CHAPTER 4
COMPUTATIONAL FRAMEWORK

4.1 Introduction

There are several appraisal models (e.g., EMA [18]) contributing in different applications such as social sciences, virtual agents, and robotics. However, none of these models have focused on the appraisal processes during collaboration. We believe appraisal plays a key role in collaboration due to its regulatory and evaluative nature. Also, collaboration induces some changes to appraisal processes due to its unique nature. For instance, although the appraisal models mostly use utility to compute the relevance of an event, we have found new cognitive components involved in determining utility because of the influence of the collaboration. These components, such as the recurrence of a belief by the human collaborator or the influence of the human collaborator's perceived emotion on the robot's decisions emphasize the fact that collaboration requires different procedures in appraisal processes. One of our contributions is to ground general appraisal concepts in the specific context and structure of collaboration.

Furthermore, we believe collaboration and appraisal have reciprocal influences on each other. In this chapter, we also talk about the influence of appraisal on collaboration through the goal management process. Also, we discuss our coping mechanism and strategies within the collaboration context. Then, we provide our computational model of three different motives used in our framework. We also briefly discuss other mechanisms in our framework. Finally, we present a crowd-

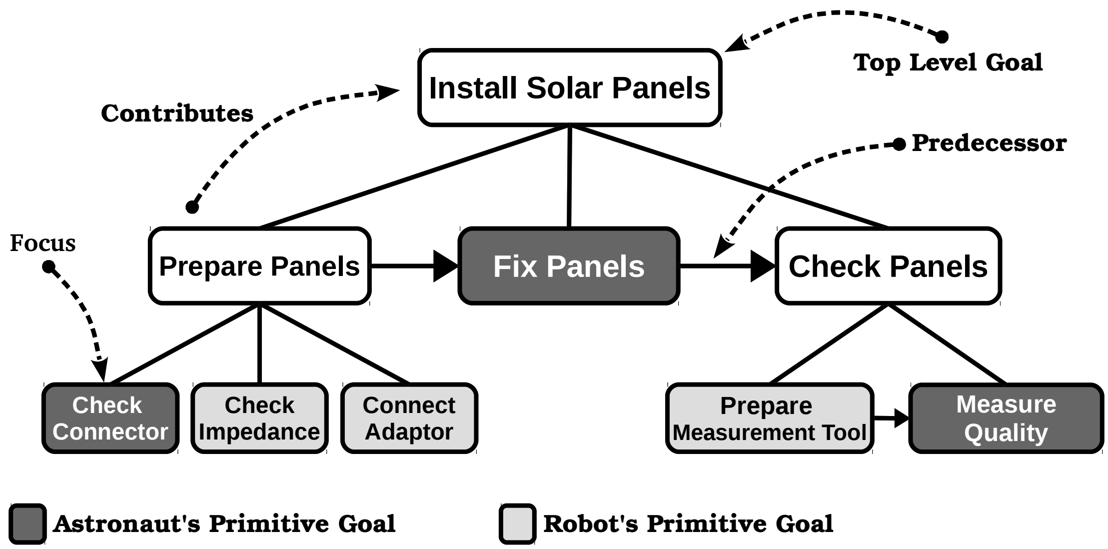

Figure 4.1: Collaboration structure (shared plan).

sourcing user-study and the results, which we conducted to validate the components of our appraisal processes.

4.2 Collaboration Mechanism

The Collaboration and Appraisal mechanisms have reciprocal influences on each other. In this section, we focus on information about the collaboration structure which will be incorporated in appraisal processes in Section 4.3. We describe some of the methods in our Collaboration mechanism which are used to retrieve information about the collaboration structure.

The Collaboration mechanism constructs a hierarchy of goals associated with tasks in the form of a hierarchical task network (see Figure 4.1), and also manages and maintains the constraints and other required details of the collaboration including 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). Collaboration also keeps track of the focus of attention, which determines the salient objects, properties and relations at each point, and shifts the focus of attention during the interaction.

Here, we describe the methods which retrieve information about the collaboration structure, and are used in our algorithms to compute the values of appraisal variables. In these methods, ε_t is the event corresponding to time t, and g_t is a given goal at time t.

- *recognizeGoal*(ε_t) returns the unique goal to which the given event (action, utterance, or emotional expression) directly contributes; it is only one goal since the robot can only do one primitive action at a time in our collaboration model, i.e, in the goal tree, a given primitive action can only directly contribute to one parent goal. The method returns *ambiguous* if it does not recognize a goal in the plan[1].

- *getGoalStatus*(g_t) returns whether g_t's status is ACHIEVED, FAILED, BLOCKED, INAPPLICABLE, PENDING, or IN PROGRESS. In our example, "Check Connector" is the current (focused) goal and it is PENDING, and the "Prepare Panels" and "Install Solar Panels" are IN PROGRESS. The focused goal is the goal that the robot currently pursues.

- *getTopLevelGoal*(g_t) returns g_t's top level goal.

- *precondStatus*(g_t) 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.

- *isLive*(g_t) returns *true* if all the predecessors of g_t are ACHIEVED and all the preconditions are SATISFIED, i.e., PENDING or IN PROGRESS goals; otherwise returns *false*.

- *isFocusShift*(g_t) returns *true* if the given goal is not the previous focus (top of the stack); otherwise returns *false*.

- *isNecessaryFocusShift*(g_t) returns *true* if the status of the previous focus was ACHIEVED; otherwise returns *false* [14].

[1] Ambiguity introduces some extra complexities which are beyond scope of this thesis.

- *isPath(g_1, g_2)* returns *true* if there is a path between g_1 and g_2 in a plan tree structure; otherwise returns *false*.

- *getContributingGoals(g_t)* returns g_t's children.

- *getPredecessors(g_t)* returns g_t's predecessors.

- *getInputs(g_t)* returns all required inputs for g_t. For example, the goal "Fix Panels" requires inputs such as *welding tool* and *panel*.

- *isAvailable(g_t)* returns whether the given input is available. For instance, whether the *welding tool* is available for the goal "Fix Panels".

- *isFocused(g_t)* returns whether the focus is on g_t.

- *getResponsible(g_t)* returns responsible agent(s) for g_t. In a dyadic collaboration, both of the agents (jointly) can be partly responsible for a nonprimitive goal, while each (self or other) is responsible for one or more primitive goals. For instance, both the Robot and the Astronaut are responsible for the non-primitive goal of "Install Solar Panels", whereas it is only the Robot who is responsible for the primitive goal of "Prepare Measurement Tool".

4.3 Appraisal Mechanism and Underlying Processes

In this section, we focus on the specific problem of appraising the *Relevance* (since other appraisals are only computed for relevant events), *Desirability* (since it discriminates facilitating and inhibitory events towards the collaboration progress), *Expectedness* (since it underlies a collaborative robot's attention), and *Controllability* (since it is associated with the agent's coping ability) of events within a collaborative interaction. There are other appraisal variables introduced in psychological [22] and computational literature [8]. We believe most of these variables can be straightforwardly added to our appraisal mechanism whenever they are required. All of the algorithms in this section use mental states of the robot (discussed in

Section 3.2.3) which are formed based on the collaboration structure (Figure 4.2). These algorithms use the corresponding recognized goal of the most recent event at each turn.

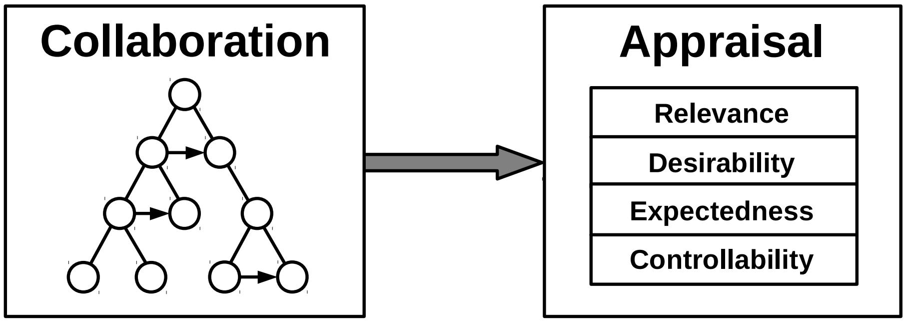

Figure 4.2: Using Collaboration structure in Appraisal (mechanisms in our framework).

4.3.1 Relevance

Relevance is an important appraisal variable since the other appraisal variables are meaningful only for relevant events. Relevance as an appraisal variable measures the significance of an event for the self. An event can be evaluated to be relevant if it has a non-zero utility [18]. However, the utility of an event is also influenced by the other collaborator's emotional expressions as the reflection of the other collaborator's mental state with respect to the status of the collaborative environment. Other appraisal models only consider the utility of an event based on the self's goal and plan.

Algorithm 1 determines the relevance of the given event with respect to the current mental state. The relevance of the event depends on the significance of the event with respect to the collaboration status, which is determined based on the utility of the event as presented in [8, 18]. Our algorithm for computing the relevance of an event during collaboration involves other factors that other appraisal models do not consider. For instance, the human's perceived emotion, recurrence of a belief, or occurrence of a belief about an unrelated goal by the human play important roles by influencing the utility of an event during collaboration. As a result, evaluating the relevance of events can cause a collaborative robot to respond effectively which can positively impact the status of the shared goal, without dedicating all its resources to every event.

After perceiving an event, the belief about that event represents the event in the robot's mental state. *recognizeGoal* returns the goal to which the current event contributes, unless it is *ambiguous*; g_t represents the shared goal at time (turn) t within the shared plan. We compute the utility ($-1 \leq \mathcal{U} \leq 1$) of the event using the values of the attributes associated with the existing beliefs, and the attributes of the motive associated with the recognized goal (see details below). We use three belief attributes (see Section 3.2.3) to compute the belief-related part of the utility:

Algorithm 1 (Relevance)

function IsEventRelevant(Event ε_t)

    $g_t \leftarrow$ *recognizeGoal*(ε_t)

    $\mathcal{U} \leftarrow$ getEventUtility(g_t)
    $\tau_t \leftarrow$ getEmotionalThreshold(g_t)

    if ($\tau_t \leq |\mathcal{U}|$) **then**
        return RELEVANT
    else
        return IRRELEVANT

- *Strength*: The extent to which the preconditions (α), postconditions (β), predecessors (λ), and contributing goals (μ) of a goal are known (SATISFIED or UNSATISFIED) makes beliefs about the goal stronger. An UNKNOWN pre and postcondition status of a goal and its predecessors and contributing goals forms weaker beliefs. For instance, if one knows all predecessors of a pursued goal (e.g., "Check Panels") are SATISFIED (i.e., "Fix Panels" and "Prepare Panels"), failure of the pursued goal will elicit one's negative emotion (due to the strong beliefs related to the goal); whereas not knowing the status of the goal-related factors (e.g., whether the Astronaut could find the tool to fix a panel) causes one to form weaker beliefs about the goal.

