20th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Atlanta, GA, 31 July - 3 August 2011.

# Generating Connection Events for Human-Robot Collaboration

Aaron Holroyd, Charles Rich, Candace L. Sidner and Brett Ponsler

***Abstract*— We have developed and tested a reusable Robot Operating System (ROS) module that supports engagement between a human and a humanoid robot by generating appropriate directed gaze, mutual facial gaze, adjacency pair and backchannel connection events. The module implements policies for adding gaze and pointing gestures to referring phrases (including deictic and anaphoric references), performing end-of-turn gazes, responding to human-initiated connection events and maintaining engagement. The module also provides an abstract interface for receiving information from a collaboration manager using the Behavior Markup Language (BML) and exchanges information with our previously developed engagement recognition module.**

## I. Introduction

Engagement [1] is a fundamental process that underlies all human interaction and has common features across a very wide range of interaction circumstances. This suggests, recognizing and generating engagement behaviors is crucial for humanoid robots that will interact effectively with humans without special training.

In our previous research [2], we identified four types of *connection events* that contribute to engagement and we developed a reusable Robot Operating System (ROS) module that automatically recognizes these events during human-robot interaction. In this paper, we report on a closely related module that *generates* connection events. Together these two modules (plus some additional testing nodes) comprise the ROS "engagement" stack, which is available under an open source license at ros.org and is the main practical contribution of this work.

The main scientific contribution of this work is the engagement generation *policies*, described in Section IV, which define the conditions under which the robot initiates specific connection events. In particular, in Section IV-C we propose a novel analysis of optimal gesture-speech pairs for object reference. For example, if a robot is asking a human to move a green triangle next to a purple square, the robot has to decide whether to look, point at either shape, do both, or do neither. These policies are based on our own [2] and others' observational study of human engagement (See Fig. 1(b)).

## II. Related Work

In the study of human behaviors that contribute to engagement, Argyle and Cook [3] documented that failure to attend to another person via gaze is evidence of lack of interest and attention. Other researchers have offered evidence of the role of gaze in coordinating talk between speakers and listeners, in particular, how gestures direct gaze to the face and why gestures might direct gaze away from the face [4], [5], [6]. Nakano *et al.* [7] reported on the use of the listener's gaze and the lack of negative feedback to determine whether the listener has grounded [8] the speaker's turn.

In terms of computational approaches, the most closely related work is that of Peters [9], which involves agents in virtual environments, and Bohus and Horvitz [10], which involves a realistically rendered avatar head on a desktop display. We share a similar theoretical framework with both of these efforts, but differ in dealing with a humanoid robot and in our focus on building reusable modules.

Mutlu *et al.* [11] have studied the interaction of gaze and turn-taking [12] using a humanoid robot. Flippo *et al.* [13] have developed a similar architecture to ours with similar concerns of modularity and the fusion of verbal and non-verbal behaviors, but for multimodal interfaces rather than robots. Neither of these efforts, however, use the concepts of engagement or connection events.

## III. Architecture

Fig. 1(a) summarizes how the engagement generation module, outlined in bold, fits into the overall architecture of a collaborative human-robot interaction system by showing the main information flows between this module and the rest of the system. Fig. 1(b) shows our robot and a human interacting in our test setup.

We assume that both the human and the robot can perform the following behaviors and observe them in the other:

- look at the other's face, objects on the table or "away"
- point at objects on the table
- nod the head (up and down)
- shake the head (side to side)

