Appraisal in Human-Robot Collaboration

Mahni Shayganfar, Charles Rich, Candace L. Sidner
Worcester Polytechnic Institute
Computer Science Department
100 Institute Road
Worcester, Massachusetts 01609
mshayganfar | rich | sidner @wpi.edu

Abstract

We have investigated the mutual influence of affective and collaboration processes in a cognitive theory to support the interaction between humans and robots or virtual agents. We have developed new algorithms for appraisal processes, as part of a new overall computational model for implementing collaborative robots and agents. We build primarily on the *cognitive appraisal* theory of emotions (Gratch and Marsella 2004) and the *SharedPlans* theory (Grosz and Sidner 1990) of collaboration to investigate the structure, fundamental processes and functions of emotions in a collaboration.

Introduction

Ronald De Sousa in The Rationality of Emotion (Sousa 1990) makes a good case for the claim that humans are capable of rationality largely because they are creatures with emotions. The idea of having robots or other intelligent agents living in a human environment has been a persistent dream from science fiction books to artificial intelligence and robotics laboratories. Collaborative robots are becoming an integral part of humans' environment to accomplish their industrial and household tasks. In these environments humans are involved in robots' operations and decision making processes. This involvement of humans influences the efficiency of robots' interaction and performance, and makes them dependent on the humans' cognitive abilities and mental states.

Our work is part of a larger effort to build robots capable of generating and recognizing humans' emotions in order to be better collaborators. In this paper, we are reporting on the specific problem of appraising events within a collaborative interaction. Our contribution is grounding the general appraisal concepts in the specific context and structure of collaboration. This work is part of the development of our framework based on our *Affective Motivational Collaboration Theory* which is built on the foundations of the *SharedPlans* theory of collaboration (Grosz and Sidner 1990) and the *cognitive appraisal* theory of emotions (Gratch and Marsella 2004).

In the rest of this paper, we are going to briefly introduce the Affective Motivational Collaboration Theory focusing on the collaboration and appraisal mechanisms as well as mental states. We are also going to provide more details about the graph representation of the robot's mental state. Finally, we describe our algorithms which we developed to compute the value of four crucial appraisal variables as part of the appraisal mechanism in our framework.

Affective Motivational Collaboration Theory

Affective Motivational Collaboration Theory is about the interpretation and prediction of observable behaviors in a dyadic collaborative interaction. The theory focuses on the processes regulated by emotional states. The observable behaviors represent the outcome of reactive and deliberative processes related to the interpretation of the self's relationship to the collaborative environment. Affective Motivational Collaboration Theory aims to explain both rapid emotional reactions to events as well as slower, more deliberative responses. The reactive and deliberative processes are triggered by two types of events: *external* events, such as the other's *utterances* and *primitive actions*, and *internal* events, comprising changes in the self's mental states, such as belief formation and emotional changes. The theory explains how emotions regulate the underlying processes when these events occur during collaboration. It also elucidates the role of motives as goal-driven emotion-regulated constructs with which an agent can form new intentions to cope with events.

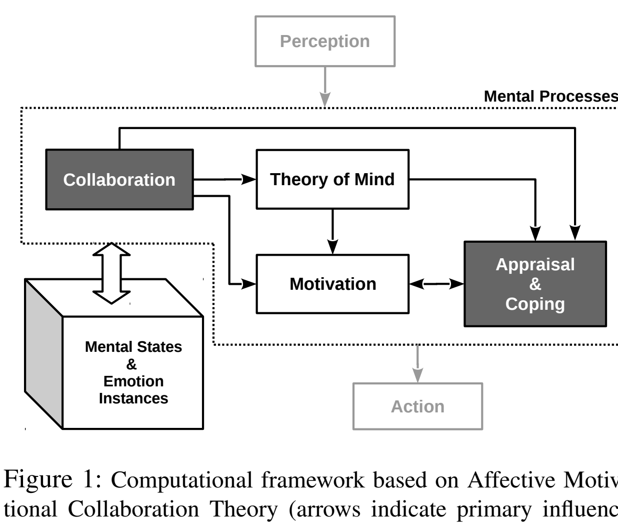

Figure 1: Computational framework based on Affective Motivational Collaboration Theory (arrows indicate primary influences between mechanisms).

Affective Motivational Collaboration Theory explains the functions of emotions in a dyadic collaboration and shows how affective mechanisms can coordinate social interactions by enabling one to anticipate other's emotions, beliefs and intentions. Our focus is on the mechanisms depicted as mental processes in Figure 1 along with the mental states. The *Mental States* includes self's (robot's) beliefs, intentions, motives, goals and emotion instances as well as the anticipated Mental States of the other (human). The *Collaboration* mechanism maintains constraints on actions, including task states and the ordering of tasks. The *Collaboration* mechanism also provides processes to update and monitor the shared plan. The *Appraisal* mechanism is responsible for evaluating changes in the self's Mental States, the anticipated Mental States of the other, and the state of the collaboration environment. The *Coping* mechanism provides the self with different coping strategies associated with changes in the self's mental states with respect to the state of the collaboration. The *Motivation* mechanism operates whenever the self a) requires a new motive to overcome an internal impasse in an ongoing task, or b) wants to provide an external motive to the other when the other faces a problem in a task. The *Theory of Mind* mechanism is the mechanism that infers a model of the other's anticipated mental state. The self progressively updates this model during the collaboration.

