the user to the system language, and output from the system to the user's language) only interacted with the other functions at single points, and were anyway beyond our technical competence to simulate, we eliminated IA and OG from the simulation. Since we were primarily concerned with interaction among the functional experts, rather than between system and user, we decided not to include the EX function. The remaining group of experts was on the one hand large enough to investigate the goals we had set, but on the other hand was small enough to manage and supervise the simulation without running into severe organizational problems.

We mentioned above that each of the experts had a set of instructions defining his or her functions (role) and how they were to be performed. These instructions included a functional role specification that was divided into the sections: overall task; input to be operated upon; output to be generated; users of the output; and, internal resources that can be used to operate upon the input. Appendix 1 shows the role specification for the problem description function, as an example of the level of detail of these instructions.

The *input* specified from whom messages could be received: in the blackboard communication mode (see below), this specified the relevant parts of the blackboard. The *output* specified the topics on which the expert was to generate hypotheses or other messages. The *user* section specified, for the actor communication mode (see below), to whom messages should be sent. And the *resources* section specified the knowledge that the experts were allowed to use in computing their messages, based on input. A major task of the observers of the experts was to make sure that the experts limited themselves to their specified functions, and especially to notice if they used other than the specified resources in generating their messages.

The two communication modes

As mentioned above, we wished, in the simulation, to consider the effect of two communication modes: blackboard and actor. These were implemented in the simulation in the following ways.

The blackboard communication mode, which can be considered an *indirect* mode of communication in which all of the experts communicate to and from a common database, the blackboard, was implemented by establishing one room, where the simulation was administered, as the blackboard. In this room, there were physically separate spaces, either real blackboards or pin-boards, corresponding to each of the experts, where the experts could place their messages, and from which the experts could read the messages that they would take as input for their processing. During the simulation, the experts, accompanied by their observers, whose role at this time was to prevent the experts from talking to one another, and from looking at forbidden parts of the blackboard, would go to this room to look for appropriate input. When a message or messages appeared on the blackboard on which they felt they could operate, they would take a note of it or them, return to their rooms, where they would generate their own message on the appropriate form, and return to the blackboard room, where they would post a copy of their message. Only the 'message' section of this copy of the message form was thus posted, although a copy of the full form went to the simulation administration. The experts would then be allowed to check whether any new messages had arrived that were relevant to their particular tasks. These interaction and communication procedures are specified in Figure 1.

The actor communication mode, in contrast, is a *direct* mode of communication, in which experts communicate directly with one another. In our simulation, this was implemented by specifying, in the role description for each expert, to whom that expert was allowed to send messages, and from whom that expert could expect to receive messages. Thus, whenever an expert generated a message, that message would be sent to the group of experts listed under 'users' in the role specification. This mode of communication was much easier to administer than the blackboard mode, since the experts in this case were directed to remain in their separate offices, all messages being circulated from expert to expert by their respective observers. In addition to the pre-specified communication of hypotheses, the experts in this commun-

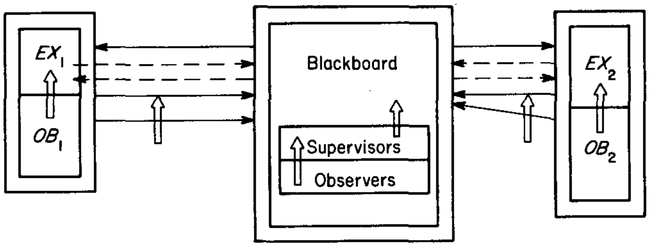

Figure 1. Structure for blackboard communication in simulation; EX_i = expert i, OB_i = observer i, ⇒ = control on activity. ▣ physically separate room, — physical communication paths, --- logical communication paths

ication mode were also allowed the possibility of sending questions or requests to other experts, including others than those on their input or output lists, if it seemed to them necessary. In such cases, they were required to state their reasons for sending the message. Again, the communication from expert to expert consisted only of the message section of the form, a copy of the whole form being sent to the central administration as well. The interaction and communication structure for this mode is shown in Figure 2.

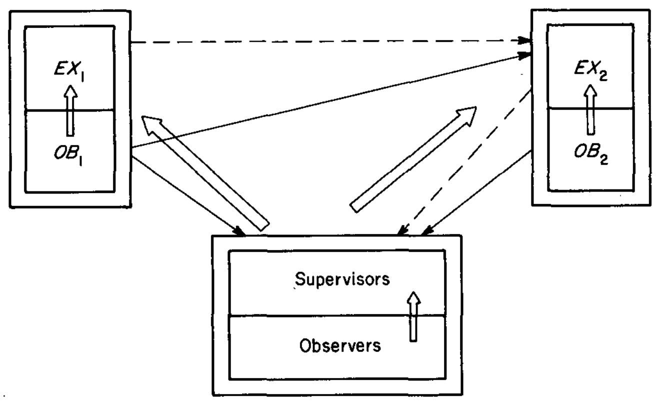

Figure 2. Structure for actor communication in simulation; symbols as in Figure 1

The control structures

We decided to test two modes of control over the communication and processing of the experts. In the first, which we called 'uncontrolled', the experts were allowed to communicate with one another without any intervention, and to process data to generate messages as they wished. This corresponds, we believe, to the distributed control situation. Thus, for instance, in the actor communication mode, uncontrolled, the experts were allowed to send any type of message to any other of the experts at any time. The individual experts were free, however, to decide whether or not to respond to any particular message or group of messages, and could generate hypotheses or messages about their particular responsibilities as they wished. Processing, then, went on without centralized control, interaction with the user being dependent upon the OG's response to individual messages, and completion of the total processing being dependent upon the user signing off.

In our second control structure, 'controlled', there was an expert designated to vet the passing of certain messages, especially those that required a response or action from some other expert. This corresponds to a situation in which there is some expert who has some view of the whole situation, who can, in effect, direct other experts to process, or to control who processes when, by controlling communication paths. In this case, interaction with the user was often dependent upon this expert, known as the 'system analyst (SA)' in the actor communication mode, and the 'blackboard analyst (BA)' in the blackboard mode.

The major role of the SA was to intercept questions or requests for information from one expert of another expert, and to decide whether: the intended recipient of the message could provide the information required, and if not, which expert could; and, whether that particular request should be honoured (that is, whether the message should actually be sent on). This role was accomplished on the basis of the SA's knowledge of the capabilities of the experts, and of its knowledge of the state of the system at the time. The role of the BA was similar, except that it basically acted as a filter on requests from the other experts to the OG to interact with the user, or to the IA for more input from the user. This was accomplished on the basis of the BA's knowledge of the state of the system at the time, and of the responsibilities of the various experts. The actions of the SA and of the BA were recorded on the same forms as were those of the other experts, and their activities were observed in the same way by another member of the simulation administration.

One other form of control was considered. This was to allow experts to post only those hypotheses that exceeded a specified certainty level, and to vary this level as a control function. We started by specifying level 3 (on the 5 point scale) as the cut-off point, but in the end did not attempt to vary this particular function, as the system worked at this level.

The questions put to the simulated IPM

Each round of the simulation had a different question put to it, based on a different problem situation, and, with one exception, concerned with different topics. In an attempt to deal with the problem of what is a 'reasonable' response, three of the queries were generated from previously collected and analysed records of real advisory interactions, in which the advisors gave the users some advice. In these cases, we could compare the original real interactions between advisor and user, with the interactions between the simulated MONSTRAT system and the simulated user. In this way we could get some idea of the validity or reasonableness of the system's response, as well as some guidance for the simulated user's interaction with the system. The other two queries were based on 'simulated' problem situations that were relatively simple, and for which we had available suitable knowledge resources. We introduced this second type because the real interactions and problem situations were very complex, and we wanted to have some fairly 'easy' and closed problems with which to begin the simulation. Therefore, we used the simulated questions for the first two rounds, held on the first day of the experiment. Each initial question put to the system was relatively brief, but the description of the underlying problem was quite detailed, in order to be prepared for possible interaction between the IPM and the person playing the user. Only that person saw the problem context description. An example simulated question and associated problem context is the following (from simulation round 2):

Initial question:

Which dishes go well with roasted wild boar?

Problem context:

The user is a woman whose husband is being considered for an important promotion in his company. He has invited the chief of the company, his immediate superior and their spouses to Sunday lunch (this Sunday). The user has decided to prepare a meal based on marinated and roasted wild boar, since she knows that the guests are fond of game, and because she has recently seen an interesting recipe for boar. She has never cooked boar, nor has she much experience with presenting game, and is concerned that the other dishes she presents will complement the boar well. The boar has already been laid in its marinade, and will be ready for roasting on Sunday morning. The user would like to serve a full meal, that is, a soup, a petit entree, two vegetables and potatoes or some grain dish with the meat, and a dessert. She has already purchased five bottles of Chateauneuf du Pape 1972 for the main dish. She has a well-equipped kitchen, and money for her is no problem. (NB, the simulation was carried out in German, so this example, and all subsequent ones, are translations).