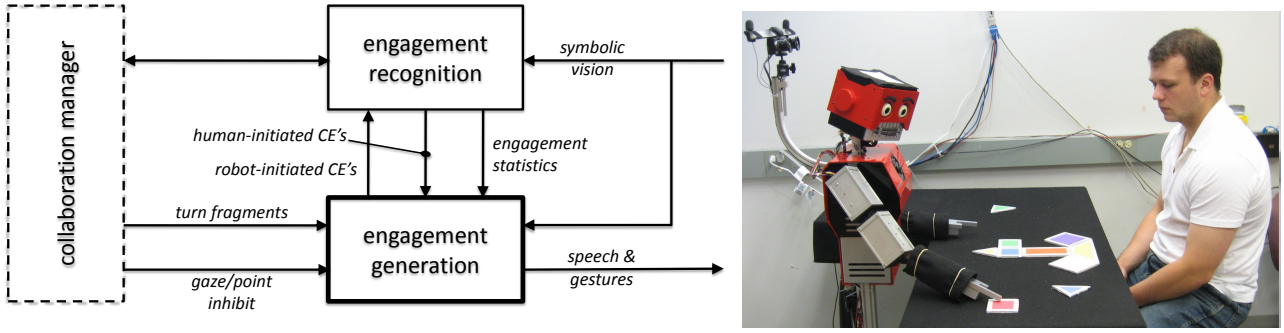
The robot also generates speech, using a text-to-speech system, that is understood by the human. However, our current system does not include natural language understanding, so the robot can only detect the beginning and end of the human's speech.

The collaboration manager in Fig. 1(a) is drawn in dashed lines to emphasize that we have made minimal assumptions about how the higher-level cognitive abilities of the robot are implemented. Furthermore, our goal is to free the collaboration manager as much as possible from having to worry about the details of the engagement process.

### A. Modules

The engagement *generation* module in Fig. 1(a) exchanges information with our previously implemented engagement

This work is supported in part by the National Science Foundation under awards IIS-0811942 and IIS-1012083. Holroyd, Rich, and Sidner are with the Computer Science Department, Worcester Polytechnic Institute, Worcester, MA 01609 (aholroyd | rich | sidner)@wpi.edu. Ponsler is with iRobot, 8 Crosby Drive, Bedford, MA 01730 bponsler@irobot.com




(a) Role of engagement generation in overall human-robot collaboration architecture.

(b) The tangram game.

Fig. 1. System Architecture

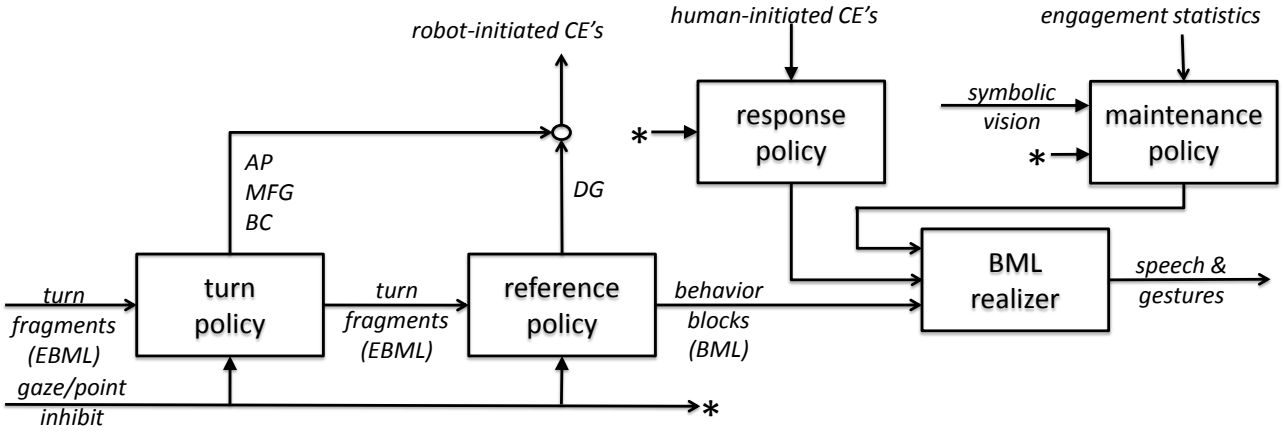



Fig. 2. Information flow inside the engagement generation module.

recognition module [2], a collaboration manager, and the robot's vision, speech and motor control subsystems.

The main job of the generation module is to generate nonverbal behavior that contributes to engagement between the human and the robot. The main input to generation is a real-time stream of *turn fragments* from the collaboration manager. These are the actions (speech, references, etc.) which the collaboration manager has decided are appropriate based on the current task state. The main output of generation is a real-time synchronized stream of speech and gestures. Note that speech is not generated but selected and passed through by the reference policy.

