Collaborative Behaviour Modelling of Virtual Agents using Communication in a Mixed Human-Agent Teamwork

Mukesh Barange, Alexandre Kabil, Camille De Keukelaere, and Pierre Chevaillier
ENIB (UEB), Lab-STICC
Brest, France
{barange, kabil, dekeukelaere, chevaillier}@enib.fr

Abstract—The coordination is an essential ingredient for the mixed human-agent teamwork. It requires team members to share knowledge to establish common grounding and mutual awareness among them. In this paper, we proposed a collaborative conversational belief-desire-intention ($C^2$BDI) behavioural agent architecture that allows to enhance the knowledge sharing using natural language communication between team members. We defined collaborative conversation protocols that provide proactive behaviour to agents for the coordination between team members. Furthermore, to endow the communication capabilities to $C^2$BDI agent, we described the information state based approach for the natural language processing of the utterances. We have applied the proposed architecture to a real scenario in a collaborative virtual environment for training. Our solution enables the user to coordinate with other team members.

Keywords—Human interaction with autonomous agents, Cooperation, Dialogue Management, Decision-Making

I. INTRODUCTION

In collaborative virtual environments (VE) for training, human users, namely learners, work together with autonomous agents to perform a collective activity [1]. The educational objective is not only to learn the task, but also to acquire social skills in order to be efficient in the coordination of the activity with other team members [2]. Effective coordination improves productivity, and reduces individual and team errors. The ability to coordinate one’s activity with others relies on two complementary processes: common grounding [3] and mutual awareness [4]. Common grounding leads team members to share a common point about their collective goals, plans and resources they can use to achieve them [3]. Mutual awareness means that team members act to get information about others’ activities by direct perception, information seeking or through dialogues, and to provide information about theirs [4].

The collaboration in a human-agent teamwork poses many important challenges. First, there exists no global resource that human team members and virtual agents can rely on to share their knowledge, whereas, in a team of autonomous agents, the coordination can be achieved through the means of a mediator, or blackboard mechanism. Second, the structure of the coordination between human-agent team members is open by nature: virtual agents need to adopt the variability of human behaviour, as users may not necessarily strictly follow the rules of coordination. In contrast, in agent-agent interactions, agents follow the rigid structure of coordination protocols (e.g., contract net protocol). Thus, the ability to coordinate with human team members requires to reason about their shared actions, and situations where team members need the coordination to progress towards the team goal. Moreover, another important characteristic of the human-human teamwork is that the team members pro-actively provide information needed by other team members based on the anticipation of other’s needs of information [5]. Thus, in a human-agent team, agents should allow human team members to adjust their autonomy and help them to progress in their task. Thus, an effective solution, supporting human-agent communications, is highly needed in a mixed human-agent teamwork. Furthermore, to exhibit the natural language communication capability, an important challenge is that the agents must take into account not only the current context of the ongoing dialogues, but also about the current context of the task and the beliefs about other team members.

This paper is the continuation and the extension of the work presented in [1]. The paper focuses on the task-oriented, collaborative conversational behaviour of virtual agents in a mixed human-agent team. Other aspects of embodied virtual agents, such as emotions, facial expressions, non-verbal communication, etc. are out of the scope of this study. As the team members must have the shared understanding of skills, goals and intentions of other team members, we proposed a belief-desire-intention based (BDI-like) agent architecture named as Collaborative-Conversational BDI agent architecture ($C^2$BDI). On the one hand, this architecture provides the deliberative behaviour for the realisation of collective activity and, on the other hand, it provides conversational behaviour for the dialogue planning to exhibit human like natural language communication behaviour for coordination. The contributions of this paper include: (1) a decision-making mechanism, in which the dialogues and the beliefs about other agents are used to guide the action selection mechanism for agents to collaborate with their team members. (2) the definition of collaborative communication protocols to establish mutual awareness and common grounding among team members; and (3) the information state based natural language processing for the task-oriented multiparty conversation. The approach consists in formalizing the conversational behaviour of the agent related to the coordination of the activity, which reduces the necessity to explicitly define communicative actions in...
the action plan of the agent. It also makes the human-agent interaction more adaptive.

In Section II, we present related work on human-agent teamwork. Section III presents different components of the proposed C²BDI architecture. The information state based context model is presented in Section IV. Section V describes the decision making mechanism of C²BDI agent that provides the interleaving between deliberation and conversational behaviour of the agent. The collaborative conversational protocols are presented in Section VI. The natural language processing in C²BDI agent is presented in Section VII. The next section illustrates how the solution fulfils the requirements of real educational scenarios. The discussion over the comparison of C²BDI agent with existing approaches is presented in Section IX. Finally, Section X summarises our positioning.

II. RELATED WORK

Both AI and dialogue literature agree upon the fact that to coordinate their activities, agents must have the joint-intention towards the group to achieve collective goal [6] and must agree upon the common plan of action [7]. Cohen and Levesque proposed the joint-intention theory, which specifies that the agents must have common intentions towards the group goal [6]. This theory does not guarantee that agents follow the same action plan. Comparing to this theory, the shared-plan theory proposed by Grosz and Kraus [7] specifies that even agents share a common action plan to achieve the group goal, it does not guarantee that agents have the commitment towards the group to achieve that goal. Both of these theories are mainly applied for the coordination among a group of artificial agents. The C²BDI architecture takes the advantage of both of these theories to establish common grounding and mutual awareness among mixed human-agent team members.

A number of human-agent team models have been proposed in the literature [8]–[10]. Rich and Sidner proposed the Collagen agent [8] and Disco for Games (D4g) that is a successor of Collagen [9], which are built upon the human discourse theory and can collaborate with a user to solve domain problems, such as planning a travel itinerary, to generate dialogue about baseball and user can communicate with agents by selecting the graphical menus. In [10], Bradshaw et al. described the teamwork notification policies based collaboration model. In their model, when an important event occurs, the agent may notify the user with respect to appropriate modality and the user’s position. To achieve collaboration between team members, Wooldridge and Jennings proposed a four stage model collaboration model [11] that includes (i) recognition of the potential for cooperation, (ii) team formation (iii) plan formation, and (iv) plan execution. Based on this model, Dignum et al. proposed an agent model and define how collective intentions from the team formation stage are built up from persuasion and information-seeking speech act based dialogues, using motivational attributes goal and intention [12]. Moreover, Blaylock and Allen proposed an agent based dialogue system by providing dialogue acts for collaborative problem solving to model communication at the utterance level, between a user and a system that focuses only on establishing coordination at the beginning of the shared activity [13]. Comparing to this approach, C²BDI agents coordinate with team members not only at the beginning but also during the realisation of the shared task. Recently, Kamali et al. [14] have proposed a theoretical framework for proactive information exchange in agent teamwork to establish shared mental model using shared-plan approach [7].

Among many other approaches, such as speech act [15] or plan-based [8], [9], [16], the information state (IS) based approach [17] is one of the prominent approaches for dialogue modelling. It contains contextual information about the current conversation. Bunt has defined the IS, which contains contextual information of dialogue that includes dialogue, semantic, cognitive, perceptual, and social context [18], [19]. This context model includes major aspects to control natural language dialogues. However, it does not include contextual information about the shared task being carried out by the agent [20]. This leads to an incoherence between dialogue context and shared task in progress. Kopp and Pfeiffer-Lessmann proposed an IS based interaction model for Max agent [20]. They considered coordination as an implicit characteristic of team members. Moreover, Bunt proposed a taxonomy of dialogue acts (DIT++) based on the dialogue interpretation theory [19]. The semantics of these dialogue acts are based on the IS based approach. This taxonomy was built mainly to annotate natural language dialogues. We are motivated to use it to understand and interpret conversation between human-agent team due to its following characteristics: (i) it is mainly used for dialogue interpretation in human-human conversation; (ii) it supports task oriented conversation; and (iii) it has become the ISO 24617-2 international standard for dialogue interpretation using dialogue acts.

III. C²BDI AGENT ARCHITECTURE

In this section, we describe components of C²BDI agent architecture that provide deliberative and conversational behaviours for collaboration (Fig. 1). The C²BDI agent architecture is based on the Belief, Desire, Intention architecture (BDI) [21] and treats both deliberative and conversational behaviours uniformly as guided by the goal-directed shared activity. The originality, compared to pure BDI, lies first on the role of dialogue, that modifies together the believes, the desire and the intentions of the agent, and second on the collaborative nature of the agent’s activity. Different components of the architecture are summarised as follow:

a) Decision Making: In C²BDI agent, the decision making includes deliberation control, and reactive behaviour control modules.