The simulation rounds

The variables in the simulation were:

- communication mode: blackboard or actor,
- control structure: controlled or uncontrolled,
- type of problem: simulated or real,
- topic of problem.

Table 3 shows how these conditions were applied to each round. We used a standard 2 × 2 design to test for effects of communication and control conditions, running five rounds because round 4 failed to run to completion, for reasons that were too complex for us to assume that they could be explained simply by that particular combination of conditions. Control by the SA in round 4 was strictly as defined in the section on control structures, but was modified in round 5 to vet only those requests or questions that the recipients did not think should be honoured, or did not understand. In addition, in this round, the observers were allowed to intercede in the activities of their experts to force them to comply strictly with their functional specification. One further variation was introduced during the course of the simulation. For the first two rounds, the initial input from the IA was distributed to all of the experts at the beginning of the simulation. On the second day, this was modified so that the RWB, RG and OG did not receive this input directly, but only via messages from the other experts who had processed the input. These two variations are discussed in the results section, below. We did not attempt to control for problem topic or type.

Table 3. Variables in the MONSTRAT simulation

Round	Communication	Control	Problem type	Problem topic
1	Blackboard	No	Simulated	Advice for buying video
2	Actor	No	Simulated	Cooking advice
3	Blackboard	Yes	Real	Tenant advice
4*	Actor	Yes	Real	Student advice
5	Actor	Yes	Real	Tenant advice

*Round 4 terminated before completion.

Summary of the conduct of the simulation

The simulation as a whole was conducted as follows. All of the participants — who were either members of the Projekt INSTRAT team, or students at the Free University of Berlin Work Unit Information Science who were participating in seminars associated with that project — were assigned their responsibilities, as expert or supervisor, in advance. On the first day of the simulation, they were brought together for two hours, during which time both general and specific instructions for behaviour during the simulation were distributed in the form of written instruction packages, and a general introduction to the simulation was given by the supervisors. The experts' roles were explained to them individually, and some training in role playing was undertaken with each. After all questions had been dealt with, the simulation began.

Each round of the simulation was initiated by a question put to the system via the IA, who was, in effect, a simulation of the user. This role was played by one of the supervisors. Since all of the experts were human beings, the output message from the IA was in natural language, German, as were all other messages in the system. This message was distributed to the appropriate experts, who then began processing, if they felt they could, and producing messages for the other experts. Whenever the OG received a message from an allowed expert to interact with the user, another of the supervisors, playing the role of OG, sent the appropriate message to the IA, who then produced, as appropriate, a new input message to the system. This message passing activity and interaction continued either until the user was satisfied with the response from the system, or until the system was demonstrably failing, at which time the supervisors could call a halt.

Each expert was constrained by instructions to produce messages only according to her/his assigned functions. Messages were posted as indicated in the discussion of communication modes, above, with copies of every message form, and of every observer's form for each message being sent to the central administration of the simulation. Interaction among the experts was restricted to communication via the message forms, as far as possible. Only the RWB had direct access to the knowledge resource or database for each round. This consisted of documentation relevant to the question and problem area for each round, which had been previously gathered by the simulation supervisors, and of which the person playing the RWB had some experience.

The first two rounds of the simulation took place during one afternoon. The results of those rounds were then briefly analysed by the supervisors of the simulation, who incorporated the changes mentioned in the previous section into the simulation procedure. The last three rounds took place on the following day, one in the morning, two after lunch. Everyone went home exhausted.

RESULTS AND DISCUSSION

Quantitative evaluations of communication and control modes

The interpretation of the data collected in the simulation presents some difficulties, which have affected the results and the conclusions which can be drawn from them. One major difficulty lies in the distinction between *types* and *tokens*, and in how the actor and blackboard communication modes are to be compared against one another by any variable which includes 'messages sent'. The second problem is the learning effect during the simulation, and its influence on the results.

The first problem concerns how one is to interpret the concept of 'messages sent'. In the actor mode, the experts sent duplicate copies of their hypotheses or requests to *all* of the other experts to whom they pertained (or for whom the sending experts thought they were relevant). The total number of copies sent of each message constitutes the *tokens* of that message *type*. But in the blackboard mode, experts sent only one copy of each message, directly to the blackboard, irrespective of how many potential users of the message there might be, so that in this mode message types and message tokens are identical. Thus, if we wish to compare the effect of the communication mode on communicative activity in terms of messages sent, we obviously cannot compare the total number of messages actually sent in the two modes. The obvious solution, for comparative evaluations, is to compare messages sent in the blackboard mode with *types* sent in the actor mode. Although there are some problems with this solution (see conclusions), it is the one we have adopted for overall comparative purposes.

The other problem, that of *learning* on the part of the humans in the simulation, is even more difficult to deal with. Obviously, the humans did learn how to manage the simulation of the experts with repetition, and obviously those running the simulations learned more about this with repetitions, and obviously the instructions to observers to vet experts' activities in the final round affected the message passing activity. We have, unfortunately, found no satisfactory way to deal with this issue, at least at the quantitative level of this part of the evaluation. Therefore, we reluctantly ignore it, and compare and cumulate as if there were no such effect.

Finally, we note that round 4 was halted prematurely, because the control expert could not handle the volume of messages being sent to it for decisions on further distribution. Although there remains some doubt that this form of control was the sole reason for the failure of this round, it is obvious that it played a major role. Because the round was not completed, we do not include it in the cumulated figures of Table 5, in which we group results by control and communication mode. On the assumption, however, that we might be able to understand what happened in round 4 in terms of our measures, we do include it in Table 4, which gives the results round by round.

Tables 4 and 5 compare the results of the simulations round by round, and by rounds grouped according to our independent variables, communication mode and control, respectively. In order to make these results more clear, in Tables 6 and 8 rank orderings of the rounds, according to our measures, are presented. These rankings are based on the ordering criteria of Table 2, and are cumulated on the assumption that all measures are equally important in assessing the effectiveness or efficiency of the control and communication variable.

If we consider first the more simple comparisons by independent variables (Table 6), we notice immediately that *control* appears to have had a significant effect in a positive direction. That is, in three of the measures, cumulated control ranks first and cumulated non-control ranks last, and on the other two measures the differences in actual values are so slight as to make the rank differences insignificant (for values, see Table 5).

For communication mode, on the other hand, the results in this table appear not to be so clear cut. With the exception of the two measures associated with use per interval (3 and 4), in which blackboard and actor modes alternated first and last ranks, these two cumulations ranked consistently in the middle, with blackboard marginally ahead of actor. The actual differences between the values of the measures on

Table 4. Quantitative profile of each simulation round

Messages / Round	1, BB no control 120 min.	2, Actor no control 120 min.	3, BB control 120 min.	4, Actor control 50 min.*	5, Actor control 50 min.
Sent**	50	61	47	23	13
Mean sent/interval	4.17	5.08	3.92	4.6	2.6
Variance sent	0.91	1.163	0.615	2.57	0.71
Used	84	71	84	31	27
Mean used/interval	7	5.92	7	6.2	5.4
Variance used	2.52	1.297	1.643	2.41	2.19
Used/sent	1.68	1.16	1.79	1.35	2.08

*Round 4 was interrupted before completion.
**Messages interpreted as *types*.

Table 5. Mean values for grouped rounds, from Table 4

Variable rounds Messages	All BB 1 + 3	All actor 2 + 5	All no control 1 + 2	All control 3 + 5
Sent	48.5	37	55.5	30
Mean sent/interval	4.045	3.84	4.625	3.26
Variance sent	0.7625	0.94	1.0365	0.66
Used	84	49	77.5	55.5
Mean used/interval	7	5.66	6.46	6.2
Variance used	2.08	1.74	1.91	1.92
Used/sent	1.73	1.62	1.42	1.935

NB. Round 4 not included because incomplete.

Table 6. Rank order for simulation rounds cumulated by independent variable for each measure and overall

Cumulated rounds Measure*	All BB 1 + 3	All actor 2 + 5	All no control 1 + 2	All control 3 + 5
1	3	2	4	1
2	2	3	4	1
3	1	4	2	3
4	4	1	2	2
5	2	3	4	1
Sum of ranks	12	13	16	8
Overall rank	2	3	4	1

*Numbers of measures identified in Table 2.

which the ranks are based do not, however, make one overconfident of the significance of the relative rankings. On measure 1, the difference between blackboard and actor is less than one-half the difference between ranks 1 and 2 and 3 and 4, and on measure 5 it is again about one-half of the other differences between adjacent ranks. It is only on variable 2, variance of messages sent, that there is a clear-cut 'victory' for blackboard mode over actor, the difference between the pair being very much larger than the other two pair-wise differences.