Mental States

A brief description of mental states is provided as prerequisite knowledge for understanding the appraisal processes. The Mental States shown in Figure 1 comprise the knowledge base required for all the mechanisms in the overall model. Mental states are conscious states of the mind providing the content for cognitive processes. Affective Motivational Collaboration Theory operates with the following Mental States: beliefs, intentions, motives, goals and emotion instances. These Mental States possess attributes, each of which provides a discriminating and unique interpretation of the related cognitive entities. The self uses Mental States' attributes whenever there is an arbitration in the internal cognitive processes. We provide more details about the attributes of beliefs and motives in this paper, since they are used in our appraisal algorithms.

Belief: *Beliefs* are a crucial part of the Mental States. We have two different perspectives on categorization of beliefs. In one perspective, we categorize beliefs based on whether they are shared between the collaborators. The SharedPlans (Grosz and Sidner 1990) theory is the foundation of this categorization in which for any given proposition the agent may have: a) private beliefs (the agent believes the human does not know these), b) the inferred beliefs of the human (the agent believes the human collaborator has these beliefs), and c) mutual beliefs (the agent believes both the self and the human have these same beliefs and both of them believe that). From another perspective, we categorize beliefs based on who or what they are about. In this categorization, beliefs can be about the self, the other, or they can be about the environment. Beliefs about the environment can be about internal events, such as outcomes of a new appraisal or a new motive, or external events such as the human's offer, question or request, and general beliefs about the environment in which the agent is situated. Beliefs can be created and updated by different processes. They also affect how these processes function as time passes.

The attributes of a belief are involved in arbitration procedures within different processes in Affective Motivational Collaboration Theory. They impact a range of these processes from the formation of new beliefs, the evaluation of an external event by the Appraisal mechanism, generation of new motives and updates on collaboration plan, to the activation of coping strategies and ultimately the self's behavior. We use the following six attributes of beliefs in our framework.

- ***Strength:*** Belief strength is about how strongly the self holds salient beliefs about an object, an entity, or an anticipated behavior.
- ***Accuracy:*** Accuracy of a belief is the relation between that belief and the truth which that belief is about.
- ***Frequency:*** The frequency of a belief is related to how regularly it appears as the result of an internal or an external event.
- ***Recency:*** The recency of a belief refers to how temporally close a particular belief is to the current state of collaboration.
- ***Saliency:*** The saliency of a belief is a cognitive attribute that pertains to how easily the self becomes aware of a belief.
- ***Persistence:*** The persistence of a belief refers to how resistant the belief is to changes.

Motive: *Motives* are mental constructs which can initiate, direct and maintain goal-directed behaviors. They are created by the emotion-regulated Motivation mechanism. Motives can cause the formation of a new intention for the agent according to: a) its own emotional states (how the agent appraises the environment), b) its own private goal (how an action helps the agent to make progress), c) the collaboration goal (how an action helps to achieve the shared goal), and d) other's anticipated beliefs (how an action helps the other). Motives can be compared on various dimensions (Sloman 1987), and they possess a set of attributes. The Motivation mechanism compares motives based on the quality of these attributes and chooses the one which is the most related to the current state of the collaboration. We have the following five motive attributes in our framework.

- ***Insistence:*** The insistence of a motive defines the "interrupt priority level" of the motive, and how much that motive can attract the self's focus of attention.
- ***Importance:*** The importance of a motive is determined by the corresponding beliefs about the effects of achieving or not achieving the associated goal.
- ***Urgency:*** The urgency of a motive defines how much time the self has to acknowledge and address that motive before it is too late.

- ***Intensity:*** The intensity of a motive determines how actively and vigorously that motive can help the self to pursue the goal if adopted; rather than abandoning the goal and ultimately the collaboration.
- ***Failure Disruptiveness:*** The failure disruptiveness attribute of a motive determines how disruptive failure is to achieving the corresponding goal.

Intention: *Intentions* are mental constructs directed at goals and future actions. They play an essential role in taking actions according to the collaboration plan as well as behavior selection in the Coping mechanism. Intentions are also involved in selecting intention-related strategies, e.g., planning, seeking instrumental support and procrastination. Intentions possess a set of attributes, i.e., *Temporal Status, Direct Experience, Certainty, Ambivalence, Affective-Deliberative Consistency* which moderate the consistency between intention and behavior (Cooke and Sheeran 2004). The details about these attributes are out of this paper's context.

Goal: *Goals* help the agent to create and update its collaboration plan according to the current private and shared goal content and structure, i.e., the *Specificity, Proximity* and *Difficulty* of the goal. Goals direct the formation of intentions to take appropriate corresponding actions during collaboration. Goals also drive the Motivation mechanism to generate required motive(s). The details about goal's attributes are also out of this paper's context.

Emotion Instance: *Emotions* in Mental States are emotion instances that are elicited by the Appraisal mechanism, e.g., *Joy, Anger, Hope, Worry*. These emotion instances include the agent's own emotions as well as the anticipated emotions of the other which are created with the help of the processes in the Theory of Mind mechanism. Each emotion has its own functionality in either the intrapersonal or interpersonal level. These emotions not only regulate the self's internal processes, but also assist the self to anticipate the other's Mental States.

Example Scenario

The example scenario shows a very short part of an interaction between a robot and an astronaut during their collaboration. Their mission is to finish installing a few solar panels together. However, the astronaut encounters a measurement tool problem:

Astronaut [turn t-1]: Oh no! Finishing the quality check of our installation with this measurement problem is so frustrating. I think we should stop now!

Robot [turn t]: I see. This is frustrating. But, I can help you with the measurement tool and we can finish the task as originally planned.

