’Man-Computer Symbiosis’ Revisited: Achieving Natural Communication and Collaboration with Computers

Neal Lesh, Joe Marks, Charles Rich, Candace L. Sidner

TR2004-053 June 2004

Abstract

In 1960, the famous computer pioneer J.C.R. Licklider described a vision for human-computer interaction that he called ”man-computer symbiosis.” Licklider predicted the development of computer software that would allow people ”to think in interaction with a computer in the same way that you think with a colleague whose competence supplements your own.” More than 40 years later, one rarely encounters any computer application that comes close to capturing Licklider’s notion of human-like communication and collaboration. We echo Licklider by arguing that true symbiotic interaction requires at least the following three elements: a complementary and effective division of labor between human and machine; an explicit representation in the computer of the user’s abilities, intentions, and beliefs; and the utilization of nonverbal communication modalities. We illustrate this argument with various research prototypes currently under development at Mitsubishi Electric Research Laboratories (USA).

IEICE 2004

INVITED PAPER

“Man-Computer Symbiosis” Revisited: Achieving Natural Communication and Collaboration with Computers

Neal LESH[†a], **Joe MARKS**[†b], **Charles RICH**[†c], *and* **Candace L. SIDNER**[†d],

SUMMARY In 1960, the famous computer pioneer J.C.R. Licklider described a vision for human-computer interaction that he called “man-computer symbiosis.” Licklider predicted the development of computer software that would allow people “to think in interaction with a computer in the same way that you think with a colleague whose competence supplements your own.” More than 40 years later, one rarely encounters any computer application that comes close to capturing Licklider’s notion of human-like communication and collaboration. We echo Licklider by arguing that true symbiotic interaction requires at least the following three elements: a complementary and effective division of labor between human and machine; an explicit representation in the computer of the user’s abilities, intentions, and beliefs; and the utilization of nonverbal communication modalities. We illustrate this argument with various research prototypes currently under development at Mitsubishi Electric Research Laboratories (USA).

key words: *Man-computer symbiosis, human-computer interaction, intelligent user interfaces, interactive optimization, natural-language processing, discourse processing, nonverbal communication.*

1. Introduction

J.C.R. Licklider (1915-1990) was one of the most influential visionaries from the early days of computing. As a program manager at the U.S. Advanced Research Projects Agency (ARPA) in the 1960s and 1970s, he championed much of the early basic research on time-shared computing and computer networking. His ARPA program also helped to establish the funding base for computer-science departments at the University of California at Berkeley, Carnegie-Mellon University (CMU), the Massachusetts Institute of Technology (MIT), and Stanford University – four departments that still wield enormous influence on the field of computing.

Licklider also gave considerable thought to how humans would eventually interact with computers [19]. More than 40 years ago he argued that computers should enable humans “to cooperate [with computers] in making decisions and controlling complex situations without inflexible dependence on predetermined programs” and “to think in interaction with a computer in the same way that you think with a colleague whose competence supplements your own.” Even today, this is a radical idea. Most computer systems augment their human operators by responding in a routine manner to explicit instructions; the human is responsible for all planning and initiative. Most research in Human-Computer Interaction (HCI) does not seek to change the current style of human-computer interaction, but merely to make it more efficient, e.g., through the use of speech input, or novel display technologies, etc. In contrast, Licklider’s vision of “man-computer symbiosis” challenges our assumptions about how humans and computers *should* interact and communicate.

Of course, if man-computer symbiosis were easy to achieve, it would have been accomplished in many systems before now! Taking our lead from some of Licklider’s observations, we argue in this paper that true symbiotic interaction requires at least the following three elements, none of which are trivial to achieve: a complementary and effective division of labor between human and machine; an explicit representation in the computer of the user’s abilities, intentions, and beliefs; and the utilization of nonverbal communication modalities. We illustrate this argument with various research prototypes currently under development at Mitsubishi Electric Research Laboratories (USA).

2. Division of Labor

> *Computing machines can do readily, well, and rapidly many things that are difficult or impossible for man, and men can do readily and well, though not rapidly, many things that are difficult or impossible for computers. That suggests that a symbiotic cooperation, if successful in integrating the positive characteristics of men and computers, would be of great value [19].*

Licklider’s advice has rarely been heeded in the design of optimization systems. Most systems for solving optimization problems, such as scheduling, routing, or

---

Manuscript received January 1, 2002.

[†] Mitsubishi Electric Research Laboratories (MERL), 201 Broadway, Cambridge, Massachusetts 02478, USA. URL: www.merl.com

a) E-mail: lesh@merl.com
b) E-mail: marks@merl.com
c) E-mail: rich@merl.com
d) E-mail: sidner@merl.com

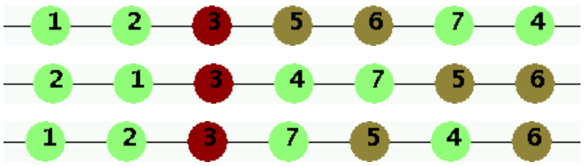
layout problems, are fully automatic. The role of the user is to specify the problem, invoke an optimization algorithm, and wait for the result.