Deliberation control: Its main task is to decide how can the agent deliberate its goal to decide which one should be pursued. The decision process is driven by the information about the goals, activity plans, Information-state and the semantic knowledge of VE and of the task.

Reactive behaviour control: It uses the multimodal perception information from perception memory to reason about whether participants are in contact with, are they visible, whether someone is talking in the group, and whether the agent have a turn to talk. It implements the limited multi-model features of YMIIR architecture [22] to manage multi-party conversation in particular to manage turn taking behaviour.

b) Knowledge Base: The organisation of knowledge in C²BDI agent allows to establish the strong coupling be-
tween decision making and the collaborative conversational behaviour of the agent. The knowledge base consists of semantic knowledge, perception memory and IS. The semantic knowledge contains semantic information that is known a priori by the agent, such as the knowledge concerning concepts, and individual and shared plans. Following the shared-plan theory [7], C²BDI agents share the same semantic knowledge about the VE and the group activity. This simplifies the planning process of agents, as agents need to construct only their local plan. Moreover, sharing the same semantic knowledge also supports proactive conversation behaviour of the agent as it allows the decision making process to identify collaborative situations and information needed by other team members. The perception memory acquires information about the state of the VE perceived by the perception module. This memory contains the belief about the state and properties of the entities in VE, and the state and actions of the team members. The IS contains contextual information about the current activity and dialogues.

c) Belief Revision: It specialises the belief revision function of BDI [21] by using the capabilities of the agent, resources used in the activity, and the Information-state. It maintains the consistency of both the knowledge base and of the Information-state by updating agent’s beliefs about the current state of the world, resources and capabilities of team members using current perceptions. In the classical BDI architecture, the belief-revision is the internal component of the decision making module, however, in C²BDI architecture, the belief-revision is placed outside. The reason behind this is that in C²BDI agent, the beliefs are updated not only from the decisions made by the agent, but also, from the information perceived by the agent.

d) Dialogue Management: The dialogue manager allows an agent to share its knowledge with other team members using natural language communication. It supports both reactive and proactive conversation behaviours, and ensures coordination of the activity. In C²BDI agent architecture, the natural language understanding (NLU) and generation (NLG) of spoken dialogues is based on the rule based approach [23]. When the agent receives an utterance, it uses NLU rules to determine the corresponding dialogue act [18], [19]. It identifies dialogue contents using semantic knowledge and contextual information from IS. The dialogue manager processes these dialogue acts and updates IS based on update rules similar to [17]. When the agent has communicative intentions, it constructs dialogue act moves and update its IS. NLG rules are used to generate natural language utterance corresponding to these dialogue moves based on the current context from IS.

e) Perception: The C²BDI agent perceives VE through the perception module. The current perceived state of VE is an instantiation of concepts the agent holds in its semantic knowledge. The perception module allows agents to enrich their knowledge, and to monitor the progress of the shared activity.

f) Behavior Realiser: The behaviour realiser module is responsible for the execution of actions and the turn taking behaviour of the agent.

IV. INFORMATION STATE BASED EXTENDED CONTEXT MODEL

In this section, we present the proposed context model that allows a C²BDI agent to store and maintain information necessary for the decision making, and natural language conversation.

The IS is primarily used in literature to control natural language dialogues [17], [19]. We extended its usage as the source of knowledge between the decision-making and conversational behaviour of the C²BDI agent to establish coherence between these two processes. The IS represents the context model of the agent, and works as an active memory that contains beliefs and intentions of the agent.

To participate in the task-oriented communication and to establish and maintain coordination among team members, the agent not only requires the current context of the dialogue and beliefs about the world, but also the information about the current context of the task, beliefs about other team members, and the collective attitudes. To acquire these information, we have extended the IS based context model of [19] by adding the task context to it (Fig.2). The extended context model includes:

- **Dialogue context**: It contains different components, that represents the features about the agents dialogue acts, and other speaker’s dialogue acts. Speaker’s dialogue acts contains the utterance received from the speaker, and the dialogueActs generated from the interpretation of the utterance. The agents dialogue acts contains the dialogue acts generated by the agent itself. The component nextMoves includes the list of dialogue moves available for the generation by the agent. Moreover, the dialogueActHistory stores the complete history about the agent’s, and other speaker’s dialogue acts, as well as about the integrated dialogue moves.

- **Semantic context**: It not only contains the agent’s beliefs about the current state of the VE, but also about the current progress of the dialogues. It contains a private component that includes following features:
  - The feature beliefs, is instantiated from concepts
InFocus, beliefs, Dialogue Context


Semantic Context

agenda, proactiveAgenda, communicativePlan, beliefs, expectations

Cognitive Context

mutual-belief

Social Context

communication-pressure

Perceptual Context

objectInFocus, agentInFocus, third-personInFocus, actionInFocus

Task Context

cooperative-info, group-goal, group-desire, group-intention, joint-goal, joint-desire, joint-intention, joint-commitment

private, task-focus, goals, desires

Figure 2: Extended Information State based Context Model

the agent holds in semantic knowledge, and updated depending on the progress of the shared task. (b) The feature agenda contains the communicative intention of the agent. These intentions are added to the agenda due to communicative intentions generated by the realisation of the collaborative task and by the social obligations carried out by the agent. (c) The proactiveAgenda stores the communicative intentions of the agent generated due to the proactive communication behaviour of the agent. The agent can proactively generate the communication intention in order to establish or maintain the cooperation with other team members, to satisfy the anticipated need of informations of self or of others, or to handle the resource sharing with other team members. The semantic context also contains the information about the expectation that the agent can have from others. Moreover, the feature communicativePlan can contain a communicative plan that an agent may have to be executed.

- **Cognitive context**: It includes the mutual belief among self and other team members as the result of the mutual awareness and common grounding. The team members communicate with each other in order to establish mutual belief among them. For example, the agent establishes with other team members the mutual belief about the collective decision of the choice of the shared goal, and also for the collective decision of the plan of action to be chosen to achieve the selected goal.

- **Social context**: It includes the information about the communication pressure such as greet open, close, etc.

- **Perceptual context**: It contains information on which the agent pays attention during conversation and during the realisation of the task. The perceptual context contains an attention stack, which includes the information about the current object in focus (objectInFocus), actor in focus (actorInFocus), and also keeps the information about the third-person in focus (thirdPersonInFocus). In contrast to [17], the agent does not only update such information from the dialogue, but also by using information acquired in its perceptual memory. This information is particularly necessary to understand the natural language utterance in particular for the resolution of pronouns and the instantiation of contextualised semantic knowledge of the agent during NLU and NLG.

  - **Task context**: It includes information about the current task in progress.

The task context is divided into two components: private, and cooperative information (cooperative-info). The private component of task context contains:

- **desires**, which contains the set of expected desires (expected state of the worlds) for the agent.

- **goals**, which contains a set of potential desires to be achieved individually or collectively.

- **task-focus**, which is a stack that contains the current intention of the agent about the task. The type of intentions in task-focus can any of the Int.To, Int (i.e., intention that), Pot.Int.To and Pot.Int.Th [7].

To ensure that each team member has a common intention towards the team goal, the cooperative-info in task context of IS includes beliefs about collective attitudes which includes: group-goal, group-desire, group-intention, joint-goal, joint-desire, joint-intention and joint-commitment. These shared mental attitudes in task context of an agent towards the group specifies that each member holds beliefs about the other team members, and each member mutually believes that every member has the same mental attitude. We distinguish between individual, group and joint mental attitudes of the agent.

The C²BDI agent constructs beliefs about these mental attitudes in collective-info of task context in a progressive manner during the process of establishing the cooperation among team members through communication. The group-goal indicates that the agent knows that all team members want to achieve the goal at a time or another. Similarly, group-desire and group-intention can be defined analogously. For an agent a group-intention becomes a joint-intention when the agent knows that this intention is shared by other team members. To form a joint-intention, a necessary condition is that the agent must have individual intention to achieve this goal. Similarly, the semantics of joint-desire and joint-goal indicates that all team members have the same group-desire and group-goal respectively, and all team members know it. Thus, these shared mental attitudes towards the group specify that each member holds beliefs about other team members, and each member mutually believes that every member has the same mental attitude.

The joint-intention only ensures that each member is individually committed to acting. The agent must also ensure the commitment of others to achieve this shared goal. Agents must communicate with other team members to obtain their joint-commitments. The agent has a joint-commitment towards the group, if and only if, each member of the group has the mutual belief about the same group-goal, the agent has the joint-intention about to achieve that goal, and each agent of the group is individually committed to achieve this goal. Hence, the IS not only contains information about the current context
of the dialogue, but also that of the collaborative task, i.e., beliefs about other team members potentially useful for the agent for its decision-making.