As shown, the robot in turn t, acknowledges the astronaut's frustration and appropriately responds to her problem. We will use this example in the Mental Graph section to clarify some of the concepts. The underlined utterance of the robot will be used in section Mental Graph.

Collaboration

The Collaboration mechanism constructs a hierarchy of goals associated with tasks in the form of a Hierarchical Task Network (see Figure 2), and also manages and maintains the constraints and other required details of the collaboration specified by the plan. These details include the inputs and outputs of individual tasks, the *preconditions* specifying whether it is appropriate to perform a task, and the *postconditions* specifying whether a just-completed task was successful (i.e., whether the corresponding goal is achieved). Collaboration also keeps track of the focus of attention, which determines the salient objects, properties and relations at each point of the collaboration. Moreover, Collaboration has the ability to shift the focus of attention during the interaction.

The Collaboration mechanism receives the data that affects the execution of individual tasks in the collaboration plan. This data is provided via the different elements of Mental States including beliefs, intentions and goals. The Collaboration mechanism generates the data that is modified or created during execution of a plan in the form of Mental States which will be used by other processes in our framework. For instance, the Appraisal mechanism uses these Mental States to evaluate the events during collaboration.

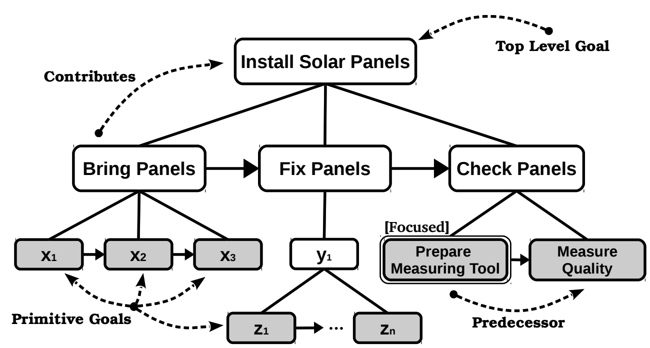

Figure 2: Collaboration structure.

Here, we briefly describe the methods which provide detail information about the collaboration structure, and are used in our algorithms to compute the values of appraisal variables.

- The *recognizeGoal()* method returns the unique goal to which the given event directly contributes irrespective of the event type (i.e., action, utterance, or emotional expression). If this method does not recognize a goal in the plan to which the event contributes, we consider the corresponding goal *ambiguous* in our processes.
- The *topLevelGoalStatus()* method returns the status of the top level goal whether it is *achieved, blocked* or *in progress*. In our example, "Install Solar Panels" is the top level goal (see Figure 2).
- The *currGoalStatus()* method returns the current goal status whether it is *achieved, blocked, unknown* or *in progress*. In our example, "Prepare Measurement Tool" is the current goal.

- The *precondStatus()* method returns the status of the precondition for the given goal whether it is *satisfied, unsatisfied* or *unknown*. For instance, the precondition for fixing a panel is whether the panel is appropriately located on its frame.
- The *doesContribute()* method returns whether the given goal contributes to another goal in the higher level of the plan hierarchy. For instance, an abstract (nonprimitive) goal of "Bring Panels" contributes to the higher level goal of "Install Solar Panels" (see Figure 2).
- The *extractContributingGoals()* method returns all the contributing goals of the given goal. For instance, as shown in Figure 2 the "Prepare Measuring Tool" and "Measure Quality" are two goals contributing to the "Check Panels" nonprimitive goal.
- The *extractPredecessors()* method returns the predecessors of the given goal. For instance, as shown in Figure 2, the "Prepare Measuring Tool" goal is the predecessor of another goal called "Measure Quality".
- The *extractInputs()* method returns all the required inputs for the given goal. For example, the goal "Fix Panels" requires inputs such as the *welding tool* and the *panel*.
- The *isAvailable()* method returns whether the given input is available. For instance, if the *welding tool* is required for the goal "Fix Panels", is it available now?
- The *isAchieved()* method returns whether the given goal is achieved. In other words, whether all the postconditions of the given goal are *satisfied*.
- The *isFocused()* method returns whether the focus is on given goal now. As shown in Figure 2, the focus is on the goal "Prepare Measuring Tool". The focused goal is the goal that the robot currently is pursuing.
- The *getResponsible()* method returns responsible agents of the given goal. In a dyadic collaboration, both of the agents can be partly responsible for a nonprimitive goal, while each agent is responsible for one or more primitive goal. For instance, both the robot and the astronaut are responsible for the nonprimitive goal of "Install Solar Panels", whereas it is only the astronaut who is responsible for the primitive goal of "Prepare Measuring Tool".

Mental Graph

In this section, we provide a graph representation of the robot's mental state for the turn t based on our example scenario. We only illustrate a part of the robot's mental states in turn t which is related to the event occurring based on the astronaut's utterances in turn t-1. This section is provided because all of our algorithms use mental graphs to compute values of appraisal variables.

In Figure 3, we illustrate all five elements of robot's mental states, i.e., belief, motive, intention, goal, and emotion, and the connections between them. The belief node $b_{t-2}^{\{1,..,n\}}$ represents n number of beliefs that are carried over from the robot's previous mental state (shown as one node for simplicity). These beliefs can be a combination of the robot's private, inferred, and mutual beliefs as well as beliefs about self, other and the environment. These beliefs have influence on motive formation and emotion elicitation processes. However, there is another node representing belief b_{t-1}^{ε} which is the belief about the event ε_{t-1} (astronaut's utterances). As we presented in our example scenario, the astronaut's utterances in turn t-1 depict a problem in a measurement tool and consequently her frustration, which can lead to an unsuccessful termination of the collaboration. This new belief also impacts the current motive m_t^{δ} and emotion e_t^{δ}. Note that the superscript δ used for motive m_t^{δ} and emotion e_t^{δ} indicates a specific value for each of these mental states that is selected from among other possible options based on an arbitration process. The motive m_t^{δ} (the need to acknowledge the astronaut's emotion) leads to the formation of a required intention i_t (acknowledge astronaut's emotion) to achieve the current goal g_t (e.g., Prepare Measurement Tool).