- *Saliency (S)*: Beliefs related to the focused goal are more salient than beliefs related to any other goal in the plan; according to Figure 4.1, if one of the collaborators is preparing a solar panel, beliefs related to all of the other *live* (PENDING or IN PROGRESS) goals (e.g. "Connect Adaptor") will be less salient than beliefs related to the focused goal, i.e., "Check Connector". Beliefs' saliency decreases according to their corresponding *live* goal's distance from the focused goal in the shared plan. *Non-live* goals will not be salient.

- *Persistence (P)*: The recurrence of a belief over time (turns) increases the persistence of the belief. Beliefs occurring only once have the lowest value of persistence. For instance, if the Astronaut repeatedly says that she can not find the measurement tool to check the connector, the Robot could pursue a new goal outside of the shared plan to acknowledge Astronaut's concern.

We also use two motive attributes discussed in Section 3.2.3 to compute the motive related part of the utility ($\mathcal{U}$):

- *Urgency (γ)*: There are two factors impacting the urgency of a motive: a) whether the goal directing the given motive is the predecessor of another goal for which the other collaborator is responsible, and b) whether achieving the goal directing the given motive can mitigate the other collaborator's negative valenced emotion. For instance, if the Robot has a private goal to fetch another panel while the Astronaut is waiting for the Robot to connect the adaptor, connecting the adaptor will be more urgent than Robot's private goal.

- *Importance (η)*: A motive is important if failure of the directing goal causes an impasse in the shared plan (i.e., no further goal is available to achieve), or achievement of the directing goal removes an existing impasse. For example, if the Robot cannot find the adaptor (an impasse to connect the adaptor), and the Astronaut provides another adaptor (external motive), the new motive becomes important to remove the impasse in the shared plan.

We provide the utility function ($\mathcal{U}$) in Equation 4.1. This function uses: saliency (S) and persistence (P) of the belief related to the recognized goal, the recognized goal's status (υ), and the aggregation of belief and motive attributes (Ψ) according to Equation 4.1.

$$\mathcal{U}(\varepsilon_t) = \begin{cases} \upsilon P \cdot S^{\Psi} & \Psi > 0 \\ 0 & \Psi = 0 \end{cases} \tag{4.1}$$

Intuitively, we use υ to generate positive and negative utility values. The υ's value becomes +1 if the status of the corresponding goal is ACHIEVED, PENDING, or IN PROGRESS, and υ's value becomes -1 if the status of the corresponding goal is FAILED, BLOCKED, or INAPPLICABLE. The P influences the value of utility only as a coefficient since recurrent beliefs are not formed frequently during collaboration. The Ψ value indicates the magnitude of the influence of beliefs and motives using their attributes. Hence, the Ψ value impacts the saliency value of beliefs exponentially, helping to differentiate between beliefs.

In equation 4.2, the subscript k refers to the *known* goal-related factors (SATISFIED or UNSATISFIED); whereas the subscript *all* includes both *known* and *unknown* goal-related factors. In this equation, both urgency (γ) and importance (η) attributes of motives can impact the outcome of the goal-related belief attributes' ratio, and ultimately the Ψ value.

$$\Psi = \frac{\alpha_k + \beta_k + \lambda_k + \mu_k}{\alpha_{all} + \beta_{all} + \lambda_{all} + \mu_{all}} + \eta + \gamma \tag{4.2}$$

$$\begin{aligned} \eta, \gamma \in \mathbb{N}, &\qquad \eta, \gamma \geq 0 \\ \alpha_k, \beta_k, \lambda_k, \mu_k \in \mathbb{N}, &\qquad \alpha_k, \beta_k, \lambda_k, \mu_k \geq 0 \\ \alpha_{all}, \lambda_{all}, \mu_{all} \in \mathbb{N}, &\qquad \alpha_{all}, \lambda_{all}, \mu_{all} \geq 0 \\ \beta_{all} \in \mathbb{N}, &\qquad \beta_{all} \geq 1 \end{aligned}$$

The significance of an event in a collaborative environment is based on the utility of the event and the human's perceived emotion. The human's perceived emotion influences the relevance of the event in the form of a threshold value τ_t. In Equation 4.3, we use the valence of the perceived emotion ($\mathcal{V}_{e_h}$) to compute τ_t.

$$\tau_t = \begin{cases} 1 - \mathcal{V}_{e_h} & \mathcal{V}_{e_h} > 0 \\ |\mathcal{V}_{e_h}| & \mathcal{V}_{e_h} \leq 0 \end{cases} \tag{4.3}$$

$$\mathcal{V}_{e_h} \in \mathbb{R}, \qquad -1 \leq \mathcal{V}_{e_h} \leq 1$$

Hence, perceiving human's positive emotion (e.g., happiness) reduces the threshold value which makes the robot find an event RELEVANT with even a slightly positive utility. Similarly, an event can be considered IRRELEVANT even though the utility has a relatively positive value, because of perceiving the human's negative emotion.

4.3.2 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 [8]. In a collaborative robot, preferences are biased towards those events facilitating progress in the collaboration. Desirability plays an important role in the overall architecture; it makes the processes involved in the other mechanisms (e.g., Motivation and Theory of Mind) and consequently the robot's mental state, congruent with the collaboration status which is a collaborative 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.

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 distinguishes between the top level goal and the current goal because the top level goal's change of status

Algorithm 2 (Desirability)

```
function IsEventDesirable(Event ε_t)

    g_t ← recognizeGoal(ε_t)
    g_top ← getTopLevelGoal(g_t)

    if (getGoalStatus(g_top) = ACHIEVED) then
        return MOST-DESIRABLE
    else if (getGoalStatus(g_top) = FAILED) then
        return MOST-UNDESIRABLE
    else if (getGoalStatus(g_top) = BLOCKED)
   (getGoalStatus(g_top) = INAPPLICABLE) then
        return UNDESIRABLE
    else if (getGoalStatus(g_top) = PENDING)
   (getGoalStatus(g_top) = INPROGRESS) then

        if (getGoalStatus(g_t) = ACHIEVED) then
            return DESIRABLE
        else if (getGoalStatus(g_t) = FAILED) then
            return MOST-UNDESIRABLE
        else if (getGoalStatus(g_t) = BLOCKED)
       (getGoalStatus(g_t) = INAPPLICABLE) then
            return UNDESIRABLE
        else if (getGoalStatus(g_t) = PENDING)
       (getGoalStatus(g_t) = INPROGRESS) then
            return NEUTRAL
```

attains a higher positive or negative value of desirability. For instance, failure of the top level goal (e.g., installing solar panel) is more undesirable than failure of a primitive goal (e.g., measuring the quality of the installed panel).

A top level goal' status must be ACHIEVED (i.e., SATISFIED postcondition) to consider the event MOST-DESIRABLE. When the goal's status is FAILED (i.e., UNSATISFIED postcondition) or BLOCKED, the associated event has the MOST-UNDESIRABLE or UNDESIRABLE values respectively. A goal is BLOCKED if any of the required goals or goals recursively through the parent goal are not ACHIEVED. An INAPPLICABLE goal is also considered as UNDESIRABLE. A goal is INAPPLICABLE if any of its predecessors are not ACHIEVED, and/or its preconditions are not SATISFIED. For

PENDING and INPROGRESS top level goals, the status of the current goal associated with the top level goal determines the status of the event ε_t. Only a non-primitive goal can have INPROGRESS status, if it has been started but is not yet completed. A goal can be PENDING if it is live, or if it is a non-primitive goal that has not been started yet. ACHIEVED current goals mark an event (ε_t) as DESIRABLE, while FAILED or BLOCKED current goals render the event associated with them as MOST-UNDESIRABLE and UNDESIRABLE respectively. PENDING or INPROGRESS current goals mark their associated events as NEUTRAL.

4.3.3 Expectedness

Expectedness is the extent to which the truth value of a state could have been predicted from causal interpretation of an event. In the collaboration context the expectedness of an event evaluates the congruency of the event with respect to the existing knowledge about the shared goal. Thus, expectedness underlies a collaborative robot's attention. The collaboration mechanism uses expectedness to maintain the robot's attention and subsequently its mental state with respect to the shared goal. Reciprocally, the appraisal mechanism uses the underlying information of the collaboration structure to evaluate the expectedness of an event [25].

In Algorithm 3 we provide the process of computing 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), which is associated with the event ε_t and its relationship with the top level goal (g_{top}).

The intuition captured here is that one expects the current goal to be finished before undertaking another activity, but the goals that can be the next focus of attention are also to be expected. Therefore, if the goal is live, the algorithm checks whether the goal has not changed, or whether the interpretation of the last event results in a necessary focus shift. Shifting the focus to a new goal is necessary when the former goal is achieved and a new goal is required. Consequently the new event is the MOST-EXPECTED one. However, even if the focus shift is not necessary, the

Algorithm 3 (Expectedness)

```
function IsEventExpected(Event ε_t)

    g_t ← recognizeGoal(ε_t)
    g_top ← getTopLevelGoal(g_t)

    if (isLive(g_t)) then
        if (¬isFocusShift(g_t)
         isNeccessaryFocusShift(g_t)) then
            return MOST-EXPECTED
        else
            return EXPECTED
    else
        if (isPath(g_t, g_top)) then
            return UNEXPECTED
        else
            return MOST-UNEXPECTED
```

new event can be considered as EXPECTED, since the corresponding goal is already live. For goals that have not yet been started (that is, are not live), the algorithm must determine how unexpected it would be to pursue one now; if the goal is at least in the plan, i.e., on the path to the top level goal, it is just UNEXPECTED while any others are MOST-UNEXPECTED.