V. DECISION MAKING MECHANISM

In C²BDI agent, decision-making is governed by information about current goals, shared activity plans, and knowledge of the agent (IS and semantic knowledge). The decision making algorithm is shown in Algo. 1.

Algorithm 1 Decision making algorithm

Require: IS, GAGT, GAPs
1: B = IS.SemanticContext.Belief
2: D = IS.TaskContext.Desire
3: I = IS.TaskContext.Intention
4: agenda= IS.Semantic-context.agenda
5: proactiveAgenda= IS.Semantic-context.proactiveAgenda
6: while GAGT is not completely processed do
7: update-perception(ρ)
8: Compute B, D, I, and update IS
9: II ⇐ Plan(P, I )
10: while !II.empty() do
11: if agenda or proactiveAgenda is not empty or the agent has received an utterance then
12: Process Conversation-Behavior()
13: Compute new B, D, I , and update IS
14: II ⇐ Plan(P, I )
15: if the task-focus contains communicative intention then
16: Process Conversation-Behavior()
17: Compute new B, D, I , and update IS
18: II ⇐ Plan(P, I )
19: Identify-Cooperative-Situation in the current plan II
20: if Cooperative-Situation is matched then
21: Process Conversation-behaviour()
22: α ⇐ Plan-action(II)
23: execute(α)

The decision making process verifies whether the agenda in IS is not empty or if the agent has received an utterance. If so, control is passed to the conversational behaviour to that supports natural language communication. After executing the communication behaviour, the agent re-evaluates its beliefs, desire and intentions because the communication can modify the mental state of the agent through the updates in its IS.

If the task-focus in task context contains the communicative intention, then also the control is passed to the conversation behaviour. This situation can occur when the agent is executing some predefined conversation plan based on the current context of the task. In this case also, the agent recomputes its desire and intentions.

Otherwise, the agent chooses the plan to be realised. If it identifies cooperative situations in the collective activity where the agent cannot progress without assistance, it requires other team members to cooperate in order to achieve shared group goal. The decision making passes the control to conversation behaviour of agent in order to make establish joint commitment towards the group to achieve the goal, or when the agent needs to share the status of the goal, i.e., the goal has been achieved, or the goal is no more achievable. This situation generates communicative intentions in the agenda or in the proactiveAgenda that cause the agent to interact with team members to share their knowledge.

The agent updates its IS if the control is passed to the conversational behaviour, and deliberate the plan to generate a new intention. Once the intention is generated, the agent selects an action to be realised and updates its task-focus in IS to maintain knowledge about the current context of the task.

In this procedure, it is important to note that the conversational behaviour of the agent can be called in one of the following situations:

- when the agenda in semantic context is not empty or when the agent receives an utterance from the user or from other agents. This is the reactive conversation behaviour of the agent that interprets the utterance by identifying its dialogue act (Sec. VII-B), integrates the effects of the generated dialogue act by updating different components of IS (Sec. VII-C), and generating appropriate dialogue move with respect to the speaker’s dialogue act (Sec. VII-D) for the generation of natural language utterance.

- when the proactiveAgenda is not empty. This situation occurs in the following conditions:
  - when the agent needs the team coordination, and wants to establish group belief towards this,
  - when the agent identifies the information need of self or of others, and wants to establish group belief by providing the information or by asking for the information, respectively. For example, this situation occurs when the agent identifies the need of the resource, or wants to provide the information about resource by knowing that the addressee needs this information.

  - when the agent executes predefined conversation plan. C²BDI agent exhibits the capability of executing preplanned conversation plans in the same was as the activity plan. However, one of the important difference between the conversation plan and the shared activity plan is that the conversation plan is executed locally by the host agent, and unlike shared activity plan, other team members do not monitor the progress of that plan. The agent deliberates the conversation plan and adds an intention Int.To to the task-focus in order to execute a conversation operation. The execution of the conversation operation results in updates in IS by construction of appropriate dialogue act and adding it to agentsDialogueActs in linguistic context, and adding corresponding communicative intention to the proactiveAgenda

The conversational behaviour allows a C²BDI agent to share its knowledge with other team members using natural language communication, and ensures the coordination of the team activity. The agent interprets and generates the dialogues based on the semantics of dialogue acts proposed in [19] using current IS. To achieve the coordination among team members, we propose collaborative conversational protocols for the agent. These protocols construct the conversational desires for the agent which, when activated, result in conversational
VI. Collaborative Conversational Protocols

As we want the agent to be proactive and cooperative, we have defined three collaborative conversational protocols (CCPs). These protocols ensure the establishment of the collaboration among team members to achieve the group-goal, and its end when the current goal is achieved. Every team member participating in a collaborative activity enters in the collaboration at the same time, and remains committed towards the group until the activity is finished. These protocols are modelled as the update operations in the IS based on the current context of the task and the dialogue.

A. CCP-1

When the agent has a new group-goal to achieve, it communicates with other team members to establish joint-commitment, and to ensure that every team member use the same plan to achieve the group-goal. Algo. 2 describes how team members collectively choose the common goal in order to establish joint-goal.

Algorithm 2 CCP1: Collective decision for Goal Choice

Require: group G, and shared goal  metavariable IS

At speaker side—:

1: if Group-Intention(G,ϕ) ∧ ¬Mutual-belief(G,ϕ) then
2: if size(Group-Goals)==1 then
3:  IS ← addTopOfProactiveAgenda Set-Q(what-team-next-goal All)
4: else if size(Group-Goals) > 1 then
5:  IS ← addTopOfProactiveAgenda Choice-Q(what-team-next-goal)
6:  IS ← addTopOfAgenda Inform(Prefer(Ai,,ϕ,Pϕ),Aj))
7: else if Receive(Choice-Q(what-team-next-goal)) then
8:  IS ← addTopOfAgenda Inform(Prefer(Ai,,ϕ,Pϕ),Aj))
9: IS ← Mutual-Belief(G,ϕ)
10: IS ← Joint-Goal(G,ϕ)
11: extract IS ← Expected(plan-choice,–,ϕ)
12: IS ← addTopOfAgenda Inform(Auto-feedback(positive-ack), All)

Similarly at receiver side—:

13: if (Receive(Set-Q(what-team-next-goal), Aj)) then
14: Receive(Choice-Q(what-team-next-goal), Aj))
15: Group-Intention(G,ϕ) ∧ ¬Mutual-belief(G,ϕ)
16: IS ← addTopOfAgenda(Inform(team-next-goal, A1,ϕ))
17: IS ← Mutual-Belief(G,ϕ)
18: IS ← Joint-Goal(G,ϕ)
19: else if Receive(what-team-next-action Ai) then
20: IS ← addTopOfAgenda(Inform(team-next-action(ϕ), Ai))

When the agent Ai has one or more group-goals to achieve (line 1), and if it has no mutual belief about them, it constructs Set-Q(what-team-next-goal) (if Ai has only one goal) or constructs Choice-Q(what-team-next-goal) (if Ai has more than one goal) dialogue act and addresses it to the group. This results in the addition of a communicative action to the proactiveAgenda in semantic context of IS. By addressing this open question, Ai allows both the user and other agents to actively participate in the conversation. If Ai receives the choice of the goal from another team member (line 7), i.e., when it receives the proposition team-next-goal, it adds a mutual belief about group-goal to its cognitive context, and the belief about joint-goal to the task context. It then confirms this choice by sending a positive acknowledgement (by constructing Auto-feedback(positive-ack)) to the speaker.

When the Ai receives Set-Q(what-team-next-goal) or Choice-Q(what-team-next-goal) from Aj, and has no mutual belief about group-goal, i.e., no other team member has already replied to the question (line 13), it can decide to reply to Aj based on its response time, and adds the Inform(team-next-action) act to agenda in IS. It chooses one of its available goals from its group-goals of IS based on its own preference rules, and informs the team by constructing Inform(team-next-goal) dialogue act. When the agent receives the choice of the goal from one of the team members that matches with its potential candidate goals, it modifies its IS by adding mutual belief about group-goal and belief about joint-goal.

Now, let us consider the case when the every team member has the joint-goal, but no joint-intention towards group to achieve joint-goal. Each team member can choose any of the available plans to achieve that goal. In this situation, the team members cannot monitor the activities of other team members, and thus, causes problems in establishing team coordination among them. To establish the joint-intention towards the group to achieve collectively chosen joint-goal, team members need to ensure that each team member will follow the same plan to achieve the joint-goal. Algo. 3 describes how team members collectively select the common plan to achieve joint-goal.

