An Approach to User Modeling in Decision Support Systems

PIOTR J. GMYTRASIEWICZ
Department of Computer Science and Engineering
University of Texas at Arlington, Arlington, TX 76019-0015
piotr@cse.uta.edu

(Received ..... ; Accepted in final form .....)

Abstract.
Drawing on our work in the area of distributed artificial intelligence, we put forth a framework for modeling a human user interacting with a knowledge-based system. We assume that the human user is situated in some decision making setting, and view the computer system as taking an active role in supporting the user's decision making and problem solving activities. The model the system has of the decision making situation and of the user can be applied to determine what the system should do, both in terms of the system's physical action, if such is possible, as well as in terms of the information that should be transmitted to the user. An important part of the user's model is the model that the user may have of the system itself, and, further, how the user may think it is being modeled by the system. Our framework, the Recursive Modeling Method (RMM), explicitly represents this nesting of models, and lets the system to coordinate with the expected actions of the human user, and to rationally choose an optimal form of communication.

1. Introduction

We investigate the interaction of a human user, assumed to be situated in some decision making setting, with a knowledge based system. In this context, we would like to view the system as taking an active role in supporting the user's problem solving and decision making activities. The central questions we would like to be able to answer are:

1. What information, chosen from a likely large number potential candidates, should the system communicate to the user?
2. What action, if such is possible on the part of the system, should be executed by the system to coordinate with the anticipated action(s) of the user?

Our work builds on a large amount of related research in human-computer interaction and artificial intelligence. The fact that modeling the user's cognitive processes, such as planning and decision making, is crucial for effective human-computer interaction has been widely recognized in the HCI literature (see, for example GOMS model Card et al. (1983) and its extensions Arend (1991) Kieras (1988)). Others have pointed out that the user may

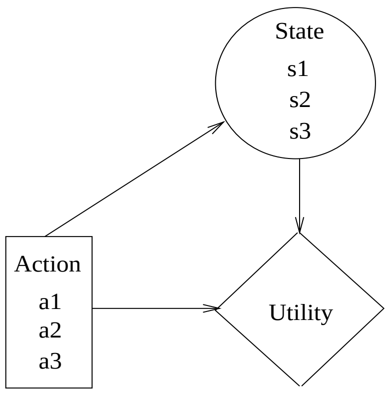

Figure 1. A Simplified Influence Diagram.

also have the model of the system Fischer (1991), and the nested character of the models within models has also been considered (see Ballim and Wilks (1991), Cohen and Levesque (1990b), Cohen and Levesque (1990a), Jameson (1989), Perrault (1990), Wilks et al. (1991) and the references therein).

As in the above work, the central issue in our research is how the system is to represent and utilize the user's model, i.e., what is known about the user's decision making situation, including the user's preferences, user's abilities for action, as well as user's beliefs about the environment and its future development. Further, what is the representation to be used for the user's model of the system, user's model of the system's model of the user, if appropriate, and so on? What role do these models play in determining the preferred action and communicative behavior between the system and the human?

While building on the existing work above, our approach additionally concentrates on the element of rational choice, formalized extensively in the field of decision theory, also pursued, for example, by Poh and Horvitz (1996), and by Jameson et al. (1995). There, the rational choice is defined as one that maximizes certain performance criterion, called the expected utility, given the system's state of knowledge, and given the goals that the system strives to achieve. In other words, we view the system's active role in supporting the user's decision making as a form of optimization, during which the information to be shared with the user is prioritized, and the highest priority message is transmitted. Similarly, physical actions, if such are possible, are prioritized, and the best one is executed.

This research was supported, in part, by Office of Naval Research grant N00014-95-1-0775, and by a research initiation grant from the Computer Science and Engineering Department at the University of Texas at Arlington.

In previous approaches Jameson et al. (1995), Poh and Horvitz (1996) Bayesian networks Pearl (1988), and their extensions – influence diagrams **?** (**?**), were used as models of the human users' decision-making situations. A simplified influence diagram in Figure 1 depicts how the possible actions of an agent (or user), may, possibly probabilistically, influence the state of the environment, and how the environment and the action impact on the utility. The wide use of influence diagrams arises form, first, their representational power; they represent the agent's abilities by listing the possible actions, agent's beliefs about the world and the way each of the available actions impacts the world state, and the agent's preferences. Second, influence diagrams represent uncertainty present in the agent's knowledge, and manage this uncertainty in a computational manner.