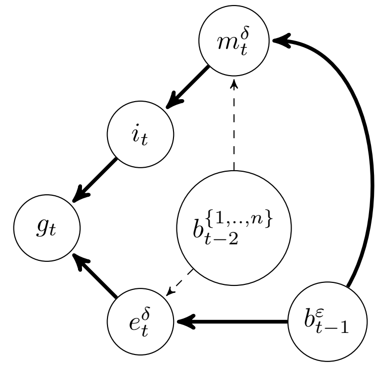

Figure 3: Part of robot's mental graph in turn t.

In our algorithms, we use *FINDPATH()* function to find the shortest path between two given node based on the Dijkstra algorithm.

Appraisal Processes

All of the algorithms in this section use a mental graph which is a directed acyclic graph constructed based on the mental states of the robot at each turn during the collaboration. They also use the content of the most recent event at each turn which includes the recognized goal associated with the event and the belief about the event. The goals can be associated with any type of event including action, utterance, or emotional expression.

We found four appraisal variables, i.e., *Relevance* (Algorithm 1), *Desirability* (Algorithm 2), *Expectedness* (Algorithm 3), and *Controllability* (Algorithm 4) to be the most important variables in a collaboration context. There are other appraisal variables introduced in psychological (Scherer, Schorr, and Johnstone 2001) and computational literature (Gratch and Marsella 2004); we do not provide algorithms for them in this paper. We believe most of these variables can be straightforwardly added to our appraisal mechanism later, e.g., *Perspective* (i.e., the view point from which an event is judged).

Relevance

Relevance as an appraisal variable measures the significance of an event for the robot. An event can be evaluated to be relevant if it has a positive utility or it can causally impact a state with a positive utility (Marsella and Gratch 2009). Relevance is an important appraisal variable since the other appraisal dimensions are only derived for the relevant events.

Algorithm 1 determines the relevance of the given event with respect to the shared goal. An event is *irrelevant* if there is no connection between the event and the current shared goal. If there is a connection, the relevance of the event will depend on the significance of the event with respect to the current collaboration status. The significance of an event is determined based on the utility of the event as it is also presented in (Gratch and Marsella 2004; Marsella and Gratch 2009). We believe although the utility of the event represents the significance of the event, the other collaborator's expressed emotion also plays a role by influencing the significance of the utility through a threshold. As a result, evaluating the relevance of the events can cause a collaborative robot to effectively respond only to the events which can positively impact the status of the shared goal without dedicating all other resources to every single event. The relevance process also benefits the information that the collaboration structure contains, e.g., shared goal.

The algorithm starts by taking the belief about the current event (b_{ε_t}) and the shared goal (g_t) associated with the current mental graph. After perceiving an event, it is the belief about that event which represents the event in robot's mental state. Continuing in line 3, the g_t represents the shared goal (in the mental graph) at time (turn) t within the shared plan. Then, we find the shortest path ($\mathcal{P}$) between the corresponding nodes of b_{ε_t} and g_t in the mental graph. As we mentioned earlier, if there is no path between the belief about the current event and the active shared goal, the algorithm finds the event *irrelevant* to the current shared goal, and terminates.

However, if there is a path between b_{ε_t} and g_t, we need a way to determine whether the event is relevant to the current collaboration status. Therefore, first, we compute the utility of the event ($\mathcal{U}$) such that ($0 \leq \mathcal{U} \leq 1$) based on the values of the attributes associated with the belief about that event (b_{ε_t}), as well as the attributes of the motive within the path $\mathcal{P}$. Yet, the significance of an event in a collaborative environment not only is based on the utility of the event, but it is also influenced by the emotional state of the other collaborator. In other words, the relevance of a belief about an event is influenced by the perceived emotion of the human collaborator. The human's emotion influences the decision about the utility of the event in form of a threshold value (τ_t). For instance, a positive expressed emotion of the human reduces the threshold value which consequently makes the robot find an event relevant with even a slightly positive utility.

We compute this threshold value (τ_t) using a Fuzzy Logic system on a three-dimensional space of the somatic markers associated with the other's expressed emotion as they are described in (Breazeal 2002). In this space every emotion can be mapped to a vector of three values, i.e., *Arousal, Valence* and *Stance*. The details about this process is out of this paper's context. Finally, we make our decision about the relevance of an event with respect to the human's emotional state. Consequently, an event can be considered *irrelevant* even though there is a path between b_{ε_t} and g_t.