Thus, on the basis of Tables 5 and 6, one might conclude that imposition of some form of communication control positively affects communication effectiveness and efficiency; and, that there might be some marginal advantage to blackboard over actor mode of communication. Figure 3 compares the effects of the independent variables in more detail, looking at the actual differences in values for each measure for each variable, for round 1, 2, 3 and 5. Normalizing these data by dividing the summed differences for each measure by the sum of the values for the measure, as displayed in Table 7, allows us to make some estimate of how significant the effect of the variable was on each measure. By taking mean values for these figures for each variable (Table 7), we can also estimate the significance of the variable overall. If we suggest, conservatively, that a change of 1 part in 10 is potentially significant, then Table 7 indicates that imposing control had no significant negative effects, and that overall, the effect of control is positive. On the other hand, the only significant effects of the actor communication mode were negative (that is, blackboard mode was significantly better), and overall there was no significant difference between the two modes, although the direction favours blackboard. Of course, this analysis does a lot with rather skimpy data, but the trends seem reasonably obvious. These data do not, however, indicate what happens when the two independent variables are combined.

Table 8 shows the relative ranking of the individual rounds by each measure. By rank order, it is fairly clear that round 3 is the most consistent, and best performer, with round 5 a clear, although less consistent second, and round 1 a consistent third. Rounds 2 and 4 seem to share last place in the ranking, with very little to choose between them other than the fact that round 4 did not manage to get to completion. The major difference between the top two performers lies in round 5's relatively poor showing on measures 3 and 4. It would perhaps be drawing too strong a conclusion on the basis of the available data to say that the combination of blackboard with control is *clearly* better than any other tested, but the direction of the data, if we accept the validity of the measures, tends this way. In any event, the top two in cumulated ranks appear to be reasonably close to one another. These data, then, tend to confirm the other analyses, that is, that control is a

Table 7. Sum of differences/sum of measure for each variable and measure (see Figure 4 for original data)

Measure* Variable	1	2	3	4	5	Mean
Communication	+0.026	−0.106	−0.107	+0.088	−0.03	−0.026
Control	+0.17	+0.22	−0.02	−0.002	+0.15	+0.102

*Numbers of measures identified in Table 2.

	No control	Control	Difference
BB	(1) 4.17	(3) 3.92	+0.25
Actor	(2) 5.08	(5) 2.6	+2.48
Difference	−0.91	+1.32	+2.73 / +0.41

Mean sent per interval (No 1)

	No control	Control	Difference
BB	(1) 0.91	(3) 0.615	+0.295
Actor	(2) 1.163	(5) 0.71	+0.453
Difference	−0.253	−0.095	+0.76 / −0.36

Variance sent (No 2)

	No control	Control	Difference
BB	(1) 7	(3) 7	0
Actor	(2) 5.92	(5) 5.4	−0.52
Difference	−1.08	−1.6	−0.52 / −2.68

Mean used per interval (No 3)

	No control	Control	Difference
BB	(1) 2.52	(3) 1.643	+0.877
Actor	(2) 1.297	(5) 2.19	−0.893
Difference	+1.223	−0.547	−0.016 / −0.676

Variance used (No 4)

	No control	Control	Difference
BB	(1) 1.168	(3) 1.79	+0.11
Actor	(2) 1.16	(5) 2.08	+0.92
Difference	−0.52	+0.29	+1.03 / −0.23

Used/sent (No 5)

Figure 3. Mean values and differences of mean values of measures for all treatments, rounds 1, 2, 3 and 5

significant positive variable, and that the blackboard mode may be slightly better than the actor mode. They furthermore seem to show that the combination of blackboard communication with control may be felicitous.

Quantitative evaluation of the experts

The quantitative data do not offer a great deal in terms of evaluating the role and performance of the experts and their interaction. Nevertheless, some general, tentative conclusions seem possible. The ratio of expert's messages used to total messages used (Figure 4) is perhaps some indication of the general 'importance' of that expert. It is necessary to be somewhat careful in interpreting these data, since there is some doubt about which experts performed which functions. We have not presented these data for individual rounds because of such problems, but hope that cumulation over all rounds will tend to even out such effects. Accepting these reservations, it is reasonably clear from Figure 4 that IA and PD were overall the most 'important' experts in this simulation, with DM being

Table 8. Rank order for each round on each measure, and cumulated

Measure*	Round 1	2	3	4	5
1	3	5	2	4	1
2	3	4	1	5	2
3	1	4	1	3	5
4	5	1	2	4	3
5	3	5	2	4	1
Sum of ranks	15	19	8	20	12
Overall rank	3	4	1	5	2

*Measures identified in Table 2.

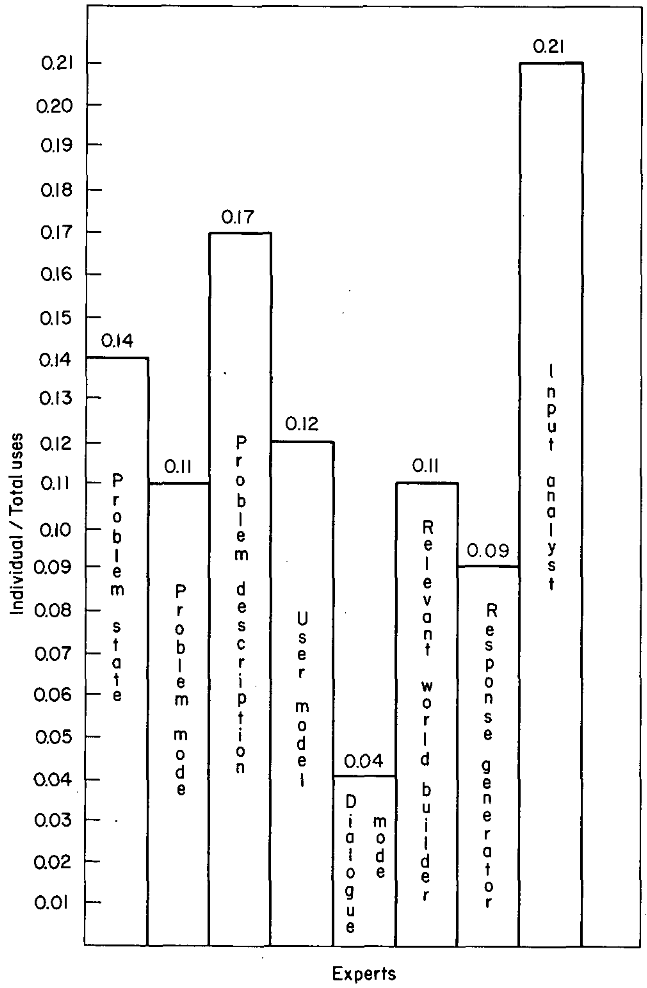

Figure 4. Ratio of uses of each expert's messages to total number of messages used, cumulated over all rounds ('effectiveness')

least important. This measure, however, is a reflection of *general* importance, that is, messages from these experts were heavily used, most probably by many of the other experts. It does not, however, say anything about the significance of any expert for successful response to the user. For instance, RG and DM, who might be exceptionally important from the point of view of the OG and the user, are least used by all of the other experts as a whole.

Thus, these data may allow us to discriminate among experts who communicate with all of the others, and those whose communication patterns are somehow restricted. From this point of view, it appears that DM, and perhaps RG have a rather restricted audience, and that IA and PD are generally useful to all of the other experts, with the other functions being intermediate. It is perhaps possible to gain a better feeling for this issue by examining the patterns of use of messages, as displayed in Appendix 2, but detailed analysis at this level is beyond the scope of this document.

We also looked at the experts' 'efficiency', that is, how successful they were in generating messages that were used by the other experts. Figure 5 displays the data for this measure, which may also be taken as an indirect indicator of importance. The problem with interpreting Figure 5 is that experts may well have sent more messages than were strictly necessary. Nevertheless, we see once again the pattern of Figure 4: that IA and PD are ranked first and second, that DM is last, and that PS and UM are perhaps marginally more 'efficient' than PM, RWB and RG.

It is certainly not a good idea to draw firm conclusions on the basis of these data, but they seem to show a consistent pattern of IA and PD, and perhaps PS and UM being generally important functions in the MONSTRAT mechanism as a whole, whereas DM, and perhaps RG and RWB are more specialized. This might imply, for instance, that the potential recipients of messages from DM, RG and RWB are perhaps more limited than those of messages from the other experts.

Message types in the simulation

A qualitative examination of the messages passed between experts in the simulation reveals that there were different *types*, despite our general restriction in the instructions to send only hypotheses. Because we did not use a formal internode language, only an informal description of these types is possible, but the categories seem reasonable given the data.