The main job of the engagement *recognition* module is to notify the generation module of *human-initiated* connection events, so that the generation module can respond appropriately. For example, when the human points to an object, engagement recognition recognizes the start of a human-initiated directed gaze event (see Section III-B), which the generation module completes by generating a robot gaze at the object. Symmetrically, generation also notifies recognition when the robot initiates a connection event, such as directed gaze. For more on engagement recognition see [2].

### B. Connection Events

In order to explain the generation policies below, we first need to briefly review the four types of connection events (CE's) that our system deals with. Note that each event type can be either human- or robot-initiated and that gesture and speech often overlap. These connection events are based on our own observational study of human engagement [2] and other human interaction research (See Section II).

*1) Directed Gaze:* In directed gaze (DG) [4], one person (the *initiator*) looks and optionally points at some object or group of objects in the environment, following which the other person (the *responder*) looks at the same object(s). This event is often synchronized with the initiator referring to the object(s) in speech, as in "please move *the blue triangle*."

*2) Mutual Facial Gaze:* Mutual facial gaze (MFG) [3] starts when the initiator looks at the responder's face. After a delay, the responder looks at the initiator's face, which completes the MFG and starts the period of mutual gaze. Note that the delay can be zero, which occurs when both parties simultaneously look at each other.

*3) Adjacency Pair:* In linguistics, an adjacency pair (AP) [14] consists of an utterance by each of two speakers, with minimal overlap or gap between them, such that the first utterance provokes the second utterance. A question-answer pair is a classic example of an adjacency pair. We generalize this concept slightly to include both verbal (utterances) and non-verbal communication acts. So for example, a nod could be the answer to a question, instead of a spoken "yes." Adjacency pairs often overlap with the gestural connection events, such as directed gaze and mutual facial gaze.

*4) Backchannel:* A backchannel (BC) is an event in which one party (the responder) directs a brief verbal or gestural communication back to the initiator *during* the primary

communication from the initiator to the responder. Typical examples of backchannels are nods and/or saying "uh huh."

## IV. Generation Policies

Fig. 2 summarizes the internal architecture of the engagement generation module, which contains four engagement generation policy components and a Behavior Markup Language (BML) realizer. Notice that the arrows entering and leaving this figure correspond to Fig. 1(a).

The *gaze/point inhibit* input from the collaboration manager is an input to all four of the policy components. This input can be used to prevent the engagement generation module from generating robot gaze or pointing behaviors when such behaviors are inappropriate due to higher-level task concerns. For example, it might be a good idea to inhibit gaze changes while the robot is watching for a mole to pop up out of its hole or to inhibit pointing while the robot is holding a lit blowtorch in its hand.

### A. Turn Fragments

A Turn fragment is the fundamental representation by which the engagement generation module provides its services to the collaboration manager. A *turn fragment* comprises:

- a turn index (first, second or third turn of an adjacency pair
- a turn status (beginning/middle/end of turn, full turn or backchannel), and
- a set of behaviors with associated synchronization constraints, where each behavior is one of either:
  - a gesture (gaze, point, nod, shake, etc.),
  - a phrase to be spoken, or
  - a reference to an object.

Turn fragments are encoded in an extension (EBML) to BML [15]. BML was chosen because it provides a rich language for expressing timing constraints between gestures and speech. Fig. 3(a) shows an example of a standard BML behavior block drawn from the tangram game described in Section V. This block will cause the robot to utter the sentence, "Please place this piece on the right side of the purple square," synchronized with initiation of a directed gaze connection event, where the gaze and pointing are synchronized with the phrase "this piece."

Fig. 3(b) is an example of a turn fragment in EBML, showing the new attributes added to BML to encode the turn index, turn status, and the new element type, discussed further in Section IV-C, to encode object references.

