Computer Methods and Programs in Biomedicine 242 (2023) 107817

Contents lists available at ScienceDirect

Computer Methods and Programs in Biomedicine

journal homepage: www.elsevier.com/locate/cmpb

Forming *We-intentions* under breakdown situations in human-robot interactions

Esteban Guerrero [a,b,*], Maitreyee Tewari [d,e,*], Panu Kalmi [a,c], Helena Lindgren [d,e]

[a] University of Vaasa, Finland
[b] School of Technology and Innovations, Information Systems Science, Finland
[c] School of Accounting and Finance, Economics, Finland
[d] Umeå University, Sweden
[e] Department of Computing Science, Sweden

ARTICLE INFO

Keywords:
We-intentions
Breakdown situations
Conflict of intentions
Repairing conflicts
Human-robot interaction
Answer set programming
Logic programming
Shared intentions
Social robots
Healthcare scenarios

ABSTRACT

Background and Objective: When agents (*e.g.* a person and a social robot) perform a joint activity to achieve a joint goal, they require sharing a relevant group intention, which has been defined as a We-intention. In forming We-intentions, breakdown situations due to conflicts between internal and "external" intentions are unavoidable, particularly in healthcare scenarios. To study such We-intention formation and "reparation" of conflicts, this paper has a two-fold objective: introduce a general computational mechanism allowing We-intention formation and reparation in interactions between a social robot and a person; and exemplify how the formal framework can be applied to facilitate interaction between a person and a social robot for healthcare scenarios.
Method: The formal computational framework for managing We-intentions was defined in terms of *Answer set programming* and a *Belief-Desire-Intention* control loop. We exemplify the formal framework based on earlier theory-based user studies consisting of human-robot dialogue scenarios conducted in a *Wizard of Oz* setup, video-recorded and evaluated with 20 participants. Data was collected through semi-structured interviews, which were analyzed qualitatively using thematic analysis. N = 20 participants (women n = 12, men = 8, age range 23-72) were part of the study. Two age groups were established for the analysis: younger participants (ages 23-40) and older participants (ages 41-72).
Results: We proved four theoretical propositions, which are well-desired characteristics of any rational social robot. In our study, most participants suggested that people were the cause of breakdown situations. Over half of the young participants perceived the social robot's avoidant behavior in the scenarios.
Conclusions: This work covered in depth the challenge of aligning the intentions of two agents (for example, in a person-robot interaction) when they try to achieve a joint goal. Our framework provides a novel formalization of the We-intentions theory from social science. The framework is supported by formal properties proving that our computational mechanism generates consistent potential plans. At the same time, the agent can handle incomplete and inconsistent intentions shared by another agent (for example, a person). Finally, our qualitative results suggested that this approach could provide an acceptable level of action/intention agreement generation and reparation from a person-centric perspective.

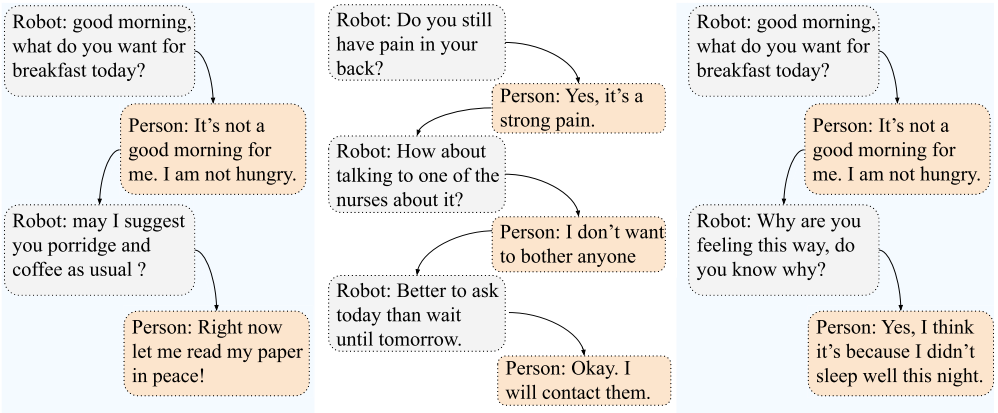
1. Introduction

A patient and a physician share the plan to form and agree upon a treatment jointly and to realize this plan. In this case, both have a *joint intention*, or so-called *We-intention* [125] for establishing the treatment plan. A We-intention is not reducible to mere personal intention or *I-intention* [128]. It is not enough for a We-intention to plan the treatment together that each patient and physician intends to plan. Such coincident intention does not even ensure that each knows of the other's intention or is appropriately committed to the joint activity itself [17].

The patient's and physician's intentions may be uncertain, leading to disagreements or conflicts that we call here *breakdown situations*. These situations are characterized by the misinterpretation or misunderstand-

* Corresponding authors.
E-mail addresses: esteban.guerrero@uwasa.fi (E. Guerrero), maittewa@cs.umu.se (M. Tewari), panu.kalmi@uwasa.fi (P. Kalmi), helena@cs.umu.se (H. Lindgren).

https://doi.org/10.1016/j.cmpb.2023.107817
Received 28 November 2022; Received in revised form 5 May 2023; Accepted 14 September 2023
Available online 20 September 2023
0169-2607/© 2023 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).



a) Avoidant behavior *b) Partial agreeing behavior* *c) Agreeing behavior*

Fig. 1. Example dialogue flows depicting avoidant, partially agreeing, and agreeing behavior.

ing (among other factors) of the We-intentions and their misalignment with I-intentions of the involved agents [17,128].

The care implications due to the pandemic worsened the already existing shortage of human-care services, moving in the direction of finding alternative solutions [47]. One such viable alternative was, for example, to use autonomous [52], telepresence- and teleoperated-robots [118,133].

To enable interaction between robotic-care providers as a physician with patients requires certain social skills [51]. Specific to the purposes of this work are the abilities of a robotic physician, enabling it to learn and differentiate models of the patient and to communicate with high-level dialogue. Researchers are already exploring interaction scenarios where a robotic physician interacts with a patient in clinical practices [118,133]. Due to inherent uncertainty in understanding mental states, breakdown situations during interactions are unavoidable, and their management in patient-physician interaction settings is a complex challenge [100].

Most people develop skills and learn to manage breakdown situations employing different strategies (*e.g.* negotiation, deliberation, even fighting [129]). However, in a patient and robotic physician joint activity, the robot needs human-like capabilities to form a joint intention with the patient, share a mutual understanding about the activity, and manage breakdown situations in collaboration with the patient [122,121].

Recent research has highlighted the importance of detecting breakdown situations and proposed taxonomies in human-robot interactions (HRI) [123,64]. Capabilities to recognize and manage breakdown situations have been considered essential and make the interaction natural [93]. Researchers have found human-like conversational styles improve acceptance of assistive agents among people with dementia and mild cognitive impairment [130]. Therefore, managing Breakdown situations becomes salient when introducing robotic physicians capable of interacting with patients.

For this work, we use the definition of a robotic-physician corresponding to the social robot as described in [121], where a social robot is defined as proactively engaging to fulfill internal goals of people [32], displaying cognitive capabilities similar to people [19,20], can distinguish other entities, and for which social interaction plays a key role [51]. Furthermore, the social robot strives to maintain a joint intention, hence, is characterized as a We-intentional agent (in the rest of the article, we interchangeably refer to robotic-physician as a social robot or simply an agent). Furthermore, the interaction by We-intentional agents (including people) is called joint activity in the rest of the text.

This paper focuses on breakdown situations arising from *uncertainty* around the 'intention' an agent aims to co-create during joint activities with a person. Such a co-creation is directed towards achieving a *group goal* by agents (including people) capable of having a 'We-intention,' which is an aim-intention that all involved agents share the belief about [126]. Researchers have argued We-intention to be a mode or an attitude that differentiates it from a case of fear [126], and contrasts with I-intentions that are *internal* to an agent. However, we cannot expect people always to form We-intention with a social robot, requiring it to manage breakdown situations.

For example, Fig. 1 depicts alternative interaction flows, where, in Fig. 1a the agent avoids (hence, the name avoidant behavior), the human's proposed We-intention (that is not to have breakfast) and continues to co-create its prior proposed We-intention even though the person is not up for it. On the other hand, and in Fig. 1b a *partially* agreeing behavior is considered when some intentions of the person coincide or align with that of agent, but there exist conflicting intentions that lead to a partially-agreeing behavior. In Fig. 1c, the robot adapts its intention with respect to what the person proposes, thus displaying an agreeing behavior. Therefore, in this work, we propose a formal framework that allows a social robot to co-create such behaviors, facilitating We-intention despite breakdown situations from internal conflicts that may emerge from an agent's I-intention.

This work aims to develop and exemplify a formal framework to manage intentions and breakdown situations. This is done based on earlier theory-based user studies of people interacting with social robots in home-care scenarios focusing on the formalization of agreement on We-intentions [102,120,121]. Following previous works [120,121], we identified and define the following three situations that can occur relating to an agreement on We-intention:

- We-intention alignment. A social robot believes that a person intends to do a joint activity, and thus such an agent intends to do its part of the activity.
- We-intention breakdown. A social robot believes that the person does not intend to do a joint activity but rather has an activity contradictory to the agent's proposed activity.
- Partial We-intention alignment. A social robot believes there is a fragmentary agreement to jointly do an activity with the person.

Two main technical challenges that we faced in this paper are: 1) the uncertainty in the shared intention (*e.g.* a robot and a person try to act jointly, and both have partial information of the intentions of each other), and 2) a partial or null congruence between the internal and shared intentions (*e.g.* the social robot's and the person's intentions are contrary). Therefore, the following research questions are addressed in this paper:

1. How can We-intentions and breakdown situations in human-robot joint activity be captured and repaired using a formal computational mechanism?

2. How could such a computational mechanism be materialized in human-robot joint activity and be perceived by people?

Research question 1 is addressed by extending a well-established framework of *rational agents* [16] with a *non-monotonic* mechanism for repairing the We-intention before the plan is defined and executed. The mechanism is exemplified (addressing Research question 2) by revisiting three dialogue scenarios and analyzing data collected in earlier studies with a particular focus on three levels of alignment defined in this paper [120,121]. The contributions of this paper are the following:

- Novel application of Answer set programming as a formal framework to form and repair We-intentions in a human-robot joint activity specific to healthcare scenarios.
- Framework founded in a human-centered methodology basing our formal framework on empirical findings.

More specifically: 1) we prove that the mental states of an agent are always consistent if an *Answer set programming* (ASP) [85] approach is used to manage We-intentions (see Proposition 2 and Theorem 1); 2) we present two mechanisms for repairing We-intention breakdown, the first one using the *Closed World Assumption* in ASP (Proposition 3), and another more restrictive using *ASP constraints* (Proposition 4).

This paper is organized as follows: the methodology and a necessary background are provided in Section 2, and the results, including the formal framework and some properties that the framework fulfills, in Section 3.1. We exemplify the formal framework and how people may perceive the agent's behavior in Section 3.2. The related work of this paper was made following a systematic literature review procedure and is presented in Section 4. We end our paper by discussing our contributions in Section 5 and conclusions in Section 6.

2. Methods

This section introduces the necessary background for characterizing We-intentions based on Tuomela's work [125,126,128], and Subsection 2.2 introduces the formal framework based on Answer set programming (ASP), basic concepts, and the theoretical background used.