In our work, we chose the representation in the form of payoff matrices, which are closely related to influence diagrams – information contained in an influence diagram can be summarized by a unique payoff matrix. For example, the information contained in Figure 1 can be summarized in a payoff matrix below:

$$\begin{array}{ccccc} & & & \text{State} & \\ & & s_1 & s_2 & s_3 \\ & a_1 & u_1^1 & u_1^2 & u_1^3 \\ \text{Actions} & a_2 & u_2^1 & u_2^2 & u_2^3 \\ & a_3 & u_3^1 & u_3^2 & u_3^3 \end{array}$$

where u_i^j is the value of the utility node for the $i-th$ action and the $j-th$ state of the world. Thus, the payoff matrix, which summarizes the information in the influence diagram by hiding the details of relationships among state variables, retains the dependence among the agent's actions, the state, and the utility. This paper reports on the way this representation can model users' decision-making while they interact with a computer system, when the state of the environment is dependent on the actions of both system and the user, and model the system's decision-making situation while it interacts with the user. We then go on to describe the ways these representations can be used to select the system's appropriate communicative and physical action.

2. An Example

Imagine a computer system interacting with a human user during a text editing session. The user is using one of many editing programs available to modify and spell check a Latex file. At some point, the system notices that a crash is imminent, and that the failure is unlikely to be anticipated by the user occupied with other tasks. Given the situation at hand, we will first consider the choice between immediate actions the system may consider:

Should the file be immediately saved under the uninformative name "new-file", or not? The answer to this question is not surprising: It depends. Mainly, it depends on what the user is expected to do. For example, if the user is expected to save the file herself under a desired name, then not saving it under the name "new-file" is preferred. However, if it happened that the user will continue spell checking and not manage to save the file in time, then saving it under "new-file" while not the best, may be preferred never-less.

In the context of examples of interactions similar to the one above, we define the concept of *computer-human coordination* precisely to mean the system's attempt to find the appropriate choice of action, given the anticipated action of the user. The decision making situation the system is facing can be summarized by a payoff matrix depicted below.

		User's Actions		
		a_1^2	a_2^2	a_3^2
Alternatives	a_1^1	-2	1	0
of the system	a_2^1	-3	2	-1

Above, the systems alternatives of saving the file as "new-file" or not are denoted as a_1^1 and a_2^1, respectively. The user's alternative behaviors, to keep on spell checking, saving file with a desired name, or doing something else, are denoted as a_1^2, a_2^2 and a_3^2, respectively. The payoffs corresponding to each of the six combinations of the system/user actions were assigned based on the perceived desirability of the positive and negative effects of the actions. It is important to note that the payoffs above represent the system's preferences, that is, they represent the desirability of the situation resulting from the joint system and user's actions from the system's point of view. Frequently, of course, these preferences will simply coincide with the preferences of the human user, and the interaction will be perfectly cooperative. In general, however, it is not necessary that the system's and the user's preferences coincide, and our representation gives us the flexibility to represent that.

Thus, in order to coordinate, the system should predict the user's action. For this purpose, the model of the user's decision-making situation can be used. One model could correspond to the user's not knowing of the approaching system crash, and thus not considering the alternative of saving the file. The payoff matrix depicting this situation is below:

		System's Actions
		a_2^1
User's	a_1^2	2
Alternatives	a_3^2	-1

On the other hand, it may be possible that the user is, in fact, aware of the approaching system crash. In this case, the user's decision making would

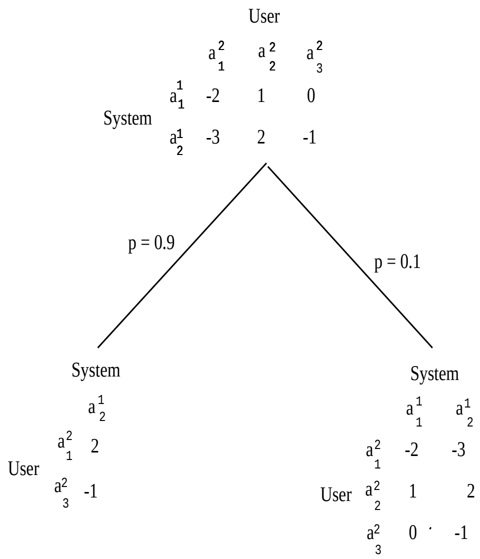

Figure 2. Recursive Model Structure in the Example of Human-Machine Coordination.