Notice that the turn fragment representation supports incremental processing of communication from the robot to the human. The collaboration manager is not required to produce an entire turn, or even a complete sentence, in a single call to the generation module. For example, the robot utterance above could be broken into two turn fragments, one for "Please place this piece," followed by one for "on the right side of the purple square."

```
<bml id="bml-example" xmlns="org.mindmakers.bml">
  <gaze id="gaze-1" target="green-triangle"/>
  <speech id="speech-1">
    <text>Please place</text> </speech>
  <gesture id="point-1" type="POINT"
    target="green-triangle"/>
  <speech id="speech-2">
    <text>this piece</text> </speech>
  <speech id="speech-3">
    <text>on the right side of</text> </speech>
  <speech id="speech-4">
    <text>the purple square</text> </speech>
  <constraint id="constraint-1">
    <synchronize ref="gaze-1:stroke">
      <sync ref="speech-1:start" /> </synchronize>
    <synchronize ref="gaze-1:stroke + 2">
      <sync ref="point-1:stroke_start" />
    </synchronize>
    <synchronize ref="point-1:stroke_end">
      <sync ref="speech-2:start" /> </synchronize>
    <synchronize ref="speech-2:end">
      <sync ref="speech-3:start" /> </synchronize>
    <synchronize ref="speech-3:end">
      <sync ref="speech-4:start" /> </synchronize>
  </constraint></bml>
```

**(a) BML behavior block.**

```
<ebml:ebml id="ebml-example"
    xmlns="org.mindmakers.bml"
    xmlns:ebml="edu.wpi.ebml"
    turnIndex="1" turnStatus="full" >
  <speech id="speech-1">
    <text>Please place</text> </speech>
  <ebml:reference id="ref-1"
      target="green-triangle"
      minReliability="0.85">
    <speechOption text="the triangle" cost="2" >
      <distractor object="blue-triangle" />
      <distractor object="yellow-triangle" />
    </speechOption>
    <speechOption text="the green triangle"
        cost="3" />
    <speechOption text="it" cost="1" />
    <speechOption text="this piece" cost="2"
                  deictic="true" >
      <distractor object="blue-triangle" />
      <distractor object="yellow-triangle" />
      <distractor object="red-square" />
      <distractor object="blue-square" />
      <!--- and the five other pieces --->
    </speechOption>
  </ebml:reference>
  <speech id="speech-3">
    <text>on the right side of</text> </speech>
  <ebml:reference id="ref-2"
      target="purple-square" minReliability="0.85">
    ...
  </ebml:reference>
  <constraint id="constraint-1">
    <synchronize ref="speech-1:end">
      <sync ref="ref-1:start" /> </synchronize>
    <synchronize ref="ref-1:end">
      <sync ref="speech-3:start" /> </synchronize>
    <synchronize ref="speech-3:end">
      <sync ref="ref-2:start" /> </synchronize>
  </constraint>
</ebml:ebml>
```

**(b) EBML turn fragment.**

Fig. 3. Behavior Markup Language examples.

### B. Turn Policy

The turn policy component (see Fig. 2) consumes turn fragments from the collaboration manager and produces turn fragments that are consumed by the reference policy, optionally adding end-of-turn facial gaze gestures. It also notifies the recognition module of the start of robot-initiated adjacency pair, mutual facial gaze and backchannel events, as appropriate. TABLE I details the actions of the turn policy, which depend on the *turn index* and *turn status* values. These attributes are removed in the turn policy's output.

| *Turn Status* | *Policy* |
|---|---|
| beginning | if first turn, initiate AP event |
| middle | |
| end | add facial gaze as last behavior<br>initiate MFG event |
| full turn | if first turn, initiate AP event<br>add facial gaze as last behavior<br>initiate MFG event |
| backchannel | initiate BC event |

TABLE I

TURN POLICY

### C. Reference Policy

The reference policy component (see Fig. 2) consumes turn fragments, possibly containing object references, from the turn policy and produces standard BML behavior blocks, which are then consumed by the BML realizer. The reference policy also notifies the recognition module of the start of robot-initiated directed gaze events, as appropriate.

An *object reference* comprises:

- an object identifier (uniquely identifying an object visible in the shared space),
- the minimum required reliability ($0 < R_{min} < 1$), and
- a non-empty set of referring phrase options, each of which specifies:
  - a phrase to be spoken (may be empty),
  - the distractor set after speaking the phrase,
  - the cost of speaking the phrase, and
  - whether the phrase is deictic (e.g. this or that).