Algorithm 1 (Relevance)

1: **function** IsEventRelevant(MentalGraph $\mathcal{G}_t$, Event ε_t)
2: $\quad b_{\varepsilon_t} \leftarrow$ getBelief(ε_t)
3: $\quad g_t \leftarrow$ *currGoal*($\mathcal{G}_t$)
4: $\quad \mathcal{P} \leftarrow$ findPath(b_{ε_t}, g_t)
5: $\quad$ **if** ($\mathcal{P} = \emptyset$) **then**
6: $\qquad$ **return** IRRELEVANT
7: $\quad$ **else**
8: $\qquad \mathcal{U} \leftarrow$ getEventUtility(b_{ε_t}, g_t)
9: $\qquad \tau_t \leftarrow$ getEmotionalThreshold($\mathcal{G}_t$)
10: $\qquad$ **if** ($\mathcal{U} \geq \tau_t$) **then**
11: $\qquad\quad$ **return** RELEVANT
12: $\qquad$ **else**
13: $\qquad\quad$ **return** IRRELEVANT

Desirability

Desirability characterizes the value of an event to the robot in terms of whether the event facilitates or thwarts the collaboration goal. Desirability captures the valence of an event with respect to the robot's preferences (Gratch and Marsella 2004). In a collaborative robot preferences are biased towards those events facilitating progress in the collaboration. An event is desirable if it somehow facilitates the state of the shared goal, or if it inhibits the state of a goal that is inconsistent with respect to the shared goal.

Desirability plays an important role in the overall architecture; it makes the processes involved in the other mechanisms (e.g., Motivation or Theory of Mind) and consequently the robot's mental state congruent with the collaboration status which is the robot's desire. Therefore, it causes the robot to dismiss events causing inconsistencies in the robot's collaborative behavior. Moreover, desirability is also crucial from the collaboration's point of view. A collaborative robot needs to know whether its own and the other collaborator's actions, utterances, and emotional expressions are desirable in terms of their consistence with the status of the current shared goal. In other words, the collaboration mechanism uses the appraisal process of desirability to coordinate what the self or the other does, says, and expresses during collaboration. Reciprocally, the appraisal mechanism and in this case the desirability process use the collaboration structure to obtain their required information.

Algorithm 2 provides a process in which the desirability of an event is computed with regard to the status of the shared goal; i.e., it operates based on whether and how the event changes the status of the current shared goal. It receives the current mental graph, $\mathcal{G}_t$, and the current event, ε_t, from input, and decides whether and how the event is desirable or undesirable. First, the algorithm checks the status of the collaboration's top level goal (lines 2 to 6), and if the the top level goal is still in progress, it continues by checking the status of the current shared goal (lines 7 to 13). If any of the top level and current shared goals are achieved in these two steps, the robot interprets the event as a desirable one. However, if any of these goals are blocked, the event will be considered undesirable by the robot.

Algorithm 2 (Desirability)

1: **function** ISEVENTDESIRABLE(MentalGraph $\mathcal{G}_t$, Event ε_t)
2:   **if** (*topLevelGoalStatus()* = ACHIEVED) **then**
3:     **return** HIGHEST-DESIRABLE
4:   **else if** (*topLevelGoalStatus()* = BLOCKED) **then**
5:     **return** HIGHEST-UNDESIRABLE
6:   **else if** (*topLevelGoalStatus()* = INPROGRESS) **then**
7:     **if** (*currGoalStatus()* = ACHIEVED) **then**
8:       **return** HIGH-DESIRABLE
9:     **else if** (*currGoalStatus()* = BLOCKED) **then**
10:       **return** HIGH-UNDESIRABLE
11:     **else if** (*currGoalStatus()* = INPROGRESS) **then**
12:       **return** MEDIUM-DESIRABLE
13:     **else if** (*currGoalStatus()* = UNKNOWN) **then**
14:       $g_{\varepsilon_t} \leftarrow recognizeGoal(\varepsilon_t)$
15:       **if** (g_{ε_t} = AMBIGUOUS) **then**
16:         **return** HIGH-UNDESIRABLE
17:       **if** (*precondStatus*(g_{ε_t}) = SATISFIED) **then**
18:         **return** LOW-DESIRABLE
19:       **else if** (*precondStatus*(g_{ε_t}) = UNSATISFIED) **then**
20:         **return** LOW-UNDESIRABLE
21:       **else if** (*precondStatus*(g_{ε_t}) = UNKNOWN) **then**
22:         $g_t \leftarrow currGoal(\mathcal{G}_t)$
23:         **if** ($doesContribute(g_{\varepsilon_t}, g_t)$) **then**
24:           **return** NEUTRAL
25:         **else if** ($\neg doesContribute(g_{\varepsilon_t}, g_t)$ **then**
26:           **return** MEDIUM-UNDESIRABLE

The algorithm continues in the case of an unknown status of the current shared goal, and checks whether the precondition(s) of the associated goal with the current event, g_{ε_t}, are satisfied (lines 17 to 21). The robot prefers the satisfied preconditions and interprets the event as desirable while unsatisfied preconditions are undesirable for the robot. For instance, a satisfied precondition of a future goal is still desirable for the robot to some extent. Note that the robot also checks the ambiguity of the associated goal with the current event (line 15). An ambiguous goal is a goal which is not recognized in the robot's plan, and it is undesirable for the robot. After all, if the preconditions of the associated goal with the current event are unknown, the robot checks whether this goal, g_{ε_t}, contributes to the current shared goal, g_t (lines 22 to 25). As a result a contributing goal will obtain a neutral desirability in comparison with a noncontributing goal which will be undesirable for the robot.

Expectedness

Expectedness is the extent to which the truth value of a state could have been predicted from causal interpretation of an event (Marsella and Gratch 2009). In the collaboration context the expectedness of an event measures the congruency of the event with respect to the existing knowledge about the shared goal. Therefore, expected events are those of which beliefs about them are congruent to the status of the collaboration since their associated goals are expected with respect to the shared plan.