Algorithm 3 CCP2: Collective decision for Plan choice

Require: group G, and shared goal  metavariable IS

At speaker side—:

1: if Joint-Goal(G,ϕ) ∧ ¬Joint-Intention(G,ϕ) then
2: if size(Plans(Ai,ϕ)) == 1 then
3:  IS ← addTopOfProactiveAgenda request(Check-Q(plan-choice), All)
4: else if size(Plans(Ai,ϕ)) > 1 then
5:  IS ← addTopOfAgenda Inform(Choice-Q(which-plan), All)
6: IS ← addTopOfAgenda Inform(Choose-Q(which-plan), All)

Similarly at receiver side—:

7: IS ← addExpected(plan-choice,–,ϕ)
8: else if receive(Inform(plan-choice, A1, Pϕ) ∧ Expected(plan-choice,–,ϕ))
9: IS ← Joint-commitment(G,ϕ) ∧ ¬Mutual-belief(G,ϕ)
10: IS ← Joint-Goal(G,ϕ)
11: extract IS ← Expected(ack,–)
12: IS ← addTopOfAgenda(Inform(team-next-action(ϕ), Ai))
13: IS ← Joint-Intention(G,ϕ) ∧ Joint-commitment(G,ϕ)
14: IS ← pushIntoTaskFocus(ϕ)
15: IS ← extract(Expected(ack,–))
16: IS ← Joint-Goal(G,ϕ)
17: IS ← Joint-Intention(G,ϕ) ∧ Joint-commitment(G,ϕ)

2014, © Copyright by authors, Published under agreement with IARIA - www.iaria.org
If the agent $A_i$ has only one plan to achieve the joint-goal (line 2), it constructs Check-$(Q$(action-plan$)) act addressing it to the group. Otherwise, if $A_i$ has more than one plan to achieve this goal (line 4), it constructs Choice-$Q$(which-plan) act and addresses it to the group. In both the cases, $A_i$ adds the communicative intention to the proactiveAgenda in IS.

When the agent receives a choice, or the confirmation of the choice of a plan, from one of the team members, it adds joint-intention to its task context. It confirms this by sending a positive acknowledgement, and constructs the belief about joint-commitment towards the group to achieve joint-goal. When the agent receives Choice-$Q$(which-plan) or Check-$Q$(action-plan), and has no mutual belief about group-intention, it constructs Inform$(plan$-choice) or Confirm dialogue act respectively, and adds corresponding intentions to agenda in semantic context of IS to inform about its plan selection. When it receives positive acknowledgement from one of the team members, it adds individual- and joint-commitment to achieve the joint-goal.

B. CCP-2

When the agent has performed all its planned actions of the shared activity, but the activity is not yet finished, the agent requests other team members to inform it when the activity will be finished. As each agent has the joint-commitment towards the group to achieve the joint-goal. That is, the team members remain committed towards the group until the goal is achieved or, the goal is unachievable. The agent can drop the goal if it believes that the goal has been achieved or no more possible. Thus, to maintain the cooperation with team members, agent can ask them to inform it if the belief about the persistent goal is modified. The protocol CCP2 is defined in Algo. 4.

Algorithm 4 CCP2

Require: Information state IS, Joint-commitment$(G, \varphi)$, $P_{\varphi} \leftarrow Plan(Prep$(G, \varphi), P_{\varphi} ))$, i.e., plan preferred by $G$ to achieve $\varphi$

---At speaker side---:

1: if Joint-commitment$(G, \varphi)$ ∧  
   $\exists a_x \in Plan(\varphi)$  |  $\forall a_y \in Plan(\varphi)$  |  $(a_x < a_y) \land Bel(DoneA_i, a_x)$

then

2: IS $\leftarrow$ addTopOfProactiveAgenda$(directive$-request$)$

3: (Inform(goal-achieved), All)

---Similarly, at receiver side: For all other agents $A_j \in G$ ---:

4: if Receive$(directive$-request$ A_j, Inform(goal$-achieved)) \land$

Joint-commitment$(G, \varphi)$

then

5: IS $\leftarrow$ addDesire$(Inform(goal$-achieved)),$ All)

In this protocol, the agent generates Directive-request$(Inform$-goal$-achieved)$ in its proactiveAgenda to ask other members to inform it when the activity will be finished. When the agent receives this dialogue act, it adds communicative goal Inform$(goal$-achieved$)$ to its agenda. The expression $A_i \prec A_j$ represents that the execution of $A_i$ is preceded by the execution of $A_j$. Other team members that receive this directive request, and contains the joint-commitment towards the group, modify their IS by adding a desire to inform about the achievement of the goal.

C. CCP-3

The agent that finished the last action of the shared activity informs other team members that the activity is terminated. The protocol CCP-3 has been described in Algo. 5.

Algorithm 5 CCP3

Require: Information state IS, Joint-commitment$(G, \varphi)$, $P_{\varphi} \leftarrow Plan(Pre$(G, \varphi), P_{\varphi} ))$, i.e., plan preferred by $G$ to achieve $\varphi$

---At speaker side---:

1: if Joint-commitment$(G, \varphi)$ ∧  
   $\exists a_x \in Plan(\varphi)$  |  $\forall a_x \in Plan(\varphi)$  |  $(a_x < a_y) \land bel(DoneA_i, a_x)$

then

2: IS $\leftarrow$ addTopOfProactiveAgenda$(Inform(activity$-finished$),$ All $)$

3: IS $\leftarrow$ addBel$(Group-Bel(G, Done(\varphi)))$

4: if Joint-commitment$(G, \varphi)$ ∧ Group-Bel$(G, Done(\varphi)) \land$

desire$(Inform(goal$-achieved))$

then

5: IS $\leftarrow$ addTopOfProactiveAgenda$(Inform(goal$-achieved$),$ All $)$

6: IS $\leftarrow$ addBel$(Group-Bel(G, always(\varphi)))$

7: IS $\leftarrow$ extract$(Joint-commitment(G, \varphi))$

8: IS $\leftarrow$ extract$(Joint-Goal(G, \varphi))$

9: IS $\leftarrow$ extract$(Mutual-Belief(G, \varphi))$

---At speaker side: For all other agents $A_j \in G$ ---:

10: if Receive$(Inform(activity$-finished$),$ A_j $) \land$ Joint-commitment$(G, \varphi)$ ∧  
   $\exists a_y \in Plan(\varphi)$  |  $\forall a_x \in Plan(\varphi)$  |  $(a_x < a_y) \land Bel(DoneA_i, a_x)$

then

11: IS $\leftarrow$ addBel$(Group-Bel(G, Done(\varphi)))$

12: if Receive$(Inform(goal$-achieved$),$ A_j $) \land$ Joint-commitment$(G, \varphi)$ ∧  
   $\exists a_y \in Plan(\varphi)$  |  $\forall a_x \in Plan(\varphi)$  |  $(a_x < a_y) \land Bel(DoneA_i, a_x)$

then

13: IS $\leftarrow$ addBel$(Group-Bel(G, always(\varphi)))$

14: IS $\leftarrow$ Po$TTaskFocus(\varphi)$

15: IS $\leftarrow$ extract$(Joint-commitment(G, \varphi))$

16: IS $\leftarrow$ extract$(Joint-Goal(G, \varphi))$

17: IS $\leftarrow$ extract$(Mutual-Belief(G, \varphi))$

The preconditions for CCP-3 are that the agent believes that it has performed the last action of the collaborative activity, and it has the joint-commitment to achieve group-goal. If these preconditions are satisfied (line 1), it constructs Inform$(activity$-finished$)$ dialogue act addressing it to the group, and adds this communicative intention to its proactiveAgenda. The predicate done$(P_{\varphi})$ represents that the plan $P_{\varphi}$ has been terminated.

When the agent receives the information that the last action of the activity has been finished (line 10), and it has the belief about joint-commitment in its task context, it constructs the group belief about the status of the plan.

When an agent has group belief that the activity is finished (line 4), and has a communicative goal Inform$(goal$-achieved$)$ to achieve (due to CCP-2), it constructs Inform$(goal$-achieved$)$ dialogue act to inform other team members that the goal has been achieved. It then adds the belief about the achievement of the goal, and removes the corresponding intention from the proactiveAgenda. The predicate always$(\varphi)$ represents that the state of the world $\varphi$ remains always true, i.e., the goal $\varphi$ has been achieved.