clearly include the option of saving the file in question. In this example we assumed that the system's and the user's preferences coincide, and the situation can be modeled as below:

		System's Actions	
		a_1^1	a_2^1
	a_1^2	-2	-3
User's	a_2^2	1	2
Alternatives	a_3^2	0	-1

We found it convenient to depict the payoff matrices arrived so far as a hierarchy in Figure 2, which we called the recursive model structure. The top matrix is the system's decision making situation, and below are two alternative models of the user, differing in one respect - in the model on the left the user does not anticipate the imminent crash, while the right model includes this possibility. The presence of both of these alternative models indicates that both are possible, as far as the system is concerned. In other words, the system is uncertain whether the user knows of the imminent crash or not, and this uncertainty is represented as two possible models of the user's decision making situation, each with an associated probability.

The alternative models of the user above are assigned probabilities, reflecting the likelihood that each of them is accurate. In our example we assumed that the user is unlikely to anticipate the imminent crash, and the corresponding model has a lower probability. The fact that the system's uncertainty as to the correctness of the user's model is explicitly represented probabilistically is, we feel, one of the strengths of our approach. It has the advantage of being based on a well developed, and well understood, formalism. Further, it opens the door for a principled approach to modifying the representation of uncertainty as to which model is correct in the face of new arriving information, based on the Bayes' theorem. Concerns frequently brought up in relation to probabilistic representation of such uncertainties are, usually, either variations on the issue of "Where do these numbers come from?", or observations that they lead the system to overestimate the accuracy of its own rating of uncertainty. Both of these issues are important, and they are treated at length in the existing literature (see Cheeseman (1985), Neapolitan (1990), Pearl (1988) and the references therein).

It is not difficult to see how the structures, like the one in Figure 2, can be solved for the optimal action (see Gmytrasiewicz and Durfee (1995), Gmytrasiewicz and Durfee (1996a) for details). Since the best action of the system depends on what the user is expected to do, the solution can proceed bottom-up, so that the results of solving the models lower in the hierarchy, determining the expected action of the user, are used to solve the models higher up, for the best action of the system. In general the solution is formulated using dynamic programming Cormen et al. (1990) . Characteristically for problems for which dynamic programming is applicable, our problem exhibits optimal substructure, i.e., the solution of the problem contains the solution(s) to subproblems, due to the recursive nesting of models as in Figure 2. Further, dynamic programming exploits the property of overlapping subproblems by avoiding repeating solutions of the same subproblems, which can reduce the computational complexity. In recursive model structures, the overlapping subproblems can appear since some of the matrices may reappear in different branches of the hierarchy of models.

As we mentioned, in case of our example above, the dynamic programming bottom-up solution corresponds to the system's making its choice (in the top matrix in Figure 2) based on the anticipated actions of the user, resulting from solving the model(s) of her decision making situation residing lower in the modeling hierarchy. This process implements the notion of computer-user coordination, defined earlier as the system's choice in anticipation of the user's action. In case of the example in Figure 2, the model of the user depicted on the left, in which she is unaware of the imminent crash, would result in her choosing action a_1^2, since it results in a higher payoff (that is, the user would keep on spell checking the file). In the alternative model, the user's action a_2^2 clearly payoff-dominates the other two, and the conclusion is that if the user knew of the coming trouble she surely would choose to save the file. Given the probabilities with which the alternative models of the user are considered correct, the overall probability distribution over actions a_1^2, a_2^2 and a_3^2, describing the user's expected behavior, is [0.9, 0.1, 0]. This probability distribution can now be propagated upwards, and used to solve the top matrix. The expected utilities of the system actions a_1^1 and a_2^1 are -1.7 and -2.5, respectively. The rational coordinated choice on the system part, given that the system judges it unlikely that the user will save her work, is to execute action a_1^1 and save the file under the heading "new-file", and expecting the payoff on -1.7.