Expectedness underlies a collaborative robot's attention by evaluating the congruence of events with respect to the structure of an existing shared plan. Congruent beliefs in a robot's mental state will lead to more consistent and effective outcomes of the processes in the overall architecture. Therefore, a collaborative robot uses expectedness to maintain its own mental state towards the shared goal. The robot will also be able to respond to unexpected but relevant events. As a result, the collaboration mechanism uses expectedness to maintain the robot's attention and subsequently its mental state with respect to the shared goal. In parallel, the appraisal mechanism uses the underlying information of the collaboration structure to evaluate the expectedness of an event.

Algorithm 3 (Expectedness)

function ISEVENTEXPECTED(MentalGraph $\mathcal{G}_t$, Event ε_t)
  $g_t \leftarrow currGoal(\mathcal{G}_t)$
  $g_{\varepsilon_t} \leftarrow recognizeGoal(\varepsilon_t)$
  if (g_{ε_t} = AMBIGUOUS) **then**
    return UNEXPECTED
  if ($g_t = g_{\varepsilon_t}$) **then**
    return EXPECTED
  else
    if ($\neg isAchieved(g_t)$) **then**
      return UNEXPECTED
    else
      if ($\neg isAchieved(g_t.parent)$) **then**
        if ($doesContribute(g_{\varepsilon_t}, g_t.parent)$) **then**
          return EXPECTED
        else
          return UNEXPECTED
      else
        if ($doesContribute(g_{\varepsilon_t}, g_t)$) **then**
          return EXPECTED
        else
          if ($isFocused(g_{\varepsilon_t})$) **then**
            return EXPECTED
          else
            return UNEXPECTED

In algorithm 3 we provide the process of the expectedness based on the shared plan and status of the shared goal. The key point in this algorithm is the status of the current shared goal (g_t) and its relationship with the goal associated with the current event (g_{ε_t}). The algorithm receives the current mental graph, $\mathcal{G}_t$, and the current event, ε_t, from input, and decides whether the current event is expected.

First, we need to extract the goal in the current mental graph and the recognized goal associated with the current event. Similar to the desirability algorithm (Algorithm 2), we check whether the g_{ε_t} is ambiguous. In the case of ambiguity in g_{ε_t}, we consider the current event unexpected since an effective collaboration requires perceivable and unambiguous goals associated with the events. We continue by the comparison of the current shared goal and the recognized goal associated with the current event with respect to the shared plan. If these two goals are not the same, it is possible that the current shared goal is already achieved. The event will be unexpected (line 10) if the current shared goal is not achieved and the current event does not refer to the same goal. However, if the current goal is achieved, it is important to see whether its parent is also achieved (line 12). This step is important because the event can be expected if the new goal contributes to the parent of the recently achieved goal. Therefore, if the parent goal in the hierarchical plan is not achieved, the contribution of the associated goal to the current event can help us to decide whether the event is expected (lines 12 to 16). However, if the parent goal is already achieved, the new goal can contribute (as a child) to the recently achieved shared goal, i.e., g_t, which is also expected (line 19). On the contrary, if the new goal does not contribute to g_t, it might be a goal in another branch in the shared plan which has received focus and should be achieved. In such a case, again, the event will be expected; otherwise we consider the event unexpected (lines 21 to 24).

Controllability

Controllability is the extent to which an event can be influenced, and it is associated with a robot's ability to cope with an appraised event (Gratch and Marsella 2004). Thus, a robot can determine whether the outcome of an event can be altered by some actions under either of the collaborators' control. In other words, controllability is a measure of a robot's ability to maintain or change a particular state as a consequence of an event.

Algorithm 4 (Controllability)

1: **function** IsEventControllable(Event ε_t)

2: $\alpha \leftarrow$ GetAgencyRatio(ε_t)
3: $\beta \leftarrow$ GetAutonomyRatio(ε_t)

4: $\lambda \leftarrow$ GetSucPredecessorsRatio(ε_t)
5: $\mu \leftarrow$ GetAvailableInput(ε_t)

6: $\mathcal{U} \leftarrow \frac{\omega_0 \cdot \alpha + \omega_1 \cdot \beta + \omega_2 \cdot \lambda + \omega_3 \cdot \mu}{\omega_0 + \omega_1 + \omega_2 + \omega_3}$

7: $\tau_t \leftarrow$ getEmotionalThreshold()

8: **if** $(\mathcal{U} \geq \tau_t)$ **then**
9: **return** CONTROLLABLE
10: **else**
11: **return** UNCONTROLLABLE

Controllability is also important for the overall architecture. For instance, the robot can choose to ask or negotiate about a collaborative task which is not controllable; it can cause the robot to interpret or predict the other's emotional state (e.g., anger if the task is blocked, i.e., uncontrollable for the other), or form a new motive to establish an alternative goal for the current uncontrollable event. In general, other mechanisms in the architecture use the appraisal process of controllability in their decision making processes; meanwhile controllability uses the information from the collaboration structure, e.g., successful predecessors of a goal.

An important determinant of one's emotional response is the sense of control over the events occurring. This sense of subjective control is based on one's reasoning about self's power. For instance, the robustness of one's plan for executing actions can increase sense of power and subsequently the sense of control. In the collaboration context, we have translated the sense of control into a combination of four different factors including a) *agency* and b) *autonomy* of the robot, as well as the ratios of c) *successful predecessors*, and d) the *available inputs* of a given goal (i.e., g_{ε_t}) in the shared plan.

In Algorithm 4, we compute the controllability of an event based on these four factors (lines 2 to 5). Algorithms 4a to 4d are used to illustrate the underlying processes of all these factors. We use weighted averaging over these four factors to compute the utility of an event in terms of controllability of the event. However, the value of all these weights are set to 1.0 for the purpose of simplicity at this stage of the project. We will adjust these weights after further investigating the influence of these factors, and implementing other mechanisms in the overall architecture. After computing the value of the utility, we compare this value to an emotional threshold similar to what we discussed in Algorithm 1. This comparison leads to our decision about the controllability of an event (lines 8 to 11).