When the agent receives the information about goal achievement (line 12), it removes the corresponding intention from the task context, and drops the communicative goal...
Inform( goal-achieved) if it has. Furthermore, it then adds the belief about the achievement of the goal, and removes the corresponding intention from the task context.

In a mixed human-agent team, the reaction time of each team member is different, and changes dynamically, the agent waits for certain time (until the threshold of its reaction time is expired) and if no team member has already replied, the agent can create an intention to reply. Otherwise, the agent simply listens to the conversation and updates its beliefs. Thus, in order to establish mutual awareness and to coordinate with other team members, the agent participates in the conversation. Once agents have established the joint-commitment, they can coordinate with other team members to achieve the group-goal. These protocols are instantiated when the decision-making identifies collaborative situations that satisfy necessary conditions of one of the CCPs to be fulfilled (Algo. 1, lines 19-21). These situations add expectations of information from other team members, which need to be satisfied. In a human-agent team, the user's behaviour is uncertain, i.e., a user may not necessarily follow these protocols. As the agent updates their beliefs using perception information, which can make expectations to be true from the observation of actions of user perceived by the agent, or from the information provided by other team members. This mechanism makes these protocols robust enough to deal with uncertainty about user's behaviour. One of the advantages of these protocols is that the dialogues for the coordination need not to be scripted in the definition of action plans.

VII. NATURAL LANGUAGE PROCESSING

The natural language processing refers to the ability to understand the natural language input utterance, integrates its meaning, and also, the ability to generate natural language utterances. The natural language understanding in C²BDI agent includes the construction of semantic form corresponding to the input utterance (Sec. VII-A), and the interpretation of the semantic form to determine its meaning in the form of dialogue acts (Sec. VII-B). The dialogue act interpretation integrates the actual meaning of the utterance to it's IS (Sec. VII-C). The agent then selects generation rules and updates its IS in order to produce new communicative intentions (Sec. VII-D), which in turn result in generation of natural language utterances. Moreover, the C²BDI agent also exhibits the capability of proactive communication (Sec. VII-E). These processes modifies the IS depends upon the role the agent plays during the conversation. In C²BDI architecture, the template rule based approach is used for the natural language generation, which uses the semantic contents of the dialogue act, and the semantic information of the VE to generation natural language utterance [23].

The IS based context model is modified by the means of applying the updated rules (Fig. 3). The Update rule consists of a set of precondition and the set of effects. The Effect defines the possible updates on IS. All preconditions must be true to apply rules which lead to apply the updates on IS defined in effect part of rules. The Rule-base which contains update rules can be classified into integrationRule and selectionRules. The former is used to integrate the meaning of received utterance during dialogue act interpretation, whereas, the later is used to update the IS in order to generate natural language utterance. The selectionRules can be further classified into reactive-UpdateRules which can be applied during the generation of reactive conversation, and the proactiveUpdateRules, which can be applied to produce the proactive conversation in the current context of the dialogue and the shared task.

![Figure 3: Information State Update Rules](image)

In the following sections, we will describe different components of natural language processing in C²BDI agent architecture.

A. Semantic form generation

The NLU focuses on the processing of the input utterance to determine its meaning. The goal is to obtain the computational form of the utterance, which can also involve the use of pragmatic aspects, and the notion of the temporality. To go further in determining the meaning, additional information is also needed to be recognised. These information or feature structure include concept types, their properties, and their relationship with other concepts in the VE, or the information about the current task. The semanticFormGenerator can obtain this information from the IS and semantic knowledge to generate the semantic form of utterance (Fig. 4).

![Figure 4: Utterance Interpretation](image)

One of the important steps is the identification of the thematic roles of different components of the utterance. Identification of these roles includes information about the sender, the addressee, and the mapping of components of the utterance to the concepts in the VE, i.e., the mapping to corresponding actions, goal, concepts, entities, their features etc. In C²BDI
agent, the approach is based on the template based rules (NLU Rules), which are processed by nabuAgent (NabuTalk agent)\(^1\). These template based rules use the cue words, and describe the syntactic structure of the utterances. The template rule is composed of lexical expressions and parametrized functional variables, organised in appropriate order to represent the syntactical structure of the utterance. Each Lexical expression represents the regular expression to describe the cue worlds, whereas the parameterised functional variables map the components of utterances to the corresponding concepts. For example, the following simplified template rule represents the syntactical structure for the utterance of the type query.

\[
\begin{align*}
\text{nlu-resource [id:should] } & \#(?:\text{should} (?:\text{ou}?) ?!?d )#) \\
\text{nlu-resource [id:I] } & [1] \\
\text{nlu-rule:} & [\{\text{should} [1] [\text{concept($\text{action}$)]}\}]
\end{align*}
\]

In this template rule, the nlu-resource represents the lexical expressions to represents the lexical unit. The nlu-rule is composed of two components input and output. The input represent the template rule for the utterance, whereas the output represents the expression for the semantic form corresponding to the utterance to be generated. The [concept(Saction)] in input represents the mapping of some string to the some action, whereas, the @concept-name(S action) in output corresponds to the name of the action obtained through [concept(Saction)].

Now, let us consider the following sequence of dialogues: input utterance :

\[A_1:: \text{ALEXANDRE: Should I place the tablet?}\]

\[S_1:: \text{SEBASTIEN: Yes, you should place the tablet.}\]

\[A_2:: \text{ALEXANDRE: Why should I do this action?}\]

Alexandre utters A1, addressing it to Sebastien. The input utterance is processed by Sebastien. The structure of the utterance A1 corresponds the template input rule, the parametrized functional variable @concept-name($\text{action}$) is then evaluated. That is, the string place the tablet is mapped to one of the action or goal using the semantic knowledge.

1) Reference Resolution: Another important step towards determining the meaning of the utterance is the reference resolution, that is when the linguistic expression refers to the previous reference, e.g., the use of pronounces, referencing to an object or an action. The result of the reference resolution is that the variables that remain free are now affected to referents. The reference resolution requires the current context of the task and the dialogue, and the use of dialogue history. In C\(^2\)BDI agent, the reference of the pronoun is resolved by using information such as whether the utterance is referencing to the speaker, the addressee or to the third person. The cue-words such as I, you, and he / she / it are used for this purpose. For example, utterance A1 contain the cue word I, thus, the receiver agent processing this utterance can identify that the speaker references herself. The agent can map the pronoun I to the identity of the speaker. The agent can use the contextual information stored in the perceptual context of IS (IV) to resolve references. The perceptual context holds information about the third person in focus, object in focus, and the action in focus during the current context of the conversation. The object resolution is done using the properties mentioned or determined by the referring expression. The action resolution refers to the action carried out by the verb or verb phrase. Solving this reference also requires the information about the current context of the ongoing activity. For the utterance A1, the generated semantic form is shown below.

\[
\begin{align*}
\text{should} & \quad \text{check-q} \quad \text{agent} \quad \text{action future} \\
\text{place} & \quad \text{tablet} \\
\end{align*}
\]

After the processing of the utterance A1, Sebastien updates its IS and the actionInFocus of perceptual context now contains the action place-the-tablet. After uttering S1, Sebastien processes the next received utterance A2 that matches with the template rule given below:

\[
\begin{align*}
\text{nlu-rule:} & [\{\text{why} [\text{should} [\text{will}]] [1] [\text{do} ] \text{alt} [\text{this} [\text{action}]] [\text{this} ]]\}]
\end{align*}
\]

Sebastien needs to resolve the action reference as the utterance A2 contains the cue word this action. Since the actionInFocus in IS of Sebastien contains the name of the action referenced in the previous utterance, it can thus resolve the action reference by referencing it to the action stored in actionInFocus, which is the place-the-tablet for the utterance A2.

B. Utterance interpretation

The utteranceInterpreter uses the semantic form of the utterance generated by semanticFormGenerator, and the current IS to determine the appropriate meaning of the utterance. The result of this step is the dialogue act corresponding to the utterance (Fig. 4). The agent uses the template based rules to determine the dialogue acts with reference to the semantic form of the dialogue. The dialogue act refers to the communicative function that can be understood in the dialogue context which also takes into account the previous dialogue utterances and the current context of the dialogue and ongoing activity. For example, consider a utterance:

\[V_1:: \text{VIRGINIE: Yes.}\]

According to the speech act theory, the utterance V1 can be considered as an assertion act [15]. However, it can be precisely modelled by dialogue act as an acknowledgement or an answer act when the interpretation is associated with the previous dialogue utterance. For example, if the previous dialogue utterance is I take the left tablet, the utterance V1 is the acknowledge in this case. However, if the previous utterance is should we assemble the shelves, the utterance V1 is the answer to the utterance of the type check-question. The identification of the dialogue act requires:

- Utterance and its semantic form
- Types of the previous dialogue acts

---

\(^1\)The NabuTalk is a commercial rule based engine that includes appropriate mechanisms to handle different NLN/NLG concepts such as utterance templates, pattern-matching, utterance understanding and generation rules.
• Current context of the dialogue, and the context of the task.