4.3.4 Controllability

Controllability is the extent to which an event can be influenced; it is associated with a robot's ability to cope with an event [8]. Thus, a robot can determine whether an event's outcome can be altered by 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.

Controllability is important for the overall architecture. For instance, the robot can choose to ask or negotiate about a collaborative task which is not controllable, or form a new motive to establish an alternative goal for the current uncontrollable event. In general, other mechanisms in the architecture use the controllability output

Algorithm 4 (Controllability)

1: **function** IsEventControllable(Event ε_t)
2: $g_t \leftarrow$ *recognizeGoal*(ε_t)

3: $\mathcal{M} \leftarrow$ GetAgencyRatio(g_t)
4: $\mathcal{R} \leftarrow$ GetAutonomyRatio(g_t)

5: $\mathcal{P} \leftarrow$ GetSuccPredecessorsRatio(g_t)
6: $\mathcal{I} \leftarrow$ GetAvailableInputs(g_t)

7: $\mathcal{V}_{e_h} \leftarrow$ getEmotionValence(g_t)
8: $\omega \leftarrow$ getWeights(g_t)

9: $\mathcal{X} \leftarrow \frac{\omega_0 \cdot \mathcal{M} + \omega_1 \cdot \mathcal{R} + \omega_2 \cdot \mathcal{P} + \omega_3 \cdot \mathcal{I}}{\omega_0 + \omega_1 + \omega_2 + \omega_3} + \mathcal{V}_{e_h}$

10: **if** ($\mathcal{X} > 0$) **then**
11: **return** controllable
12: **else**
13: **return** uncontrollable

in their decision making processes; meanwhile controllability uses information from the collaboration structure, e.g., predecessors of a goal.

An important determinant of one's emotional response is the sense of control over occurring events. 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 one's 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_t) in the shared plan.

In Algorithm 4, we partially compute the controllability of an event based on the above four factors. We use weighted averaging of these factors to determine their impact on the controllability of an event (line 9). The value of all these weights are set to *1.0* for the purpose of simplicity (**getWeights**). These weights can be adjusted after further investigating the influence of these factors, and implementing other mechanisms in the overall architecture. We believe that the human's perceived emotion also impacts the controllability of an event (**getEmotionValence**). The $(-1.0 \leq \mathcal{V}_{e_h} \leq 1.0)$ is the valence value of the human's perceived emotion. Positive emotions, e.g., happiness, possess positive values, and negative emotions, e.g., anger, have negative values. The magnitude of this value can change with respect to the intensity of the perceived emotion. Thus, a positive controllability value indicates that an event is CONTROLLABLE; otherwise UNCONTROLLABLE.

GetAgencyRatio: *Agency* is the capacity of an individual to act independently in a given environment. In a collaborative environment collaborators are sometimes required to act independently of each other. Hence, they need to have some internal motives that are formed based on their own mental states rather than motives that are reinforced by the other. These internal motives will lead the collaborators to acquire new intentions when required. If the robot's mental state possesses only an internal motive supporting the recognized goal, we consider a maximum agency value denoted as $\mathcal{M}$ in Algorithm 4 (i.e., $\mathcal{M} = 1.0$); otherwise we consider the minimum agency value (i.e., $\mathcal{M} = 0.0$).

GetAutonomyRatio: *Autonomy* is the ability to make decisions without the influence of others, and 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, $\mathcal{R}$ denotes the value of autonomy with regard to the goal g_t. This value $(0.0 \leq \mathcal{R} \leq 1.0)$ is the ratio of the number of goals contributing to g_t for which the robot is responsible over the total number of contributing goals, if the goal associated with the current event is a nonprimitive goal. However, if the associated goal of the current event corresponds to a primitive goal the value of $\mathcal{M}$ would be 0.0 or 1.0. In general, higher autonomy leads to a more positive value of controllability.

GetSuccPredecessorsRatio: The structure of a shared plan contains the order of the required *predecessors* of a goal. Predecessors of a goal, g_t, are goals that the collaborators should achieve before trying to achieve goal g_t. We use the ratio of successfully achieved predecessors of the recognized goal over the total number of predecessors of the same goal. If all of the predecessors of the given goal are achieved, then $\mathcal{P} = 1.0$ which is the maximum value for $\mathcal{P}$. On the contrary, failure of all of the predecessors will lead to $\mathcal{P} = 0.0$. Therefore, a higher $\mathcal{P}$ value positively impacts the value of controllability for the current event.

GetAvailableInputs: Finally, *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. We compute the ratio of the available required inputs over the total required inputs of the goal associated with the current event. This value (denoted as $\mathcal{I}$ in Algorithm 4) will be bound between 0.0 and 1.0. Similar to the other factors in the controllability process, the closer the value of $\mathcal{I}$ gets to 1.0, the more positive impact it has on the overall controllability value of the event.

In summary, the output of these four appraisal processes serves as critical input for the other mechanisms of the Affective Motivational Collaboration Framework, shown in Chapter 3. By providing adequate interpretation of events in the collaborative environment, the appraisal mechanism enables the robot to carry out proper collaborative behaviors.

4.4 Goal Management

A collaborative robot needs to be able to regulate and manage shared goals during collaboration. Emotion has a crucial influence on this goal management process. In this section, we provide a cost function that we use to choose the goal in the shared plan with the lowest cost value out of a set of alternative goals. This cost function is a) based on the goal attributes, b) with respect to the reverse appraisal of the perceived emotion, and c) the appraisal of the collaborative environment.

Goals represent an important part of the context during collaboration. However, not all goals are appropriate to pursue at the moment, depending on conditions. In fact, it can be destructive for a collaboration to pursue a good goal in a poor context. Therefore, a collaborative robot must be able to manage shared goals during collaboration. The goal management process has a critical influence on a collaborative robot's behavior by maintaining or shifting the focus of attention to an appropriate goal based on the collaboration status.

Changes in a collaboration environment alter the balance of alternative goals. These changes can reflect the collaborators' internal changes and the influence of their actions. In a collaboration environment, emotions represent the outcome of underlying mental processes of the collaborators. Emotions have many different functions [23] including goal management. Goal-oriented emotions such as anger, frustration and worry regulate the mental processes influenced by one's internal goals. In our example in this section, a robot and an astronaut are collaborating to install solar panels. When one of the astronaut's goals is blocked, the robot must manage the shared goals in order to prevent failure of the collaboration. By using reverse appraisal [6] of the astronaut's emotion and its own appraisal of individual goals, the robot is able to successfully shift the focus of attention from the blocked goal (eliciting worry in the astronaut) to an appropriate one to maintain the collaboration. A similar example is provided our conducted user study, which is explained in Chapter 5.

Here, we describe the goal management process in our framework using an astronaut-robot collaboration example. We introduce the goal management process based on a cost function including the influence of affective appraisal and reverse appraisal processes. Goal management is a crucial part of our investigation of the reciprocal influence of appraisal on a collaboration structure (see Figure 4.3).

As we mentioned earlier, we use four appraisal variables including: relevance, desirability, expectedness and controllability. The outcome of each appraisal process

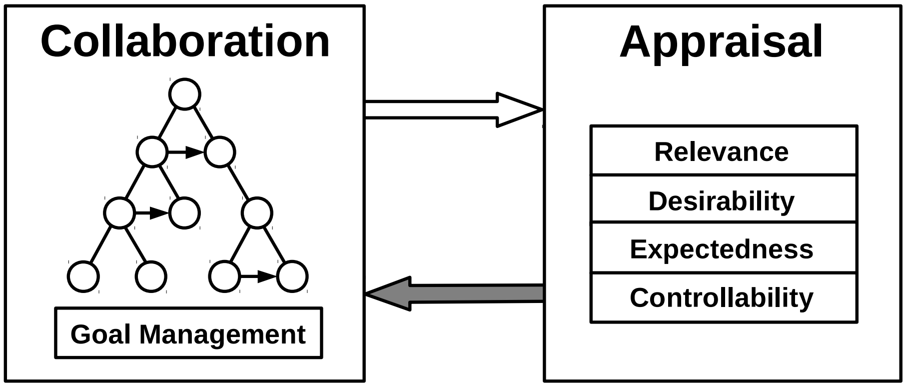

Figure 4.3: Using Appraisals' outcome to influence Collaboration structure (mechanisms in our framework).

is a specific value for the corresponding appraisal variable. The vector containing these appraisal variables can be mapped to a particular emotion instance at each point in time when requied. Moreover, the functions of emotions, such as goal management, in a social setting and the meaning of the collaborator's perceived emotion in collaboration context are also important.

A collaboration structure provides a hierarchy and constraints of the shared goals in the form of a shared plan (Figure 4.11) which contains both the robot and the human collaborator's goals. The robot pursues the goals for which the robot is responsible in the shared plan. However, there can be several live goals available for the robot to pursue at each point in time during collaboration. A goal is live if all of its predecessors are achieved and all of its preconditions are satisfied. Therefore, a collaborative robot requires a mechanism to choose between a set of live goals. We believe appraisal processes are crucial to choose between the available live goals, since the appraisals are the immediate outcome of the robot's assessment of the collaboration environment.