ASP provides methods to account for 'uncertainty' for knowledge representation and allows non-monotonic reasoning. Non-monotonic reasoning draws tentative conclusions, which can be retracted in the presence of new evidence or facts. This is important for co-creating We-intentions as they evolve during the interaction, and the agreement about them may change as the interaction unfolds.

The intention control mechanism was defined in a high-level algorithm which was evaluated as a first step by applying this to a subset of the scenarios defined and evaluated in [120,121]. Three human-robot healthcare scenarios were used as a benchmark in this evaluation. The methodology to develop those three scenarios is presented in Subsection 2.3.

2.1. We-intentions concepts

The concept of joint intentions is central to the We-intentions theory, which drives the agents' acting together. Therefore, to define joint intentions between agents, we follow the work of Tuomela [125–127]. In Tuomela's work, a *joint intention* is referred to as "action intention", which can be achieved by agents performing joint actions. These actions are preconditioned on involved agents having 1) a We-intention to perform the actions, and 2) they mutually believe those actions can be achieved. Such We-intentional agents, during a joint activity, can commit to their *private* or *internal* intentions, in other words, form an *I-mode* attitude or a mutually agreed *We-mode* attitude. Let us define the two types of intentions more formally depending on the intentional mode in a formal representation used in our framework. We define *internal*, *external*, and *We-intentions* during an interaction between a social robot and a person, as follows:

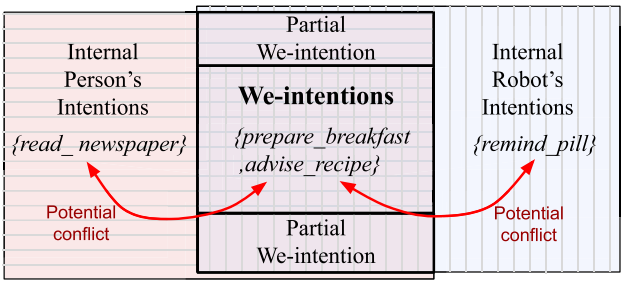


Fig. 2. We- and I-intention scenarios in a human-robot interaction.

Definition 1 *(Internal intentions)*. Let $\mathcal{I}^{ag}$ be the full set of an agent's intentions, then the *internal intentions* of an agent is a set $I^{ag}_{Int} \subseteq \mathcal{I}^{ag}$.

Similarly, we can define the agent's external and We-intentions *with respect to* a person as follows:

Definition 2 *(External intentions)*. Let $\mathcal{I}^{per}$ be the set of a person's intentions. Then we say that the *external intentions* of an agent is a set $I^{ag}_{Ext} \subseteq \mathcal{I}^{per}$.

Then we define the We-intentions:

Definition 3 *(We-intentions)*. Let $\mathcal{I}^{ag}$ and $\mathcal{I}^{per}$ be the sets of (full) intentions of an agent and a person. Then a *We-intention* is given by: $I^{ag}_{We} \subseteq \mathcal{I}^{ag} \cap \mathcal{I}^{per}$.

Example 1. Suppose a social robot (as an agent) aims to support the health and well-being of a person through two internal intentions: 1) reminding to take pills, and 2) advising healthy cooking recipes, *i.e.* $\mathcal{I}^{rob} = \{remind_pill,\ advise_recipe\}$. Now, let us suppose that a person starts an interaction with a social robot in a kitchen, saying that she intends to prepare breakfast and read the newspaper, *i.e.* $\mathcal{I}^{per} = \{prepare_breakfast,\ read_newspaper\}$.

Then, in this specific scenario, three sets of alignment, breakdowns due to conflict, and partial alignment of intentions can be found (see Fig. 2):

- We-intentions alignment: $I^{rob}_{We} \equiv I^{per}_{We}$

  $I^{rob}_{We} \equiv I^{per}_{We} = \{prepare_breakfast,\ advise_recipe\}$
- We-intentions breakdown: between I^{rob}_{Int} and I^{per}_{Int}

  $I^{rob}_{Int} = \{remind_pill\}$

  $I^{per}_{Int} = \{read_newspaper\}$
- Partial We-intentions alignment: between I^{rob}_{Int} and I^{per}_{Int}

  $I^{rob}_{Int} = \{advise_recipe\}$

  $I^{per}_{Int} = \{prepare_breakfast,\ read_newspaper\}$

Moreover, we can see that the subset $I^{per}_{Ext} = \{remind_pill\}$ conflicts with every other intention of the person.

The previous example presents two types of intention relations: a potential *intention alignment* and potential *intention breakdowns* due to conflict and partial alignment of We-intentions.

Definition 4 *(Intention breakdown)*. Let $i_a \in I^{ag1}_{We}$ and $i_b \in I^{ag2}_{Int}$, be two intentions from agents $ag1$ and $ag2$, we say that there is a breakdown

situation because i_a is in *conflict* with i_b if there is a semantic evaluation in which the set $\{i_a, i_b\}$ is inconsistent. Noted as $\texttt{BrkDwn}(i_a, i_b)$.

We define an alignment relationship between two intentions as:

Definition 5 *(Aligned intention)*. Let $i_a \in I_{We}^{ag1}$ and $i_b \in I_{Int}^{ag2}$ be two intentions, if the set $\{i_a, i_b\}$ is consistent, then we say that i_a and i_b are semantically aligned. Noted as $\texttt{ALIGN}(i_a, i_b)$.

These relationships have a *semantic* perspective, *i.e.* two intentions may or may not be in conflict if an interpretation of those intentions leads to a semantic disagreement if they belong to the same set of intentions. In this paper, we do not suggest any particular formal computational mechanism for such semantic interpretation; however, we use techniques from *natural language processing* (NLP) in our implementation (see [117] for a review).

Agreeing and avoiding intentions for forming We-intentions

In Tuomela's work [128], different requirements are necessary for a We-intention formation, such as a collective of agents and mechanisms for sharing intentions, beliefs, and task distributions. However, in healthcare scenarios, the relationship between a social robot and a person is not symmetric, *i.e.* it is expected that the person's intentions and desires have greater importance or relevance than a social robot's. Therefore, we explore three types of intention acceptance during We-intention breakdown situations:

Definition 6 *(Agents intention acceptance types)*. Let $ag1$ and $ag2$ be two agents forming a We-intention, with $i_a, i_c \in I_{We}^{ag1}$, $i_b \in I_{Int}^{ag2}$, and i_a, i_b have a breakdown $\texttt{BrkDwn}(i_a, i_b)$, and I_{Int}^{ag2} are preferred than I_{Int}^{ag1}. Then, $ag1$ can be an:

- Agreeing agent, *accepts* an intention despite a potential internal conflict, by integrating the external intention into its own intentions: $I_{Int}^{ag1} \cup \{i_b\}$
- Avoidant agent that constraints (*blocks*) an external intention to avoid a breakdown due to an internal conflict: $I_{Int}^{ag1} \cup \{i_b^{bloq}\}$
- Partial agreeing agent, which accepts part of the intentions of the other agent and blocks other intentions: $I_{Int}^{ag1} \cup \{i_b^{bloq}\} \cup \{i_c\}$

where i_b^{bloq} symbolizes the generation of a constraint that disables the intention to be used by the agent when it is integrated into its knowledge base.

In the next section, we will formalize the notions of the types of agents presented in Definition 6; we will use Answer set programming as a mechanism for representing the accepting and blocking intentions during We-intention breakdown situations.

2.2. Theoretical framework background

In this section, we introduce the necessary background to characterize We-intentions in a person-agent scenario.

Syntax

In this paper, we assume that every agent has a knowledge base encoded using an *extended logic program* (ELP) [56], which is a set of rules with the form: $L_1, \ldots, L_l \leftarrow L_{l+1}, \ldots, L_m, not\ L_{m+1}, \ldots, not\ L_n (n \geq m \geq l \geq 0)$ where each L_i is a positive/negative *literal*. *not* is *negation as failure* (NAF) [56] (*e.g. not A* represents uncertainty to draw a conclusion about atom A). $\neg A$ represents negative information (*w.r.t.* A). The symbol "," represents disjunction. The left-hand side of a rule is called the *head*, and the right-hand side is the *body*. A $head(r)$, $body^+$ and $body^-$ represent literals $L_1, ..., L_t$, $L_{t+1}, ..., L_m$ and $L_{m+1}, ..., Ln$, respectively. A rule r is a *constraint* if $head(r) = \emptyset$; and r is a *fact* if $body(r) = \emptyset$. A program P is NAF-free if $body^-(r) = \emptyset$ for every rule r in P [110].

Semantics

In this paper, we use *answer set semantics* [56], an extension of *Stable model semantics* (STB). For STB, if a Lit is the set of all ground literals of an ELP, and a set $S \subseteq Lit$, then given a ground rule r if $body^+(r) \subseteq S$ and $body^-(r) \cap S = \emptyset$, then implies that $head(r) \cap S = \emptyset$. In particular, S satisfies a ground integrity constraint r with $head(r) = \emptyset$ if either $body^+(r) \subseteq S$ or $body^-(r) \cap S = \emptyset$. S satisfies a ground program P if S satisfies every rule in P. When $body^+(r) \subseteq S$ (*w.r.t.* $head(r) \cap S = \emptyset$), it is also written as $S \models body^+(r)$ (*w.r.t.* $S \models head(r)$).

Definition 7 *(Answer set function* $\texttt{AS}$*)*. Let P be a NAF-free ELP, a set $S \subseteq Lit$ is an answer set of P if S is a minimal set such that: 1) S satisfies every rule from the ground instantiation of P, and 2) $S = Lit$ if S contains a pair of complementary literals L and $\neg L$. The rule $r^S : head(r) \leftarrow body^+(r)$ is included in the *reduct* P^S if $body^-(r) cap S = \emptyset$. Then, S is an answer set of P if S is an answer set of P^S. In this paper, the set of all answer sets of P will be written as $\texttt{AS}(P)$.[1]

In this paper, the difference between *not* P and $\neg P$ is essential whenever we cannot assume that the available positive information about P is complete, *i.e.* when the *closed world assumption* (CWA) does not apply to P [56].

Definition 8 *(CWA)*. Let $x \in P$ be an atom x, we use $\texttt{CWA}(x)$ to denote the following operation to x: $\neg x \leftarrow not\ x$

Example 2 *(Applying* $\texttt{CWA}$*)*. Let $I_{Int}^{rob} = \{remind_pill\}$ be an internal intention of a social robot. Then, if we apply $\texttt{CWA}$ to the atom *remind_pill*, we will obtain $\texttt{CWA}(remind_pill) = \{\neg remind_pill \leftarrow not\ remind_pill\}$, which has the intuitive reading: "*if there are no evidence that the pill was reminded, then it is assumed that the pill reminder was not given*".

We will use $\texttt{CWA}$ as a mechanism for dealing with an agreeing agent behavior (Definition 6). In the same context, we can *block* a specific atom, for example, x, by adding the rule $\bot \leftarrow x$ into the program.

Example 3 *(Blocking an atom)*. Let $I_{Int}^{rob} = \{remind_pill\}$ and $I_{Int}^{per} = \{read_newspaper\}$ be two agents' intentions, then if the social robot is forced to accept the external intention, having a consistent knowledge base P, it can block *read_newspaper* as follows: $P \cup \{\bot \leftarrow read_newspaper'\}$.