The agent identifies the communicative function of the utterance, the identity of the speaker and addressee, and construct the logical form which constitute the relevant contents of the dialogue act. If the dialogue act is successfully constructed, then the utterance and the dialogue act are added to the addresseeDialogueAct component of the linguistic context in IS. For example, let us consider the utterance $A_1$ uttered by Alexandre. Sebastien processes the utterance, and identifies the associated dialogue act as an information seeking check-question-agent-next-action act, as Alexandre (the speaker) seeks the validity of the proposition that its next action is place-the-tablet. That is, the communicative function of the dialogue act is check-question-agent-next-action, whereas, the contents of the dialogue act includes the information about the dimension (task), speaker (Alexandre), addressee (self), and the logical form (check-question-agent-next-action Alexandre "place the tablet").

### C. Dialogue Act Interpretation

The result of the dialogue interpretation process is the integration of the meaning of the dialogue utterance to the context model. The formal model of dialogue act interpretation is described in Fig. 5. The dialogue act interpreter selects the update rules from the IntegrationRules that can be applied to the IS based on the current context of the dialogue. The dialogue act interpretation uses the current state of IS, the dialogue act, and the semantic knowledge for the evaluation of the preconditions of these rules. The successful interpretation of the dialogue act results in updates of different parts of the IS.

![Diagram of dialogue act interpretation](image)

#### Figure 5: dialogueAct Interpretation

1) IS update when agent processes received utterance:
Successful interpretation of the incoming utterance results in the processing of the DAs. Processing of the task-oriented dialogue acts provokes the changes in the semantic context of IS. This processing results in creating the belief about the speaker’s belief, and updating the expectation of information in semantic context. The team members communicates with each others in order to establish the mutual awareness between team members. Establishing the mutual belief provokes the changes in the cognitive context. Processing of the social obligation acts will create the social pressure in social context. A successful interpretation of the utterance also results in updating the linguistic context by adding new dialogue acts to the addressee’s dialogue acts. Moreover, if the utterance references to an object, addressee, sender, third person, or to the action, the perceptual context is updated. Moreover, the team members can cultivate efficient team coordination through dialogues to achieve the team goal. During this process, team members construct beliefs about different collective attitudes such as group goal, joint goal, joint commitment etc, and modifies the cooperative-Info component of the task context.

To endow C^2BDI agents with multiparty conversation, the updates mechanism takes into account the effects of communication on the shared mental model of team members. Consider that an agent $A_i$ has received the utterance $U_i$ from the speaker $S_j$, and $(A_i, S_j) \in G$. The utterance $U_i$ contains the proposition $P$. The UtteranceInterpretation has identified the dialogue act $D_i$ corresponding to the utterance $U_i$.

**a) Processing of Information-Providing-Function:** The algorithm for the context update during the processing of dialogue acts of the type Information-Providing-Function is given below:

1) If the semantic form generation or Utterance interpretation of utterance $U_i$ is failed Then
   - No updates in IS.
   - Exit.
2) If the communicative function of $DA_i$ is Information-Providing-Function Then
   - If utterance $U_i$ is addressed to the agent $A_i$ itself, Then
     Construct mutual-belief about the speaker’s belief on $P$ in Cognitive context.
     Else
     - If utterance $U_i$ is addressed to the group G, Then
       Construct group-belief about the speaker’s belief on $P$ in CooperativeInfo of task context.
     Else
     - The receiver agent is an overhearer, thus,
       Construct belief about the speaker’s belief on $P$ in semantic context.
   - If utterance $U_i$ is addressed to $A_i$ or to the group G Then,
     o If the agent has a negative belief about $P$, i.e., if it believes $\neg P$, Then
       Drop $\neg P$ from semantic context.
     o If agent has the weak belief about $P$, Then
       Drop the weak belief about $P$ from semantic context
     o Adopt the belief $P$, i.e. create the belief about $P$ in semantic context.
   - If the agent $A_i$ has an expectation about $P$ from speaker, Then
     If the expectation about $P$ is satisfied, Then
     o Drop expectation about $P$ from semantic context.
     o Generate acknowledgement.
3) Copy $DA_i$ to the dialogueActHistory in dialogue context.
4) Remove $DA_i$ from addresseeDialogueActs in dialogue context.

**b) Processing of Information-Seeking-Function:** The algorithm for the context update during the processing of dialogue acts of the type Information-Seeking-Function is given below:

1) If the semantic form generation or Utterance interpretation of utterance $U_i$ is failed Then
generate utterance in response to the incoming utterance as
utterance, the intention is removed from the
After the successful processing of the intention to generate the
and the generated utterance are also stored in dialogue history.
respectively. After the generation of utterance, the dialogue act
the utterance is addressed to an addressee or to a group
cognitive context or task context, depending upon whether
the communicative function of \( DA_i \) is \textit{Information-Seeking-Function} \textbf{Then}
\begin{enumerate}
\item \textbf{If} \textit{utterance} \( U_i \) is addressed to the agent or to the group
\textbf{Then}
\begin{enumerate}
\item Construct \textit{mutual-belief} in cognitive context about
the speaker’s intention that the addressee provides
information about \( P \).
\item Create \textit{Pot.Int.To} to reply about \( P \) to speaker.
\item Add this \textit{Pot.Int.To} to the \textit{agenda} in semantic context
\item Keep \( DA_i \) in addresseeDialogueActs
\end{enumerate}
\end{enumerate}
\item \textbf{Else}
\begin{enumerate}
\item The receiver agent \( A_i \) is an overhearer, therefore,
Construct \textit{belief} in semantic context about the speaker’s
intention that the addressee provides information about \( P \).
\end{enumerate}
\end{enumerate}
3) Copy \( DA_i \) to the dialogueActHistory in dialogue context

\section*{D. Select and Update for Reactive Reply}

At this stage, we consider that the agent has successfully interpreted the dialogue act associated with input utterance, and have updated the IS. In order to decide how to reply, the agent selects the update rules from reactiveUpdateRules that can be applied. The selection of the rules depends upon the intention in \textit{agenda}, previous speaker’s dialogue act, and the current IS.

The application of selected rules and the generation of the utterance in response to the input utterance also results in updating different components of IS. Generation of utterance with information transfer function results in updating the cognitive context or task context, depending upon whether the utterance is addressed to an addressee or to a group respectively. After the generation of utterance, the dialogue act and the generated utterance are also stored in dialogue history. After the successful processing of the intention to generate the utterance, the intention is removed from the \textit{agenda}.

We now describe the context update algorithm when agent generate utterance in response to the incoming utterance as follows:

1) \textbf{If} Top of \textit{agenda} is not empty
\begin{enumerate}
\item \textbf{If} if top of agenda contains \textit{Pot.Int.To}
\begin{enumerate}
\item \textbf{If} the evaluation of conditions for \textit{Pot.Int.To} is
succeed, then
upgrade \textit{Pot.Int.To} to \textit{Int.To}
\item \textbf{Else}
\begin{enumerate}
\item pop \textit{Pot.Int.To} from top of agenda
\item remove \( DA_i \) from addresseeDialogueActs in
dialogue context
\item Exit.
\end{enumerate}
\end{enumerate}
\end{enumerate}
2) Select update rules for which preconditions are true in current dialogue context and intention.
3) \textbf{If} selected rules > 0, then
\begin{enumerate}
\item \textbf{ForEach} updateRule in selected rules
apply update effects to IS
\item generate and add next dialogue moves to nextMoves in
linguistic context
\item Pop \textit{agenda}
\item \textbf{ForEach} dialogueMove in nextMoves
\begin{enumerate}
\item process dialogueMove to generate NL utterances
\item If dialogueMove corresponds to the information-
transfer function, then
\begin{enumerate}
\item If generated utterance is addressed to a particular addressee, then
Construct the mutual belief with the addressee
\item that the provided information is true.
\end{enumerate}
\end{enumerate}
\end{enumerate}
4) \textbf{Else}
\begin{enumerate}
\item Construct the group-belief with the group the
provided information is true.
\item Clear nextMoves;
\item remove \( DA_i \) from addresseDialogueActs in dialogue context
\item add generated dialogue to the agentDialogueActHistory
\end{enumerate}

The agent evaluates the of conditions for \textit{Pot.Int.To} before agent adopts it as \textit{Int.To}. To do so, the agent verifies if this intention corresponding to the previous input utterance can be processed in current context of the dialogue and task. In the case when the previous utterance was addressed to the group, the agent verifies if any other agent has already replied. If so, the agent drops the intention, as the information need of the speaker has already been satisfied.