For instance, Figure 4.4 shows a non-primitive "Prepare Panels" goal decomposed into three primitive goals. Therefore, if "Prepare Panels" is live, its primitive goals can be pursued by the responsible agent. In our example, the astronaut is responsible for the "Check Connector" goal; the robot is responsible for the remain-

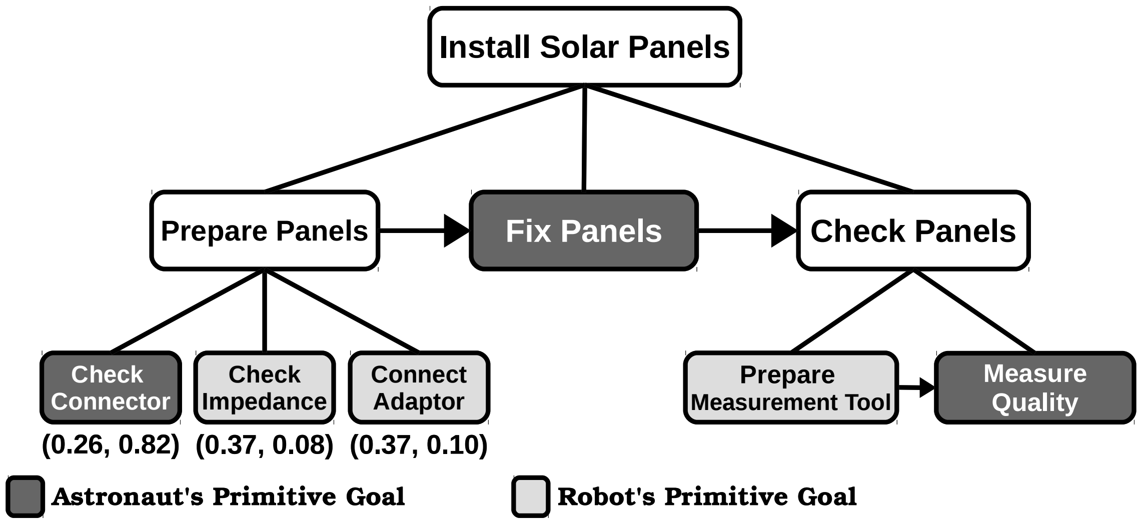

Figure 4.4: Cost values indicated by tuples with (second number) and without (first number) the influence of emotions.

ing two primitive goals. According to the collaboration mechanism in our overall framework, "Check Connector" is in focus, with the astronaut pursuing this goal. Suddenly, however the astronaut tells the robot that she can not find the connector and she is *worried* about failure of this goal. The robot's response to this situation will be explored below as we discuss details of our cost function.

Equation 4.4 shows the function to calculate the cost of each live goal. The base in the equation calculates the cost of pursuing any given goal. The three functions used to calculate the cost are: *proximity* $P(g)$, *difficulty* $D(g)$, and *specificity* $S(g)$ (see equations 4.6 to 4.8).

$$Cost(g) = \left(\omega_0.P(g) + \omega_1.D(g) + \omega_2.\frac{1}{S(g)+1}\right)^{\Gamma} \tag{4.4}$$

For simplicity in this example, we assume equal values for the weights: ω_i=1.

$$\Gamma = -C[(R_r + 1)D_r + \alpha(R_h + 1)D_h] \tag{4.5}$$

The exponent part of our cost function (Equation 4.5) captures a) the influence of the human's perceived emotional instance, and b) the influence of self appraisal of the given goal. $R_h \in [0,1]$ and $D_h \in [-1,1]$ are the relevance and desirability values respectively, which are based on the *reverse* appraisal of the human's perceived emotion. For instance, if the astronaut is *worried*, D_h is negative, e.g., -0.8 (depending on how undesirable the event is according to reverse appraisal), and R_h will be 1 for the active goal and its value descends to 0 for other live goals depending on their distance to the active goal in the shared plan (e.g., 0.1).

$R_r \in [0,1]$ and $D_r \in [-1,1]$ are relevance and desirability values, provided by the *self* appraisal functions for all of the live goals. For instance, for the active goal for which the astronaut was *worried*, D_r can be positive, e.g., 0.8 (depending on the self's desirability appraisal function); R_r can be 1, since the active goal is relevant for the robot. These values will change for the other live goals depending on how relevant they are with respect to the collaboration status (e.g., 0.9 and 0.8). Finally, $C \in [1,\infty)$ is a constant (e.g., 2) used to control the influence of affect on cost value. It is negative since undesirability (negative values) should increase the cost. $\alpha \in [1,\infty)$ is another constant (e.g., 3) used to control the importance of reverse appraisal relative to self appraisal.

The *proximity* of a goal indicates how far the goal is from the current active goal in the shared plan. It is calculated by the distance function (Equation 4.6) which returns the number of edges between the current active goal g_{act}, and the given goal g in the shared plan. In our example, $P(g)$ is 2 for both "Check Impedance" and "Connect Adaptor" goals.

$$P(g) = max\{1, distance(g_{act}, g)\} \tag{4.6}$$

The *difficulty* of a goal is a function of three parameters (Equation 4.7) which consider the difficulty based on a) topology of the shared plan tree (domain independent), and b) the amount of effort required to pursue a given goal (domain dependent). The $\sum pred_e(g)$ is the sum of efforts that all the *predecessors* of a given goal g require. The $\sum desc_e(g)$ is the sum of efforts that all the *descendants* of a given goal g require. The effort values represent the amount of effort for the goals with respect to the domain. In our example, we assume the values of all the goal efforts are 1 for simplicity. The $H(g)$ is the height of the given goal g. The heights of all primitives under "Prepare Panel" goal are 0 in our example.

$$D(g) = \Big(H(g) + 1\Big) \times \left[\sum_{m=0}^{M} pred_e(g) + \sum_{n=0}^{N} desc_e(g)\right] \tag{4.7}$$

The *specificity* of a goal is the function of *depth* (distance from the root) and *degree* (number of children in the graph) of a given goal g. The first non-primitive goal (root) is the least specific goal, and the primitives (leaves) are the most specific goals. As calculated based on Figure 4.4, the values of $S(g)$ for the three primitives under the "Prepare Panels" are 2.

$$S(g) = \frac{depth(g)}{degree(g) + 1} \tag{4.8}$$

The tuples below the goals in Fig. 4.4 indicate the cost value of each goal. The first number in each tuple is the normalized cost value without the influence of the affective part of the cost function, i.e., the exponent is equal to 1 in Equation 4.4. The second number of each tuple indicates the normalized value of the cost including the influence of affective appraisal and the astronaut's perceived emotion.

Based on our cost function, the cost of completing the primitive goal "Check Connector" is 0.82 (see Figure 4.4). As shown, when affect is not considered the cost is 0.26; the negative emotion of the astronaut (worry) significantly increases the cost of the current goal, and also impacts the other two primitive live goals under the same parent. Therefore, instead of insisting on pursuing the same blocked goal which has caused the astronaut's negative emotion, the robot can mitigate the astronaut's emotions by adapting to her worry. The robot shifts the focus of attention to "Check Impedance" to maintain progress and prevent failure of the collaboration.

We use our proposed cost function in our goal management algorithm to integrate affective appraisal into the collaboration mechanism in our framework. A similar situation is used in our conducted user study (see Chapter 5) to evaluate the human's perception of the robot's behavior as a result of the goal management process.

4.5 Coping Mechanism and Strategies

We have implemented the Coping mechanism to determine how the agent would respond to events using our framework. Our Coping mechanism includes a set of coping strategies that can be triggered based on different conditions (see Figure 4.5). All of these coping strategies are known in the literature. Some of our coping strategies, i.e., *planning*, *active coping* and *seeking social support for instrumental reasons*, are categorized as problem-focused and others, i.e., *acceptance*, *mental disengagement*, and *shifting responsibility*, are categorized as emotion-focused startegies as described in [8]. We selected these six coping strategies since they let our agent demonstrate distinct behaviors with respect to the output of the appraisal mechanism and the agent's mental state in our framework. The behaviors and underlying processes associated with these coping strategies are described as follows.

4.5.1 Planning

The *planning* coping strategy works based on the shared plan and the task structure introduced as an input to our framework. The task structure includes the hierarchy and ordering of the tasks, the required inputs of each task as well as the preconditions and postconditions of individual tasks. We use this task structure to create our shared plan which includes the primitive and non-primitive goals that our agent and its collaborator want to achieve throughout their collaboration. Therefore, our agent executes actions related to its own goals based on this shared plan, and uses the same shared plan to associate goals and their status with the human collaborator. To achieve a goal the agent is required to execute an action, and to execute an action the agent needs to have the right intention. In our framework, whenever this coping strategy is activated the Coping mechanism provides the selected intention to the Action mechanism. The Action mechanism executes an action based on the given intention to achieve the corresponding goal in the shared plan.

4.5.2 Active Coping

The *active* coping strategy can provide one or all of the following three different intentions with respect to whether this coping strategy is activated and the required conditions are provided. Firstly, this coping strategy can provide an intention to *acknowledge* the human's emotions. For instance, if the human expresses an emotion with negative valence, the agent can acknowledge human's negative emotion accordingly. Secondly, the active coping strategy can provide an intention to *respond* to the human if the human asks a question. Currently, in our framework, the agent can respond to the human if the human asks the agent: a) what input is required to achieve a goal, b) how to do a task to achieve a goal, c) to achieve a goal, d) who is responsible to achieve a given goal. For instance, if the human asks the agent to achieve a goal, the active coping strategy forms an intention to either accept the human's proposal (if achieving the given goal is controllable for the agent), or reject the human's proposal (if it is not controllable for the agent). Thirdly, the active coping strategy can form an intention to *delegate* a task to the human collaborator. The intention for task delegation can be formed if the agent fails to achieve its own goal, and the human's perceived emotion is not negative. As mentioned earlier, any or all of these intentions can be formed if active coping is selected. The agent acts accordingly by passing these intentions to the Action mechanism. For instance, if the human is frustrated about a failure that occurred when using a tool to perform its own task and asks the agent whether the agent can provide its own tool, the active coping strategy forms a new intention to acknowledge the human's frustration and responds to the human by providing the right tool (input) to use and fulfill the task. In this example, there will be no new intention to delegate a new goal to the human since the agent perceives the human's negative emotion.

4.5.3 Seeking Social Support for Instrumental Reasons

The *seeking social support for instrumental reasons* strategy forms new intentions for the agent whenever the agent needs the human's help and needs to ask questions from the human collaborator to make progress in collaboration. The questions that our agent can ask are the reciprocal of those questions that the human can ask and the human can respond as we mentioned above. Therefore, our agent can ask a) what input is required to achieve a goal, b) how to do a task to achieve a goal, c) the human to achieve a goal, d) who is responsible to ahcieve a given goal. Reciprocally, again, the agent expects the human collaborator to accept or reject the agent's proposals. In our framework, whenever this strategy is activated the agent consider's human's perceived emotion. For instance, if the human is worried about the outcome of a task failure, the agent does not form an intention to ask questions about any of the above cases and consequently prevents asking for more help.