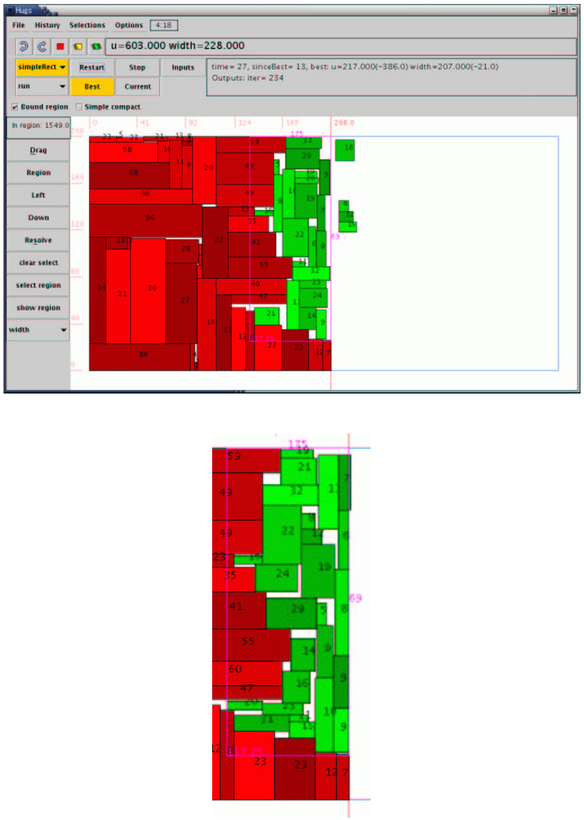
There are, however, several reasons to involve people in the process of optimization. Interactive optimization leverages people's skills in areas in which people currently outperform computers, such as visual perception, strategic thinking, and the ability to learn. Furthermore, while automatic algorithms typically solve an oversimplified formulation of a real-world problem, users can steer an interactive algorithm based on their preferences and knowledge of real-world constraints. Finally, people can better trust, justify, and modify solutions that they help construct than they can automatically generated solutions.

The Human-Guided Search (HuGS) project at MERL is an ongoing investigation into how to design human-in-the-loop optimization systems. In the HuGS framework, users can manually modify solutions, backtrack to previous solutions, and invoke, monitor, and halt a variety of optimization algorithms. However, the most distinctive aspect of the HuGS framework is the mechanism whereby human users can influence combinatorial search: they can constrain and focus optimization algorithms using *mobilities*, a general mechanism that allows users to visually annotate elements of a solution in order to guide a computer search. In particular, the user can assign each element one of three mobilities: high, medium, or low. The optimization algorithm can only apply transformations to the current solution that move at least one high-mobility element, any number of medium-mobility elements, and no low-mobility elements.

The following simple example illustrates the use of mobilities. Suppose an optimization problem contains seven elements and the solutions to this problem are all possible orderings of these elements, and the only transformation used by the optimization algorithm is to swap the position of adjacent elements. Suppose the user has assigned element 3 low mobility (shown in red), elements 5 and 6 medium mobility (shown in yellow), and the rest of the elements high mobility (shown in green):

An optimization algorithm can swap a pair of adjacent elements only if at least one has high mobility and neither has low mobility. It is limited to the space of solutions reachable by a series of such swaps, including:





u=603.000 width=228.000

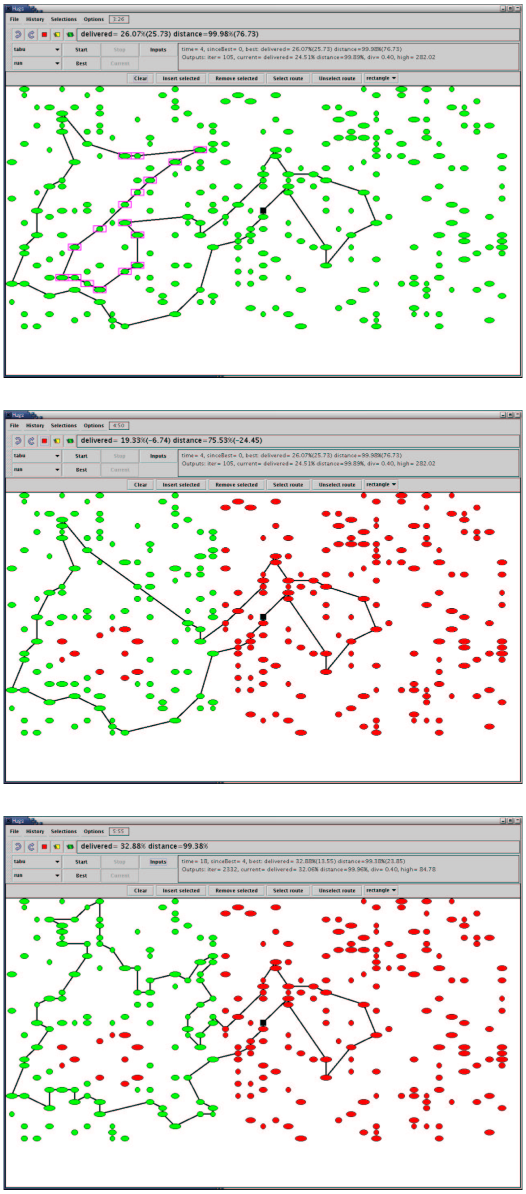
time= 27, sinceBest= 13, best: u=217.000(-386.0) width=207.000(-21.0)
Outputs: iter= 234

Fig. 1 Interactive system for an industrial cutting-stock problem, 2D Rectangular Strip Packing. In the first image, the user has selected a region to which an optimization algorithm can be applied, and has assigned high mobility (shown in green) to most of the rectangles in that region, those that the user thinks are poorly packed. The other rectangles are assigned low mobility (shown in red). The second image shows a blowup of the selected portion of the packing after the optimization algorithm has run for a few seconds. By focusing on a small region and subset of the rectangles, a better packing for the problematic region is found quickly, thereby improving the overall solution.