2.3. Benchmark scenarios and people's perception of breakdown situations

To address our second research question, we adopt three video-recorded dialogue scenarios from previous work [120,121] to exemplify the formal computational mechanism directing the robot in human-robot joint dialogue activities. The selected scenarios illustrate situations when We-intention breaks down (We-intention conflict and We-intention partial alignment), and aligns (We-intention alignment). Furthermore, the scenarios embed situations when the social robot and the person display agreeing, partially agreeing, and avoidant behavior. The collected data obtained through interviews with participants viewing reflecting on the recorded dialogues were analyzed with a particular focus on the three kinds of behavior (agreeing, partially agreeing, and avoidant behavior). The results were expected to illustrate a tentative user's experience of We-intention alignment and breakdown situations.

Selected scenarios

The first scenario illustrates **We-intention alignment** where a social robot and a person begin the day by interacting about how the person is feeling and what can be done about it if they are not feeling well.

[1] In this paper we use the ASP solver *DLV system* [82].

The social robot adapts its behavior to the human's suggested topic (see Fig. 3). The second scenario illustrates the **We-intention breakdown** presented in Fig. 4, where the social robot and person talk about the person not sleeping well because of pain. The dialogue leads to the person rejecting the social robot's proposal to address the pain issue, thus causing a breakdown of We-intention. In the third scenario embedding **We-intention partial alignment** exemplified in Fig. 5, the person and the social robot talk about pain, and the person again rejects the robot's suggestion. Still, when the social robot provides a supporting argument, the person accepts the robot's proposal, thus, resulting in a partially aligning of We-intentions.

Study setup, data selection, and analysis

We refer to the two persons who acted in the WoZ setup as volunteers and people who participated in the study, viewing the recorded scenarios as participants.

20 participants in the age range of $(23-72)$ were recruited for the study. For analysis purposes, we categorize the participants aged 23-40 as younger participants (YA) and those between 41-72 as older participants (OA). Each group has 10 participants, with six women and four men each.

The study was conducted remotely, where the participants watched audio and video recordings and participated in a semi-structured interview. The recordings were constructed in a Wizard of Oz (WoZ) setup with two volunteers interacting separately with a Nao robot. The joint activities were authored dialogues on daily living healthcare situations and were performed in a lab turned into a home environment.

The interview contained the following questions: (1) What goal did the social robot and volunteer have in this dialogue? (1.2) Did you notice any mismatch between those goals? (2) What sort of behavior did the social robot display? and (3) What kind of behavior did the volunteers display?

The analysis focused on whether the agents' (volunteers and social robot) goals matched or mismatched and whether we could categorize the social robot's and volunteer's behavior as agreeing, partially agreeing, or avoidant.

Data were analyzed qualitatively using Thematic Analysis (TA). TA is a qualitative method used to derive "patterns of meaning" referred to as themes in a data set. TA is considered a rigorous, systematic, and accessible approach to coding and theme development [23]. To apply TA, we followed the following steps: (1) familiarization with the transcribed data with research questions in mind. (2) Identification of *codes* corresponding to participants´ perception of the person and the social robot's 'intention' and 'behavior' aspects. The codes were based on keywords such as 'conflict,' 'ignored,' 'intention,' 'goal,' 'behavior,' etc. (3) The codes were categorized into agreeing, partially agreeing, or avoidant behavior; and when there was a breakdown or alignment of We-intention. TA was performed by one of the authors using Taguette software [104].

3. Results

The results include a novel formal computational mechanism enabling agents to deal with We-intention formation under uncertainty (Subsection 3.1) and exemplifications of the framework based on a user study of human-robot dialogues (Subsection 3.2).

3.1. Theoretical results - the adjustable intentionality framework

In this paper, we characterize an agent as an entity with *beliefs*, *desires*, and *intentions* [105,16]. As a shortcut notation, we will use $\mathcal{I}$ to note the intentions of any agent, instead of $\mathcal{I}^{ag}$ as a mechanism for generalization; we will use the specific notation $\mathcal{I}^{ag}, \mathcal{I}^{per}, I^{ag}, I^{per}$ when we describe joint activities.

Definition 9 *(Joint belief-desire-intention framework)*. Let $\mathcal{B}$ be a set of agents' beliefs, $\mathcal{D}$ be a set of desires, and $\mathcal{I}$ be the joint intentions of an agent Ag. Then, a *joint BDI framework* is a tuple $JBDI = \langle \mathcal{B}, \mathcal{D}, \mathcal{I} \rangle$, where $\mathcal{I} = I_{Int} \cup I_{Ext}$.

In this setting, a We-intention is the set $\mathcal{I} = I_{Int} \cup I_{Ext}$, and an agent uses $JBDI$ framework to generate consistent *plans* considering their and the intentions of other agents, commit to one of them, and execute it. Such procedure is performed by a *control loop* in Algorithm 1, which iterates until the agent is *active*.

Control loop specification

In this loop, functions such as intend(), cooperate(), or plan() (lines 12, 18, and 20 respectively) among others, are formally implemented and described in terms of logic programming procedures, as is presented in Table 1. The control loop starts (lines 1, 2, and 6) with the initialization of atoms and sets of atoms. The iterative reasoning process initiates with a fact-obtaining phase (line 9), where F is a set of facts ($bod\,y(r) = \emptyset$, of a given rule r), which are joined to the set of initial beliefs $\mathcal{B}_0$ to update them (line 10), and generate a new set of desires based on the initial set of intentions. Then, the loop starts with the manipulation of intentions (highlighted lines in blue in Algorithm 1), which is the core of our contributions.

Table 1 provides a simplified explanation of important functions for intention management (first column) that are used in the presented control loop. The second column is the actual formalism in terms of Answer set programming.

Intention generation and reparation

In line 12 of Algorithm 1, intend is the process for generating the intentions of the agent. In this paper, the answer sets obtained from function AS are considered *potential intentions*. In line 13, the agent obtains the intentions that other agents share.

The line 14 of our control loop is the primary evaluation procedure for intention reparation, which is when an external intention is not part of the agent's internal intention set, and at the same time, the external intention has a full or partial conflict with the existent intentions, then, a reparation intention process starts (repairCooperation()) with three alternatives (options **OP**) for repairing the We-intention breakdown:

OP1 ***Agreeing scenario***: the agent applies CWA for every atom that is not congruent with its internal intentions. For example, an atom $x \in I_{Ext}$ that is $x \notin I_{Int}$, then the *agreeing agent* accepts the new intention atom without making inconsistent its already defined intentions. It adds the following sets of rules: $\{ x \leftarrow not\ \neg x,\ \neg x \leftarrow not\ x \}$.

OP2 ***Avoidant scenario***: the agent creates *constraints* for every atom that is not congruent with its intentions. Continuing with the example in OP1, an *avoidant agent* adds the following rule $\bot \leftarrow x$ for every $x \in I_{Ext}, x \notin I_{Int}$.

OP3 ***Partial agreeing scenario***: the agent accepts and constrains parts of the external intention set. For example, let $x, y \in I_{Ext}$ and $\mathtt{BrkDwn}(\{x, y\}, I_{Int})$, then it can be the case that $I_{Int} \cup \{\bot \leftarrow x\} \cup \{\neg y \leftarrow not\ y\}$, blocking x and using CWA with y.

In this paper, we call *cooperate* (line 18) the process of adoption of external intentions. This mechanism consolidates a set of potential intentions that always is consistent (see properties in the next section), whether other agents' intentions were repaired or not in a previous step. Such a consolidated set of intentions is shared by the agent (line 21), and finally, the control loop ends with a selection of one intention that the agent is committed to (line 20), which will be executed (line 22).

In the following, we present novel formal properties of our control loop based on answer sets.

Properties of an answer set-based control loop

In the previous section, we present an extension of a "classic" BDI control loop with novel characteristics using the Answer set program-

input : $JBDI = \langle \mathcal{B}, \mathcal{D}, \mathcal{I} \rangle$; // Joint BDI framework
output: π^* ; // A plan executed

1 $I_0, \mathcal{B}_0$; // Initial states of the agent
2 $\mathcal{I} \longleftarrow \emptyset$; // We-intentions set
3 $I_{Int} \longleftarrow \emptyset$; // Internal intentions
4 $I_{Ext} \longleftarrow \emptyset$; // External intentions
5 $I^r_{Ext} \longleftarrow \emptyset$; // Repaired external intentions
6 $F \longleftarrow \emptyset$; // Empty facts
8 **while** *active* **do**
9 $F \longleftarrow$ getFacts(); // Perception of facts
10 $\mathcal{B} \longleftarrow$ update$(F, \mathcal{B}_0)$; // Update beliefs
11 $D \longleftarrow$ wish$(\mathcal{B}, I_0)$; // Desire generation
12 $I_{Int} \longleftarrow$ intend$(\mathcal{B}, D, I_{Int})$; // Internal intentions
13 $I_{Ext} \longleftarrow$ getJointIntent(); // Getting the others' intentions
14 **if** $\exists\, i \in I_{Ext} | i \notin I_{Int}$, *and* $\mathit{BrkDwn}(i, I_{Int})$ **then**
15 $I^r_{Ext} \longleftarrow$ repairCooperation(I_{Ext}); // Intention reparation
16 $\mathcal{I} \longleftarrow$ cooperate(I_{Int}, I^r_{Ext}); // Repaired We-intention
17 **else**
18 $\mathcal{I} \longleftarrow$ cooperate(I_{Int}, I_{Ext}); // We-intention
19 **end**
20 $\pi \longleftarrow$ plan$(B, \mathcal{I})$; // Planning
21 share$(\mathcal{I})$; // Share intentions
22 execute(π); // Plan execution
23 **end**

Algorithm 1: Adjustable intention control loop. Highlighted lines from 12 to 18 are the main focus of this paper.

Table 1
Map of mental state functions of an agent's control loop (Algorithm 1), and formal procedures in Answer set programming.

Control loop function	LP procedure
getFacts()	$F \subseteq \mathcal{B} = \{x, \ldots, y\}$ where $x, \ldots, y$ are facts
update(x, B)	$B \setminus x' \cup x \mid x' \in B, x \notin B$
wish$(\mathcal{B}, I_0)$	$P \subseteq F \cup R$
intend$(\mathcal{B}, D, I_I nt)$	$\mathtt{AS}(P)$
getJointIntent()	$F \cup F' \mid F' = I_{Ext}$
repairCooperation$(I_E xt)$	**OP1**. $I^r_{Ext} \cup \{\neg x \leftarrow not\ x\} \mid \forall x \in I_{Ext},\ x \notin I_{Int}$ **OP2**. $I^r_{Ext} \cup \{\bot \leftarrow x\} \mid \forall x \in I_{Ext},\ x \notin I_{Int}$ **OP3**. $I^r_{Ext} \cup \{\bot \leftarrow x\} \cup \{\neg y \leftarrow not\ y\} \mid x, y \in I_{Ext},\ x, y \notin I_{Int}$ and $\mathtt{BrkDwn}(\{x, y\}, I_{Int})$ for **OP1**,**OP2** and **OP3**
cooperate(I_{Int}, I_{Ext}) or cooperate(I_{Int}, I^r_{Ext})	$\mathtt{AS}(P) \cup I_{Ext}$ or $\mathtt{AS}(P) \cup I^r_{Ext}$
plan$(\mathcal{B}, \mathcal{I})$	$\mathtt{SEL}_x, \alpha(\mathtt{AS}(P) \cup I_{Ext})$ or $\mathtt{SEL}_x, \alpha(\mathtt{AS}(P) \cup I^r_{Ext})$

ming approach. In this section, we present the formal properties of our framework. We start with a set of fundamental axioms defining key relationships in the We-intention formation.