The reference policy processes turn fragments by "passing through" the standard BML elements, removing the object reference elements, and replacing them with a choice of one of the phrase options, possibly with the addition of a gaze and/or pointing gesture.

Fig. 3(b) is an example of a turn fragment input to the reference policy which, under certain conditions, produces the BML output shown in Fig. 3(a). Notice that the reference policy has replaced each object reference (`ref-1` and `ref-2`) by a speech element chosen from its speech options (`speech-2` and `speech-4`, respectively), and added a gaze (`gaze-1`), a pointing gesture (`point-1`) and a synchronization constraint.

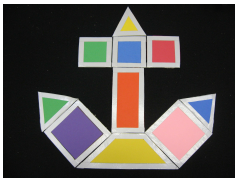


Fig. 4. Tangrams.

*1) Distractor Sets:* The concept of a *distractor set*, which is central to the reference policy, originates in natural language generation, specifically the generation of referring phrases [16]. Given some universe of objects, such as the tangram pieces on a table, a referring phrase and a target object, the distractor set is the set of objects that are consistent with the referring phrase, but are *not* the target.

For example, for the universe of ten tangram pieces in Fig. 4, which includes a blue, a yellow and a green triangle, the referring phrase "the triangle" (first speech option for `ref-1` in Fig. 3(b)) has two distractors, whereas the distractor set for "the green triangle" (the second speech option) is empty. As we will see below, we generalize the concept of distractor set to gestures, such as gaze and pointing. The distractor set for pointing is often empty, but if the pointer is large and/or far away, even pointing can be ambiguous.

*2) Optimal Gesture-Speech Pair:* The reference policy chooses the minimum cost combination of gesture and speech which satisfies the minimum reliability (see below) required for the given object reference. Formally, this is the gesture-speech pair $(j, s) \in J \times S$, which is

$$\underset{R(j,s) \geq R_{min}}{\operatorname{argmin}} \; C(j, s) \tag{1}$$

- $R(j, s)$ is the reliability function
- $C(j, s)$ is the cost function
- $S$ is the set of speech options ($\lambda$ is the empty phrase),
- $J = \{\lambda, G, DG, DGP\}$, and
  - $\lambda$ means no gesture,
  - $G$ means a gaze gesture (DG optional),
  - $DG$ means a directed gaze (DG required),
  - $DGP$ means a directed gaze with pointing.

If no gesture-speech pair satisfies the minimum required reliability, then a most reliable pair is chosen and an error message is returned to collaboration.

*3) Reliability:* The reliability of an object reference is intuitively the likelihood that it will be correctly understood. The *minimum required reliability* ($R_{min}$) may differ between object references due, for example, to the difference in importance for the collaborative task of correctly identifying the object.

In a world of perfect communication channels, the reliability of a referring phrase would depend only on the size of the distractor set—if it is zero, then the reliability is 1. However, in the real world, communication channels have noise, which means that the net communication reliability is always less than 1. The same analysis applies to the "channel" reliability of gestures—the lighting could be poor, the view could be partially obscured, etc., all of which could contribute to misunderstanding an otherwise unambiguous gaze or pointing.

Formally, the reliability of a gesture-speech pair is

$$R(j, s) = \frac{1}{|D(j, s)| + \frac{1}{r(j,s)}} \tag{2}$$

- $D(j, s)$ is the distractor set after performing $(j, s)$, and
- $0 < r(j, s) < 1$ is the channel reliability of $(j, s)$.

To understand the logical structure of this formula, consider first the hypothetical case where the channel reliability is 1. The overall reliability is then the likelihood that the correct object will be chosen by random chance out of the set which includes the target object and the distractors. On the other hand, suppose the distractor set is empty; the overall reliability is then just the channel reliability.