Note that setting element 3 to low mobility essentially divides the problem into two much smaller subproblems. Also, while medium-mobility elements can change position, their relative order cannot be changed. Mobility constraints can drastically reduce the search space; for this example, there are only 12 possible solutions, while without mobilities, there are 7! = 5040 possible solutions.

The mobility mechanism has proven its value in several contexts. We have incorporated it into exhaustive and heuristic optimization algorithms for a wide variety of problems, and studied people's ability to guide these optimization algorithms [1], [14], [25]. Additionally, we have developed the HuGS Toolkit, Java software which supports the quick development of interactive optimization systems in our paradigm [15]. Figures 1 and 2 show screenshots from two sample applications.

Our experiments show that human interaction



```
delivered= 26.07%(25.73) distance=99.98%(76.73)
time= 4, sinceBest= 0, best: delivered= 26.07%(25.73) distance=99.98%(76.73)
Outputs: iter= 105, current= delivered= 24.51% distance=99.89%, div= 0.40, high= 282.02
```

```
delivered= 19.33%(-6.74) distance=75.53%(-24.45)
time= 4, sinceBest= 0, best: delivered= 26.07%(25.73) distance=99.98%(76.73)
Outputs: iter= 105, current= delivered= 24.51% distance=99.89%, div= 0.40, high= 282.02
```

```
delivered= 32.88% distance=99.38%
time= 18, sinceBest= 4, best: delivered= 32.88%(13.55) distance=99.38%(23.85)
Outputs: iter= 2332, current= delivered= 32.06% distance=99.96%, div= 0.40, high= 84.78
```

Fig. 2 Interactive system for vehicle-routing optimization. In this system the goal is to find a truck route that makes the maximal number of deliveries to customers (represented as ellipses whose areas indicate the sizes of their orders) without exceeding a given driving distance. The image sequence demonstrates the user manually modifying a solution and then invoking a focussed search. The first image shows a route found in a few seconds by a tabu search algorithm. Because routes that nearly self intersect are rarely optimal, the user has selected some customers to be removed from a problematic section of the route. The second image shows the new route after the user has removed those customers and assigned mobilities to focus the optimization algorithm. Customers just removed from the route are assigned low mobility (shown in red) to ensure that they will not be added back to the route. In addition, distant customers are also assigned low mobility, since they are not relevant to this local fix. The remaining customers – ones near the relevant area but not part of the original problematic section of the route – are assigned high mobility (shown in green). The third image shows the route found after the user has again invoked the optimization algorithm for a few seconds. It is significantly more efficient than the original route.

Fig. 3 The Optimization Table. We have found that interactive optimization applications benefit from the use of a projected, tabletop display shown in Figure 3, which we call the Optimization Table. We project an image down onto a whiteboard. This allows users to annotate candidate solutions by drawing or placing tokens on the board. In addition, several users can comfortably use the system together. Once again, this is a development anticipated by Licklider: "[I]nformal, parallel arrangements of operators, coordinating their activities through reference to a large situation display, have important advantages over the arrangement, more widely used, that locates the operators at individual consoles and attempts to correlate their actions through the agency of a computer [19]."

can significantly improve the performance of optimization algorithms. Human guidance can improve the performance of an exhaustive optimization algorithm for the capacitated-vehicle-routing-with-time-windows problem to the point where the interactive algorithm is competitive with the best previously reported algorithms [1], [25]. A second set of experiments showed that human-guidance of a more sophisticated optimization algorithm (called tabu search) could produce results in 10 minutes equivalent to, on average, 70 minutes of search without human guidance [14]. Additionally, our interactive system was able to achieve the best performance we know of on benchmark problems for the 2D Rectangular Strip Packing Problem [18].