4.5.4 Acceptance

The *acceptance* coping strategy forms an intention to drop the intention of pursuing a goal. In our framework, if this strategy becomes activated, the intention to pursue the current goal will be dropped; see Figure 4.5. The acceptance strategy also forms an intention to inform the human collaborator about the agent's decision on not pursuing the current goal.

4.5.5 Mental Disengagement

The *mental disengagement* coping strategy forms new intention to lower the negative emotional charge associated with a goal in the event of a failure or an impasse. We use our goal management algorithm (see section 4.4) as the underlying process used as the result of selecting this strategy to dissociate the collaboration process and subsequently disengage the collaborator from a negative event (e.g., failure to achieve a goal). This disengagement helps the agent to lower the utility of an unsuccessful goal achievement attempt and focus on other achievable goals with respect to their costs to facilitate progress of collaboration. In our framework, this coping strategy forms an intention to run the goal management process. As the result of mental disengagement activation, the mechanism also forms another intention to inform the human about the outcome of the goal management process, i.e., whether the agent proposes switching to pursue another goal with lower cost, or if there is not much the agent can do since there is no other goal with a lower cost to pursue. The process and example of choosing another goal with a lower cost are shown in section 4.4.

4.5.6 Shifting Responsibility

The *shifting responsibility* strategy forms new intention to shift the blame from the agent to another entity. In our framework, we use this strategy to mitigate the influence of negative events causing negative emotions in the agent or the human collaborator. For instance, if this strategy becomes activated as a result of a failure, a new intention will be formed to blame the third person who provided the input (if the task needed a tool as an input) or the other collaborator. It can also form an intention to give the credit to the human collaborator to mitigate human's negative emotions.

4.5.7 Activation of Coping Strategies

In our Coping mechanism, there are three components involved as the activation criteria for each coping strategy. The first criterion is the conjunction of emotion valences of the self and the other collaborator (see Emotion Valence column in Figure 4.5). For instance, if the valence of the human collaborator's emotion is *negative* **and** the valence of the agent's emotion is also *negative*, the active coping, the acceptance, and the mental disengagement coping strategies are the coping strategy candidates that have potential to become activated if the other activation criteria also exist for any of them. For example, if the human collaborator is frustrated and the agent's elicited emotion is guilt, the three above mentioned coping strategies become potential candidates to be selected as the agent's active coping strategy. The second criterion is the need for the agent to cope with an event. The values of our three different motives (i.e., *satisfaction*, *achievement*, and *external*) are involved in the decision of whether there is a need for a particular coping strategy to become activated. We use conjunction of satisfaction motive's value with the disjunction of achievement and external motives. For instance, if we have highly negative values for all three motives for the potential candidates of coping strategies based on the example we mentioned above, the acceptance coping strategy will be selected as the strategy with the highest need for the agent. For example, this kind of condition can occur when the agent fails doing its own task and pursuing the current goal (negative satisfaction motive), and can not find another goal to overcome the impasse (negative achievement motive). The details about how the motive values are computed is presented in Section 4.6. Finally, the ability to cope with an event is the third criterion that impacts the decision of whether the selected coping startegy can be activated. The controllability of an event represents whether the agent is able to control the situation occurring with the given event. In our example, if the agent finds the event uncontrollable, the acceptance coping strategy becomes activated (see Figure 4.5).

4.6 Motivation Mechanism

As we discussed in Chapters 2 and 3, motives are goal-driven emotion-regulated constructs indicating an urge related to their goal. There are several motives in psychological and computational literatues as we reviewed in Chapter 2. However, none of these computational models have particularly focused on the application of motives in the collaboration context. In our framework, we have implemented

Coping Strategy	Emotions (AND)		Need [a AND (b OR c)]			Ability
	Other	Self	Satisfaction Motive (a)	Achievement Motive (b)	External Motive (c)	Controllability
Planning	Neutral | Positive	Any	–/+	*high* +	*high* +	High
Active Coping	Any	Neutral | Negative	–/+	*med* +	*med* +	High
Seeking Social Support for Instrumental Reasons	Neutral | Positive	Any	–/+	*low* +	*low* +	Low
Acceptance	Negative	Negative	*high* –	*high* –	*high* –	No
Mental Disengagement	Neutral | Negative	Neutral | Negative	*low/med* –	*low/med* –	*low/med* –	No
Shifting Responsibility	Neutral | Positive	Negative	*high* –	–/+	–/+	No | Low

Figure 4.5: Conditions for selecting coping strategies

three computational models of motives for *satisfaction*, *achievement*, and *external* motives. We use the values of these three motives in other mechanisms including the Coping mechanism as we described in Section 4.5 and show in Figure 4.5.

4.6.1 Satisfaction Motive

The satisfaction motive indicates the satisfaction level with the collaboration for the agent and its human collaborator. The satisfaction motive process maintains the value of *satisfaction drive* throughout the collaboration. The satisfaction drive is the quantitative weighted accumulation of desirability values between -1 and +1 over time. For instance, if the desirability values of the agent's appraisal over three consecutive turns are {0.75, 0, -0.25}, and their corresponding weights are {0.25, 0.5, 1.0}, the satisfaction drive value will be (0.25)(0.75) + (0.5)(0) + (1.0)(-0.25) which is -0.0625. Notice that the latest desirability values get higher weights. Intuitively, it is because older desirable events have less influence on overall desirability and consequently the satisfaction level of the collaboration. The same process computes the satisfaction drive values for the agent and the human collaborator. Only the sources of desirability values are different, i.e., appraisal for the agent and reverse appraisal for the human collaborator. Then, the satisfaction motive process computes the difference between the current and the previous satisfaction drives, called delta of satisfaction drive value, δ_{sat}. As shown in equation 4.9, we use the δ_{sat} value in all three functions to compute the overall satisfaciton motive's value $\mathcal{M}_{sat}$. We also use three different functions with respect to the valence value of the the human collaborator's perceived emotion. Our satisfaction motive's model has three user defined parameters $\mathcal{S}_{sat} \in [0, 1.5]$, i.e. strength of motive, $\mathcal{B}^{\mathcal{L}}$ where $\mathcal{B}$ is the base parameter of the function in $(1, \infty)$ and $\mathcal{L}$ is the exponential parameter of the same function in $(0, \infty)$; together $\mathcal{B}$ and $\mathcal{L}$ define *unsatisfiability* value. In our framework, we set the $\mathcal{S}_{sat}$ value to 1.5, the $\mathcal{B}$ to 3.0, and the $\mathcal{L}$ to 2.0.

$$\mathcal{M}_{sat}(\varepsilon_t) = \begin{cases} arctan(\mathcal{S}_{sat} \times \delta_{sat}) & valence = 0 \\ \mathcal{B}^{\mathcal{L} \times (\delta_{sat} - 1)} & valence > 0 \\ -\mathcal{B}^{-\mathcal{L} \times (\delta_{sat} + 1)} & valence < 0 \end{cases} \tag{4.9}$$

Intuitively, if the human collaborator does not express any emotion, the satisfaction motive's value can vary between -1 and +1 (blue curve in Figure 4.6). However, if the agent perceives positive emotion, there will be no negative satisfaction value since the other collaborator is in positive state of mind (red curve in Figure 4.6), and in contrast, if the agent perceives negative emotion, the satisfaction motive value only changes between -1 and 0 (green curve in Figure 4.6) with respect to how satisfied the agent is according to the status of its own goals during collaboration.

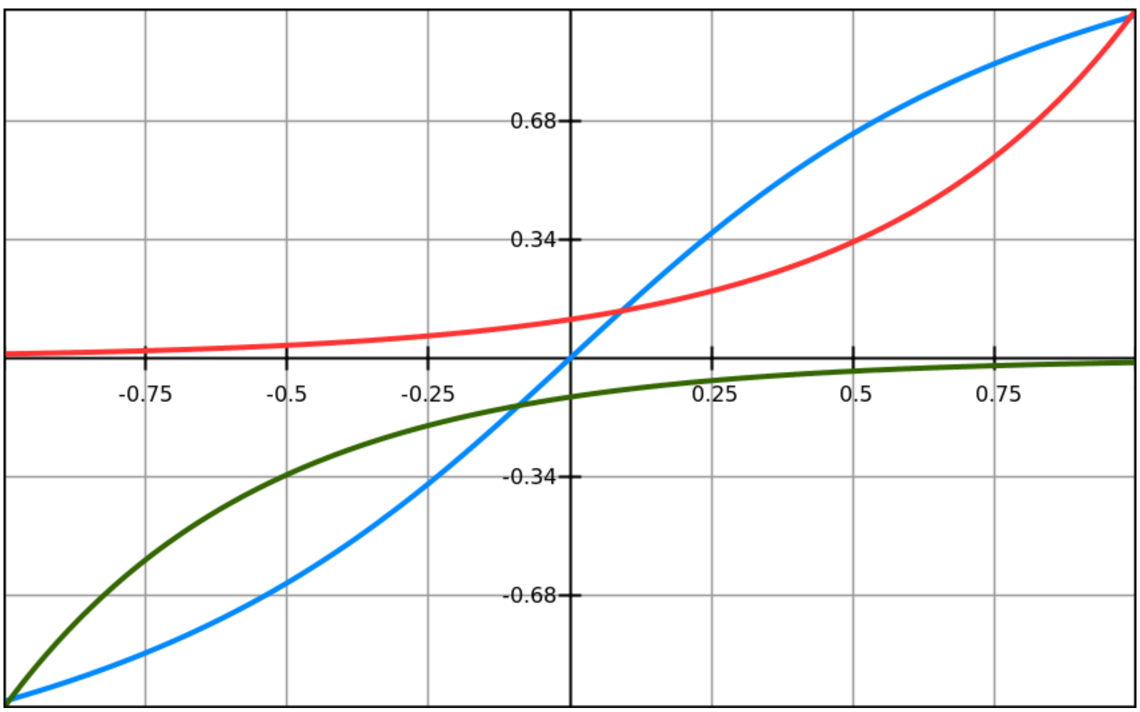

Figure 4.6: Three functions of satisfaction motive (blue: valence = 0, red: valence = positive, green: valence = negative). The x-axis indicates the satisfaction drive's delta value in [-1, +1], and the y-axis indicates the magnitude of satisfaction motive in [-1, +1].