At this point it may be in order to briefly comment on the, seemingly strong, assumption in the above analysis that the user will always be able to recognize and execute her rational actions. While the concept of rationality as the behavior that maximizes the expected utility of action is purely normative, it also provides a good guidance to the majority of actually observed human behavior. As experimentally verified in various settings (see for example Tversky and Kahneman (1981), Busemeyer and Townsend (1993)) humans do end up choosing among their alternative actions with probabilities that reflect the expected utilities of these alternatives in the normative sense. Thus, alternatives that have significantly higher expected utility than their next competitors are almost always chosen by human decision makers. However, if some alternatives are close competitors, in the sense that their expected utility is similar, the human subjects tend to guess and, for example, in a case of two alternatives, choose them with 0.5/0.5 probability. One of the simplest techniques, suggested by Busemeyer and Townsend (1993), that allows one to account for the occasional irrationality of the human user is to introduce another model, with associated probability reflecting the extend to which the user's choice is likely to be the irrational, say, random, in the situation at hand. Our approach in RMM allows for this addition as a uniform probability distribution model directly in the recursive model structure.

3. Rational Communicative Behavior

This section will address the issue of rational communicative behavior. What we have in mind, in the context of a computer system interacting with a human user, is the general tool that the system can use to choose what messages to transmit to the user, and to choose the best form that these messages should take.

Continuing with our example developed in the Section 2, the questions we would like to be able to answer are: What is the best message that the system can send to the user? Is it, as we would expect, a message saying "System going down in 10 seconds", which we shall call M_1? Or is it a message "Saving the current file under "new-file" heading", which we shall call M_2? Or maybe the message should not even be in English at all, like in the case of "System padnie za 10 sekund"*, which we shall call M_3?

Our general approach was outlined in Gmytrasiewicz et al. (1991), Gmytrasiewicz and Durfee (1996b). The basic idea is to investigate the change of the payoff that can be expected, resulting from the change of the user's state of knowledge brought about by a communicative act. The value, or the expected utility, of transmitting a message can then be identified as the net change of the utility expected in the interaction. This measure, in turn, can be used to choose the most valuable message to be transmitted, thus resulting in a rational communicative behavior.

We identify the transformation of the state of knowledge due to a communicative act as the act's *pragmatic meaning*, following Clark (1992), Grice (1957), Schiffer (1972). The change in the state of knowledge can, in turn, change the decision-making situations of the recipient, as well as the speaker, of the message. We call the transformation of the decision-making situation due to a message the decision-theoretic (DT) pragmatics of the message. DT pragmatics of a message differs in a subtle way from its pragmatic meaning. In the example situation of human-computer interaction above, the message "It is 3:47pm in Sydney, Australia", if sent by the system to the user, could change the user's knowledge about the time in Sydney, and thus could have a nontrivial pragmatic meaning. However, this message would not change the decision-making situation the user is facing at the moment in any way, and thus its DT pragmatics would be an identity transformation.*

An example of a message with nontrivial DT pragmatics is the message M_1, "System going down in 10 seconds", depicted in Figure 3. Figure 3 illustrates that the result of transmitting M_1 is the elimination of the possibility, i.e., the model, according to which the user does not know about the imminent crash.

With DT pragmatics of the message M_1 as postulated in Figure 3, its impact on the expected utility can be computed. Recall that the utility expected from the interaction before communication was -1.7. The new utility, expected if M_1 were to be transmitted, can be computed in a similar bottom-up fashion from the resulting structure on the right in Figure 3, and results in the value of 2. It turns out, that unlike in the case when there is no communication, the system can now count on the user to save her file under the desired name. The difference between the utilities is the amount gained due to the message M_1, whose value is $2-(-1.7) = 3.7$.

Let us now investigate the expected effects of the message M_2, "Saving the current file under "new-file" heading". Its DT pragmatics is as depicted in Figure 4; the user's decision making situation is still uncertain, since the

* System going down in 10 seconds, in Polish.

* Note that the same message may have nontrivial DT pragmatics, as well as a considerable value, in a different decision-making situation.