3. Representing the User

> *[Compare] instructions ordinarily addressed to intelligent human beings with instructions ordinarily used with computers. The latter specify precisely the individual steps to take and the sequence in which to take them. The former present or imply something about incentive or motivation, and they supply a criterion by which the human executor of the instructions will know when he has accomplished his task. In short: instructions directed to computers specify courses; instructions directed to human beings specify goals [19].*

Since Licklicker wrote these words, a large body of empirical knowledge and computational theory has been accumulated regarding how human beings communicate and collaborate with one another. The goal

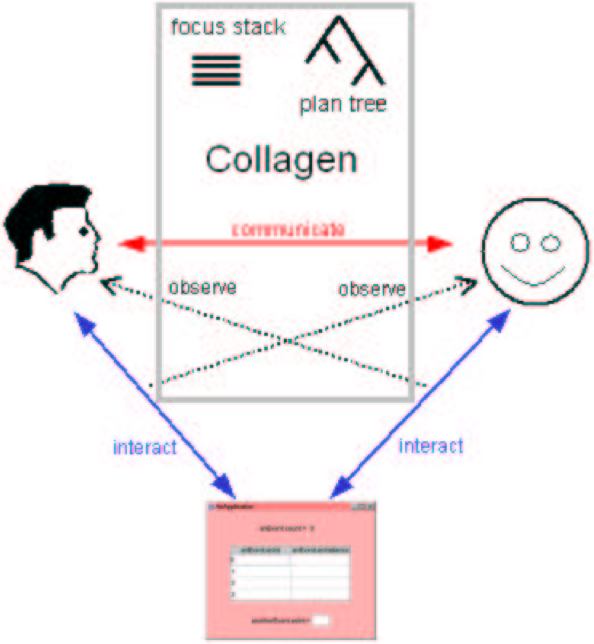


Fig. 4 The COLLAGEN Collaboration Manager.

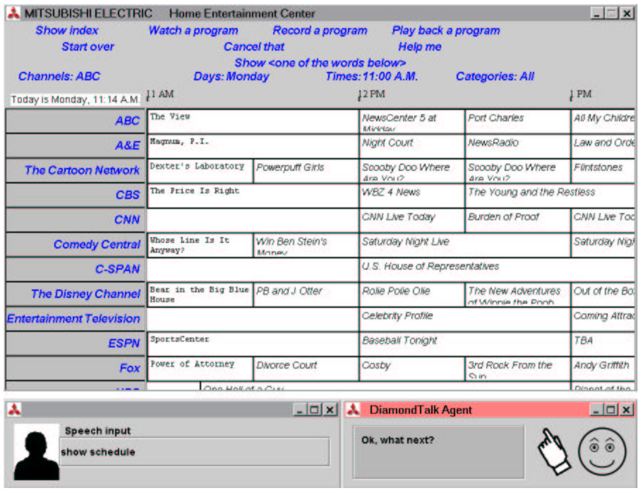
of the COLLAGEN (for COLLaborative AGENt) project at MERL has been to apply this work to human-computer interaction. In particular, we have focused on the SharedPlan theory of human collaborative dialogue by Grosz, Sidner, and Lochbaum [8], [9], [20].

The practical output of the COLLAGEN project has been the development of an application-independent *collaboration manager* of the same name, implemented in Java. A collaboration manager is a software component which mediates a collaborative interaction between a software agent and a human user (see Figure 4). The capabilities of a collaboration manager extend those of what is traditionally called a "dialogue manager" by including a representation of the goals and plans of the agent and user.

COLLAGEN has and is currently being used both inside and outside MERL to build prototype human-computer collaborative dialogue systems for a wide range of applications including [24]: air travel planning, airport flight-path planning, email, a computer-aided design tool, power-plant operator training, multimodal web form filling, a programmable home thermostat, the personal video recorder (PVR) interface shown in Figure 5, and the robot described in Section 4.

3.1 Dialogue State Representation

The participants in a collaboration derive benefit by combining their talents and resources to achieve common goals. However, collaboration also has its costs. When people collaborate, they must usually communicate and expend mental effort to ensure that their actions are coordinated. In particular, both participants must maintain some sort of mental model of the status of the collaborative tasks and the conversation about them—we call this model the *dialogue state*.



Channels: ABC
Days: Monday
Times: 11:00 A.M.
Categories: All

Today is Monday, 11:14 A.M.	11 AM		12 PM		1 PM
ABC	The View		NewsCenter 5 at Midday	Port Charles	All My Childre
A&E	Magnum, P.I.		Night Court	NewsRadio	Law and Orde
The Cartoon Network	Dexter's Laboratory	Powerpuff Girls	Scooby Doo Where Are You?	Scooby Doo Where Are You?	Flintstones
CBS	The Price Is Right		WBZ 4 News	The Young and the Restless	
CNN			CNN Live Today	Burden of Proof	CNN Live Tod
Comedy Central	Whose Line Is It Anyway?	Win Ben Stein's Money	Saturday Night Live		Saturday Nigh
C-SPAN			U.S. House of Representatives		
The Disney Channel	Bear in the Big Blue House	PB and J Otter	Rolie Polie Olie	The New Adventures of Winnie the Pooh	Out of the Bo
Entertainment Television			Celebrity Profile		Coming Attrac
ESPN	SportsCenter		Baseball Tonight		TBA
Fox	Power of Attorney	Divorce Court	Cosby	3rd Rock From the Sun	Andy Griffith

Speech input
show schedule

Ok, what next?

Fig. 5 Collaborative Agent for Personal Video Recorder

Among other things, the dialogue state tracks the beliefs and intentions of both participants in the collaboration and provides a focus-of-attention mechanism for tracking shifts in the task and conversational context. All of this information is used by each individual to help understand how the actions and utterances of the participant contribute to the common goals.