4.6.2 Achievement Motive

The achievement motive drives the agent's need to achieve a goal during the collaboration. According to the literature, e.g. [19], the achievement motive is based on the estimation of success probability and the difficulty of achieving a goal. In our framweork, we compute the probability of success as the multiplication of the *controllability* and *expectedness* appraisal values. Intutively, the more controllable and expected the events are, the probability of successful achievement of their related goal is higher.

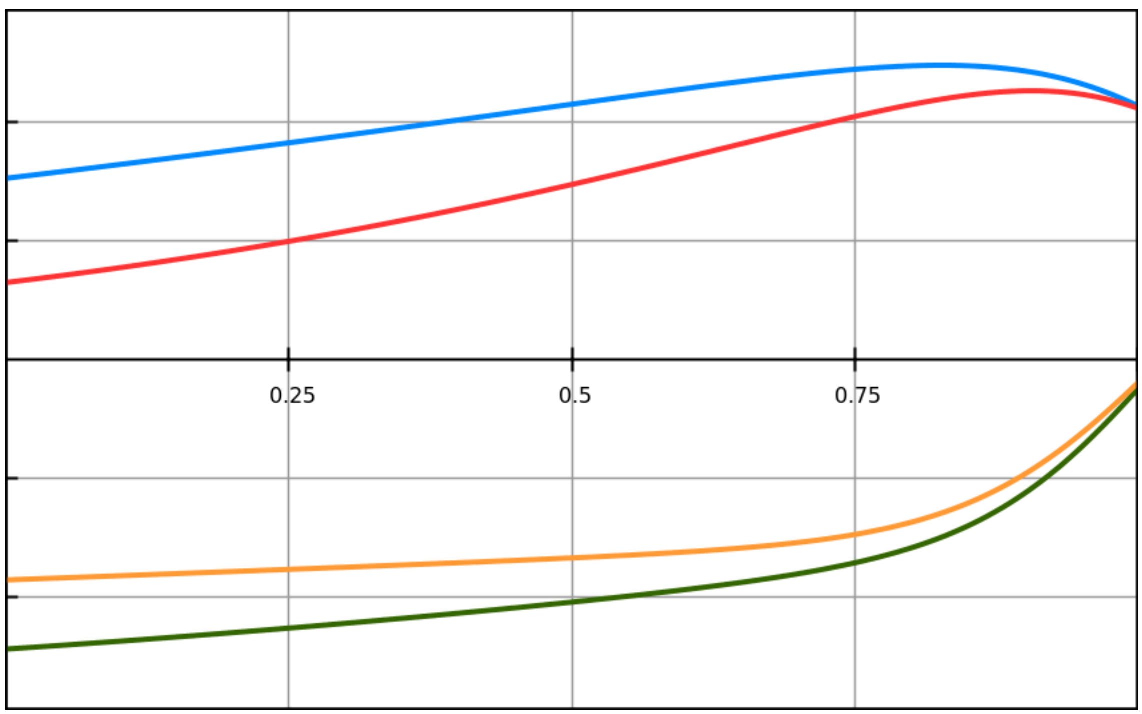

Figure 4.7: Two functions of the achievement motive (blue: valence = +1, red: valence = 0, green: valence = -1, orange: valence = close to zero from negative side). The x-axis indicates the success probability value of achieving a goal which is in [0, +1], and the y-axis indicates the magnitude of acheivement motive in [-1, +1].

In our framework we use two sigmoid-based functions to compute the achievement motive's value. These functions values change based on the probability of success and valence of the human collaborator's emotion. We use Equation 4.10

when the perceived emotion of the human has positive or zero valence value, and we use Equation 4.11 when the perceived emotion of the human has a negative valence value. As shown in Figure 4.7, when the value of the valence changes between 0 and +1, the output of $\mathcal{M}^{+}{}_{ach}$ function changes between the red and the blue lines respectively. Conversely, when the value of the valence changes between -1 and a small negative number (close to zero), the output of $\mathcal{M}^{-}{}_{ach}$ function changes between the green and the orange lines.

$$\mathcal{M}^{+}{}_{ach}(\varepsilon_t) = \frac{2.0}{1 + e^{(2.0 - valence) \times (1.05 - p(success))}} - \frac{1.0}{1 + e^{(12.0 - valence) \times (1.2 - p(success))}} \tag{4.10}$$

$$\mathcal{M}^{-}{}_{ach}(\varepsilon_t) = \frac{1.0}{1 + e^{(0.5 + valence) \times (1.05) - p(success)}} - \frac{1.0}{1 + e^{(12.0 + valence) \times (p(success) - 1.02)}} \tag{4.11}$$

By intuition, as the probability of success increases the agent is more motivated to achieve a goal and this motive gets higher when the human's emotion is positive or at least neutral. The human's negative emotions cause lower values of achievement motive since taking care of and acknowledging the human's negative emotion should have higher priority for a collaborative agent than achieving a goal.

4.6.3 External Motive

The external motive drives the agent's need to achieve a proposed goal by the human collaborator during the collaboration. In our framweork, the external motive is also based on the estimation of success probability and the difficulty of achieving a goal, but this goal is proposed by the human collaborator. The probability of success for the external motive is computed the same way as the achievement motive's probability of success, i.e. the multiplication of *controllability* and *expectedness* appraisal values.

The only differnce is that we use Equations 4.10 and 4.11 in reverse order for the external motive; i.e. we use Equation 4.11 when the valence of human's perceived emotion is positive, and Equation 4.10 when the valence of the human's perceived emotion is negative or zero.

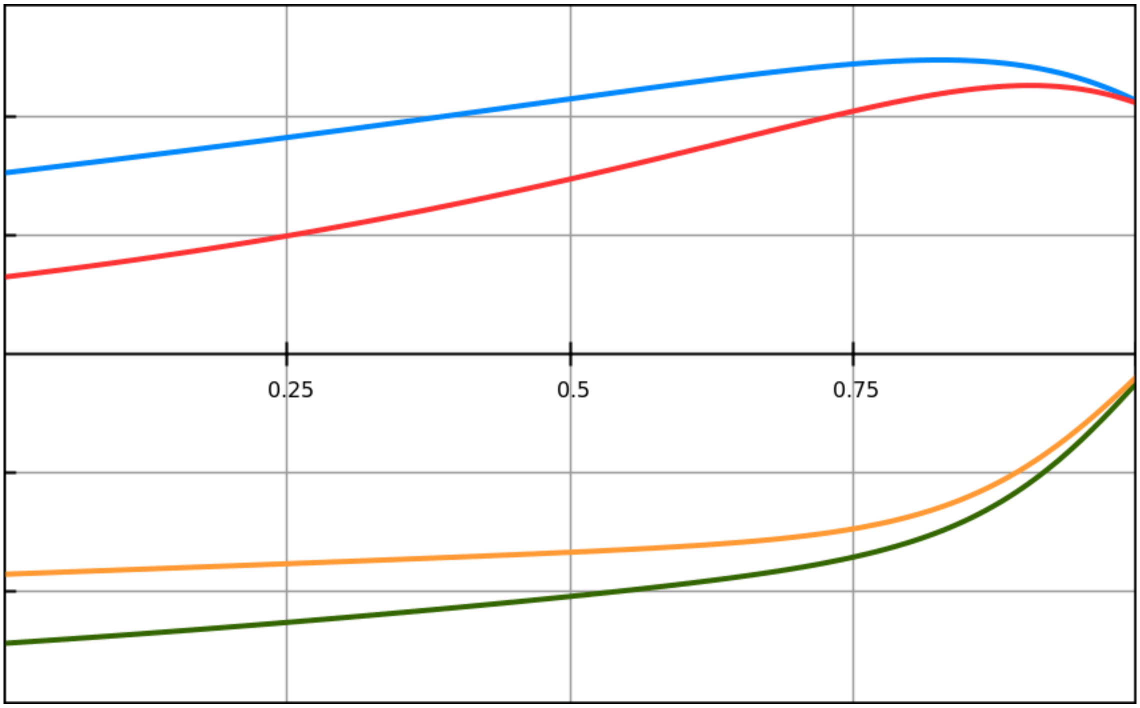

Figure 4.8: Two functions of external motive (blue: valence = -1, red: valence = 0, green: valence = +1, orange: valence = close to zero from negative side). The x-axis indicates the success probability value of achieving a proposed goal which is in [0, +1], and the y-axis indicates the magnitude of the acheivement motive in [-1, +1].

Intuitively, when the human proposes a new goal while expressing a negative emotion the agent should be more motivated to acknowledge human's proposal and pursue the proposed goal to mitigate human's negative emotion and maintain the collaboration.

4.7 Theory of Mind

The Theory of Mind mechanism uses the same collaboration structure and functions as well as appraisal processes to form anticipated beliefs about the human's mental and emotional states. The agent uses the collaboration structure during the human's turn to compute appraisal values with respect to the human's current emotional state and the current goal in the shared goal structure. The outcome of the reverse appraisal forms beliefs about the anticipated mental and emotional state of the human collaborator.

We use the same *relevance*, *expectedness* and *controllability* algorithms for the reverse appraisal as those algorithms we described in Section 4.3. In these three algorithms the Theory of Mind mechanism substitutes the agent's required goal and its corresponding constraints and information with the human's goal and its corresponding information which is provided to the agent within the shared plan structure. In other words, since our agent knows about the human's goals (as part of the shared plan), it can apply the human goals to the same algorithms during the human's turn of the collaboration. However, only for the reverse appraisal of *desirability* we chose to simply use the valence value of the human's perceived emotion and interpret negative, neutral and positive valence values as undesirable, neutral and desirable values respectively. In this way, our agent could directly infer whether the occurrence of the current event and its corresponding goal is desirable for the human. The outcome of all of these processes is a vector of reverse appraisal values that could be used by other mechanisms in our framework.