Agency is the capacity of an individual to act independently in any given environment. In a collaborative environment sometimes collaborators are required to act independently of each other. Hence, they need to have some internal motives that are formed based on their own mental state rather than being reinforced by the other. These internal motives will lead the collaborators to acquire new intentions towards new goals whenever it is required. In Algorithm 4a, if there is a path between the belief about the current event and the recognized goal associated with that event, we extract the motive within the acquired path. We consider maximum agency value denoted as α in Algorithm 4 (i.e., $\alpha = 1.0$) if the robot's mental state possesses an internal motive towards the recognized goal; otherwise we consider the minimum agency value (i.e., $\alpha = 0.0$) for no motives or external motives only. Note that the process of forming new internal motives is out of this paper's context.

Autonomy is the ability to make decisions without the influence of others. Autonomy implies acting on one's own and being responsible for that. In a collaborative environment, tasks are delegated to the collaborators based on their capabilities. Therefore, each collaborator is responsible for the delegated task and the corresponding goal. In Algorithm 4, β denotes the value of autonomy with regard to the event

Algorithm 4a (Get Agency Ratio)

function GETAGENCYRATIO(Event ε_t)
$b_{\varepsilon_t} \leftarrow$ GETBELIEF(ε_t)
$g_{\varepsilon_t} \leftarrow$ *recognizeGoal*(ε_t)
if $(g_{\varepsilon_t} =$ AMBIGUOUS$)$ **then**
return 0.0
$\mathcal{P} \leftarrow$ FINDPATH$(b_{\varepsilon_t}, g_{\varepsilon_t})$
if $(\mathcal{P} \neq \emptyset)$ **then**
$\mathcal{M}_{g_t} \leftarrow$ *currMotive*$(\mathcal{P})$
if $(\mathcal{M}_{g_t} \neq \emptyset)$ **then**
if $(\mathcal{M}_{g_t} \cdot type =$ INTERNAL$)$ **then**
return 1.0
else
return 0.0
else
return 0.0
else
return 0.0

(ε_t). This value is the ratio of the number of the contributing goals to g_{ε_t} for which the robot is responsible over the total number of contributing goals to g_{ε_t} (see Algorithm 4b). If the goal associated with the current event corresponds to a nonprimitive task, the algorithm checks the responsible agent for each primitive task contributing to the nonprimitive one and returns a value of which $(0 \leq \beta \leq 1)$. However, if the associated goal of the current event corresponds to a primitive task the value of β would be 0 or 1. In general, higher autonomy leads to a more positive value of controllability.

Algorithm 4b (Get Autonomy Ratio)

function GETAUTONOMYRATIO(Event ε_t)
$g_{\varepsilon_t} \leftarrow$ *recognizeGoal*(ε_t)
if $(g_{\varepsilon_t} =$ AMBIGUOUS$)$ **then**
return 0.0
$\Phi_g \leftarrow$ *extractContributingGoals*(g_{ε_t})
for all $\phi_g^i \in \Phi_g$ **do**
if (*getResponsible*$(\phi_g^i) =$ SELF) **then**
$count_{self} \leftarrow count_{self} + 1$
return $count_{self}$ / $\Phi_g.total()$

The structure of a shared plan accommodates the order of the required predecessors of a goal. Predecessors of a goal, g, are other goals that the collaborators should achieve before trying to achieve goal g. In Algorithm 4c, we use the ratio of successfully achieved predecessors of the recognized goal (g_{ε_t}) associated with the current event over the total number of predecessors of the same goal. This ratio (denoted as λ in Algorithm 4) is another factor used to compute the controllability of an event. If all of the predecessors of the given goal are already achieved, then $\lambda = 1$ which is the maximum value for λ. On the contrary, failure of all of the predecessors will lead to $\lambda = 0$. Therefore, higher λ value positively impacts the value of controllability for the current event.

Algorithm 4c (Get Succeeded Predecessors Ratio)

function GETSUCPREDECESSORSRATIO(Event ε_t)
$g_{\varepsilon_t} \leftarrow$ *recognizeGoal*(ε_t)
if $(g_{\varepsilon_t} =$ AMBIGUOUS$)$ **then**
return 0.0
$\Theta g \leftarrow$ *extractPredecessors*(g_{ε_t})
for all $\theta_g^i \in \Theta_g$ **do**
if (*isAchieved*(θ_g^i)) **then**
$count_{achieved} \leftarrow count_{achieved} + 1$
return $count_{achieved}$ / $\Theta g.total()$

Inputs of a task are the required elements that the collaborators use to achieve the specified goal of the task. These inputs are also part of the structure of a shared plan. In Algorithm 4d, we extract the required inputs of the associated goal with the current event, and check whether all the required inputs are available for the goal g_{ε_t}. The outcome will be the ratio of the available required inputs over the total required inputs of the goal associated with the current event. This value (denoted as μ in Algorithm 4) will be bound to 0 and 1. Similar to the other factors in the controllability process, the closer the value of μ gets to 1, the more positive impact it has on the overall controllability value of the event.