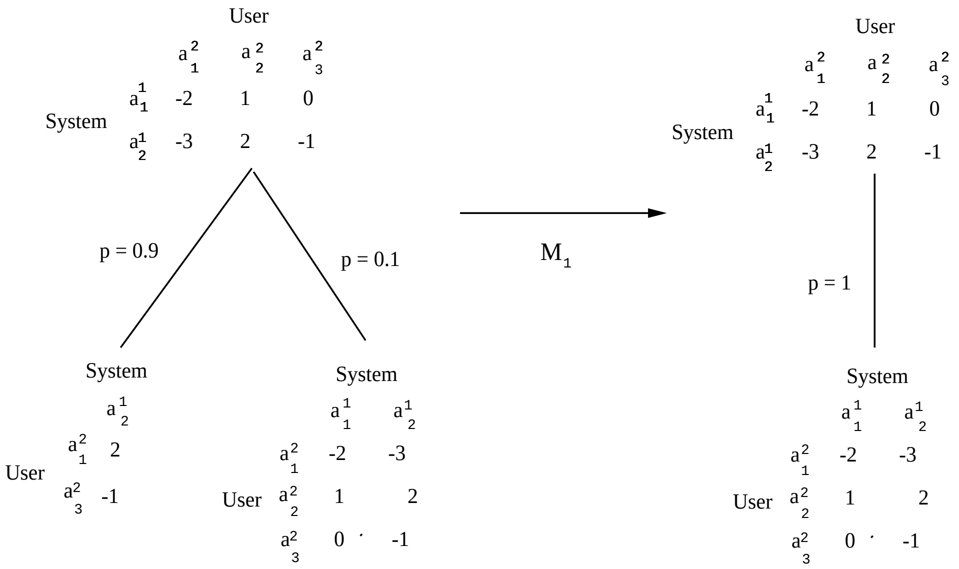

Figure 3. DT Pragmatics of Message M_1.

message does not clarify the reason for the system's saving the file. It is not difficult to verify that message M_2, as modeled in Figure 4, does not change the user's expected behavior: according to the payoffs displayed the user's rational behaviors are the same as before. The value of sending M_2 is, therefore, zero.

The above example brings out the simplifying assumption we made in our calculation of the values of messages: It is myopic. In other words, we compute the value of the message looking only one step ahead. We did not consider the fact that, for example, an intelligent user, after having received message M_2, would investigate the reason for the system to save the file, possibly ask for clarification, and find out that the system will go down shortly. All of this would impart positive value to the message M_2, but these future factors were not considered in our calculation here. Even if they were, however, the value of M_2 would likely remain below the value of M_1.

Let us now look at the message M_3. The fact that M_3 does not use English as the communication medium should not prevent us form analyzing the value of this message. As could be expected, this value depends of the likelihood, q, the system attaches to the possibility that the user speaks Polish (for simplicity we'll assume that the user cannot speak both languages). DT pragmatics of M_3 are depicted in Figure 5.

It is not difficult to compute the value of M_3 as a function of the parameter q: For q values less than 4/9 the value the new expected utility is