The most important message type, in terms of quantity and effect, was, of course, the type *hypothesis*, which was the normal output for most experts. Hypotheses were defined as messages that were suppositions, propositions

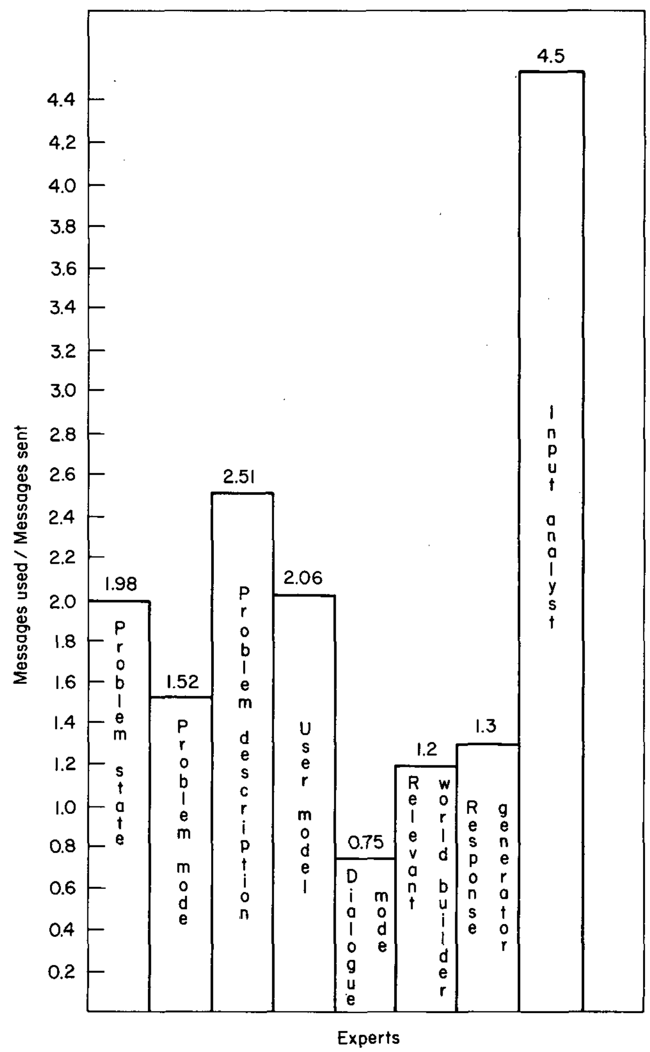

Figure 5. Ratio of messages used to messages sent by each expert, cumulated over all rounds ('efficiency')

assumed for the sake of argument, theories to be proved or disproved by reference to facts or hypotheses from other experts, or provisional explanations of anything, based on the expert's input and internal resources. Hypotheses always had degrees of certainty associated with them. An example of this message type, from the first message sent by the user model expert in simulation round 3, is:

> The user does not live alone — certainty value 5 [absolutely certain]

A second message type that often appeared was the *question*, which informally means an enquiry or interrogation of one expert by another, with the expectation that the enquirer be provided with an explication, supplement, judgement or decision. The complement to this message type is the *answer*, which is the response to a message of type question. In some cases, at least, this message type is equivalent to the type hypothesis. This message type pair was legal in our simulation only in the actor communication mode, although it occurred, in an indirect fashion, in the blackboard mode as well. By indirect, we mean that questions could be asked only by posting messages within the expert's own portion of the blackboard, which is perhaps relevant to the expert(s) of whom the question is posted, or who might be interested in the question. This means that it is not certain that the relevant expert actually finds the question, nor that the expert responds to it. Again, if an answer to a question is produced, it can only be posted on the blackboard, so that the interested expert must somehow discover which other expert has actually managed to answer the original question. This is obviously a cumbersome mode of communication in a system in which messages of type question and answer are necessary.

On the other hand, in the actor communication mode, messages of type question and answer arise naturally, and can be communicated easily, since this is a direct expert-to-expert communication mode. Such a communication mode favours these message types, since the question–answer cycle is a special process between two experts (nodes, functions) without any need for direct influence of external procedures at the time of the cycle. In the simulation itself, the human experts tended to disobey the rules of the blackboard communication mode, and to post questions (generally undirected to specific other experts) explicitly as such. An example of such a message pair, from round 3 of the simulation, between the problem mode and problem state experts, is given below:

> PM: What does the user have in mind with this question? Does s/he only want to know whether the comparison flats are appropriate, or does s/he want to know whether the rent increase is justified?
> PS: 1. The user needs information about the criteria year built, size and rent level, and their mutual interdependence for comparison flats in general — certainty level 5.
> PS: 2. In addition, an evaluation of the given comparison flats according to these criteria — certainty level 5.

Two other message types that occurred in the simulation were *facts* and *rules*. A fact in our environment is essentially a chunk of knowledge that can be used directly, without any process that generates or activates it. In other contexts, facts can be considered as data. Rules were interpreted as principles, standards, codes of regulation, and specifically in our environment as processes by means of which facts can be generated. Both of these message types occurred only in messages from the Relevant World Builder, which were abstracts from the Knowledge Resource, or in messages from the Response Generator, as interpretations of equivalent messages from the RWB. An example of the message type fact, also from round 3 is:

> RWB 1: The legal basis is paragraph 2, Interpretation of the Renter's Handbook, 195,200.

and of a combination of fact and rule, again from round 3:

> RG 2: Specify the data for the comparison flat according to type, fittings, condition, location and size. The comparison criteria for flats are type, size, fittings, condition and location (Paragraph 2(1)2, and Publication BM No. 200, numbers 1–4 and 6); with the condition that the flats must be specified by name and address.

These five message types were sufficient to categorize all of the messages passed in all five rounds of the simulation. However, it seems that for all message types it might be necessary to introduce some method for distinguishing between different *versions* of any one message. That is, for any particular expert, the same general message was often, over time, amended, extended, or in general modified to some different version. A common instance of this phenomenon was the modification of certainty factors, another was the progressive specification of an hypothesis.

Qualitative evaluation of communication patterns

The basic data for analysis of communication patterns in the simulation are the generation and use of messages by the experts. One way to represent these data is by a chart or graph of messages sent or used, over time. Appendix 2 is the time-line representation for communication in round 3 of the simulation, as an example of the data we have analysed. In these data, we have discerned two different patterns of interaction between the experts. To get more insight into the processes that took place, and to understand and describe these interactions, we have developed a formal model of message types and the message system. We present this model below, and then use it to understand the patterns that occurred in the simulation.

Message types as actions

We begin by reducing the general concept of message types to the single notion of actions rather than messages, just in order to be able to investigate their patterns of interaction. We do this by defining the set of all possible actions, by means of which we will then be able to define act/act schemata. In the following expressions, the outer braces of the formulae represent set definitions, where we assume I, J, K, L, M, N, X, Y are subsets of N, with N the set of unsigned integers.

We define:

$$\text{QUESTIONS } (Q) = \{Q_i \mid 0 \leq i \leq I\}$$

which is the finite set of all admissible questions with respect to the resources of the experts, as discussed in the section on experts and their roles;

$$\text{ANSWERS } (AN) = \{AN_j \mid 0 \leq j \leq J\}$$

which is the finite set of all admissible answers with regard to questions and with respect to the resources of the experts;

$$\text{HYPOTHESES } (H) = \{H_k \mid 0 \leq k \leq K\}$$

which is the finite set of all admissible hypotheses with respect to the resources of the experts;

$$\text{FACTS } (F) = \{F_l \mid 0 \leq l \leq L\}$$

which is the finite set of all admissible facts with respect to the resources of the experts;

$$\text{RULES } (R) = \{R_m \mid 0 \leq m \leq M\}$$

which is the finite set of all admissible rules with respect to the resources of the experts.

Then the finite set of all admissible actions with respect to the resources of the experts, can be defined by

$$\text{ACTIONS} = Q \cup AN \cup H \cup F \cup R$$

thus

$$\text{ACTIONS } (A) = \{A_x \mid 0 \leq x \leq X\}$$

In these definitions we always use the attributes finite and admissible. The first assumption is made because it is difficult to handle infinite sets in simulation environments. The second is introduced to make possible actions responsive to available resources.

Having defined actions, we define

$$\text{EXPERTS } (E) = \{E_n \mid 0 \leq n \leq N\}$$

to be the finite set of all specified experts in the MONSTRAT system, and

$$\text{EVENTS } (M) = A \times E$$

thus,

$$M = \{M_y \mid 0 \leq y \leq Y\}$$

to be the finite power set of actions and events.

Then, the message system for a given set of events can be defined as the pair

MESSAGE SYSTEM = (EVENTS, PRECEDENCE)

where

PRECEDENCE (P) is a partial order on sets of events, such that

$$P: 2^{(M)} \rightarrow 2^{(M)}$$

The precedence constraint thus defines partial or complete causality.

Patterns of interaction in act/act schemata

Now the patterns or flow of actions can be represented by an augmented graph theoretic model, which consists of the act/act schemata, each of which is a message system, or component of a message system. The basic elements are nodes and arcs, where the nodes represent the set of actions related to the set of experts, and the arcs represent the effect of a certain action on some other action, by one or more of the experts, that is, the precedence relation.

Thus, in the following figures, P is represented by the arrow '→'.