Proposition 1 *(Axioms of person-agent We-intention formation). Let ag and per be two agents where per represents a person, with sets of intentions* $\mathcal{I}^{ag}$ *and* $I^{per} \subseteq \mathcal{I}^{per}$*. The following axioms define We-intentions relations:*

- $\mathcal{I}^{ag} \neq \emptyset$*, Intentional agent*
- $\mathcal{I}^{ag} \cap \mathcal{I}^{per} = \emptyset$*, Breakdown scenario*
- $I^{per} \cap \mathcal{I}^{ag} \neq \emptyset$*, Partial We-mode*
- $I^{per} \equiv \mathcal{I}^{ag}$*, Full We-mode*
- $\mathcal{I}^{per} \cap (\mathcal{I}^{ag} \cup I^{per} *) \neq \emptyset$*, Person's intention adaptation (CWA application)*
- $\mathcal{I}^{per} \cap (\mathcal{I}^{ag} \setminus I^{per} *) \neq \emptyset$*, Person's intention inhibition*

Proposition 1 establishes the initial conditions for a We-intention formation. The first axiom defines a desirable characteristic of an intentional agent, where the set of intentions of an agent should not be empty. The rest of the axioms are consequences of considering breakdowns and agreements among intentions, which is the key characteristic for decision-making in Algorithm 1.

Consistent mental states using answer sets approach:

We start by showing a key property in the process of intention sharing from the perspective of an agent that receives an intention (not the initiator of the joint intention).

Proposition 2 *(Consistent shared intentions). Let* Ag_1 *and* Ag_2 *be two agents with knowledge bases encoded in logic programs* P_1 *and* P_2*, respectively. If each agent uses an answer set approach to generate their intentions (Algorithm 1), then every set of the shared intentions (partial intentions) is consistent.*
See Proof A in Appendix section A.

The importance of Proposition 2 lies in the fact that when sets of a stable model are used in shared mental states such as intentions or partial intentions, they are always consistent, meaning that under our approach, two agents cannot have uncertain atoms that may generate misinterpretations.

Theorem 1 *(Consistently shared states). Let* Ag_1 *be an agent with encoded information in a program* P_1*. If an answer set process is used for interpreting any mental state (w.r.t. BDI), then such mental state representation is consistent.*
See proofs in Appendix A.

Generalizing from Proposition 2, Theorem 1 establishes a clear difference between our answer set approach and previous approaches. Unlike other contemporaneous control loops for rational agents, Algorithm 1 guarantees consistency when used to encode mental states.

Repairing shared intentions

In Algorithm 1 line 15, we presented a mechanism of shared information manipulation that an agent can use when there is a partial convergence among internal and shared intentions.

Proposition 3 *(Incompatible atom elimination using CWA). Let* P_1 *and* P_2 *be two encoded knowledge bases of two agents. If* $\exists x \in P_2$ *and* $x \notin P_1$*, we say that* x *is* incompatible *w.r.t.* P_1*. Then, by adding the rules:* $x \leftarrow not\ \neg x$ *and* $\neg x \leftarrow not\ x$ *into* P_1*, the incompatible atom* x *is eliminated, being* $P_1 \cup \{x \leftarrow not\ \neg x\} \cup \{\neg x \leftarrow not\ x\}$ *consistent.*

There is the practical importance of Proposition 3, which is when an agent needs to "adopt" an external atom as part of its knowledge base (see **OP1** in Table 1), maintaining at the same time its knowledge base consistency. The proposed method for repairing incompatibilities defines an agreeing behavior of such an agent, meaning that agents using this strategy will always accept a shared atom. From the perspective of We-intentions in [128], Proposition 2 *i.e.* OP1 process in Algorithm 1, is a necessary condition to establish a *full-blown* joint intention, which is the case of every participant symmetrically having the same relevant We-intention [128]. On the other hand, if an agent restricts a shared atom instead of adopting it, then it creates a logic programming constraint, which makes such an atom impossible to be true.

Proposition 4 *(Restricting atoms with constraints). Let* P_1, P_2 *be two encoded knowledge bases of two agents* Ag_1, Ag_2. *If* $\exists x \in P_2$ *and* $P_1 \cup \{\bot \leftarrow x\}$, *then* x *will not be true in* Ag_1.

The previous proposition is a strong restriction for accepting any atom. The procedure presented in Table 1 as **OP2** is for agents that reduce the margin of cooperation.

Computational complexity of the framework

In this section, we address the *computational complexity* of the proposed framework, mainly the associated costs of different parts of Algorithm 1. To this end, we present the approximate computational cost of intention generation considering well-established *asymptotic upper bounds*. In this context, we are not interested in the specific computational cost (time) that a function or the entire Algorithm 1 has in a given programming implementation. Instead, we are focused on delineating approximate upper boundaries, which is a more general exercise with practical implications. Finally, we summarize a set of heuristics that can be used to cope with "costly" functions in our framework.

Notation We use standardized computational complexity notation [69]. Appendix B introduces a background of computational complexity theory in logic programming and the corresponding notation.

Approximate computational cost analysis of functions in our control loop

- **getFacts**() (Line 9). Perception of facts can be considered as a non-complex task in the Algorithm 1 setting, *i.e.* its cost is linear ($O(n)$) and dependent on the environment. The rationale for this assumption is that obtaining facts from the environment does not imply *search* or more computationally complex tasks. However, since the late 1990s, it has been well-known that the (time) cost of *sensing* in social robots depends on the type of environment (see the work of Kinny, Georgeff, and Henlder in [73] to assess optimal sensing considering static and dynamic worlds).
- **update**() (Line 10). The LP procedure of the beliefs update function $B \setminus x' \cup x \mid x' \in B, x \notin B$ can be considered a *belief revision* procedure in the logic programming literature [2,4,35,66]. In that context, the computational cost depends on the associated tasks to the update operator ($\cup$ in Line 10) and the type of modification that such operation implies, for example, in [9,34,66] (among others), a comparison between *answer set models* [86] is performed to evaluate the *equivalence* of programs (in the sense of *strong equivalence* [87]). Which is a *decision problem* of the form: given B and x', is x' true in all answer sets of B? which is the decision whether to update B with x' or not. In [48] was identified that using *cautious reasoning*, the complexity of *aggregate* operators (functions) brings the cost to Π_2^P considering disjunctive programs. Other updating/aggregate operators and their complexity analysis using different underlying representations and semantics have been reported in [31,44]. This in-depth analysis of the update operator will be part of our future work.
- **wish**() (Line 11). This procedure is oriented to bring about a relevant desire (or goal) from a defined set of goals (following the "standard" BDI model [18]). There are several mechanisms how wish could be practically implemented, as a search process and preference-elicitation mechanism. These can be considered polynomial-time algorithms, or at least there are polynomial-time reductions of such problems. However, regarding search mechanisms, the existence of polynomial-time decision algorithms alone does not ensure that a corresponding search problem can be solved efficiently or correctly [50]. Other non-monotonic reasoning mechanisms for implementing a wish function have been proposed in [59] using *formal argumentation theory*, which, under the *stable argumentation semantics* [41] and under a *credulous* perspective is non-deterministic NP (see technical details in [42]).
- **intend**() (Line 12). In Algorithm 1, the intention generation is performed by a function AS(), which is the generation of answer sets (stable model [56,86]). Such operations are *NP-hard* problems with several variants and reductions. Moreover, it has been proved that other less *costly* mechanisms can generate equivalent answer sets under certain underlying representation restrictions (see [38,39]).
- **getJointIntent**() (Line 13). This function integrates the internal intentions of the agent with the external intentions of other agents. In this context, the computational cost is not significant, considering the cost of other functions.
- **repairCooperation**() (Line 15). A key part of the paper is the repairing options OP1-OP3. In any of these options, the suggested functions can be considered as logic program updates, more specifically as ASP model updates, in which, as it was mentioned in update(), the upper boundary can be Π_2^P for certain underlying knowledge representations. However, despite the apparent high cost, a simple addition of a rule to an ASP model is not a complex process that can be performed in polynomial time, given that it does not imply equivalence verification as in "standard updates" of logic programs *e.g.* [66].
- **cooperate**() (Line 18). This function is a joining programs mechanism that can be performed in *linear* time, depending on the size of the added set *i.e.* the intention to be assimilated.
- **plan**() (Line 20). This function is performed for an *answer set planning* mechanism, which differs from *satisfiability planning* in that it uses logic programming rules instead of propositional formulas [86], then, the answer sets for that program represent different possible *evolution* or scenarios. In [124], a review of answer set planning mechanisms, the authors compile several ASP-related mechanisms for planning, showing several heuristics are used to reduce the computational cost of the planning task. We also acknowledge that some ASP planning mechanisms have been used to model the behavior of agents and multi-agents in a review article [43]. Then, in general, plan as an ASP planning mechanism can be considered NP-hard in the worst-case scenario.

In summary, the computational cost of Algorithm 1 can be reduced to the intention generation cost (intend() - Line 12) and the cost of planning (Line 20). If cost(x) is a function that retrieves the approximate computational cost (time) of every procedure in Algorithm 1, the total cost ($TOTAL$) can be approximated to the following expression:

$$\begin{aligned} TOTAL = O(&\text{getFacts}(n), \text{update}(n), \text{wish}(n), \\ &\text{intend}(n), \text{repairCooperation}(n), \text{cooperate}(n), \\ &\text{plan}(n)) \leq O(\text{intend}(n)) \\ = O(n^k)k \geq 1 \end{aligned} \tag{1}$$

The strength of the BDI models lies in their use of heuristics, which attacks the complexity of the problem with domain-independent strategies that allow it to make decisions with as much information as possible given the resources that are available [116]. In this sense, different heuristics have been proposed to reduce the computational cost of specific procedures inside BDI-like control loops. In Appendix B.2,

Table 2
Results of the We-intention alignment scenario. In the left column are presented some lines in Algorithm 1, and the right column shows the output of selected functions in the control loop.

Line	Output of Algorithm 1 for the agreeable robot scenario
9-13	$F = \{$ morning;in_kitchen;says_wants_breakfast $\}$ $I_{Int} = \{$ take_breakfast :- says_wants_breakfast . take_breakfast :- in_kitchen, morning, not breakfast_taken. read :- in_kitchen, has_newspaper. read :- says_wants_read. $\}$ $I_{Ext} = \{$ remind_pill :- morning, not pill_taken. advise_breakfast :- in_kitchen, morning. $\}$
14	BrkDwn({remind_pill},{read})
18	$\mathcal{I} = \{$ take_breakfast :- says_wants_breakfast . take_breakfast :- in_kitchen, morning, not breakfast_taken. advise_breakfast :- in_kitchen, morning. $\}$ $\mathcal{B} = \{$ morning;in_kitchen;says_wants_breakfast. $\}$
20	$\mathcal{B} = \{ \pi = \{$advise_breakfast$\} \}$

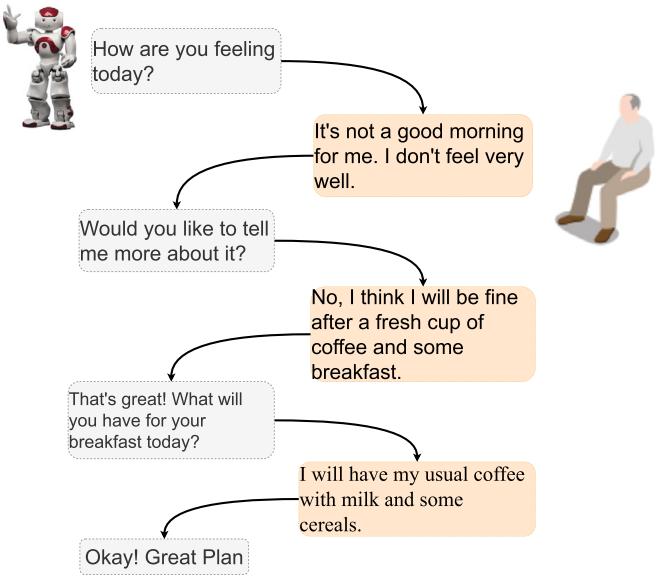


Fig. 3. Scenario 1, depicting We-intention alignment, *i.e.* the agreeing robot (based on Figure 3 in [121]).

we present a non-exhaustive list of computational heuristics that can be (re)used in Algorithm 1 to build potentially less costly implementations.