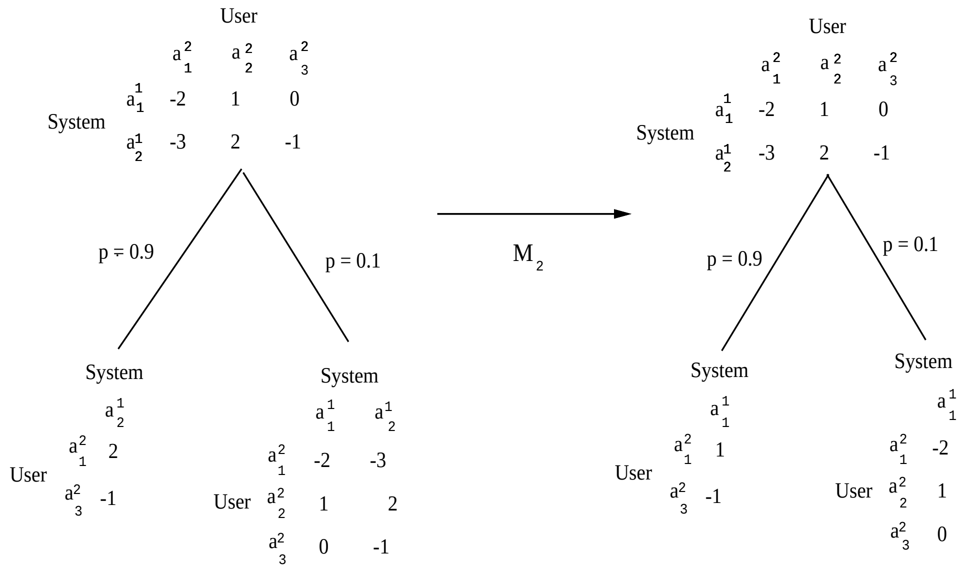

Figure 4. DT Pragmatics of Message M_2.

$2.7q - 1.7$, and the value of M_3 is $2.7q$. For q greater than $4/9$ the new expectation is $4.5q - 2.5$, and the value of M_3 is equal to $4.5q - 0.8$. The last expression matches in value the message M_1 for $q = 1$, as expected.

Let us note that for the message M_3, as well as for M_2 the effects of the transmission are not known with certainty; the user's state of knowledge was uncertain, as was her ability to understand the communication language used. In spite of this, decision-theoretic pragmatics, as postulated by our method, is well defined, and the value of the various messages considered can be precisely computed.

The examples in this section are intended to illustrate a general features of our methodology: The values of alternative messages, given their chosen transmission medium, language used, as well as volume, brightness, and so on, can be precisely computed using our method. These values can be used to prioritize the alternative communicative acts. Executing the act of the highest expected utility results in a rational communicative behavior between the computer system and the human user.

4. Conclusion

We have presented a novel method a computer-based decision support system can use to model a human user during an interaction. Our method allows the system to rationally coordinate its activity with the expected actions of the user. Further, it allows the system to compute the value of a number of

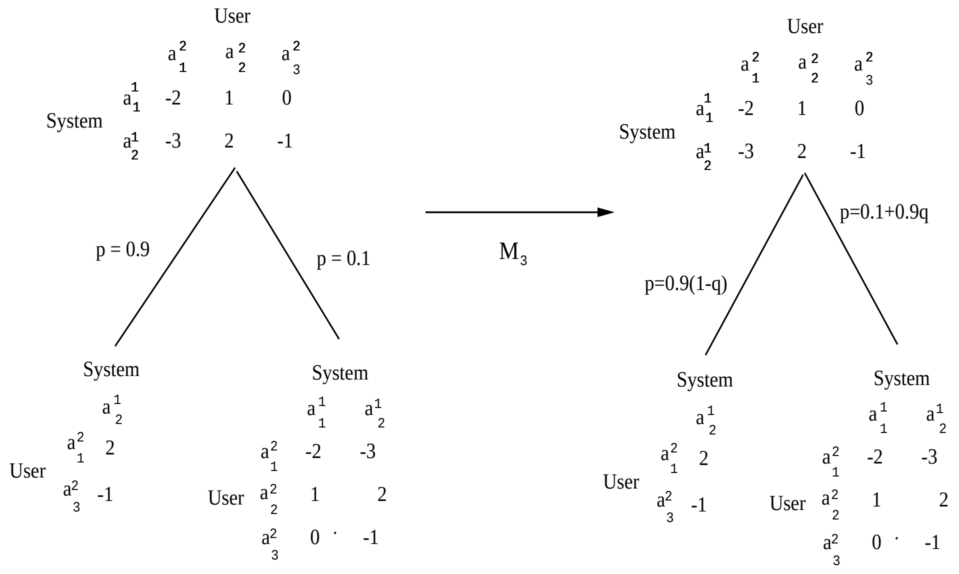

Figure 5. DT Pragmatics of Message M_3.

alternative communicative acts that the system could consider. The alternatives could include various possible contents to be transmitted, as well as alternative communication modalities, communication languages used, and so forth. The execution of the communicative act that maximizes the computed expected utility enables rational communicative behavior between the system and the human.