To simplify implementation of this policy, we assume that the channel reliabilities of gestures and speech are independent and can be characterized by simple parameters, i.e.,

$$r(j, s) = r(j) + r(s) \tag{3}$$

$$r(s) = \begin{cases} \lambda & : 0 \\ otherwise & : r_s \end{cases} \quad r(j) = \begin{cases} \lambda & : 0 \\ G & : r_G \\ DG & : r_{DG} \\ DGP & : r_{DGP} \end{cases}$$

The subscripted $r$ parameters can be tuned depending on the robot's configuration and relation to the human. Furthermore, we expect in general that $r_G < r_{DG} < r_{DGP}$.

The distractor set after performing a gesture-speech pair can be decomposed as

$$D(j, s) = D_g(j) \cap D_p(j) \cap D_s(s) \tag{4}$$

- $D_g(j)$ is the distractor set after executing the gaze portion, if any, of $j$
- $D_p(j)$ is the distractor set after executing the pointing portion, if any, of $j$
- $D_s(s)$ is the distractor set after uttering the possibly-empty referring phrase $s$

The distractor sets after executing a gesture also depend on the robot's configuration and relation to the human. Currently, we compute $D_g(j)$ by intersecting the gaze cone of the robot with the table top and including all objects that fall within the resulting ellipse (except the target object). We compute $D_p(j)$ by similarly intersecting a wide cylinder aligned with the pointing forearm of the robot with the table top. If there is no gaze or no pointing in $j$, or $s$ is empty, then the respective distractor sets are the universe minus the target object. As we have seen above, the distractor sets associated with each non-empty speech option, $D_s(s)$, are provided as input to the generation module by the collaboration manager as part of the object reference.

*4) Cost:* The cost of a gesture-speech pair is given as

$$C(j, s) = k_g C_g(j) + k_p C_p(j) + k_s C_s(s) \tag{5}$$

- $C_g(j) \geq 0$ is the cost of moving the robot's head from its current position to gaze at the given object,
- $C_p(j) \geq 0$ is the cost of moving the robot's appropriate hand from its current position to point at the target,
- $C_s(s) \geq 0$ is the cost of uttering the given referring phrase, and
- $C_g(\lambda) = C_p(\lambda) = C_s(\lambda) = 0$

Currently, we compute $C_g(j)$ and $C_p(j)$ by adding together the number of degrees of rotation required in each of the joints to achieve the required movement. The $k_g$, $k_p$ and $k_s$ coefficients in the cost formula above are parameters of the engagement generation module, which can be used to tune the policy for different robots and configurations. $C_s(j)$ is provided as input to the generation module as part of the object reference.

To generalize this policy to optimize sequences of object references, we add the robot state as a third argument to the cost functions, so that the optimal sequence of gesture-speech pairs $(j_i, s_i)$ for a sequence of object references is

$$\underset{R(j_i, s_i) \geq R_{min_i}}{\operatorname{argmin}\ (j_i, s_i)} \sum_{i=1..n} C(j_i, s_i, T(j_{i-1})) \tag{6}$$

- $T(j_i)$ is the robot state after executing gesture $j_i$ and
- $T(j_0)$ is the starting state of the sequence

### D. Response Policy

Both the reference and turn policies, discussed above, concern robot-initiated behavior (mostly at the behest of the collaboration manager). In contrast, the response policy component (see Fig. 2) deals with the robot's response to the start of human-initiated directed gaze and mutual facial gaze events. The current policy for how the robot should respond to these behaviors is simple: the robot always looks where the human directs and meets the human's facial gaze, except when gaze/point inhibition is active. Notice that the response policy does not include either providing the second turn of a human-initiated adjacency pair or responding with a backchannel to a human utterance. Both of these cases require the higher cognitive abilities (collaboration manager) of the robot because they depend on task context.

### E. Maintenance Policy

Because engagement is about how people "*maintain* ... their perceived connection," [1] a key component of the generation module is the engagement maintenance policy. Currently, this policy is only concerned with the robot's gaze. In the future (see Section VI), we plan to add nodding and other nonverbal behaviors to the robot's engagement maintenance repertoire.

The maintenance policy (see Fig. 2) is organized as the following set of priorities regarding where the robot looks (starting with the highest priority):

- obey gaze inhibit input
- obey gaze output of reference and turn policies
- track the human's hands if they are moving
- a quick glance at the human's face if the elapsed time since the last connection event exceeds the mean time between connection events (MTBCE) [2]
- track the human's face

These priorities are based on our own observational study of human engagement [2] and other human gaze interaction research cited above. The two external inputs to the maintenance policy (see Fig. 2) are symbolic vision, which supports the hand tracking, and statistics from the engagement recognition module, which provide the MTBCE.