In order to turn a computer into a collaborator, we therefore need a formal representation of dialogue state and algorithms for updating and using it. The dialogue state representation used in COLLAGEN, illustrated in Figure 6, is a partial implementation of Grosz and Sidner's SharedPlan theory.

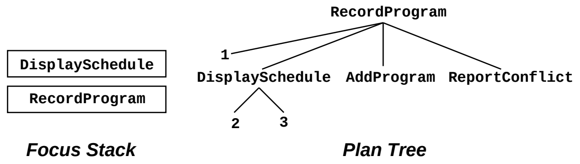
COLLAGEN's discourse state consists of a stack of goals,[†] called the *focus stack*, and a *plan tree* for each goal on the stack. The top goal on the focus stack is the "current purpose (goal)" of the dialogue. A plan tree in COLLAGEN is an (incomplete) encoding of a partial SharedPlan between the user and the agent. For example, Figure 6 shows the focus stack and plan tree immediately following the discourse events numbered 1–3 on the right side of the figure.

The annotated, indented execution trace on the right side of Figure 6, called a *segmented interaction history*, is a compact textual representation of the past, present, and expected future states of the dialogue. We originally developed this representation to help us debug agents and COLLAGEN itself, but we have also experimented with using it to help users visualize what is going in a collaboration (see discussion of "history-based transformations" in [23]).

The numbered lines in a segmented interaction history are simply a log of the agent's and user's utterances

---

[†] Technically, a stack of focus "spaces," each of which contains a goal.



Scheduling a program to be recorded.
1 User says "I want to record a program."
Done successfully displaying the recording schedule.
2 Agent displays recording schedule.
3 Agent says "Here is the recording schedule."
Next expecting to add a program to the recording schedule.
Expecting optionally to say there is a conflict.

Fig. 6 Dialogue State and Segmented Interaction History for PVR.

and primitive actions. The italic lines and indentation reflect COLLAGEN's interpretation of these events. Specifically, each level of indentation defines a *segment* whose purpose is specified by the italicized line that precedes it. For example, the purpose of the toplevel segment in Figure 6 is *Scheduling a program to be recorded.*

Unachieved purposes that are currently on the focus stack are annotated using the present tense, such as *scheduling*, whereas completed purposes use the past tense, such as *done*. Note in Figure 6 that a goal is not popped off the stack as soon as it is completed, because it may continue to be the topic of conversation, for example, to discuss whether it was successful.

Finally, the italic lines at the end of each segment, which include the keyword *expecting*, indicate the steps in the current plan for the segment's purpose which have not yet been executed. The steps which are "live" with respect to the plan's ordering constraints and preconditions have the added keyword *next*.

```
public recipe RecordRecipe achieves RecordProgram {
   step DisplaySchedule display;
   step AddProgram add;
   optional step ReportConflict report;
   constraints {
      display precedes add;
      add precedes report;
      add.program == achieves.program;
      report.program == achieves.program;
      report.conflict == add.conflict;
   }
}
```

Fig. 7 Example Recipe in the PVR Task Model. Definition of the recipe used in Figure 6 to decompose the non-primitive `RecordProgram` goal into primitive and non-primitive steps. COLLAGEN task models are defined in an extension of the Java language which is automatically processed to create Java class definitions for recipes and act types.

Scheduling a program to be recorded.
1 User says "I want to record a program."
Done successfully displaying the recording schedule.
2 Agent displays recording schedule.
3 Agent says "Here is the recording schedule."
4 User says "Ok."
Done identifying the program to be recorded.
5 Agent says "What is the program to be recorded?"
6 User says "Record 'The X-Files'."
Next expecting to add a program to the recording schedule.
Expecting optionally to say there is a conflict.

Fig. 8 Continuing the Interaction in Figure 6.

3.2 Interpretation and Generation

COLLAGEN updates its dialogue state after every utterance or primitive action by the user or agent using Lochbaum's interpretation algorithm [20] with extensions to include plan recognition [16] and unexpected focus shifts [17].

According to Lochbaum, each event should be explained as either: (i) starting a new segment whose purpose contributes to the current purpose (and thus pushing a new purpose on the focus stack), (ii) continuing the current segment by contributing to the current purpose, or (iii) completing the current purpose (and thus eventually popping the focus stack).

An utterance or action contributes to a purpose if it either: (i) directly achieves the purpose, (ii) is a step in a recipe for achieving the purpose, (iii) identifies the recipe to be used to achieve the purpose, (iv) identifies who should perform the purpose or a step in the recipe, or (v) identifies a parameter of the purpose or a step in the recipe. These last three conditions are what Lochbaum calls "knowledge preconditions."

A *recipe* is a goal-decomposition method (part of a task model). COLLAGEN's recipe-definition language (see Figure 7) supports partially ordered steps, parameters, constraints, pre- and post-conditions, and alternative goal decompositions.

COLLAGEN also uses the dialogue state to generate behavior, which we illustrate by briefly describing how PVR agent produces the underlined utterance on line 5 in Figure 8, which continues the interaction in Figure 6.