In our future work, we will investigate the effective methods for compiling deep user models into shallow, easy to use stereotypes, and methods of belief revision and learning in the context of the RMM formalism developed (for preliminary results see http://dali.uta.edu). We will further draw on the cognitive psychology literature, and on the theoretical approaches in AI and distributed system research, as well as in game and decision theories, to possibly integrate these disparate approaches more fully into our representation. Finally, we are planning on investigating the applications of the methodology presented here in the specific domains of industrial and managerial decision making.

References

Arend, U. (1991). Analysing complex tasks with an extended GOMS model. In Tauber, M. J., and Ackerman, D., eds., *Mental Models and Human-Computer Interaction*. North Holland.

Ballim, A., and Wilks, Y. (1991). *Artificial Believers*. Earlbaum Associates, Inc.

Busemeyer, J. R., and Townsend, J. T. (1993). Decision field theory: A dymanic cognitive approach to decision making in an uncertain environment. *Psychological Review* 432–457.

Card, K., Morgan, P., and Newell, A. (1983). *The Psychology of Human-Computer Interaction.* Elsevier.

Cheeseman, P. (1985). In defense of probability. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence*, 1002–1009.

Clark, H. H. (1992). *Arenas of Language Use.* Chicago: The University of Chicago Press.

Cohen, P. R., and Levesque, H. J. (1990a). Persistence , intention and commitment. In Cohen, P. R., Morgan, J., and Pollack, M. E., eds., *Intentions in Communication.* MIT Press.

Cohen, P. R., and Levesque, H. J. (1990b). Rational interaction as the basis for communication. In Cohen, P. R., Morgan, J., and Pollack, M. E., eds., *Intentions in Communication.* MIT Press.

Cormen, T. H., Leiserson, C. E., and Rivest, R. L. (1990). *Introduction to Algorithms.* The MIT Press.

Fischer, G. (1991). The imortance of models in making complex systems comprehensible. In Tauber, M. J., and Ackerman, D., eds., *Mental Models and Human-Computer Interaction.* North Holland.

Gmytrasiewicz, P. J., and Durfee, E. H. (1995). Formalization of recursive modeling. In *In Proceedings of the First International Conference on Multiagent Systems, ICMAS'95.*

Gmytrasiewicz, P. J., and Durfee, E. H. (1996a). Rational interaction in multiagent environments: Coordination. *Submitted for publication.* Available in postscript from http://www-cse.uta.edu/ piotr/piotr.html.

Gmytrasiewicz, P. J., and Durfee, E. H. (1996b). Rational interaction in multiagent environments: Communication. *Submitted for publication.* Available in postscript from http://www-cse.uta.edu/ piotr/piotr.html.

Gmytrasiewicz, P. J., Durfee, E. H., and Wehe, D. K. (1991). The utility of communication in coordinating intelligent agents. In *Proceedings of the National Conference on Artificial Intelligence*, 166–172.

Grice, H. P. (1957). Meaning. *Philosophical Review* (LXVI):377–388.

Jameson, A., Shaefer, R., Simons, J., and Weis, T. (1995). Adaptive provision of evaluation-oriented information: Tasks and techniques. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 1886–1893.

Jameson, A. (1989). But what will the listener think? belief ascription and image maintenance in dialog. In Kobsa, A., and Wahlster, W., eds., *User Models in Dialog Systems.* Springer-Verlag.

Kieras, D. E. (1988). Towards a practical GOMS model methodology for user interface design. In Helander, M., ed., *Handbook of Human-Computer Interaction.* Elsevier.

Neapolitan, R. E. (1990). *Probabilistic Reasoning in Expert Systems.* John Wiley and Sons.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufman.

Perrault, C. R. (1990). An application of default logic to speech act theory. In Cohen, P. R., Morgan, J., and Pollack, M. E., eds., *Intentions in Communication.* MIT Press.

Poh, K. L., and Horvitz, E. J. (1996). A graph-theoretic analysis of information value. In *Proceedings of the Twelfth Conference on Uncertainty on Uncertainty in Artificial Intelligence (UAI-96).*

Schiffer, S. (1972). *Meaning.* Clarendon Press.

Tversky, A., and Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science* 453–458.

Wilks, Y., Barden, J., and Wang, J. (1991). Your metaphor or mine: Belief ascription and metaphor interpretation. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, 945–950.