3.2. Application and perception of intention control loop

In this section, the intention control loop is exemplified using three scenarios embedding We-intention alignment, breakdown, and partial alignment in dialogues between a social robot and volunteers. Furthermore, Algorithm 1 is applied to the three scenarios, and study participants' experiences and perceptions about them are summarized.

Scenario 1 *(We-intention alignment - the agreeing robot)*. **Preconditions:**

- The volunteer (person) is unaware of the social robot's dialogue or intention.
- The *social* robot is unaware of the volunteer's intention.
- The environment is static. There are no changes in the kitchen that affects the location of the social robot.
- The social robot has pre-defined programs capturing a common health and well-being scenario.

The social robot is situated in a corner near the breakfast table. In the morning, the volunteer arrives in the kitchen to have breakfast. The social robot greets and asks how the volunteer is feeling. The volunteer indicates that they are not feeling well. The volunteer says that having breakfast could improve their health situation. The social robot agrees and aligns itself to support the breakfast preparation. The social robot presents some (We-intentions) alternatives. The volunteer responds, and the social robot selects the joint activity. The social robot concludes the dialogue after the volunteer responds to the breakfast selection (Fig. 3 illustrates the unfolding of the dialogue).

Post-conditions:

- The volunteer changed the topic of the dialogue.
- The social robot adapted and ends the dialogue.
- The environment remains static.

In Table 2, we present the results of using Algorithm 1 in Scenario 1. We limit our attention to specific lines of the control loop, which represents the person's perspective of the scenario, *i.e.* we interpret *internal intentions* from the perspective of the person, then, I_{Int} represents intentions of the person, I_{Ext} the agent's intentions, and F the perceived information from the social robot's sensors. In line 14, we highlight the potential semantic breakdown between internal and external intentions, and lines 18 and 20 show the intentions' alignment output.

Participant's potential perception of We-intention alignment and agreeing behavior: Half of the participants observed and commented on the volunteer's and robot's agreeing behavior.

Older participants described the social robot's agreeing behavior by characterizing it as being cooperative and displaying care towards the human: *"Cooperative and efficient. We can't expect more from the robot."* Another older participant described the social robot's interaction as being soft and appropriately situated: *"The interaction with the people was very soft. He reacts, turns around, and comes back in a polite way. The time-lapse was good, and it was not right away."*

The young participants describe the social robot displaying empathy, positivity, and care for the person participant, *"...robot seems to have a little bit of empathy."* Half of the younger participants commented on how the volunteers in the scenario overall displayed agreeing behavior by recognizing the social robot's presence and by adapting their dialogue towards it: *"I guess they were being kind of polite to it. They said thank you, it's a good idea like they were almost in on it..."*.

However, the participants noted that the older volunteer was accepting, paid attention, and treated the social robot almost like a person compared to the young man; *"older lady was perfect she knew when was her turn."* and *"the lady behaved more normal compared to the younger person."*

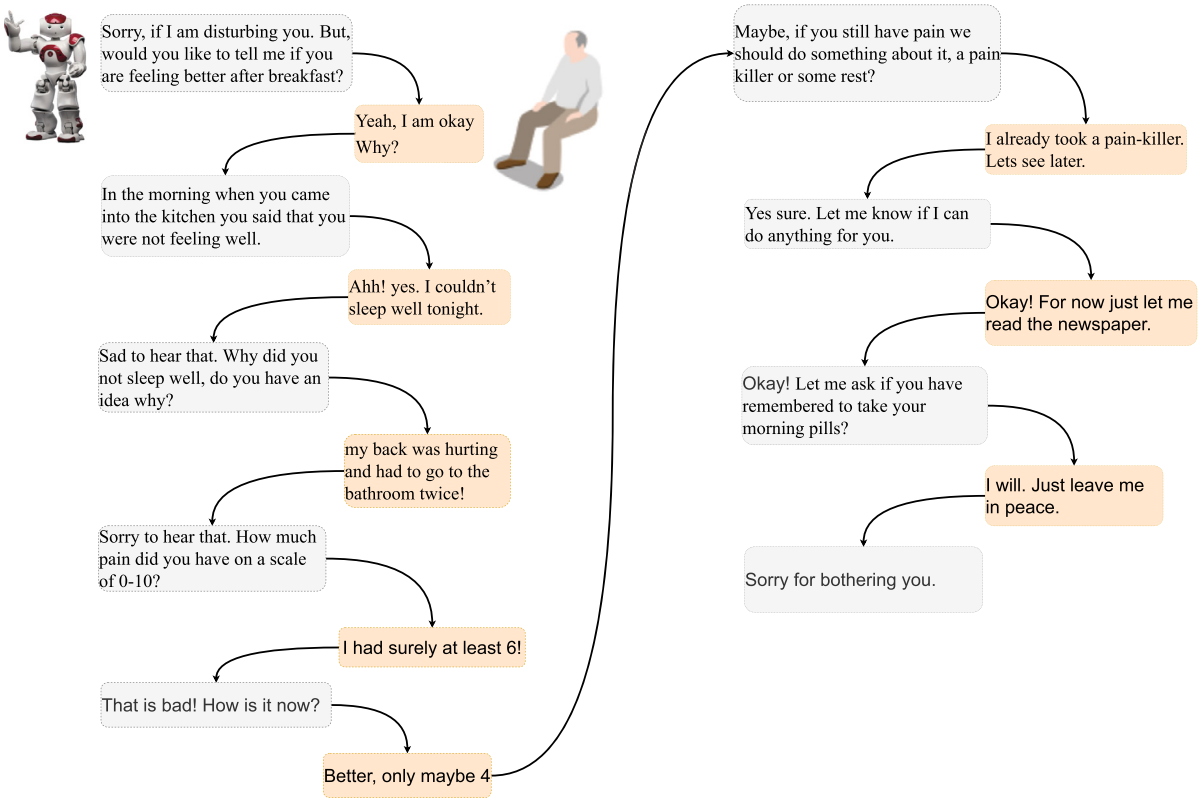


Fig. 4. Scenario depicting intention breakdown between the person and a social robot (based on Figure 5 in [121]).

Scenario 2 *(We-intention breakdown - the avoidant robot)*. **Preconditions:**

- The volunteer (person) is unaware of the social robot's dialogue or intention.
- The social robot is unaware of the volunteer's intention.
- The environment is static. No changes in the kitchen affect the location of the social robot.
- The social robot has pre-defined programs capturing a common health and well-being scenario.

The volunteer is reading newspaper in the kitchen. The social robot follows up on why the volunteer did not feel well in the morning. The volunteer responds that they did not feel well in the morning because of sleep issues due to back pain. The social robot asks to estimate the pain level. The volunteer indicates that the pain is high. Then, the social robot suggests doing something about it. The volunteer rejects it and asks the social robot to leave and let them continue reading their newspaper in peace (refer to Fig. 4 for details about the scenario).

Post-conditions:

- The volunteer rejects continuing with the topic of the dialogue.
- The social robot apologizes and ends the dialogue.
- The environment remains static.

In Table3 are presented selected lines of Algorithm 1 applied to the avoidant robot scenario. The pre- and postconditions show a breakdown scenario.

Participant's potential perception of We-intention breakdown and avoidant behavior:

A majority of the participants (n = 14) commented on how the volunteers caused We-intention breakdown such as being bothered or indicating they wanted to be left alone while watching TV or reading their newspaper; *"the man is a little upset.. he is upset because he wants to stay alone and watch his TV..."*. This breakdown situation was caused by the volunteers when telling the social robot to leave them alone (Fig. 4): *"at some point, they feel that it has been becoming too intrusive ah just asking too many questions... so they just ask it to leave them alone."* See also Table 3.

Most of the We-intentions breakdowns were observed in relation to the volunteers in the recordings, while only three participants observed a breakdown in the robot's We-intention. This was in the situation when it kept talking about sleep and pain while the volunteers were doing their daily activities, such as reading or listening to the TV news: *"When they say let me read in peace or just watch TV, and the robot ask questions that are not relevant of this situation."* Furthermore, one older participant suggested that even though the volunteers were annoyed about the social robot's continued talking, they should listen to it in order to solve their ongoing problems: *"maybe the robot reminded them about what they should do try to solve it even though they were annoyed."*

More than half (seven) of the young participants reflected on how the volunteers displayed avoidant behavior towards the social robot in different ways. Participants noticed the volunteers did not just avoid the social robot when they were occupied with other activities, but they also did not talk directly to it even when they were not busy, as illustrated by the following comments: *"You know when they were standing for breakfast in both the cases the robot was ignored."*; *"I think that they did not look the robot in the eyes... And did not talk directly to it. Sometimes, they looked at the newspaper, just did not look the robot in the eyes."*

On the other hand, only one older participant found the volunteers or the social robot displaying avoidant behavior. By contrast, six younger participants expressed how the social robot ignored the volunteer by either not responding to their introduced topic, avoiding the volunteer's requests illustrated by the comment: *"I have my pain here can you see?' But he never answered to that, never acknowledged that it could see or not, just keep going with the conversation."*, or persisting on its own topic: *"He was too persistent in describing the pain."*

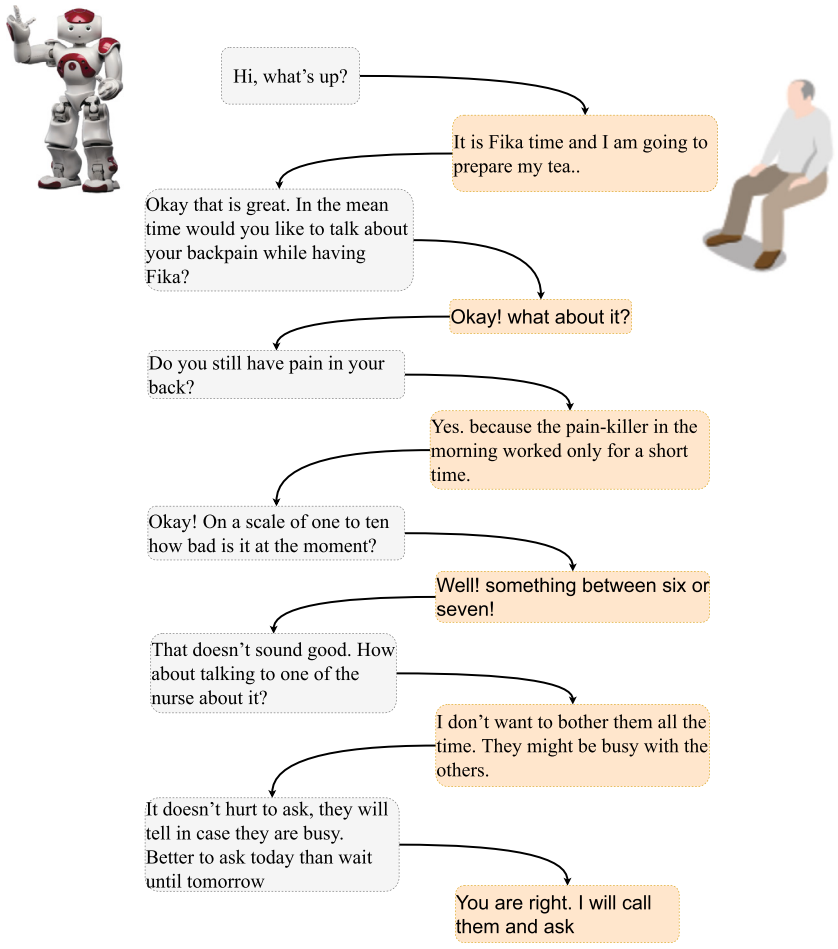


Fig. 5. Scenario depicting a partial alignment of intentions. In this setting, fragmented intentions are common between the volunteer and the social robot (based on Figure 7 in [121]).