COLLAGEN's dialogue generation algorithm is essentially the inverse of its interpretation algoritm. Based on the current dialogue state, it produces a prioritized list, called the *agenda*, of (partially or totally specified) utterances and actions which would contribute to the current discourse purpose according to cases (i) through (v) above. For example, for the dialogue state in Figure 6, the first item on the agenda is an utterance asking for the identity of the program parameter of the AddProgram step of the plan for RecordProgram. This utterance starts a new segment, which is then completed by the user's answer on line 6.

4. Nonverbal Communication

> *If computing machines are ever to be used directly by top-level decision makers, it may be worthwhile to provide communication via the most natural means, even at considerable cost [19].*

4.1 Gestures in Interaction

Gestures are fundamental to human interaction. When people are face-to-face at near or even far distance, they gesture to one another as a means of communicating their beliefs, intentions and desires. When too far apart or in too noisy an environment to use speech, gestures can suffice, but in most human face-to-face encounters, speech and gesture co-occur. According to the claims of [21], they are tightly intertwined in human cognition. However cognitively entangled, people gesture freely, and the purposes of those gestures are the topic of this section.

Gestures convey intentions, beliefs and desires, that is, information about the individuals who use them. Gestures are made with every part of the body: hands, face, posture of the body, body stance, legs, feet. Facial gestures have been carefully studied to interpret emotion and intentions to deceive [6], while others have considered the use of facial muscles (such as an eye raise) to convey belief as well as intention [7], [13]. Hands are expressive, providing gestures of the newness information to the conversation (so called beat gestures), as well as iconic, metaphorical and deictic indications, the last to point to objects in the environment [2]. Body posture and stance can be used to convey emotion (as can legs and feet), but body posture also signals major boundaries in the units of conversation [3].

Gestures provide many types of information about the individual to the conversation partner. One additional type of information they provide has been noted among scientists who study human interaction. Gestures convey engagement, that is, the attentiveness of one partner to the other during their interaction. The human-robot interaction team at MERL is focused on the nature of engagement in human-machine interaction.

4.2 The Nature of Engagement

Engagement is the process by which two (or more) participants establish, maintain, and end their perceived connection to one another (e.g., Figure 9). This process includes: making initial contact, negotiating their collaboration during the conversation, assessing the intentions of the other participants in remaining engaged, evaluating whether to stay involved, and ending the interaction [27]. In circumstances where conversational participants are not face-to-face (notably on the telephone), engagement must be undertaken and maintained by the vocal capabilities of the participants. But when face-to-face, gesture is a significant means of performing this task. Engagement, or demands for engagement from one's partner, can be conveyed by all parts of the body. However, gaze, head movement, arm movements and body stance are the principal means of doing so.

Fig. 9 Face-to-face conversation

Gaze to the conversational partner, and away for brief periods or to undertake related tasks, is a major indication of engagement. Wandering gaze, or gaze to unrelated items or persons, signal a loss of engagement, and may signal a desire to end the interaction. Head movements provide even stronger engagement evidence than gaze, most likely because head movement is more physically complex and cannot be undone as easily. Arm movements to point at relevant objects not only tell the conversational partner what is of interest, but also where to pay attention in the interaction. They can also be used to get attention when a partner's interest has wandered. The direction of the front of one's body and the trajectory of movement also indicate one's primary focus of attention. When a conversational partner must direct his or her body to something besides a conversational partner, some other means of indicating ongoing engagement must be conveyed instead.

Engagement tells the partner that the interaction has been understood so far (the grounding conditions of dialogue [5], and also that the interaction is to go forward (or not). However, because people live in a world that is not perfectly predictable, they have to divide their attention between the conversational partner, and noticing what's going on in the world around them. In a quiet, benign, office environment, this division may seem unnecessary, but much of human life occurs in less benign circumstances where it may be necessary

Fig. 10 Robot and User Conversation

to notice that nothing dangerous is headed one's way or at least that the other people are are not demanding one's attention or behaving dangerously. Furthermore, people often multitask between a conversation and some other undertaking that may or may not be related to the conversation; for example, nearly everyone has washed the dishes while talking with a conversational partner about something else. The task that requires attention of one's visual system then must be balanced with attention to the conversational partner.

4.3 Robots with Engagement

Can robots be built that take advantage of non-verbal gestures, including those that convey engagement? The answer is yes, if one is willing to engineer a number of different systems and algorithms together to provide motor behavior, fused sensory input and models of conversation together with rules for engagement. Our most recent efforts center on a robotic penguin that converses with people and collaboratively presents a demo of a new hardware invention from MERL. Figure 10 shows the robot in conversation with a user.

The architecture of this robot divides its overall behavior into a brain and a body. The brain models the conversation and the overall collaboration that is being pursued, using the COLLAGEN collaboration manager described in Section 3. The body gathers sensory data, interprets it and provides it to the `brain`. It accepts back commands for what to do, and also takes note of the current state of the conversation in order to choose some of its next moves with its sensory devices. A diagram of the architecture available in Figure 11.