The simplest act/act schema (schema 1) is represented in Figure 6. We say that this is the most simple because there is a direct sequence of actions and consequent actions of some experts: an action of one expert initiates or causes the action of another expert, which causes the action of another expert, and so on. This schema 1 is thus a directed non-weighted graph with 1:1 relations where cycles are allowed and where each node contains additional information about the producer/user of the action. In this case, we can say that *only* the event set $\{E_i, A_i\}$ precedes the event set $\{E_{i+1}, A_{i+1}\}$.

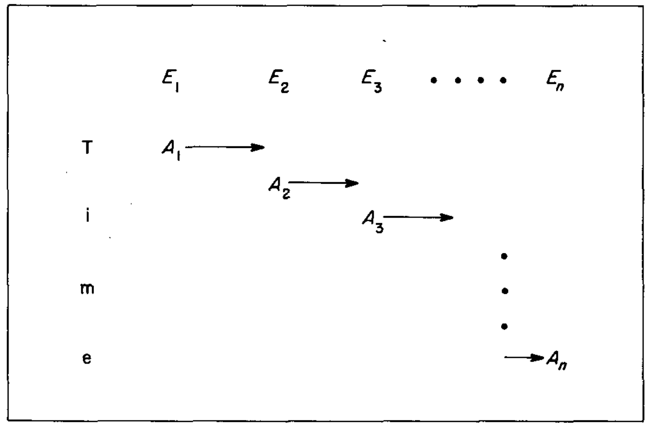

Figure 6. Act/act schema 1: simple nonrecursive sequence with 1:1 relation; E_i = expert i, A_j = action j

Figure 7, representing act/act schema 2, is a directed, non-weighted graph with cycles and information about the producer/user of the action, as the other schemata, but in this case N:1 relations occur. This reflects the situation in the system where several actions together are the preconditions for action being taken by a specific expert. In this case, the event set that precedes the event (E_5, A_5) is $\{(E_3, A_3), (E_4, A_4)\}$, which yields the N:1 (in this case 2:1) relation.

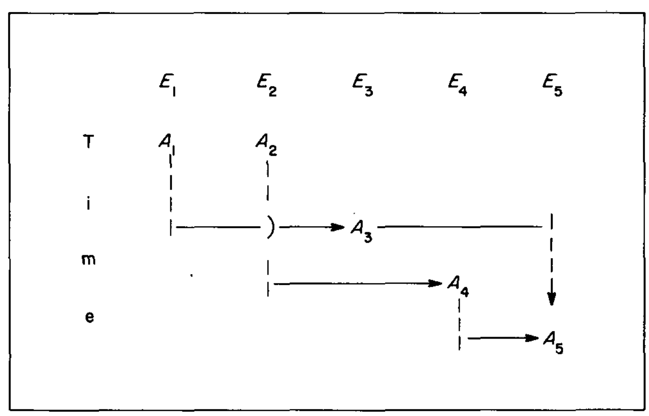

Figure 7. Act/act schema 2: simple nonrecursive sequence with N:1 relation

Figure 8, representing act/act schema 3, is a graph with all of the characteristics of the others, except that in this case 1:M relations occur. This is the situation, for instance, when the action of one expert is the cause of (precedes) the actions of several other experts. In this case, the event (E_2, A_2) precedes the event set $\{(E_3, A_3), (E_4, A_4), (E_5, A_5)\}$, thus the 1:$M$ relation, $M = 3$.

Act/act schemata 1–3 are all concerned with situations in which the action of an expert takes place without regard

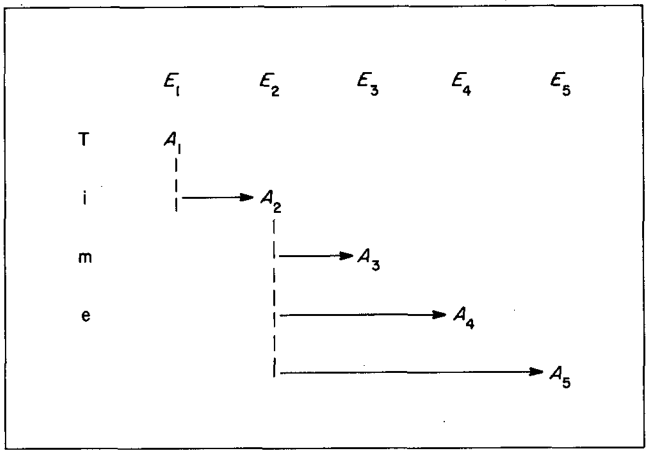

Figure 8. Act/act schema 3: simple nonrecursive sequence with 1:M relation

to any subsequent actions of that expert. But for some action types, and in some specific situations, it is possible that the action of an expert is taken *just so that* that expert can take some subsequent action. If the action, for instance, is of the type question, this will almost certainly be the case. For such situations, we can define act/act schema 4 (Figure 9), the directed, cyclic, non-weighted graph, as in the other schemata, but with recursion. The simplest such schema, recursion without nesting, involves only three events with two experts. Considering, for instance, from Figure 9 the three events (E_1, A_1), (E_2, A_6) and (E_1, A_7) in isolation from the rest of the graph, we obtain the sequence

$$(E_1, A_1)\ [M_1] \rightarrow (E_2, A_6)\ [M_6] \rightarrow (E_1, A_7)\ [M_7]$$

In this case of recursion the 1:1 relation holds. Introducing one level of nesting, however, immediately generalizes the situation to the N:M relation, N and $M > 1$. Such nesting occurs whenever the recursive sequence requires some intermediary event. In Figure 9, for instance, M_1 precedes M_6, which precedes M_7, in that M_6 is the answer to the question M_1, which allows the event M_7. But, (E_3, A_5) $[M_5]$ also precedes M_6, in that it is the answer to question (E_2, A_2) $[M_2]$, and is thus a necessary precursor to event

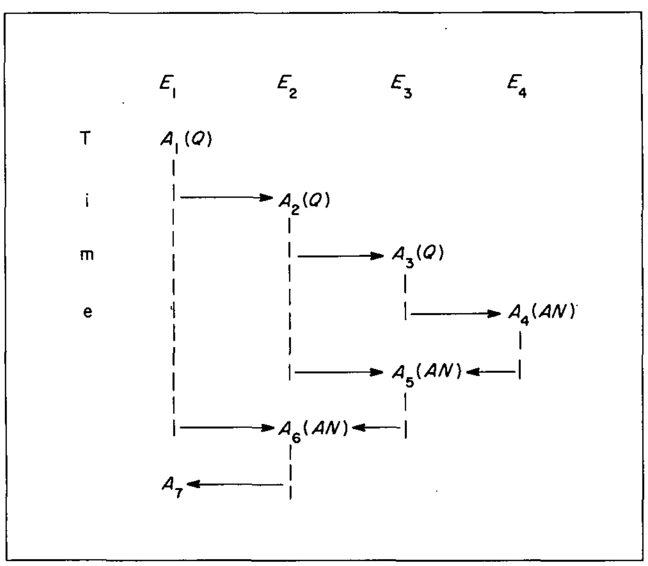

Figure 9. Act/act schema 4: generalized, nested recursion with N:M relations

M_6. Therefore, the event set $\{M_1, M_5\}$ precedes M_6, the 2:1 relation. But, since M_1 precedes both M_2 and M_6, the 1:M relation also holds. Thus, any nesting of recursive sequences immediately results in the N:M relation. Although such recursive sequences can be nested to arbitrary depth, only the first event triple can stand in 1:1 relation, if considered in isolation from the rest of the sequence, all others necessarily being at least 2:2. One way to consider the property of recursion, as here defined, is that the action of some particular expert is necessary to (precedes) a subsequent action of that expert. But this order of precedence is always expressed by at least one intermediary event.

By means of the act/act schemata defined above, we can now describe, simply by superposition of the defined basic elements, all those patterns of communication that occurred in the simulation, as represented in the example message/time chart of Appendix 2.

Description of interactions in the simulation

If we consider, for example, the message history of simulation round 3 (Appendix 2), we can see that the simple act/act schemata described in the previous section do indeed occur, and that they seem to be sufficient to describe all of the patterns of interaction. Furthermore, by use of this formalism we can say a few things about the interactions that are not otherwise immediately apparent.

In the chart of Appendix 2, the conventions used are as follows:

- each column represents the activity of the designated expert during the simulation,
- each event in the simulation is designated by the acronym for that expert and the serial number of the expert's action, e.g., RG1 means the first action (message) produced by the response generator,
- '>' following an event specification means that that message was produced as output by that column's expert at that time,
- '>' preceding an event specification means that that column's expert noted, at that time, the specified message as input that generated some subsequent action,
- once an event has been noted as input by an expert, no subsequent mentions of that event are noted.

One striking characteristic of round 3 is that there are no examples of schema 1 that are longer than one step; that is, there are no cases in which an event had only one precedent, which preceded only it, and in which that event preceded only one subsequent event, which itself preceded only one further event. Thus, the simple, single causal chain,

$$M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow$$

does not exist in this simulation round, and occurred in other rounds only between the response generator and dialogue mode experts. This leads to the conclusion that most experts depend upon input from a number of other experts in order to compute their functions, and that few experts' functions are relevant only to one other expert. It also appears to demonstrate that the functions we have defined do not lead to extended simple causal chains, but rather that such simple chains are confined to local environments (pairs or at most triples of experts).