4.8 Perception and Action

As described in Chapter 3, the Perception and Action mechanisms are not part of our theoretical work. Therefore, we only implemented these mechanisms to the extent to which they could help us to run and test our framework. The Perception mechanism only redirects the input values from the system's users to the framework. For instance, in our conducted user study described in Chapter 5, the Perception mechanism only receives the valence of human's emotion from the input and provides it to the framework. On the other hand, the Action mechanism executes some functions based on the intentions formed and provided by the Coping mechanism described in this section. We group all of these functions into three categories in our framework. The first group of functions includes all of the functions capable of executing some actions with respect to the domain. The second category includes all of the functions involved in revealing the agent's utterances by writing on the screen or conveying through the agent's voice and text to speech systems. The last category includes all of the functions to express the agent's emotion. The emotions can be expressed through colors, emoticons, voice and text. For example, in the user study described in Chapter 5, we expressed the agent's emotions by using emoticons and utterances through the text on the screen as well as the agent's voice.

4.9 Emotion Instances

We have also implemented 10 different emotion instances that can be elicited by the agent or anticipated from the human during collaboration in our framework (see Figure 4.10). These emotion instances have meanings in social context and more specifically in collaboration. There are two components involved in selecting a particular emotion: appraisal variables and collaboration context.

We use the outcome of the four appraisal processes discussed in section 4.3 to determine the potential emotion instance to be elicited (if the agent wants to express an emotion), or to anticipate a potential emotion from the human collaborator (if the human response is anticipated). The outcome of appraisal processes can be one of the values presented in Figure 4.9 with respect to the corresponding process.

We also use the collaboration context as our second determinant to select a particular emotion. We define the collaboration context based on: *goal achievement*,

<table>
<tr><th>Appraisal Variable</th><th>Relevance</th><th>Desirability</th><th>Expectedness</th><th>Controllability</th></tr>
<tr><td rowspan="5">Values</td><td rowspan="2">RELEVANT</td><td>HIGH_DESIRABLE</td><td>MOST_EXPECTED</td><td>HIGH_CONTROLLABLE</td></tr>
<tr><td>DESIRABLE</td><td>EXPECTED</td><td rowspan="2">LOW_CONTROLLABLE</td></tr>
<tr><td rowspan="3">IRRELEVANT</td><td>NEUTRAL</td><td>UNEXPECTED</td></tr>
<tr><td>UNDESIRABLE</td><td rowspan="2">MOST_UNEXPECTED</td><td rowspan="2">UNCONTROLLABLE</td></tr>
<tr><td>HIGH_UNDESIRABLE</td></tr>
</table>

Figure 4.9: Appraisal values.

goal failure, *proposal of a goal*, *acceptance of the proposed goal*, and *rejection of the proposed goal*. All of these situations can occur by either of the collaborators, i.e., agent or human (see Figure 4.10). There is only one exception and it is when the desirability value is neutral the associated emotion to the event is always neutral without considering the collaboration context and the values of other appraisal variables.

As an example, if the agent finds an event *uncontrollable*, *unexpected*, *undesirable* and *relevant* as the result of the human's proposal of a new goal to the agent (in the agent's turn), the elicited emotion instance will be *worry* which can be expressed by the agent to indicate the agent's concern. Similarly, the agent will anticipate *worry* for the human if the same appraisal values are computed while for instance the agent rejects the human's proposal of the new goal (in the human's turn).

4.10 Experimental Scenario

We developed an experimental scenario in which participants were asked to carry out a sequence of hypothetical collaborative tasks between themselves and an imaginary friend, Mary, in order to accomplish their shared goal. To minimize the background knowledge necessary for our test subjects, we used a simple domestic example of preparing a peanut butter and jelly sandwich, and a hard boiled egg sandwich for a hiking trip. The tasks did not require the participants to solve problems; rather, the tasks were part of simple daily activities that should be familiar to all participants.

#	Emotion Instance	Context		Relevance	Desirability	Expectedness	Controllability
1	Neutral				*NEUTRAL*		
2	Joy	human	*HUMAN_ACHIEVED*	*RELEVANT*	*DESIRABLE* *HIGH_DESIRABLE*	*EXPECTED* *MOST_EXPECTED*	
		agent	*AGENT_ACHIEVED*				
3	Sadness	human	*HUMAN_FAILED*	*RELEVANT*	*UNDESIRABLE* *HIGH_UNDESIRABLE*	*EXPECTED* *MOST_EXPECTED*	*UNCONTROLLABLE*
		agent	*AGENT_FAILED*				
4	Gratitude	human	*AGENT_ACCEPTED* *AGENT_ACHIEVED*	*RELEVANT*	*DESIRABLE* *HIGH_DESIRABLE*	*EXPECTED* *MOST_EXPECTED*	
		agent	*HUMAN_ACCEPTED* *HUMAN_ACHIEVED*				
5	Positive Surprise	human	*AGENT_PROPOSED* *AGENT_ACCEPTED* *AGENT_ACHIEVED*	*RELEVANT*	*DESIRABLE* *HIGH_DESIRABLE*	*MOST_UNEXPECTED*	
		agent	*HUMAN_PROPOSED* *HUMAN_ACCEPTED* *HUMAN_ACHIEVED*				
6	Negative Surprise	human	*AGENT_PROPOSED* *AGENT_REJECTED* *AGENT_FAILED*	*RELEVANT*	*UNDESIRABLE* *HIGH_UNDESIRABLE*	*MOST_UNEXPECTED*	
		agent	*HUMAN_PROPOSED* *HUMAN_REJECTED* *HUMAN_FAILED*				
7	Anger	human	*AGENT_REJECTED* *AGENT_FAILED*	*RELEVANT*	*HIGH_UNDESIRABLE*	*EXPECTED* *MOST_EXPECTED*	*UNCONTROLLABLE*
		agent	*HUMAN_REJECTED* *HUMAN_FAILED*				
8	Worry	human	*AGENT_PROPOSED* *AGENT_REJECTED* *HUMAN_FAILED* *AGENT_FAILED*	*RELEVANT*	*UNDESIRABLE* *HIGH_UNDESIRABLE*	*UNEXPECTED*	*UNCONTROLLABLE*
		agent	*HUMAN_PROPOSED* *HUMAN_REJECTED* *HUMAN_FAILED* *AGENT_FAILED*				
9	Frustration	human	*AGENT_PROPOSED* *AGENT_FAILED*	*RELEVANT*	*UNDESIRABLE*	*EXPECTED* *MOST_EXPECTED*	*UNCONTROLLABLE*
		agent	*HUMAN_PROPOSED* *HUMAN_FAILED*				
10	Guilt	human	*HUMAN_FAILED*	*RELEVANT*	*UNDESIRABLE* *HIGH_UNDESIRABLE*	*EXPECTED* *MOST_EXPECTED*	*LOW_CONTROLLABL* *HIGH_CONTROLLABLE*
		agent	*AGENT_FAILED*				

Figure 4.10: Conditions for selecting emotion instances

4.10.1 Hypothesis

We conducted this user study to test our hypothesis that humans and our algorithms will provide similar answers to questions related to different factors used to compute four appraisal variables: relevance, desirability, expectedness, and controllability.

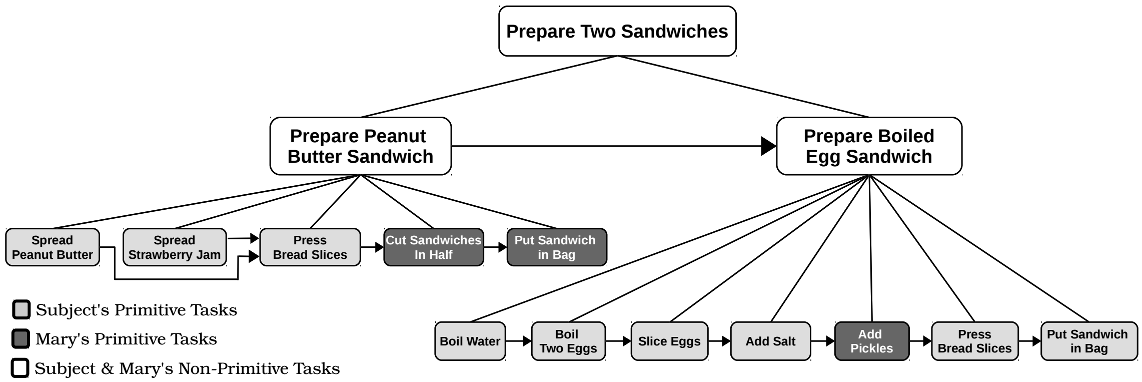

Figure 4.11: Collaboration task model for the evaluation.

4.10.2 Procedure

We conducted a between-subject user study using an online crowdsourcing website – CrowdFlower[1]. We had a questionnaire for each appraisal variable. There were 12 questions (including 2 test questions) in the controllability and expectedness questionnaires, 14 questions (including 2 test questions) in the desirability questionnaire, and 22 questions (including 3 test questions) in the relevance questionnaire.

We provided textual and graphical instructions for all questionnaires; Figure 4.11 shows the corresponding task model. The instructions, provided in the Appendix, presented a sequence of hypothetical collaborative tasks to be carried out by the test subject and an imaginary friend, Mary, in order to accomplish their goal of preparing two sandwiches. We also provided a simple definition and an example of each appraisal variable. The collaboration structure and the instructions were the same for all questionnaires. The questions introduced specific situations related

[1]http://www.crowdflower.com

to the shared plan, which included blocked tasks and failure or achievement of a shared goal. Each question provided three answers which were counterbalanced in the questionnaire. We provided an option like C in all questions (see Figure 4.13), because we did not want to force participants to choose between two options when they did not have a good reason. There were two questions designed based on each factor that we use in our algorithms (see Section 4.3). The questions were randomly placed in the questionnaire. Figure 4.13 shows an example question from the relevance questionnaire which was designed to test whether participants perceive saliency as a factor in relevance. The input for our algorithms was the task model depicted in Figure 4.11.

4.10.3 Participants

Each participant group originally had 40 participants. We limited the participant pools to those with the highest confidence level on the crowdsourcing website in the United States, Britain, and Australia. Test questions were included to check the sanity of the answers. We eliminated participants providing wrong answers to our sanity questions, and participants with answering times less than 2 minutes. The final number of accepted participants in each group is provided in Table 4.1.