The robot uses vision algorithms of [22], [28] to find the people in a room and to track the head movements of one of the people, who is the robot's conversational partner (CP). To determine the CP, the robot listens for a voice, and co-locates that voice with one of the faces it sees in the room. It looks at the person's face as it moves around in front of it. The robot turns away from the CP when it must point to an object in the demo, and it also looks to be certain that the person tracks its pointing behavior. Accompanying this movement is a spoken dialogue between the robot and the person about the demonstrated hardware with speech recognition accomplished using IBM Via Voice). Wing gestures (the penguin has wings, not arms and hands) indicate the presence of new information in the utterance it is speaking. [4]. Using algorithms to detect headnods and head shakes [12], the robot notices when the person nods instead of saying `uh-huh` as well as combining information from `yes` and `no` together with head nods and head shakes. The robot also can tell when the person fails to observe itself or the demo for extended periods of time, and will query the person if she or he does not take a turn in the dialogue when expected.

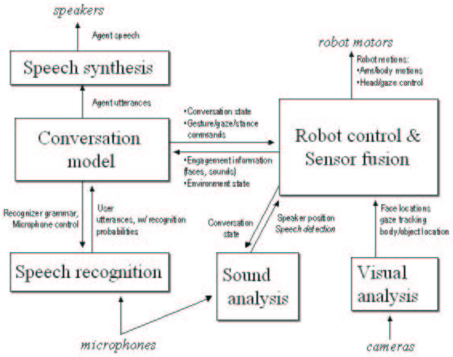


Fig. 11 Architecture of Humanoid Robot

Does this robot's non-verbal gestural behavior have an impact on the human partner? The answer is a qualified yes. In experiments with 37 human subjects, all subjects were found to turn their gaze to the robot whenever they took a turn in the conversation, an indication that the robot was real enough to be worthy of engagement [26]. Furthermore, in a comparison between the robot using its full engagement behavior versus one that just stared woodenly straight ahead, participants using the fully active robot looked back at the robot significantly more whenever they were attending to the demonstration in front of them. The participants with the active robot also responded to the robot's change of gaze to the table significantly more than the other subjects. These subjects unconsciously considered the robot a partner to keep engaged with.

Far more research on engagement and creating engagement with robots remains to be accomplished. Among the many questions to be considered: Is engagement affected by the type of utterance in the conversation? How do humans stay engaged when directing most of their attention to a task in the environment? How can robots be built to better integrate language production and engagement behaviors? By study of human-human interaction and by study of the robots built with engagement abilities, future human and robot interactions will feature natural, expressive and easy to use capabilities.

5. Conclusion

We have all grown accustomed to a certain style of human-computer interaction (HCI): using window, icon, menu, and pointer (WIMP) metaphors, users give instructions to a computer-based device; the device responds by conveying simple status information. Perceptual interaction (e.g., via speech-, vision-, or haptics-based methods) may change the input and output modalities, but not the essence of this interaction.

Will this style dominate for decades to come, or is there an alternative notion of how humans and computers can communicate? We believe there is, and that it was articulated more than 40 years ago by Licklider. Licklider's basic idea is that humans and computers will communicate to collaborate. In this paper we have described some examples of collaborative HCI; other examples can be found in the recent research literature on intelligent user interfaces (e.g., see the proceedings of the ACM Conference on Intelligent User Interfaces (IUI) [10]). While it is happening later than Licklider had predicted, we share his optimism that a collaborative-HCI revolution will eventually happen.