Scenario 3 *(We-intention partial alignment)*. **Preconditions:**

- The volunteer (person) is unaware of the social robot's dialogue or intention.
- The social robot is not aware of the volunteer's intention.
- The environment is static. There are no changes in the kitchen that affects the location of the social robot.
- The social robot has pre-defined programs capturing a common kitchen scenario where health and well-being dialogues can be developed.

In the evening, the volunteer enters the kitchen and prepares tea/coffee for themselves. The social robot asks about the pain. The volunteer indicates they still have pain. Then, the social robot suggests they contact a nurse. The volunteer rejects the social robot's suggestion. The social robot provides a supporting argument for contacting the nurse. Then, the volunteer accepts, and the corresponding dialogue unfolds as depicted in Fig. 5.

Post-conditions:

- The social robot tries to convince the volunteer to see the nurse.
- The volunteer gets convinced and ends the dialogue.
- The environment remains static.

In Table 4, the output of selected lines of Algorithm 1 is presented considering as *input* an answer set program representing the intentions of a social robot and a person in Scenario 3. Summarizing, in line 18 of Table 4 the constraint `:- advise_call_nurse.` is added to avoid inconsistency in the set of intentions $\mathcal{I}$. In line 20, the plan generated by the Answer set programming mechanism suggests a coherent plan, avoiding potential intentions' breakdown.

Participant's potential perception of We-intention partial alignment: Most of the participants found the partial alignment of We-intention as natural. However, one older participant reported that after some initial disagreement, the older volunteer changed their attitude towards the robot: *"Both these people seem to be fine. The lady was less in a mood but I think she warmed up to the robot."*

4. State-of-the-art

This section follows a systematic literature review methodology to explore related work. We use a search-and-review well-established method introduced in [75] involving the following steps:

- Search questions (SQ) definition. We considered the following two search questions, SQ1: what formal mechanisms for sharing intention using Answer set programming or logic programming have been proposed for rational agents? and SQ2: what formal methods allowing rational agents to repair intentions have been proposed?
- Keyword selection. We used a set of keywords in different databases with minor syntactic changes, such as database-specific words. The keywords were the following: ALL = (INTENTION AND (SHARING OR SHARED OR JOINT) AND ("ANSWER SET PROGRAMMING" OR "LOGIC PROGRAMMING" OR "ANSWER SETS")).

Table 3
Results of a scenario where intention breakdown happens between a person and a robot. The left column is the Algorithm 1 selected lines. Right column shows the outputs of some control loop functions.

Line	Output of Algorithm 1 for the intention breakdown scenario
9-13	$F = \{$ morning;in_kitchen;has_newspaper;be_irritated; says_wants_be_alone;says_agree_take_pill;pain_pill_taken; says_feels_pain $\}$ $I_{Int} = \{$ read :- in_kitchen, has_newspaper. read :- says_wants_read. agree_take_pill :- says_agree_take_pill. be_alone :- says_wants_be_alone. feel_pain :- says_feels_pain, not pain_pill_taken. $\}$ $I_{Ext} = \{$ remind_pill :- morning, not pill_taken. advise_pain_killer :- feel_pain. ask_about_pain :- feel_pain. reduce_interaction :- be_alone. $\}$
14	BrkDwn({remind_pill,advise_pain_killer,ask_about_pain},{be_alone}) BrkDwn({remind_pill},{read}))
18	$\mathcal{I} = \{$ read :- in_kitchen, has_newspaper. read :- says_wants_read. agree_take_pill :- says_agree_take_pill. be_alone :- says_wants_be_alone. feel_pain :- says_feels_pain, not pain_pill_taken. advise_pain_killer :- feel_pain. ask_about_pain :- feel_pain. :- remind_pill. reduce_interaction :- be_alone. $\}$ $\mathcal{B} = \{$ morning;in_kitchen;has_newspaper;be_irritated; says_wants_be_alone;says_agree_take_pill;pain_pill_taken; says_feels_pain $\}$
20	π = {advise_pain_killer;ask_about_pain;reduce_interaction}

Table 4
Results of the use of Algorithm 1 in a partial alignment of intentions scenario. The left column is the Algorithm 1 selected lines. Right column shows the outputs of some control loop functions.

Line	Output of Algorithm 1 for a partial alignment of intentions scenario
9-13	$F = \{$ evening;in_kitchen;preparing_tea;has_newspaper; says_feels_pain;doubtful $\}$ $I_{Int} = \{$ prepare_tea :- in_kitchen, evening. feel_pain :- says_feels_pain, not pain_pill_taken. not_bother_nurses :- says_not_bother_others, doubtful. $\}$ $I_{Ext} = \{$ advise_call_nurse- feel_pain. ask_about_pain :- feel_pain. give_extra_information :- doubtful. $\}$
14	BrkDwn({advise_call_nurse},{not_bother_nurses})
18	$\mathcal{I} = \{$ prepare_tea :- in_kitchen, evening. feel_pain :- says_feels_pain, not pain_pill_taken. not_bother_nurses :- says_not_bother_others, doubtful. ask_about_pain :- feel_pain. give_extra_information :- doubtful. :- advise_call_nurse. $\}$ $\mathcal{B} = \{$ evening;in_kitchen;preparing_tea;has_newspaper; says_feels_pain;doubtful $\}$
20	π = {give_extra_information;ask_about_pain}

- Selection criteria definition. We consider the following selection and rejection criteria: 1) Approaches connected with non-monotonic reasoning, 2) articles with formal and empirical contributions, 3) approaches using models similar to the BDI, and 4) papers in the English language. Rejection criteria: 1) workshop papers and 2) articles published in less-recognized publishers and databases.
- Databases selection. We used the following article databases: Web of Science, IEEE Xplore, ACM Digital Library, SpringerLink, ACL Anthology, and Scopus.

Based on this methodology, we found 101 *potential* papers as follows: Web of Science (n = 16), IEEE Xplore (n = 1), ACM Digital Library (n = 75), SpringerLink (n = 3), ACL Anthology (n = 1) and Scopus (n = 4). We removed duplicates and papers not fulfilling the mentioned criteria, then we obtained a list of 16 papers.

Major findings We found that the work of Sakama et al. in [110] is the closest to our approach. In that paper, the authors studied specific semantics for two cooperation cases *i*) $\mathtt{AS}(P) \cup \mathtt{AS}(Q)$, and *ii*) $\mathtt{AS}(P) \cap \mathtt{AS}(Q)$ being P and Q two *extended disjunctive programs* [56] called *generous* and *rigorous* coordination respectively [110]. Sakama & Inoue's work

is focused on *belief coordination* rather than intention manipulation. We identified other approaches using ASP for *belief change* mechanisms [35,111], explained next.

For knowledge representation, answer set programs are rarely static; rather, modification by correcting, adding, or coalescing the programs is natural. This change can be studied and facilitated by belief-revision of the knowledge base. However, the non-monotonic nature of answer sets makes this change of beliefs difficult. Motivated by the limitation of answer sets in revising beliefs, the authors in [35] reformulate belief change similar to that in propositional logic by using strong equivalence characterized by SE models. Where strong equivalence gives rise to substitution principle, indicating that two strongly equivalent programs P and Q, $P \cup R$ and $Q \cup R$ have the same answer sets for any program R.

In [111], the authors formulate 'interactions' between answer sets in multi-agent systems, categorized by cooperation, competition, norms, and subjection. These interactions happen in an agent society using belief revision represented as answer set programs. These answer set programs were then defined for cooperation, where given two programs P_1 and P_2, $\Phi \subseteq Lit$, and answer set $S \in AS(P_1)$, $T \in AS(P_2)$, cooperation is when $S \cap \Phi = T \cap \Phi$. Other types of cooperation were defined as acceptance, adaptation, and concession. The competition was defined as $S \cap T \cap \Phi = \emptyset$ using the same programs and answer sets of cooperation. Other types of the competition were benefits and precedence. Similarly, norms and subjection were defined.

Our review showed that several approaches dealt with mental states in cooperative agents, for example, the work of Jennings, Wooldridge, Kinny, Dignum, Ancona & Mascardi, and Cohen (see [7,26,37,68,74,132]).

Jennings has defined models similar to We-intentions using the concept of commitment and conventions for distributed systems. The author defines commitment as a 'pledge to take a certain course of action' and convention to be 'a means for monitoring of commitment when circumstances change' for communities of agents [68].

Dignum et al. [37] provide a formal framework to cooperate and construct teams using argumentation theory dialogues types proposed by [129]. The authors define the framework as composed of the following: (1) potential recognition of agents by the initiating agent suitable for the overall goal and how they can be integrated with a team, and (2) team formation resulting in a collective intention to achieve the overall goal.

Kinny et al. [74] situate their work for agents with a repertoire of plans specifying goals that can be decomposed into sub-goals. A means/end analysis approach was taken, where an agent can select a plan from its repertoire instead of first generating plans from basic principles. To adopt a plan, an agent has a partial commitment or intention enabling stability in the presence of a dynamic environment. A formal framework for the planning of cooperative activities by MAS was provided. Their framework extended the beliefs with mutual belief about the environment and actions of the other agents; goals became joint goals and plans became joint plans as the means to satisfy joint goals; and, intentions were transformed to joint intentions as a commitment to joint plans. Their formal framework presents how agents as a collective can successfully achieve joint intentions. Furthermore, the authors also discuss possible failures during the execution of joint intentions arising from primitive actions, role plans, or changes in the beliefs, intentions, and goals of a team member.