\section*{E. Proactive conversational behaviour}

When the agent identifies the need of the collaboration with other team members or has identified the information need of other team members or of self, the agent can create an intention to communicate with other agents individually, or collectively, depending on the current context of the task. The agent models the proactive conversation behaviour in two steps, which are the construction of dialogue acts, and generation of next dialogue moves.

1) \textit{conversation operation}: The agent executes conversation operation, which can be abstract operations such as \textit{askOperation}, \textit{informOperation}, \textit{directiveRequest}, \textit{greetOperation} etc. An extract of the conceptual model of conversation operation is shown in Fig. 6. The conversation operation can be executed if the preconditions are satisfied. The execution of the conversation operation, constructs the appropriate dialogue act, and updates the IS of the agent by first, adding the generated dialogue act to the \textit{agentDialogueActs} of linguistic context, and second, it adds the associated intention \textit{Pot.Int.To} to the \textit{proactiveAgenda} in semantic context.

![Figure 6: Conversation operation](image-url)

2) \textit{IS Update for the Proactive conversational Intention}:
If the IS contains an intention in \textit{proactiveAgenda}, the agent processes it. The algorithm for the context update for the proactive utterance generation is described as follows.

1) \textbf{If} top of \textit{proactiveAgenda} is not empty, Then \textbf{If} top of proactiveAgenda contains \textit{Pot.Int.To} , Then
2) Select update rules for which preconditions are satisfied in current dialogue context and intention

3) If selected (rules > 0), Then
   • ForEach updateRule in selected rules
     ◦ Apply update effects to IS
   • Add generated next dialogue moves to nextMoves in linguistic context
   • If communicative function of $aDA_i$ is Information-Seeking function, Then
     ◦ Add expectation about $P$ from addressee
   • Pop proactiveAgenda
   • ForAll dialogueMove in nextMoves
     ◦ Process NextMove (dialogueMove, IS, $aDA_i$) to generate natural language utterances
   • Clear nextMoves();

4) Remove $aDA_i$ from agentDialogueActs in dialogue context

The proactive conversation behaviour of the agent is driven by the information need, or by the cooperative situations where agent need to cooperate with other team members in order to achieve shared team goal. If the top of the agenda contains $Pot.Int.To$, the agent evaluates it. In this case of proactive conversation, the agent verifies if the information need of other agent or of its own, is already satisfied. If so, it drops the intention. Similarly, the agent also drops the intention to communicate if it identifies that the need of cooperation has been satisfied. Otherwise, the agent upgrades the $Pot.Int.To$ to $Int.To$ in order to select update rules from the selectionRules to update IS and to generate next moves. The agent selects rules from proactiveUpdateRules, which can be applied to IS, depending upon the current communicative intention, current task context, and the generated dialogue act.

The successful generation of proactive utterance addressed to an addressee (group), creates the mutual belief (group-belief) between the speaker and the addressee (group) about the speaker’s information need or of addressee, depends upon the current context of the task. If the communicative function of the dialogue act is information transfer function, the speaker creates the mutual belief (group-belief) with the addressee (group) that the proposes information is true. However, if the communicative function of the dialogue act is information seeking function, the speaker creates an expectation of the information from the addressee (group).

VIII. IMPLEMENTATION

The technical architecture of $C^2$BDI agent is mainly composed of dialogue manager and Unity3D interface, which has been presented in [24]. Each $C^2$BDI agent is associated with a virtual human and controls its behaviors. User interacts with VE through her avatar. $C^2$BDI agent sends service messages to the associated virtual human to perform actions chosen by the decision-making module or by the dialogue manager (turn-taking behavior). The rendering system realises the requested actions and sends action events (begin, end) towards corresponding $C^2$BDI agent. The conversation manager deals with automatic-speech-recognition (ASR) and text to speech synthesis (TTS). The message manager handles the dispatching of perception information and service messages.

Let us now consider a motivational scenario where three agents (may include both virtual or real), named as Virginie, Sebastien, and Alexandre need to assemble a furniture. To do so, they need to choose tablets from the table (Fig. 7) and place them on shelves (Fig. 8). Following sequence of dialogues describe a typical interaction between them where a user plays the role of Alexandre.

![Figure 7: Furniture Assembly Scenario: before tablet selection](image-url)
Figure 8: Furniture Assembly Scenario right: before choosing tablet position

Figure 9: Partial view of Furniture Assembly plan shared between team members.

The challenging scenario includes some important characteristics such as collaborative situations to establish common grounding (S1,U1,S2,S3,U2), handling resource conflicts (V1,U3,V2,), dynamic environment (agents manipulate objects, e.g., move tablet), interleaving between communication and actions (agents utter and perform action S4,U3,V3,U4), mixed initiative dialogues (V1,U3,V2 or U4,V3,U5), and both reactive (V3) and proactive (S1,V1) communications.

At the beginning, both user and virtual agents have a goal "place-tablet-on-the-shelf". As this goal is shared among team members, it becomes the group-goal (Fig. 9). A subset of knowledge of agents is shown in Table I.

Since, Sebastien has a group-goal as "place-tablet-on-the-shelf" in its IS, but has no mutual belief about that goal, the decision making process identifies this collaborative situation that fulfills conditions of CCP-1 (Algo. 1, line 19). The CCP-1 generates Set-Q(team-next-goal) dialogue act (Algo. 2, line 3), and adds the corresponding intention to the agentDialogueActs. Processing of this intention (Sec. VII-E) generates natural language utterance S1.

Sebastien interprets utterance U1 as Inform(team-next-goal "place-tablet-on-the-shelf") dialogue act. As Sebastien has the same group-goal, it creates mutual-belief about group-goal, and generates positive acknowledgement S2 for Alexandre. The snapshot of current state of Sebastien’s IS is given in Table II. Virginie passively listens to the conversation and updates its IS following CCP-1. Now, to ensure that the each team member will follow the same action plan, Sebastien constructs Check-Q(plan-choice) dialogue act considering that team members have only one plan "place-tablet-plan" to achieve the current group-goal, and generates S3.

When both, Sebastien and Virginie receive response U2 from Alexandre, they construct the joint-intention as well as joint-commitment towards the group-goal and update their IS. The decision making process, now, deliberate the plan and computes the new intention as grasp-tablet (Table III). Sebastien chooses the large-tablet as the resource is explicitly defined with the action. Virginie needs to perform explicit resource acquisition, as only the resource type is defined for its action which is dependent on Alexandre’s choice (Fig. 9). As two instances of "Tablet-narrow" are available (Fig. 7), and if Virginie has no belief about Alexandre’s choice, it constructs Set-Q(what-resource-choice) to ask Alexandre to choose one of the tablets (V1). When Alexandre specifies its choice (U3), Virginie chooses the other one (V2). After executing last action "place-tablet" by Sebastien from his plan, and as the shared

| TABLE I: Snapshot of IS for Virginie and Sebastien before initialisation of CCP-1 |
|------------------------------|---------------------------------|
| Information State | \( H_0 \) (Virginie) | \( H_0 \) (Sebastien) |
| Task-Context | cooperative-info("group-goal ("place-tablet-on-the-shelf")") | cooperative-info("group-goal ("place-tablet-on-the-shelf")") |

| TABLE II: Snapshot of IS for agent Sebastien after establishing joint-goal |
|------------------------------|---------------------------------|
| Information State | \( H_3 \) (Sebastien) |
| Cognitive-Context | mutual-belief("group-intention ("place-tablet-on-the-shelf")", "group-goal ("place-tablet-on-the-shelf")") |
| Task-Context | cooperative-info("group-goal ("place-tablet-on-the-shelf")", "joint-goal ("place-tablet-on-the-shelf")") |

| TABLE III: Snapshot of IS of Virginie after establishing joint-commitment |
|------------------------------|---------------------------------|
| Information State | \( H_1 \) (Virginie) |
| Cognitive-Context | mutual-belief("group-intention ("place-tablet-on-the-shelf")", "group-goal ("place-tablet-on-the-shelf")") |
| Task-Context | cooperative-info("group-goal ("place-tablet-on-the-shelf")", "joint-goal ("place-tablet-on-the-shelf")", "joint-commitment ("place-tablet-on-the-shelf")") |

\( \text{Intention}("grasp-tablet") \)
activity is not yet finished, it utters S5 following CCP-2. When Alexandre asks Virginie about its choice of position (U4), Virginie interprets this utterance as Set-Q(what-resource-choice) and informs its choice (V3). Once Alexandre places the tablet (U5) which is the last action of the shared plan, Virginie informs all the team members that the goal is achieved (V4) following CCP-3.