References

[1] D. Anderson, E. Anderson, N. Lesh, J. Marks, B. Mirtich, D. Ratajczak, and K. Ryall. "Human-guided simple search." In Proc. of AAAI 2000, pages 209–216, 2000.
[2] J. Cassell. "Nudge nudge wink wink: elements of face-to-face conversation for embodied conversational agents." In Embodied Conversational Agents, J. Cassell, J. Sullivan, S. Prevost, and E. Churchill (eds.), MIT Press, 2000.
[3] J. Cassell, Y.I. Nakano, T.W. Bickmore, C.L. Sidner, and C. Rich. "Non-verbal cues for discourse structure." In Proc. of the 39th Annual Meeting of the Association for Computational Linguistics (ACL 2001), pp. 106–115, July 2001.
[4] J. Cassell, H. Vilhjlmsson, and T.W. Bickmore. "BEAT: the behavior expression animation toolkit." In Proc. of ACM SIGGRAPH 2001, pp. 477-486, 2001.
[5] H.H. Clark. Using Language. Cambridge University Press, 1996.
[6] P. Ekman. Emotion in the human face. Second Edition. Cambridge University Press, 1982.
[7] C. Goodwin. "Gestures as a resource for the organization of mutal orientation." Semiotica, 62(1-2): 29-49. 1986.
[8] B.J. Grosz and C.L. Sidner. "Attention, intentions, and the structure of discourse." Computational Linguistics, 12(3): 175-204. 1986.
[9] B.J. Grosz, and C.L. Sidner. "Plans for discourse." In Cohen, P. R.; Morgan, J. L.; and Pollack, M. E., eds., Intentions and Communication. Cambridge, MA: MIT Press. 417-444. 1990.
[10] URL: www.iuiconf.org
[11] W.L. Johnson, J.W. Rickel, and J.C. Lester. "Animated pedagogical agents: face-to-face interaction in interactive learning environments." International Journal of Artificial Intelligence in Education, 11: 47-78, 2000.
[12] A. Kapoor and R. Picard. "A real-time head nod and shake detector." In Proc. of the Workshop on Perspective User Interfaces, MIT Media Lab Technical Report #544, 2001.
[13] A. Kendon. "Some functions of gaze direction in social interaction." Acta Psychologica, 26: 22-63, 1967.
[14] G. Klau, N. Lesh, J. Marks, and M. Mitzenmacher. "Human-Guided Tabu Search." In Proc. of AAAI 2002, pp. 41-47, 2002.
[15] G. Klau, N. Lesh, J. Marks, M. Mitzenmacher, and G.T. Schafer. "The HuGS platform: A toolkit for interactive optimization." In Proc. of Advanced Visual Interfaces (AVI 2002), pp. 324-330, 2002.
[16] N. Lesh, C. Rich, and C. Sidner. "Using plan recognition in human-computer collaboration." In Proc. 7th Int. Conf. on User Modelling, pp. 23-32, 1999.
[17] N. Lesh, C. Rich, and C. Sidner. "Collaborating with focused and unfocused users under imperfect communication." In Proc. 9th Int. Conf. on User Modelling, pp. 63-74, 2001.
[18] N. Lesh, J. Marks, A. McMahon, and M. Mitzenmacher. "New Exhaustive, Heuristic, and Interactive Approaches to 2D Rectangular Strip Packing." In submission, 2003.
[19] J.C.R. Licklider. "Man-Computer Symbiosis." IRE Transactions on Human Factors in Electronics, volume HFE-1, pp. 4-11, March 1960. Copies of this article can be found on several websites.
[20] K.E. Lochbaum. "A collaborative planning model of intentional structure." Computational Linguistics 24(4), pp 525–572, 1998.
[21] D. McNeill. Hand and mind: what gestures reveal about thought. University of Chicago Press, 1992.
[22] L.-P. Morency, A. Rahimi, and T. Darrell. "Adaptive view-based appearance models." In Proc. of IEEE CVPR 2003, Vol. 1, pp. 8-20, 2003.
[23] C. Rich, and C. Sidner. "COLLAGEN: A collaboration manager for software interface agents." User Modeling and User-Adapted Interaction 8(3/4): 315-350. 1998.
[24] C. Rich, C. Sidner, and N. Lesh. "Collagen: Applying collaborative discourse theory to human-computer interaction." AI Magazine, 22(4), pp. 15–25, 2001.
[25] S. Scott, N. Lesh, and G. W. Klau. "Investigating human-computer optimization." In Proc. of ACM CHI 2002, pp. 155-162, 2002.
[26] C.L. Sidner, C.D. Kidd, C. Lee, and N. Lesh. "Where to look: a study of human-robot interaction." in Proc. of the International Conference on Intelligent User Interfaces (ACM IUI 2004), pp. 78-84, 2004.
[27] C.L. Sidner and M. Dzikovska. "Human-robot interaction: engagement between humans and robots for hosting activities." In Proc. of the IEEE International Conference on Multimodal Interfaces (ICMI 2002), pp. 123-128, October 2002.
[28] P. Viola and M. Jones. "Rapid Object Detection Using a Boosted Cascade of Simple Features." In Proc. of IEEE CVPR 2001, pp. 905-910, 2001.

Joe Marks Prior to joining MERL in 1994, Joe Marks worked at Digital Equipment Corporation's Cambridge Research Laboratory. As a researcher at MERL, Joe's primary focus was on computer graphics, user interfaces and heuristic optimization. Joe also has a strong interest in teaching. He was an adjunct lecturer in the Division of Engineering and Applied Sciences at Harvard University from 1991 to 1996. Joe became Associate Director of the MERL Research Lab in 1999 and Director in 2000.

Candace Sidner Candy Sidner is an expert in user interfaces, especially those involving speech and natural language understanding, and human and machine collaboration. At the MERL Research lab, she is currently working human robot interaction and especially on engagement between humans and robots. Dr. Sidner was Chair of the 2001 International Conference on Intelligent User Interfaces and is a past President of the Association for Computational Linguistics. She is also a Fellow and past Councilor of the American Association for Artificial Intelligence.

Neal Lesh Neal Lesh's research efforts aim to improve (or at least ease) the interaction between people and computers. His research projects at the MERL Research lab include interactive optimization (the HuGS project), collaborative interface agents (the COLLAGEN project), and collaborative navigation of digital data (the Personal Digital Historian project). Before coming to MERL, he was a graduate student at the University of Washington with Oren Etzioni, and and a postdoc with James Allen at the University of Rochester.

Charles Rich The thread connecting all of Dr. Rich's research has been to make interacting with a computer more like interacting with a person. As a founder and director of the Programmer's project at the MIT Artificial Intelligence Lab. in the 1980s, he pioneered research on intelligent assistants for software engineering. Dr. Rich joined MERL in 1991 as a founding member of the Research Lab. For the past several years, he has been working on a technology, called Collagen, for building collaborative interface agents based on human discourse theory. Dr. Rich is a Fellow and past Councilor of the American Assoc. for Artificial Intelligence. He was Chair of the 1992 Int. Conf. on Principles of Knowledge Representation and Reasoning, Co-Chair of the 1998 National Conf. on Artificial Intelligence, and Program Co-Chair the 2004 Int. Conf. on Intelligent User Interfaces.