Similarly, we notice that, from the point of view of almost any event in round 3, the N:1 relation holds, with the exception of actions of the input analyst, which are usually sufficient to be the sole precedent of another event. In the case of the IA, however, we also note that the $1{:}M$ relation, $M > 1$, holds consistently. For other experts this is not so consistently the case, there being a number of instances of single use of a message, and quite a few examples of messages that are not used by any other expert at all. Schema 4 appears several times in this round, but only, as far as can be judged at this level of analysis, without nesting. However, it rarely takes place as the 1:1 model, usually being rather $N{:}1$, the initiator of the sequence being only one of the precursors to the eventual response. Considering then only one or two step sequences, schema 1 appears only rarely, and usually with specific pairs of experts, schema 2 is the norm, schema 3 appears fairly often, and schema 4 at times, but without explicit nesting. Many events are participants in superpositions of schemas 2 and 3, leading to the large number of instances of the $N{:}M$ relation among three events (N & $M > 1$).

By looking at individual experts, and at the temporal sequences of events, analysis by schemata can tell us some more about the interaction patterns that occurred. For instance, we can see that some experts tended toward different schemata as their 'normal' type. Although our results are not strong enough in this simulation to characterize each expert in this way, some can be so identified (e.g., OG with schema 1). Strong characterization of particular schemata with particular experts could lead to communication architectures that take account of these specific characteristics.

We also note that the initial activity of the system involves a small group of experts, triggered by an hypothesis from the IA, who communicate intensively with one another. Eventually, this analytic activity, associated in all rounds with UM, PD, PS, PM, and to some extent with DM, reaches the synthetic experts, RWB and RG, whose activity is with one another and with the other experts. This synthetic activity appears to reinstigate, but to a lesser degree, the processing of the analytic experts. The OG begins to function only after events from the synthetic experts have taken place. The OG's activity may lead to new hypotheses from the IA; this triggers a second wave of activity among the experts, following the same general pattern, but with less complexity. Throughout, the controlling expert, when present, maintains a strictly monitoring and repressive function, preventing direct messages to the OG from experts other than RG and DM. Round 3 exemplifies this pattern, which occurred in all of the simulation rounds, of initial analytic complexity, followed by synthesis and further, simpler analysis, followed by output, and when possible by a second or further such sequences reinitiated by the IA. This general pattern is illustrated in Figure 10. Although this pattern tends to become simpler over time, simplicity measured by the numbers of experts actively involved in interaction with the others, and by the degree of the relations for schemas 2 and 3 and the extent of their overlap, it is also the case that once an expert begins activity, it continues until the end of the entire interaction.

Thus, analysis of event interaction by use of the act/act schemata leads to a fairly regular pattern, which indicates a particular temporal sequencing of actions by specific experts, which shows a progressive decrease of complexity of interaction over time, and which indicates that some pairs or triples of experts (notably RG, OG and DM)

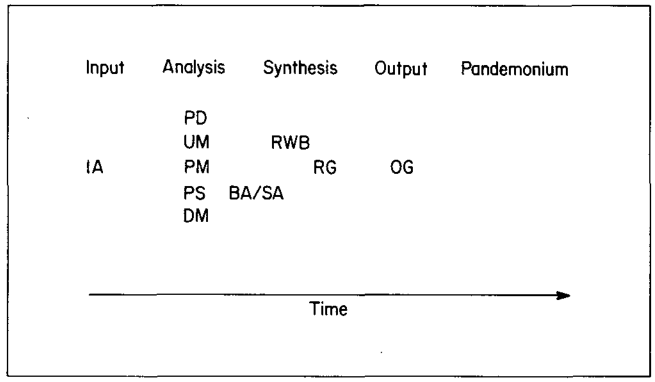

Figure 10. Order of initiation of activity of experts in MONSTRAT simulation

evolve as distinct societies. Furthermore, this general pattern appears to be re-established, albeit in less complex form, with each input to the system as a whole. On a strictly intuitive basis, viewing the general patterns of each simulation round, communication mode seemed not to affect this pattern, but imposition of control appeared to result in the interaction becoming simpler more quickly than in the rounds without control. We hesitate to make too much of this last result, however, because of the influence of the learning effect on the conduct of the simulation.

Summary of results

Referring to the goals of the simulation, our results discussed above appear to have answered our initial questions as follows.

- The overall MONSTRAT model seems to work. That is, each round of the simulation, except round 4, led to a reasonable response to the initial query, and the reasoning and response in rounds 3 and 5 were similar to those of the advisors in the real interactions from which the problems were derived. Similarly, the involvement and interaction of experts, as seen in both quantitative and qualitative evaluations of the simulation, indicate that the functions we have postulated, with the possible exception of DM, are necessary and probably sufficient for such an intermediary mechanism, and that they do interact in complex, non-linear and non-hierarchic ways. Finally, message types other than hypothesis appeared regularly, and may therefore be necessary in such a system.
- Blackboard communication seemed to have a marginal advantage over actor communication by the quantitative measures, but in other respects these two modes appeared to perform equally well.
- The imposition of some sort of control had a positive effect in terms of the quantitative measures, and may also have led to earlier achievement of simple interaction patterns than in the uncontrolled rounds. The combination of blackboard communication with control seemed, on the quantitative measures, to be especially effective.
- Groupings of experts, in terms of interaction with one another, and in terms of times of their actions, were demonstrated. Some experts were shown to be more general in their utility to the remainder than others, by both quantitative and interactional analysis, and a

general pattern for interaction in the system was evident.

Given these results, the questions remain: what do they imply for the understanding and development of this type of model? and, what do they imply for the architecture of systems based upon it?

CONCLUSIONS

System architecture

Communication mode

Considering first the issue of communication structure, the results tend toward blackboard rather than actor mode of communication, but are not wholly convincing on this point. There are, however, some other factors that we have not yet considered that might be helpful in drawing conclusions about the appropriate communication mode.

When thinking about MONSTRAT from a computing, rather than only logical, point of view, the number of distinct communication paths between experts becomes important. There seems to be some inherent structural efficiency of the blackboard mode in this respect. That is, for a fully interconnected system in the actor mode, each node must have two links (sending and receiving) with every other. Thus, for n nodes, there will be $n(n-1)$ links (if directionality is considered, otherwise half that number). On the other hand, full interconnection via the blackboard mode, in which the blackboard plays an intermediary role, implies in the bidirectional case only $2n$ links, otherwise only n . This assumes, of course, that reading the blackboard is a *single* communicative act, no matter which, or how many messages on the blackboard are eventually used by the reading expert. The problem with this assumption is that reading the blackboard is a complex activity, which imposes its own overheads, which may be of the same order as those associated with the fully interconnected model. From this point of view, for the case of full interconnection, or a high degree of interconnection, it appears that the blackboard mode has the advantage, but for a system in which interconnection is minimal, the actor mode may be structurally more efficient.

Another factor to consider is decision making procedure in the system; that is, how it is decided that a message is relevant to a particular expert. From this point of view, the advantage to the blackboard mode is that it is possible to monitor many messages *at the blackboard*; this is perhaps a suitable role for a controlling function, which would remove this decision burden from both sending and using expert. The disadvantage to the blackboard mode is the potential complexity of the monitoring function(s). The disadvantage to the actor mode in this factor is that it requires that decisions be made either by sender or recipient as to the relevance of a message. In the former case this means that the sending experts must have some knowledge of the functions and requirements of all the other experts; in the latter, it means that the experts must continually be monitoring input, as well as doing their computing. The advantage to the actor mode is that it allows messages to be delivered directly, without a mediating function.

These two factors indicate that blackboard mode is probably more efficient in circumstances in which many experts are likely to use the message of another, but in which there is some doubt as to its relevance, and that the actor mode is likely to be more efficient in cases where there are a limited number of potential recipients, and in which it is reasonably certain that the message is relevant. These characteristics may, indeed, explain some of the results of the simulation, and offer guidance on overall system design.

Both quantitative and qualitative results indicated that some experts are generally important, others only in specific circumstances or for specific other experts. It appears, furthermore, that the bulk of the communication activity is centred on those generally important experts, but that its conclusion is dependent upon the specific ones. This may explain why, even though round 3, blackboard with control, was individually the best performer, overall there was little to choose between blackboard and actor modes. Perhaps the response to this is to tailor the architecture to take account of the differences between experts. Given the results, an optimum system design for communication would be one in which the basic pattern is blackboard, specifically for communication among the analytic experts and between them and the synthetic, with actor mode for communication among the specialized small societies of experts (e.g., DM and RG with OG, RG and RWB with one another).