A. Evaluation

We wanted to see the contribution of conversation in a teamwork from the point of view of the user. The main aim is to see (a) the effects of conversation to establish effective team coordination, and (b) characteristics of verbal interaction with team members. We conducted the experiment in two phases. The first phase had 9 participants (group-1). Each participant was asked to perform the assembly of furniture with two virtual agents (Virginie and Sebastien) having CCPs disabled. In the second phase, 12 participants (group-2) were asked to do the same, but the virtual agents (Virginie and Sebastien) had the CCPs enabled. The participants were 3rd year engineering students between 21-23 years old. After the experiment, each participant had to respond to a questionnaire by assigning the ratings between 0 to 5 (0 means completely disagree, 5 means completely agree).

Figure 10 shows the experimental result. We found that 66% of the participants of the group-1 were agreed that the conversation with team members facilitated (mean value 2.9) them to achieve the goal. Whereas, 83% participants in the group-2 found that the conversation facilitated them (mean value 3.2) to achieve group goal. 55% participants in group-1 found that the agents took into account their participation (mean value 3), however, 75% in the group-2 found the same (mean value 3.6). The reason is that the C³BDI agent takes into account the uncertainty of user behaviour, and CCPs are flexible enough to deal with this situation.

The conversation with team members was more constructive when the agents had CCPs enabled features (group-1 mean value 3, group-2 mean value 3.5). That is, virtual team members motivated them to coordinate with each other to achieve the goal. 55% participants in the group-1 found the communicative interaction informative (mean 2.9), whereas, 83% participants in the group-2 found it informative (mean value 3.65) as the virtual agents provided them information proactively. The reason is that the virtual agents engaged the participants in collective decision making, such as shared goal selection, plan selection, and provided the necessary information in proactive manner to perform the shared task.

Furthermore, 55% in the group-1 considered that their interaction with virtual agents was not spontaneous (mean 2.22), whereas, 75% in the group-2 found it more spontaneous (mean 3.34), that is 22.4% of gain value for spontaneous interaction. They found that virtual team members initiated the conversation driven by the information needs of participants. Moreover, both the groups found the conversation with the agents coherent (mean value 3.2, and 3.3 in group-1 and group-2 respectively) to the task. However, participants in the group-2 indicated that sometimes the conversational interaction was not amusing (mean value 2.28). The reason for this less amusement was that the participants experienced the similar conversational interaction at the time of the initialisation of each new group-goal. Nevertheless, participants in group-2 admitted that the agents’ cooperative conversational behaviour helps them to effectively achieve the shared team goal.

B. Integration with Virtual Agent

The C³BDI architecture has been integrated with the interaction model for virtual and real human [25] on the GVT platform [26] for learning of a procedure for the industrial maintenance [27]. This scenario describes a maintenance procedure in a plastics manufacturing workshop. The scenario consists in the replacement of a mould in a plastic injection moulding machine (Fig. 11). This specific intervention requires a precise coordination of tasks between two workers: the setter and the machine operator. The use of autonomous agents allows the learner to execute the learning procedure. The user interacts with VE by controlling his avatar thanks to a tracking system of the body and hands (Fig. 12).
IX. DISCUSSION

The proposed work is done based on the theoretical framework of joint intention, shared plan, and collaborative problem solving approach. These approaches aimed to specify the mental states (belief, goal, intention) during the collaboration, whereas our approach focused on the practical use of natural language dialogues for cooperation in human-agent teamwork. Moreover, these models do not specify how their model looks like. In contrast, we described an extended Information State based context model. Our belief that the team members require the belief about other members in order to establish collective intention towards the group to achieve shared team goal is close to the theoretical framework of Dignum and Dunin [28], and Dunin and Verbrugge [29] for the teamwork in multi-agent systems. In their approach, an initiator agent identifies the potential for collaboration of each team members and tries to form a team by asking confirmations from other team members, and thus follows the master state mechanism. However, in our approach, each team member participates in collective decisions (such as the choice of a group-goal, the choice of the shared plan to achieve that goal). Moreover, team members also provide opportunities and motivations for other team members (including the user) to participate in the natural language conversation in order to establish efficient coordination among them.

The context model of C\textsuperscript{2}BDI agent is inspired by the context models proposed in Traum and Larsson [17], Keizer and Morante [30], and Bunt [19]. However, it has significant differences with their context model. The context models in [30] and [19] include the system belief and user’s belief in semantic context and in cognitive context respectively. However, in C\textsuperscript{2}BDI agent, the semantic context contains the beliefs about the agent’s own beliefs, and the beliefs about other team members. Moreover, the task-context in C\textsuperscript{2}BDI agent contains collective attitudes in the cooperative information (cooperative-info), which includes information necessary to establish and maintain coordination with other team members. Furthermore, the context models presented in [17] and [30] only accommodate an agenda that holds the communicative intentions of the agent. However, in the context model of C\textsuperscript{2}BDI agent, the semantic-context contains agenda and proactiveAgenda to store the intentions generated due to reactive and proactive conversation behaviour respectively. Moreover, the C\textsuperscript{2}BDI agent also manages the intentions to perform actions in task-focus in the task-context explicitly. Thus, the IS not only contains the current context of the dialogue but also the ongoing task of the agent.

Most of the dialogue system support two party conversation, however, the conversational behaviour of C\textsuperscript{2}BDI agent deals with multiparty conversation as the agent can play different roles (i.e., speaker, addressee, or overhearer) during the conversation. The information state is mainly used in these approaches to handle the conversation, and can be updated during dialogue processing. In contrast, in C\textsuperscript{2}BDI architecture, the information state is updated during the dialogue processing, but also during the deliberation of the task. Comparing with the context model for Max agent proposed by Kopp and Pfeiffer-Lessmann [20], in which the cooperation is considered as an implicit characteristic of agents, C\textsuperscript{2}BDI agents exhibit both reactive and proactive conversational behaviours, and explicitly handle cooperative situations through natural language communication between team members taking into account the user in the loop.

The proposed behavioural architecture can be improved in many ways. For the simplicity, we considered that an utterance contains only one communicative function. However, dialogue utterances often have multiple communicative functions, such as answering a question but also providing feedback on the understanding of the question, and also taking the turn [19]. Like Bunt [31], we are convinced that the taking into account both of these features require to define more precise update semantics for dialogue acts. Furthermore, the C\textsuperscript{2}BDI agent architecture does not take into account different modalities of interaction, such as facial expressions, emotions, gesture, gaze. However, these are linked in language production and perception, with their interaction contributing to felicitous communication [32]. It will be interesting to integrate these modalities in order to improve believability, usability, and coverage of interaction in a mixed human-agent teamwork.

X. CONCLUSION

The proposed behavioural architecture C\textsuperscript{2}BDI endows the agents in the collaborative VE with the ability to coordinate their activities using natural language communication. This capability allows users and agents to share their knowledge with their team members. The architecture ensures the knowledge sharing between team members by considering the deliberative and the conversation behaviours, not in isolation, but as tightly coupled components, which is a necessary condition for common grounding and mutual awareness to occur. The collaborative conversational protocols we proposed enable agents to exhibit human-like proactive conversational behaviour that helps users to participate in the collaborative activity. We proposed the information state based approach for natural language processing, in which the semantic information about VE and the shared plans is used as knowledge source. Moreover, we described the context update mechanisms to integrate the effects of both reactive and proactive conversation, bases on the role played by the team members during conversation. Furthermore, user experience also confirms the advantages of collaborative conversational behaviour of agents for the efficient team coordination in human-agent teamwork. While the implemented scenario already shows the benefits of
the solution, the behaviour of the agents could be enriched both in terms of collaborative team management and in terms of natural language dialogue modelling. Particularly, it would be interesting to endow agents with problem solving capabilities to select their communicative intentions, or to engage themselves into information seeking behaviours and negotiation rounds, as observed in human teamwork [33].

ACKNOWLEDGMENT

This work was partly supported by the ANR (Corvette project ANR-10-CORD-012).

REFERENCES