## V. Validation

As validation of the generation policies and implementation above, we have developed a simple human-robot demonstration, which we call the "tangram game" (see Fig. 1(b)), that naturally includes all four types of connection events. Our humanoid robot, Melvin, was built by Michaud *et al.* at U. Sherbrooke (Canada). We used Morency's Watson system [17] for human face and gaze tracking and detecting head nods and shakes, and OpenCV to implement symbolic vision. Since the focus of our research is on engagement and collaboration, we simplified the robot's vision problem as much as possible using distinct colors and shapes. Our collaboration manager was specially written for this game.

The game starts with the tangram pieces randomly arranged on the table. The goal of the game is to arrange the pieces to form the anchor shape shown in Fig. 4, but only the robot knows the target shape. The robot mostly

| Question | Oper. Avg. | Deg. Avg. | Std. Err. | p-value |
|---|---|---|---|---|
| 1) Melvin seemed more like a human than a robot. | 3.43 | 2.27 | .55 | $< 0.05$ |
| 2) Melvin looked at the table and the puzzle pieces at appropriate times. | 6.21 | 4.47 | .73 | $< 0.05$ |
| 3) Melvin always looked at me in a natural way. | 4.57 | 2.73 | .58 | $< 0.01$ |
| 4) Melvin looked at me at appropriate times. | 6.21 | 2.40 | .46 | $< 0.001$ |
| 5) I always knew what object Melvin looked at. | 6.07 | 3.80 | .83 | $< 0.05$ |
| 6) I could easily tell which objects Melvin looked at. | 5.71 | 3.40 | .72 | $< 0.01$ |
| 7) I looked at Melvin's face often. | 6.07 | 5.00 | .50 | $< 0.05$ |
| 8) I made eye contact with Melvin frequently. | 5.43 | 4.00 | .62 | $< 0.05$ |
| 9) I always knew what object Melvin pointed at. | 6.57 | 3.13 | .72 | $< 0.001$ |
| 10) I could easily tell the object that Melvin pointed to. | 6.43 | 3.07 | .70 | $< 0.001$ |

TABLE II
QUESTIONNAIRE RESULTS

directs the human with utterances and gestures such as in Fig. 3(a). However, there is also opportunity for limited human initiative when the robot says, "Please point at the next piece you want to move."

Using the tangram game, we conducted a between-subjects study with 29 participants, all students at WPI ages 18 to 26, in two conditions: operational and degraded. The operational condition uses the complete system described in this paper. In the degraded condition, all of the policies are disabled, and the robot is always looking up to the right[1]. After completing the tangram, each participant was given a questionnaire using a Likert scale ranging from 1 (strongly disagree) to 7 (strongly agree). TABLE II compares these questions in the two conditions using a two-tailed p-test.

| | Oper. | Deg. |
|---|---|---|
| male | 10 | 12 |
| female | 4 | 3 |

TABLE III
PARTICIPANT DISTRIBUTION

This experiment shows that the generation policies made a significant improvement in the robot's performance as a collaborator. Question 1 attests to the robot being perceived as more human-like. Questions 2 through 4 show proof of Melvin performing the correct actions for a fluent and natural interaction. The remaining questions show that the participants believed that they understood Melvin's behaviors. These results support our belief that to interact with people, robots should have fluent and natural behaviors that are similar to, if not exactly like, human behaviors. The results also show that people will reciprocate to connection events that the robot initiates.

## VI. FUTURE WORK

Our first order of business is to conduct some pilot studies to tune the various parameters in the policies, such as the cost coefficients in Section IV-C.4. Also, there are a number of small improvements in architecture and policies we would like to explore. For example, as mentioned in Section IV-D, we currently require the collaboration manager to handle the generation of backchannels, such as confirmatory nods. We would like to find a way to off-load this behavior to the generation module. We would also like to explore using our robot's—albeit limited—abilities for emotional expression (eyebrows and mouth) to contribute to engagement.