Control

The issue of control, at least as we interpreted it, seems to have been settled by the results of the simulation. That is, with the exception of round 4, imposition of control on communicative activity always had a positive effect. Unfortunately, round 4 was a rather large exception, since it was the only round that failed to produce a result. There seem to be two important factors in this result. The first is that by this round of the simulation, the people who were the experts had become quite confident in their role playing, perhaps even overconfident. There is some evidence that they had begun consistently to overstep their assigned functions, and they certainly had begun to send more messages of the type question than in previous rounds. The second factor was the level of control, which was quite strict and also quite strongly centralized, requiring a decision by the controlling expert on whether any question should be honoured, and if so, by whom. The combination of these two factors seems to have led to the clogging up of the simulation in the SA bottleneck. Because we responded to the situation by modifying both factors for round 5, it is not completely clear which was most important. Given the results of the other rounds, however, it seems fair to conclude that such strict control is not necessary, especially in the blackboard mode, but that the excess activity of the humans in the simulation masked what would probably have been a positive influence. We conclude, therefore, considering also the arguments about the monitoring function above, that moderate control of communication activity is necessary in such a system.

The other control issue, that is, how is it decided when the system has reached a successful conclusion, has been dodged, or rather, the answer has been assumed. The simulation, as it ran, distributed processing control among all the experts, allowing, in effect, the user, standing outside the mechanism, to make the decision as to when to halt the system. Although there is some vestige of such control in our BA and SA, and in the direct link between RG and OG, by and large it appears safe to say that distributed control worked in the simulation, and that it ought to work in any implementation of it.

Feedback

Embedded in the general issues of communication and control are the problems of feedback, of reliability of messages, and of action on incomplete data. The question of feedback is a problem primarily because of the different message types that we found. In this, we refer to the issue of when, and under what conditions, receipt of a particular type of message must be acknowledged, triggering the generation of a second message. If, for instance, one expert generates a question, that expert may need to know whether the message reached the chosen destination, whether the recipient is willing and/or able to generate an answer, whether the question has been referred to yet another expert, what delays might be expected, and so on, then a *formal* feedback mechanism would be appropriate. If the original generator of the message does not receive such formal feedback, then it will be a difficult task for it to decide, for instance, whether to attempt to generate an hypothesis based on incomplete information.

Such formal feedback can be implemented easily in the actor communication mode, but is rather more complex in the blackboard mode. The latter allows for posting messages in the expert's own area, and reading messages in predefined relevant areas. If these messages are hypotheses, there is no problem. But if they are questions or requests, then it is necessary to build in another message type, say acknowledgement, in order to take account of the need for formal feedback. This might, of course, be accomplished, as in our simulation, by a special monitoring expert that controls questioning activity. In any event, the formal feedback issue implies additional constraints on system architecture beyond those evident in the bare results of the simulation.

Reliability of messages

There is also a problem with respect to *substantive* feedback. Generally it must be assumed, as it was demonstrated in some cases in the simulation, that an expert can compute totally incorrect hypotheses, without the possibility of any other expert stopping this process. This follows from the functionally defined distribution of tasks: each expert is a specialist for that specific function, and no other expert can replace it in performing that function. In our simulation, this effect was partially overcome because the experts were human beings, each of whom had greater 'resources' (knowledge, intelligence) than the experts they were simulating. They thus tended to moderate performance of their own functions according to their knowledge of the situation as a whole. But in the wholly mechanized system, we cannot expect this type of effect, which implies that there might be a need for some sort of substantive feedback which corrects aberrant processing.

The most obvious means is the natural one (in such a system) of correction by the weight of conflicting evidence. The problem here is that if the initial hypotheses are very incorrect, and then are subsequently used as basic input for processing by other experts, who then also produce incorrect hypotheses, positive feedback could result, in which finally nothing is working properly. Fortunately, the results of the simulation indicate that, even when one or several experts have submitted evidently false hypotheses, the system as a whole has not been biased by them. Indeed, in some cases, such experts eventually corrected themselves. But given the nature of the simulation, we must say that this is still an open issue, and that some sort of more rigorous substantive feedback will probably be necessary.

Such feedback might be accomplished by some sort of context-sensitive content analysis of messages, that is, some means of comparing the contents of different messages to insure that they 'fit together', or of checking the consistency of a single message. We might find some evidence for such methods in the protocols of the simulation, but have not yet carried out such analysis. We can say that content-related feedback and interaction among the experts is still only partially understood, and so is only capable of being partially modelled and implemented. It appears that it may be possible to build in mechanisms that allow individual experts to adapt and correct their processing based on their own resources. But it is very much an open question how such judgement could be performed by any other expert, even a system expert, since no expert could have a complete model of any other.

Weak information

Finally, the issue of *weak* information has some architectural implications. All of the experts in the simulation, at one time or another, experienced difficulties in processing for various of the following reasons:

- Appropriate input could not be expected from all potentially relevant experts.
- Output from a specific expert is vague or incomplete.
- Output from a potentially relevant expert is delayed.
- Output expected from one expert is provided by one or more other experts.

In such cases, it seems appropriate to have a mechanism, initiated by the experts themselves, or by a system expert, for modifying the importance of specific messages by associating weights with them. Such a mechanism could, of course, be subject to substantial problems of the sort discussed in the section on feedback, above. But, if such a mechanism were triggered only on the basis of *formal* parameters, then it would probably not be too difficult to *rank* the output appropriately (using, say, certainty values), and to define a level at which to *cut* the output stream of an expert, or the input that an expert heeds. Such a *rank/cut* mechanism seems necessary in any system of the type we have considered, and indeed appeared to have been used, at least informally, by the human experts in our simulation.

Summary

In summary, our results appear to indicate the following for architecture of a MONSTRAT-type system. It should be mixed communication mode, primarily blackboard for analytic experts, and partially actor for synthetic experts. It should incorporate different message types and procedures for dealing with such types in differential ways. It needs some means for enhancing processing when necessary, and some mechanisms for stabilizing processing. It also needs some function for controlling some aspects of interaction among the experts: this was the SA or BA in our simulation. A number of the procedures and mechanisms that seem necessary could be provided by such a controlling function as a natural part of its activities, while others could well be distributed among the individual experts. Thus, overall, we opt for a highly mixed architecture, with functions and mechanisms built into it that are not substantive, as are those associated with our original experts, but which are formally necessary for efficient operation.

The MONSTRAT model

The results of the simulation have indicated that our overall model seems generally appropriate to the situation. The experts all performed, most of them were used by others, there were no obviously lacking functions, the general form of interaction was non-linear and non-hierarchic, and appropriate responses to the user were achieved. These were factors in the model that we hoped to test via the simulation. We have also learned some new things about this model that were not so readily apparent before the simulation. First, we have managed to discover that there are some natural societies of experts in the functions that we have stipulated, in terms of how they communicate with one another and when they communicate with one another. Second, we have found that there is some temporal ordering of functions, at least as far as when they are able to start processing in earnest. Furthermore, we have found that the complexity of interaction in the system decreases over time, as results of processing become known, in a regular way. All of these results have some strong implications for the MONSTRAT model in general and how it might be used, and for what further research should be done.

We can now see that the model can be successfully partitioned into particular groups. This tells us something not only about potential system architectures, but also about how an intermediary mechanism *ought* to act, when dealing with a human user. The emphasis on initial analysis, and quick identification of types within the individual functions as a spur to processing by all experts, implies that interaction with the user, to obtain information relevant to the analytic functions, is important at the beginning of the human–machine interaction, not only for those experts, but also for the synthetic experts, who will rely on the earlier hypotheses. This has implications not only for human–machine interaction, but also for human–human information interaction.

These results also appear to indicate where further development of the model needs to take place. It is quite obvious that the specification of the functions is not yet precise enough for any reasonable system. What we do know now is something about the relationships between the various functions, which gives us some clues as to what kinds of research to undertake in order to achieve appropriate specification. For instance, specification of any function would seem to depend very strongly on the uses to which other functions put it. We now have a better idea of who uses whom, so we can, for instance, study the relevant aspects of such interactions empirically, limiting them to the societies of functions that we have identified. Communication among the experts is evidently much more complex than we had assumed, with a variety of stabilizing and controlling mechanisms. These need further to be identified and described, and if possible modelled. Each function now appears to be itself a small distributed system, with its own knowledge sources operating on its input. We need now to look at the types of knowledge used, and especially at the stereotypes that drive the separate functions. Some experts seem to have been much more important than others. It is necessary to see if this is really the case, if some functions override all others in eventual system response, or if perhaps this has happened because too large a responsibility has been devolved onto such functions.

There are many more areas of potential development. What is more interesting than listing them here is to note that they seem to lead to ways to consider the problem of the information interaction in general. In this sense, we can see even more clearly that the MONSTRAT model appears to be not only an appropriate tool for the design of appropriate information provision mechanisms, but also a strong structure for understanding and studying how information systems, whether human–human or human–machine work. This simulation, we think, provides a good example of this strength of the distributed expert approach to information provision.