Mascardi and Ancona [7] propose a cooperative BDI framework, an extension addressing the problem of events in BDI generating empty plans. Coo-BDI manages empty plans by defining *cooperativity* in agents by exchange of plans. Cohen et al. [26] define joint intention as *commitment* of a group (specifically, multi-agent systems (MAS)) to achieve joint actions. Agents with joint intentions were defined as dynamically situated with an incomplete or inconsistent set of beliefs, goals that can change, and actions that can fail. Furthermore, agents do not share their beliefs and goals, creating a necessity for communication. Such communication was facilitated by alternative definitions of speech acts such as request and assert [115]. The work was developed in [24,25] to allow people and agents to perform joint actions to achieve joint intentions. Specific to collaborative dialogues, authors in [24] redefine the general slot-filling mechanism with intents. In [25], a "collaborative multi-model planning based" dialogue system facilitating human-agent joint goals was presented. The dialogue system reasons about its own and others' goals and beliefs. Belief reasoning was performed by modal Horn-clause, and the planning mechanism considers speech acts as actions affecting the mental states allowing an agent to plan speech acts such as request, question, recommend, and those associated with emotions. The work in [103] applies a logic programming framework to enable agents with reasoning mechanisms to collaborate and participate during information-seeking dialogues. The framework represents actions, states, and knowledge with extended logic programs, and the reasoning mechanism is based on well-founded semantics with explicit negation. Authors in [97] provide "operational semantics" for updating agents' mental states when creating a theory of mind (ToM) in a MAS context.

Table 5
Approaches from the agents' literature where mental attitudes are shared for building cooperation.

Reference	Shared mental states	Formal language	Repairing
[13]	Actions, $\mathcal{B}$, plans	*AgentSpeak*$\mathcal{L}$	No
[98]	$\mathcal{BDI}$	Classical logic	No
[74]	Actions, $\mathcal{BDI}$, plans	Classical logic	No
[68]	$\mathcal{BDI}$	Classical logic	No
[26]	Joint commitments (intentions)	Classical logic	No
[60]	$\mathcal{BD}$	Logic programming	No
[7]	Plans	Propositional logic	No
[37]	Plans, pre-plans	Propositional logic	No

However, in these approaches mentioned so far, the analysis and formal representation of breakdowns (see Scenario 3) is not considered, apart from that of Kinny's work in [74] and Sarkadi's work [112], where, they applied a probabilistic method to represent uncertainty for agents when creating a theory-of-mind (ToM) about others.

Regarding SQ2, what formal methods for repairing intentions between rational agents been proposed? We only found one article using the keywords mentioned in those databases. In [135], the authors use CWA (Definition 8) to define restrictions in a knowledge base to maintain the *correctness* of specific atoms in a particular domain. Using answer set constraints for intention repairing is a novel approach for practical matters in multi-agent systems (MAS).

In Table 5 we summarize approaches related to our work where different mental states are shared. We found that the related approaches presented in Table 5 present models for cooperation or collaboration formation, but none of them have proposed computational mechanisms for *repairing* potential inconsistency and incompleteness of external mental states. Moreover, the use of classical logic as an underlying formalism of representation is common (see *Formal language* in Table 5). Contrary to these models, we use extended logic programs to handle uncertainty in the shared mental states, additionally, we addressed the problem of repairing cooperation depending on the agent's attitude, which has been addressed with the use of AgentSpeak-$\mathcal{L}$ in [13], and extended logic programs in [61].

5. Discussion

In this paper a formal computational mechanism is presented, which equips social robots with the capabilities to form a We-intention with a person, also in the presence of incomplete and inconsistent intentions. The mechanism is exemplified in three dialogue scenarios involving a person and a social robot. These dialogue scenarios were enacted in a WoZ setup and participants observed them. In a semi-structured interview after observing the scenarios, participant perspectives on the three different types of We-intention alignments were gathered and analyzed.

The following subsections discuss the results, limitations of the methods applied, and compared with the related work.

5.1. Sharing and repairing intentions in the agent literature

In the agent literature, specifically in approaches using cognitive architectures of agents we found three major trends:

1. Mental states are shared between agents without handling uncertainty in the information representation. Our results suggest (Table 5) that most of the formal approaches for sharing mental states between agents use classical logic variants (see [26,68,74,98]), which are comparatively less rich to capture uncertainty than approaches using ELP, or AgentSpeak-$\mathcal{L}$ (*e.g.* [103,13,60,109]). Consequently, some computational mechanisms cannot address issues related to the formation and reparation of We-intentions under incomplete and inconsistent information from other agents.
2. The use of the Belief-Desire-Intention (BDI) model introduced in [16] is ubiquitous in the agents' literature. Our review found that most papers in shared intention and intention formation consider the BDI model as a key for the agents' decision-making process (see [13,60,68,74]). Therefore, they propose control loops with similar characteristics as of Algorithm 1; for example, they started with initial empty sets of intentions and plans, the control loops are always active receiving *percepts*, and the main intention and planning generation is based on an algorithmic BDI manipulation. In this sense, our proposed algorithm has the advantage that it can be deployed in frameworks such as JaCaMo [14]. However, at the same time, such control loops require further consistency analysis to avoid *no-rational* decisions, for example, when an agent We-intends to do X with others. However, it does not believe that it can achieve it. In this sense, Bratman in [16] showed that there is no *irrationality* if a person intends to do X, and yet does not believe that s/he will do it, this can be labeled as *intention-belief-incompleteness*. However, it can be considered *irrational* to intend X and believe that s/he will *not* X, the so-called *intention-belief-inconsistency* [16, p. 38-39]. Recently, in the agents' literature, some authors have considered *rationality principles* as key for agent-based persuasive technology [62].
3. Reparation of intentions in joint activity formation is not considered in the agent literature. As our review suggests, formal processes for modifying the structure of intentions before they are assimilated or rejected by an agent have not been proposed. Most approaches in sharing mental states literature assume two distinct options, *full acceptance* or *full rejection* of external intentions. However, as our partial alignment scenario suggests, there exist settings where an agent may partially adopt and reject intentions from other agents. Consequently, our formal mechanism to deal with partial scenarios can be seen as a novel mechanism and applicable for different scenarios in person-robot cooperation.

Furthermore, based on the review results, we found that *intention sharing* is an active and well-established research track in the agents' community but not in the answer set and logic programming fields. In the agents' field, authors from social sciences have inspired several formal computational mechanisms, such as the work of Searle, Bratman, and Tuomela (see [17,114,128] among others). Our framework follows the seminal work of Bratman, in [16] to define a reasoning *control loop*. Tuomela's work on *We-intentions* is particularly close to our approach. Tuomela defines a We-intention as an "attitude" required to have a joint intention. Tuomela in [125] described two types of attitude, one aiming to achieve a mutually agreed *group* goal, as can be illustrated by our *agreeing agent*, or motivated by *private* goals illustrated by our *avoidant agent*.

In [110], authors explore collective semantics using ASP, contrasting to our approach, we did not consider preferences, leaving open the notion of selection as a function `SEL` in Algorithm 1. Furthermore, the novelty of our approach compared to other approaches such as [7,26,37,68,74,132] lies in dealing with breakdowns of mental states such as intentions by analyzing and considering the conflicts and partial alignment of We-intention in our control loop mechanism.

5.2. Formalizing We-intentions

In Section 3.1, our mechanism aiming to formalize a We-intention formation under uncertainty was presented as a major theoretical contribution of this paper.

The mechanism differs in two aspects from traditional agent literature, specifically, those based on the BDI model [16,106]. First, it considers intentions from other agents (which is not performed in most of the BDI approaches). Second, it provides practical mechanisms to accept intentions to allow the execution of joint activity. Then, we can argue that it follows a *mirror model* of agency [101] where agents share knowledge and task agreements require simultaneous, mirrored mental models.

Despite the differences between our framework and others, we found a common characteristic among ASP approaches that we shaped in Proposition 2 and Theorem 1, in which all the ASP-based methods to reason about mental states always generate a consistent output. These results align with the ASP literature and highlight the importance of our framework and this line of research.

Regarding the two knowledge representation mechanisms used to integrate and isolate specific intentions as sets of atoms, Proposition 3 and Proposition 4 established the two main processes used in Algorithm 1 to deal with potential breakdown situations, *i.e.* inconsistencies in the intentions set $\mathcal{I} \subseteq I_{Ext} \cup I_{Int}$. We know that such transformations cannot be generalized to other underlying formalisms different from ELP and similar, where negation as failure is considered. Nevertheless, we can see that our framework can be implemented in well-established platforms for agent design, such as JaCaMo.

5.3. Repairing We-intentions in a person-robot healthcare scenario

In dialogues between a person and a social robot, cognitive capabilities are required to manipulate, reason with, and mediate mental objects such as beliefs, desires, and intentions. Furthermore, a social robot or agent is expected to manifest appropriate behavior and situated knowledge in a social situation. Inherent to social encounters is the process of negotiation about a shared view on a situation, where knowledge needs to be updated. In this work, we focus on managing the alignment, breakdown, and partial alignment of intentions between a person and a social robot. Whereas previous approaches [88,8] have been focused on alignment and breakdown scenarios while disregarding a partial agreement to perform a joint activity.

In HRI literature, several theoretical and empirical approaches have been proposed to manage breakdowns, which are seen as "errors" in the social robot's behavior; for example, while the authors in [113] identify and manage perceptual errors of the robot, the work by [90] addresses grounding problems and impossible actions in navigational tasks involving robots and people. In a similar scenario of task-oriented HRI, the author [108] identifies a robot's errors by comparing its behavior with social signals of people.

In contrast, Human-centric AI shifts the focus from designing and conceiving agents as task-fulfilling machines to those that collaborate, enhance, and empower people to achieve their goals [96]. A fundamental challenge in human-centric AI is to capture and manage the understanding, more specifically, establish semantic grounding and alignment within the context of a person's action (e.g., [119]). To address this challenge, there is an increasing focus on how to capture different levels of interpretation of information in relation to the social context [28].

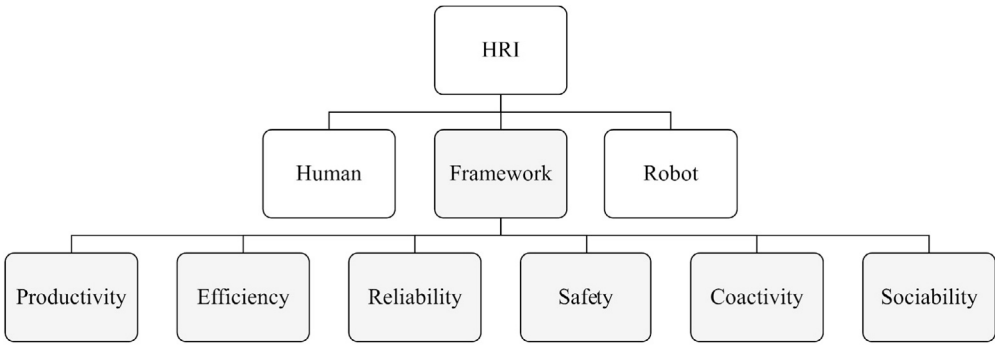


Fig. 6. Taxonomy of metrics for evaluating implementations of HRI instances using our framework.

For evaluation purposes in this paper, HRI scenarios were crafted by employing the activity-theoretical framework, which provides a multi-layered and systemic view on human activity [12,83,84].

The crafted scenarios were enacted in a WoZ setup. WoZ setup was chosen for the following reasons: (1) it facilitates prototyping and rapid development of complex scenarios [11]. (2) allows for compensating technical limitations of robots to perceive and conduct joint activities in natural and dynamic environments of the human, and (3) in support of this method, researchers have provided criteria to set up these studies and guidelines to report them within the HRI field [107].

Design implications guided how to create such an understanding of meaning [120], which was further extended into a layered taxonomy for 'understanding' [121].

This work addresses the fourth design implication in [120] relating to co-creating goals and shared intention, and the proposed strategies, expanded in [121] to create knowledge about We-intentions and manage associated conflicts.