Algorithm 4d (Get Available Input Ratio)

function GETAVAILABLEINPUTRATIO(Event ε_t)
$g_{\varepsilon_t} \leftarrow$ *recognizeGoal*(ε_t)
if $(g_{\varepsilon_t} =$ AMBIGUOUS$)$ **then**
return 0.0
$\mathcal{X}_g \leftarrow$ *extractInputs*(g_{ε_t})
for all $\chi_g^i \in \mathcal{X}_g$ **do**
if (*IsAvailable*(χ_g^i)) **then**
$count_{available} \leftarrow count_{available} + 1$
return $count_{available}$ / $\mathcal{X}_g.total()$

Related Work

Our work is built on the general notion of appraisal theory while it is focused on its application in human-robot collaboration. Computational appraisal models have been applied to a variety of uses including contributions to psychology, robotics, AI, and Cognitive Science. For instance, in (Marsella and Gratch 2009) EMA is used to generate specific predictions about how human subjects will appraise and cope with emotional situations. Furthermore, appraisal theory has also been used in robots' decision making (Gonzalez, Malfaz, and Salichs 2013), or in their cognitive systems (Hudlicka 2007; Marinier III and Laird 2008). Additionally, in the virtual agents community, empathy and affective decision-making is a research topic that has received much attention in the last two decades (Velàsquez 1997; Paiva et al. 2004; McQuiggan and Lester 2007; Pontier and Hoorn 2013). However, EMA and several other examples in artificial intelligence and robotics which apply appraisal theory do not focus on the dynamics of collaborative contexts (Marsella and Gratch 2009; Kim and Kwon 2010; Adam and Lorini 2014; Rosenbloom, Gratch, and Ustun 2015).

The computational collaboration model in our work is closely influenced by the SharedPlans theory (Grosz and Sidner 1990). However, our algorithms are also compatible with other collaboration theories, e.g., Joint Intentions theory (Cohen and Levesque 1991), or STEAM as an hybrid collaboration theory (Tambe 1997). These theories have been extensively used to examine and describe teamwork and collaboration. Yet, they have never been combined, as they are in our work. We believe a systematic integration of collaboration theories and some theories like appraisal theory can help us describe the underlying collaboration processes leading to the existing collaboration structures.

References

Adam, C., and Lorini, E. 2014. A BDI emotional reasoning engine for an artificial companion. In *Workshop on Agents and multi-agent Systems for AAL and e-HEALTH (PAAMS)*, volume 430, 66–78. Springer.

Breazeal, C. 2002. *Designing Sociable Robots*. MIT Press.

Cohen, P., and Levesque, H. J. 1991. *Teamwork*. SRI International.

Cooke, R., and Sheeran, P. 2004. Moderation of cognition-intention and cognition-behaviour relations: A meta-analysis of properties of variables from the theory of planned behaviour. *British Journal of Social Psychology* 43:159–186.

Gonzalez, A. C.; Malfaz, M.; and Salichs, M. A. 2013. An autonomous social robot in fear. *IEEE Transactions Autonomous Mental Development* 5(2):135–151.

Gratch, J., and Marsella, S. C. 2004. A domain-independent framework for modeling emotion. *Cognitive Systems Research* 5(4):269–306.

Grosz, B. J., and Sidner, C. L. 1990. Plans for discourse. In Cohen, P. R.; Morgan, J.; and Pollack, M. E., eds., *Intentions in Communication*. Cambridge, MA: MIT Press. 417–444.

Hudlicka, E. 2007. Reasons for emotinos: Modeling emotinos in integrated cognitive systems. In Gary, W. D., ed., *Integrated Models of Cognitive Systems*, volume 59. New York: Oxford University Press. 1–37.

Kim, H.-R., and Kwon, D.-S. 2010. Computational model of emotion generation for human-robot interaction based on the cognitive appraisal theory. *Journal of Intelligent and Robotic Systems* 60(2):263–283.

Marinier III, R. P., and Laird, J. E. 2008. Emotion-driven reinforcement learning. In *CogSci 2008*.

Marsella, S. C., and Gratch, J. 2009. EMA: A process model of appraisal dynamics. *Cognitive Systems Research* 10(1):70–90.

McQuiggan, S. W., and Lester, J. C. 2007. Modeling and evaluating empathy in embodied companion agents. *International Journal of Human-Computer Studies* 65(4):348–360.

Paiva, A.; Dias, J.; Sobral, D.; Aylett, R.; Sobreperez, P.; Woods, S.; Zoll, C.; and Hall, L. 2004. Caring for agents and agents that care: Building empathic relations with synthetic agents. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 1*, 194–201.

Pontier, M., and Hoorn, J. F. 2013. How women think robots perceive them - as if robots were men. In *International Conference on Agents and Artificial Intelligence (ICAART-2)*, 496–504.

Rosenbloom, P. S.; Gratch, J.; and Ustun, V. 2015. Towards emotion in sigma: From appraisal to attention. In Bieger, J.; Goertzel, B.; and Potapov, A., eds., *Artificial General Intelligence*, volume 9205 of *Lecture Notes in Computer Science*. Springer International Publishing. 142–151.

Scherer, K. R.; Schorr, A.; and Johnstone, T. 2001. *Appraisal Processes in Emotion: Theory, Methods, Research*. Oxford University Press.

Sloman, A. 1987. Motives, mechanisms, and emotions. *Cognition & Emotion* 1(3):217–233.

Sousa, R. D. 1990. *The Rationality of Emotion*. MIT Press.

Tambe, M. 1997. Towards flexible teamwork. *Journal of Artificial Intelligence Research* 7:83–124.

Velàsquez, J. D. 1997. Modeling emotions and other motivations in synthetic agents. In *Proceedings of the 14th Nnational Conference on Artificial Intelligence AAAI-97*, 10–15.