ACKNOWLEDGEMENT

This study was supported by the German Federal Ministry for Research and Technology under Grant No IDA 0160–4.

REFERENCES

1 **Belkin, N J, Seeger, T and Wersig, G** 'Distributed expert problem treatment as a model for information system analysis and design' *J. Info. Sci.* Vol 5 (1983) pp 153–167

2 **Chandrasekaran, B** 'Special issue on natural and social metaphors for distributed problem solving' *IEEE Trans. Syst., Man & Cybern.* Vol SCM–11 (1981) pp 1–96

3 **Davis, R** 'Report on the workshop on distributed AI' *SIGART Newsletter* (1981) pp 42–52

4 **Smith, R G** *A Framework for Distributed Problem Solving* University of Michigan Press, USA (1981)

5 **Schindler, S and Spaniol, O** *Kommunikation in verteilten Systemen – Anwendungen und Betrieb* GI: NTG-Fachtagung, Federal Republic of Germany (1983)

6 **Smith, R G and Davis, R** 'Frameworks for cooperation in distributed problem solving' *IEEE Trans. Syst., Man & Cybern.* Vol SCM–11 (1981) pp 61–69

7 **Wesson, R, Hayes-Roth, F, Burge, J W, Stasz, C and Sunshine, C A** 'Network structures for distributed situation assessment' *IEEE Trans. Syst., Man & Cybern.* Vol SCM–11 (1981) pp 5–23

8 **Erman, L D and Lesser, V R** 'The HEARSAY-II system: a tutorial' in *Trends in Speech Recognition* **W A Lea (Ed)** Prentice-Hall, USA (1979) pp 361–381

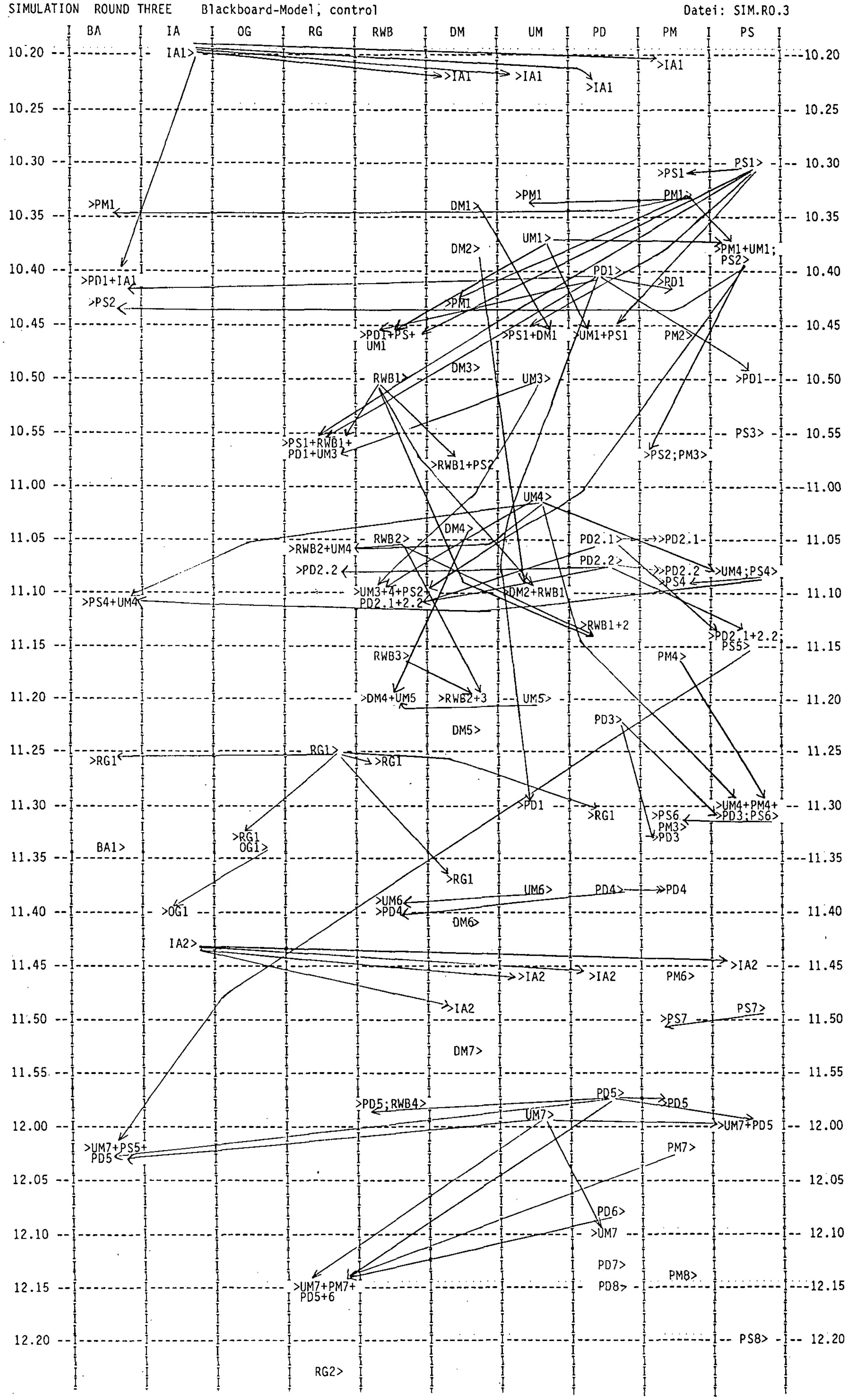

Figure 11. Blackboard model, control

9 **Lawrence, P R and Lorsch, J W** *Organization and Environment* Harvard University Press, USA (1967)
10 **Uhr, L** *Pattern Recognition, Learning and Thought* Prentice-Hall, USA (1973)
11 **Hewitt, C** 'Viewing control structures as patterns of passing messages' *Artif. Intell. (Netherlands)* Vol 8 (1977) pp 323–364
12 **Brooks, H M and Belkin, N J** 'Using discourse analysis for the design of information retrieval interaction mechanisms' *Research and Development in Information Systems. Proceedings Sixth Annual International ACM SIGIR Conference* ACM, USA (1983) pp 31–47
13 **Wersig, G and Hennings, R-D** 'The intellectual architecture of information systems' *IRFIS 5. Proceedings Fifth International Research Forum in Information Science* North Holland, The Netherlands (in press)
14 **Belkin, N J and Windel, G** 'Using MONSTRAT for the analysis of information interaction' *IRFIS 5. Proceedings Fifth International Research Forum in Information Science* North Holland, The Netherlands (in press)

APPENDIX 1: SPECIFICATION OF THE PROBLEM DESCRIPTION EXPERT

Problem description

The main task of this expert is the analysis of the user's problem according to the following subfunctions:

1. Problem type, e.g., procedural, decision making, cooking, etc.
2. Problem structure, e.g., well or ill structured, well or poorly understood, few or many concepts, with or without connections, etc.
3. Problem topic.
4. Problem environment, e.g., restrictions on the topic in connection with time, extent of desired answer, formulation of question, etc.
5. Problem context, that is, the relationship between the specific problem, as it is understood at the moment, and the user's life situation: why the question has been posed, and why the problem is a problem.

Input*

1. Input analyst, user model, relevant world builder, problem description (4, 5)
2. Input analyst, problem description (1), problem state
3. Problem description (2), input analyst, user model, relevant world builder
4. Input analyst, user model, problem state, problem description (1)
5. Input analyst, user model, problem description (3)

*Numbers following, and in each further subsection, refer to subfunctions specified above.

Output

1. Hypotheses about problem type
2. Hypotheses about problem structure
3. Hypotheses about problem topic
4. Hypotheses about problem environment
5. Hypotheses about problem context

Users

1. Relevant world builder, response generator, problem mode, dialogue mode, problem description (2, 4)
2. Relevant world builder, problem description (3)
3. Relevant world builder, response generator, problem description (5)
4. Relevant world builder, response generator, problem description (1)
5. Response generator, problem mode, user model, problem description (1)

Resources

1. Sets of problem types
2. Sets of problem structures
3. Sets of problem topics
4. Sets of environment types
5. Sets of context types

Procedures

Analysis:

1–5. According to input, resources and rules for correlating the two, for each subfunction.

Evaluation:

1–5. Assignment of certainty values to each of the hypotheses generated for each subfunction, according to the given scale.

APPENDIX 2. TIME-LINE REPRESENTATION OF MESSAGE PASSING ACTIVITY IN SIMULATION ROUND 3

In Figure 11, the conventions used are as follows:

- Each column represents the activity of the designated expert during the simulation.
- Each event in the simulation is designated by the acronym for that expert and the serial number of the expert's action, e.g., RG1 means the first action (message) produced by the response generator.
- '>' following an event specification means that that message was produced as output by that column's expert at that time.
- '>' preceding an event specification means that that column's expert noted, at that time, the specified message as input that generated some subsequent action.
- Once an event has been noted as input by an expert, no subsequent mentions of that event are noted.