Table 4.1: Number of participants

appraisal variables	# of participants
Relevance	29
Desirability	35
Expectedness	33
Controllability	33

4.11 Results

Each question in our questionnaires was designed based on different factors that we use in our algorithms (see Section 4.3). For each of the four questionnaires we provide an example question, and describe how each question relates to a specific factor within the corresponding algorithm. The input for our algorithms was the task model depicted in Figure 4.11. The complete list of questions is provided in the Appendix. Additionally, we provide the p-value for each question, using a binomial distribution, with a probability of success of 0.33, which is the probability of selecting the right answer if the participant is simply guessing.

4.11.1 Expectedness

Figure 4.13 shows an example question from the expectedness questionnaire. In this example, with respect to Algorithm 3 (line 6), option A is more expected because the task related to this option provides the next available task in the focus stack (see the task model in Figure 4.11). Although the task in option B is part of the existing task model, it is considered as unexpected by our algorithm, since it is not live in the plan. We provided option C to determine whether the participants will similarly differentiate between these two options. This question was presented to the participants to determine whether their decision for the expectedness of this event is similar to the output of the expectedness algorithm. For this question, the human decision was 97% similar to the algorithm's output.

Results for the expectedness questionnaire are presented in Figure 4.12 (Refer to Expectedness summary table). As shown in this table, there is strong evidence that the results are not random; in fact, for questions 1-6 and 9-10, human participants showed between 67 and 100 % agreement with our algorithms, with p-values of ¡¡0.001 when compared with a random population. Questions 7 and 8 were the only two questions that did not show a statistically significant p-value. It should be noted that these questions are comparing equally expected or equally unexpected situations, none of which our algorithms would consider most-expected or most-unexpected.

Question	Factor	Equivalent Condition	Number of Matching Answers	*p*-Value
1	Live goal vs. Necessary focus shift	No	31	« 0.001
2	Live goal vs. Not part of shared plan	No	32	« 0.001
3	Live goal vs. Not part of current branch	No	27	« 0.001
4	Necessary focus shift vs. Not part of shared plan	No	33	« 0.001
5	Necessary focus shift vs. Not part of current branch	No	32	« 0.001
6	Not part of shared plan vs. Not part of current branch	No	24	« 0.001
7	Live goal	Yes	14	0.093
8	Not part of current branch	Yes	14	0.093
9	Necessary focus shift	Yes	22	« 0.001
10	Not part of shared plan	Yes	29	« 0.001

Figure 4.12: Expectedness results.

4.11.2 Controllability

Figure 4.15 shows an example question from the controllability questionnaire. The algorithm's output is option B, and is determined by Algorithm 4 (line 3), similarly to the expectedness example above. In this example, option B is more controllable than option A, because the self over total ratio of the responsibility of the predecessors of the given task (see *Autonomy* in Section 4.3.4) is higher than the ratio in

Imagine you have pressed the two slices of bread (one covered with strawberry jam and one covered with peanut butter) together and passed it to Mary. Which of the following two actions is **more expected**?

A. Mary puts the given sandwich into a zip lock bag after cutting it in half.

B. Mary puts some pickles on another slice of bread.

C. Equally expected.

Figure 4.13: Example expectedness question.

option A; i.e., self is responsible to spread peanut butter on one slice of bread and strawberry jam on another slice of bread. In this question, the humans decision was 90% in agreement with the algorithm's output.

Question	Factor	Equivalent Condition	Number of Matching Answers	*p*-Value
1	Agency	No	28	« 0.001
2	Autonomy (contributors)	No	17	0.009
3	Autonomy (predecessors)	No	30	« 0.001
4	Succeeded predecessors ratio	No	19	0.001
5	Available inputs	No	30	« 0.001
6	Agency	Yes	30	« 0.001
7	Autonomy (contributors)	Yes	24	« 0.001
8	Autonomy (predecessors)	Yes	18	0.003
9	Succeeded predecessors ratio	Yes	23	« 0.001
10	Available inputs	Yes	25	« 0.001

Figure 4.14: Controllability results.

Results for the controllability questionnaire are presented in Figure 4.14 (insert controllability summary table). As shown in the table, the p-value is ¡0.01 for each of the ten questions. The two questions with the lowest human agreement with the algorithms both relate to autonomy of the participants with 52% and 55%.

Imagine you want to make a peanut butter sandwich. Which of the following two actions is **more controllable**?

A. You can spread the peanut butter on one slice of bread and you need Mary to spread strawberry jam on the second slice of bread.

B. You can spread the peanut butter on one slice of bread and strawberry jam on the second slice of bread.

C. Equally controllable.

Figure 4.15: Example controllability question.

4.11.3 Desirability

Figure 4.17 shows an example question from the desirability questionnaire. The output based on the Algorithm 2 (line 14) is option C, since in both option A and option B, the focus goal has been achieved successfully. Therefore, in this example, both options A and B are desirable. The humans' decision was 77% in agreement with the algorithm's output in this question.

Question	Factor	Equivalent Condition	Number of Matching Answers	*p*-Value
1	Top level goal is failed	No	35	« 0.001
2	Top level goal is achieved	No	29	« 0.001
3	Predecessors or preconditions of the top level goal	No	35	« 0.001
4	Focus is achieved	No	34	« 0.001
5	Focus is failed	No	35	« 0.001
6	Predecessors or preconditions of the focus	No	35	« 0.001
7	Pending or in-progress focus	Yes	16	0.040
8	Top level goal is failed	Yes	23	« 0.001
9	Predecessors or preconditions of the top level goal	Yes	19	0.003
10	Focus is achieved	Yes	20	0.001
11	Focus is failed	Yes	21	« 0.001
12	Predecessors or preconditions of the focus	Yes	27	« 0.001

Figure 4.16: Desirability results.

The results of the desirability questionnaire are presented in 4.16 (insert desirability summary table). As shown in the results table, the p-value is less than 0.05 for all of the desirability questions. However, an interesting trend is that human participants had a level of agreement of 83%-100% when the algorithm's output selected one alternate as more desirable than another alternate. When the algorithm's output chose option C (i.e. rating two situations as equally desirable), the human participants only showed 46%-77% agreement. This may indicated that a higher level of granularity is required in the algorithm when evaluating options with similar levels of desirability.

Which of the following two actions is **more desirable**?

A. Imagine you pressed two slices of bread together with peanut butter and strawberry jam on them, and passed them to Mary. Mary cuts the peanut butter sandwich in half and puts them in the zip lock bag.

B. Imagine you want to make the egg sandwich. You have sliced the eggs, put them on one slice of bread, salted them, and waiting for Mary to put some pickles on your eggs. Mary puts some pickles on your eggs.

C. Equally desirable.

Figure 4.17: Example desirability question.

4.11.4 Relevance

In the example shown in Figure 4.19, with respect to Algorithm 1, option A is relevant because of Mary's perceived negative emotion (see Equation 4.1). Although option B is relevant (since it achieves the next goal in the shared plan), 83% of participants consider it as less relevant than option A; we believe this is due to the effect of Mary's perceived negative emotion which also generates a higher utility value in our relevance algorithm. Another question also tested belief saliency. However, the options provided only related to the shared plan (i.e., no human emotions in the options). In this case 87% of participants chose the option that accomplished the next goal in the shared plan. Interestingly, when confronted with a negative emotion from their collaborator, human participants deviated from the shared plan and found their collaborator's emotion more relevant than the original plan. It is noteworthy that in both the absence and the presence of emotions the participants chose the more salient option with respect to our definition of saliency, which was not referenced or provided in the questionnaire.

The complete summary of results for the relevance questionnaire is provided in 4.18 (Insert summary table for relevance). As shown in the table, all questions show

Question	Factor	Equivalent Condition	Number of Matching Answers	*p*-Value
1	Belief Saliency	No	25	« 0.001
2	Belief Strength	No	13	0.063
3	Belief Recency	No	28	« 0.001
4	Motive Insistence	No	25	« 0.001
5	Motive Urgency	No	19	« 0.001
6	Motive Intensity	No	21	« 0.001
7	Goal Proximity	No	20	« 0.001
8	Goal Specificity	No	23	« 0.001
9	Belief Saliency	Yes	26	« 0.001
10	Belief Strength	Yes	22	« 0.001
11	Belief Recency	Yes	21	« 0.001
12	Motive Insistence	No	26	« 0.001
13	Motive Urgency	Yes	29	« 0.001
14	Motive Intensity	Yes	29	« 0.001
15	Goal Proximity	Yes	24	« 0.001
16	Goal Specificity	Yes	26	« 0.001
17	Belief Saliency	No	17	« 0.001
18	Motive Insistence	No	3	0.995
19	Goal Proximity	No	4	0.982

Figure 4.18: Relevance results.

59%-100% agreement with our algorithms and statistically significant p-values except for questions 2, 18 and 19. Question 2 addresses belief strength. Questions 18 and 19 address motive insistence and goal proximity, respectively; both of these questions present situations in which participants must choose whether an intense emotional circumstance, or adherence to the collaboration plan is more relevant (refer to the questionnaire provided in the Appendix). Our algorithms choose that the strong emotional circumstance will be more relevant; however, human participants generally selected adherence to the collaboration plan to be more relevant.

Imagine you have made the peanut butter sandwich and passed it to Mary to cut it in half. Which of the following two actions is **more relevant**?

A. Mary starts crying since she cut her finger with a knife.

B. You begin to boil the water to boil the eggs for your second sandwich.

C. Equally relevant.

Figure 4.19: Example relevance question.

4.12 Discussion

As shown in the results tables in sections 4.11.1 through 4.11.4, the human participants agreed 100% on some questions, while on some other questions there was a much lower level of agreement. Our results indicate that people largely performed as our hypothesis predicted. The p-values obtained based on a binomial distribution show the probability of human participants' answers being generated from a random set. The very small p-values indicate that the data set is not random; in fact, the high percentage of similarity confirms our hypothesis and shows that the algorithms can help us to model appraisal in a collaboration. The very low level of agreement on a handful of questions may indicate algorithm components that require further refinement before implementation.