Finally, there is much more work to be done on strategies and policies for initiating and terminating engagement. A (very) challenging test case for this work would be for a robot to start up a conversation with a chosen person at a cocktail party, and then later gracefully end the conversation.

[1]A pilot study showed that if the robot looked straight ahead or down at the table, participants sometimes concluded that the robot was intentionally making eye contact or paying attention to their actions.

## REFERENCES


[1] C. L. Sidner, C. Lee, C. Kidd, N. Lesh, and C. Rich, "Explorations in engagement for humans and robots," *Artificial Intelligence*, vol. 166, no. 1-2, pp. 104–164, 2005.

[2] C. Rich, B. Ponsler, A. Holroyd, and C. Sidner, "Recognizing engagement in human-robot interaction," in *Proc. ACM Conf. on Human-Robot Interaction*, Osaka, Japan, Mar. 2010.

[3] M. Argyle and M. Cook, *Gaze and Mutual Gaze*. Cambridge University Press, New York, 1976.

[4] A. Kendon, "Some functions of gaze direction in two person interaction," *Acta Psychologica*, vol. 26, pp. 22–63, 1967.

[5] S. Duncan, "Some signals and rules for taking speaking turns in conversations," *Journal of Personality and Social Psychology*, vol. 23, no. 2, pp. 283–292, 1972.

[6] C. Goodwin, "Gestures as a resource for the organization of mutual attention," *Semiotica*, vol. 62, no. 1/2, pp. 29–49, 1986.

[7] Y. Nakano, G. Reinstein, T. Stocky, and J. Cassell, "Towards a model of face-to-face grounding," in *Proc 41st Meeting of Assoc. for Computational Linguistics*, Sapporo, Japan, 2003, pp. 553–561.

[8] H. H. Clark, *Using Language*. Cambridge: Cambridge Univ. Press, 1996.

[9] C. Peters, "Direction of attention perception for conversation initiation in virtual environments," in *Proc. 5th Int. Conf. Intelligent Virtual Agents*, Kros, Greece, 2005, pp. 215–218.

[10] D. Bohus and E. Horvitz, "Models for multiparty engagement in open-world dialog," in *Proc. SIGDIAL 2009 Conference*, London, UK: Association for Computational Linguistics, Sept. 2009, pp. 225–234.

[11] B. Mutlu, T. Shiwa, T. Kanda, H. Ishiguro, and N. Hagita, "Footing in human-robot conversations: How robots might shape participant roles using gaze cues," in *Proc. ACM Conf. on Human-Robot Interaction*, San Diego, CA, 2009.

[12] D. Crystal, *The Cambridge Encyclopedia of Language*. Cambridge, England: Cambridge University, 1997.

[13] F. Flippo, A. Krebs, and I. Marsic, "A framework for rapid development of multimodal interfaces," in *Proc. 5th Int. Conf. Multimodal Interfaces*, Nov. 2003, pp. 109–116.

[14] H. Sacks, E. A. Schegloff, and G. Jefferson, "A Simplest Systematics for the Organization of Turn-Taking for Conversation," *Language*, vol. 50, no. 4, pp. 696–735, 1974.

[15] H. Vilhjálmsson, N. Cantelmo, J. Cassell, N. E. Chafai, M. Kipp, S. Kopp, M. Mancini, S. Marsella, A. N. Marshall, C. Pelachaud, Z. Ruttkay, K. R. Thórisson, H. Welbergen, and R. J. Werf, "The behavior markup language: Recent developments and challenges," in *Proc. of the 7th Intl. Conf. on Intelligent Virtual Agents*, ser. IVA '07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 99–111.

[16] C. Gardent, "Generating minimal definite descriptions," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ser. ACL '02. Morristown, NJ, USA: Association for Computational Linguistics, 2002, pp. 96–103.

[17] L. P. Morency, A. Rahami, and T. Darrell, "Adaptive view-based appearance model," in *IEEE Conf. on Computer Vision and Pattern Recognition*, Madison, WI, June 2003, pp. 803–810.