The activity-theoretical definition of focus shift was applied [12,46], which explains how people move the focus from the *main activity* with an 'objective' driven by a need to, for instance, to an *alternative activity* about the tool being used, the social norms or regulations being employed, or other components of the activity system [46] as illustrated in [121]. In the case of agents, such focus shifts can be caused when agents pursue and prioritize their own intention, represented in this work as internal 'I-intention', which may cause conflict or motivate partial agreement of We-intention. Therefore, three behaviors enacted in the scenarios presented in [121,120] were used to exemplify the theoretical construction of managing focus shifts during formation of We-intention under uncertainty. Managing focus shifts consisted of employing avoidant behavior, partial alignment of We-intention, and alignment of We-intention, respectively.

The participants were observed to attribute most of the We-intention breakdowns and avoidant behavior to the volunteers in the scenarios. Human's avoidant attitude towards the robot was commented on, the general view was that such behavior is acceptable.

We observed that participants had few comments about the partial alignment of We-intention. One interpretation that could be derived is that participants perceived such cases of We-intention as natural and suggested the agent adapts to the human's trail of thinking. If so, a consequence could be that partial alignment could be interpreted as "good enough" cases of repairing perceived conflicts. However, further studies are needed to better understand how partial alignment of We-intention can be formed and expressed between people and social robots.

Participants' recommendations to adapt and improve social robot's behavior

Some older and younger participants provided suggestions to improve the social robot's agreeing behavior. For example, the social robot can deliberate about health issues such as sleep, physiological and psychological well-being. Preferably, with a slower pace, embedding some chit-chatting in between, and choosing to interrupt when the person is either unoccupied or transitioning between activities.

Recommendations were also provided on how a social robot can manage conflict of We-intention or when the person displays an avoidant behavior. In such situations the social robot should disengage or halt the dialogue and return at a later time.

Evaluation of We-intention-based systems

Theoretical frameworks as presented here, provide the foundations for designing real-world systems, however, the evaluation of their performance, robustness, human impact, etc. can be a challenge given that such assessment requires specific context-based variables to be evaluated. In the human-robot interaction literature, different metrics have been proposed to assess HRI applications such as the work in [134] where safety bounds and heuristics were proposed. In [27] the authors reviewed methods for evaluating *quality* in the human-robot interaction where a classification of safety metrics was proposed, and in a similar way to the taxonomy of HRI metrics proposed in [95].

Regarding human-robot collaboration, other articles have proposed metrics to evaluate the type of interaction, for example, the paper in [63] where HRI communication metrics were presented, based on how a technology conforms to specific HRI communication tasks considering the following categories: the extent of usage, flexibility, duration, among others. In the same line of research, the work presented in [29] reviewed common metrics for HRI and team work.

Some of the aforementioned metrics for evaluating HRI implementations can be adapted to assess our framework. In this sense, we extended the work presented in [95] to propose a five-dimension metric to evaluate our framework. The five dimensions are *productivity, efficiency, reliability, safety, co-activity,* and *sociability* (see Fig. 6).

In this proposed taxonomy, we considered that metrics for assessing effectiveness, productivity, task difficulty, and time operations are evaluations of the *productivity* of a system. In our formal framework, a productivity metric can be the measurement of the joint activity achievement considering the joint intention.

Efficiency, in our framework can be evaluated considering the competence for human-robot interaction, and the time to complete a joint task. These tasks depend on the implementation of Algorithm 1, specifically the mechanisms for generating the intentions and plans. In this sense, we presented in Appendix B some potential mechanisms to improve the efficiency of our approach.

Reliability metrics, in general, are oriented to evaluate the robustness of an HRI system [95]. In our setting, the number of intention discrepancies can provide a metric for evaluating how reliable is the cooperation mechanism proposed in our algorithm.

Safety is a dimension that requires an assessment of the context, and intentions of a person in HRI. In our framework, safety considerations were considered during the WoZ experimentation, being this methodology a well-established process for evaluating HRIs in controlled settings.

Co-activity is a metric to evaluate the cognitive state of the robot and the interaction [95], in this sense, the evaluation of joint activities with and without a robot (or agent) could provide a benchmark for evaluat-

ing frameworks such as the proposed in this paper. The evaluation of cognitive states in an HRI setting is part of our future work.

Finally, the *sociability* dimension of an HRI could be evaluated by considering not only the achievement of joint activities but the reactions and responses of a person during and after the joint activity with a robot or an agent. The robotics literature have recently focused on those qualitative approaches to evaluate such dimension, see for example the review presented in [33,76].

5.4. Limitations

Regarding our formal framework in this article, we are aware of three main limitations: 1) *computation complexity*: we have not analyzed the complexity and tractability of the Algorithm 1 proposed here, we would like to extend this work with such evaluation; 2) *User model and Theory of Mind*: our framework requires a dynamic representation of the person and the social robot; and 3) *activity representation*: a dynamic representation of the complex joint activity enacted in the collaboration is needed. In order to repair We-intention, also mechanisms for co-constructing shared knowledge about the situation are required [120].

There are limitations associated with exemplifying our scenarios and qualitative results presented in this work. Primarily, the scenarios illustrating breakdown, partial, and full-alignment of We-intention were selected from a previous user study with participants' perception of dialogues conducted between volunteers and a social robot [120,121] in a WoZ setup. These scenarios provide only an exemplification and a third-person view of our proposed formal computational mechanism. Such limitation presents a future work to implement and evaluate the formal mechanism proposed here with HRI studies.

The empirical findings presented in this work are potential comments of future users based on qualitative analysis of a small sample (20) of interview transcripts. Such empirical findings have the limitation of being less rigorous due to dependence on the researcher's interpretation and are usually based on a small data set. Furthermore, these empirical results are difficult to generalize and reproduce. Future work in this regard aims to strengthen the results presented here by taking a quantitative research approach.

6. Conclusions and future work

In this paper, we addressed the challenge of *aligning* the intentions of two agents (for example, a person and a social robot) when they try to achieve a joint intention.

We generalized the problem involving two agents aiming to provide a formal framework to be used as a descriptive mechanism of different interaction types, for instance, in a patient-physician dialogue, between two social robots, and in a healthcare setting involving a patient and a social robot. Our starting point was a well-established theory of collective intentions presented in the work of Tuomela et al. [125,128], stating that when two agents act aiming to achieve a joint goal, they require sharing a relevant group intention, the so-called *We-intention*, which differs from the internal intentions named *I-intention*. Therefore, this paper's primary challenge was establishing a general computational mechanism to represent alignments, breakdowns, and partial alignments of intentions.

We presented a formal computational mechanism for knowledge representation and reasoning, enabling a social robot with three desired capabilities:

- Assessing potential inconsistent intention, *w.r.t.* internal and joint (external) intentions of the agent.
- Repair breakdowns and incomplete external intentions in order to resolve inconsistent mental states, and
- Make a plan using the repaired (if necessary) joint intention.

Two major formal contributions we made in this paper to the agent community:

1) a well-defined formal framework for sharing intentions between agents (including person agents) with desirable practical characteristics such as: *i*) it extends from the BDI model using a standard syntax, implying their implementation in already existing agents' platforms (*e.g.* JaCaMo [13]); *ii*) it handles uncertainty and incompleteness of the agents' mental states (when sharing); and *iii*) the process for joint intention generation can be extended to other mental states, *e.g.* beliefs and desires.

2) Our repairing mechanism deals with well-known issues of agent cooperation, endowing an agent with the capability to behave as an agreeing, avoidant, or partially agreeing agent, meaning that it can decide if it accepts entirely, restricts, or partially accepts potential joint intention.

The framework was exemplified in three scenarios illustrating the three behaviors, and participants' perceptions of volunteers and the social robot's behaviors were analyzed. A particularly interesting observation was that very few occasions of partial alignment were commented on by the participants. Partial alignment of intentions has been disregarded in the software agents field and overlooked by the original work of Tuomela. The empirical findings in this work indicate that this approach could provide an acceptable level of agreement from the perspective of the human, which will be studied in future work.

The generality of the proposed framework, its computational cost, and the adaptability of the introduced control loop make our contributions suitable to apply to other scenarios than human-robot joint activities, for example, using Algorithm 1, *i.e.intention adaptable agents* to improve the personalization in human-agent joint activities, such as it was shown for financial literacy coaching systems [58].

Additional future directions can be explored. First, theoretically exemplifying the control loop and answer set programs to other scenarios mentioned in [121]. Other future work would relate to implementing the theoretical work done in this article to facilitate joint activities between people and robots. Finally, a further theoretical formalization can be done to represent knowledge about emotions, mood, norms, and rules on how to act, and the distinction between facts and procedures to provide mechanisms to also co-create shared knowledge about a situation, which would further equip an agent with strategies to manage and repair We-intention in human-agent collaboration.

Declaration of competing interest

The authors of this article declare that have no competing financial interests or personal relationships that could have influenced the work reported in this article.

Acknowledgements

The authors from the University of Vaasa, Guerrero, and Kalmi, were partially funded by two projects: 1) Digiconsumers, supported by the Academy of Finland Strategic Research Council, project number: 327241, and also the European Regional Development Fund (A75418), and 2) the InvestiGame project (Improving Financial Literacy and Pro-Environmental Choices Using Game-Based Pedagogy and Virtual Reality), supported by the Nasdaq Nordic Foundation. Research conducted by Tewari and Lindgren was partially funded by Marcus and Marianne Wallenberg Foundation (Dnr MMW 2019.0220) and by The Humane-AI-Net excellence network funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 952026.

Appendix A. Proofs

Proof (Consistent shared intentions). Proposition 2. Let $S \subseteq \mathtt{AS}(P_2)$ be the set of intentions of Ag_2, it follows that a Gelfond-Lifschitz transformation [56] of P_2, *i.e.* P_2^M for any set M of atoms from P_2 where

deletion of 1) each rule that has a *not* literal in its body, and 2) all negative literals in the bodies of the remaining rules, then we say that P_2^M is negation-free, and its stable model is a minimal *Herbrand model* of P_2. In an *indirect proof* (by contradiction), we need to prove that under this transformation, there exist two atoms $x, x' \in P_2^M$ conflicting, *i.e.* $\texttt{BrkDwn}(x, x')$ (see Definition 4). $\texttt{BrkDwn}(x, x')$ implies that $x \not\equiv_{sem} x'$ where $\not\equiv_{sem}$ implies a semantic difference. So, this implies that one of the atoms x' or x is a negative literal, which contradicts the initial Gelfond-Lifschitz transformation. If x, x' are intentions of different agents, *e.g.* Ag_1, Ag_2 then we can say that under Gelfond-Lifschitz transformation the shared intentions are consistent.

Proof (Consistent shared states). Let us assume that $S \subseteq \texttt{AS}(P_2)$ is a set of mental states, *e.g.* beliefs, desires, or intentions of a given agent. Then by using an AS function corresponding to the Gelfond-Lifschitz transformation [56] of P_2, *i.e.* deleting each rule that has a *not* literal in its body, and all negative literals in the bodies of the remaining rules, then we say that P_2^M is negation-free and its stable model is a minimal *Herbrand model* of P_2. By contradiction, we need to prove that there exist two mental states $x, x' \in P_2^M$ conflicting, *i.e.* $\texttt{BrkDwn}(x, x')$, which contradicts the initial Gelfond-Lifschitz transformation.

Proof (Incompatible atom elimination using CWA). Let us assume that $\texttt{AS}(P_2) = S$ (part of agent Ag_2) is a consistent set of literals. We want to show that there is no $x \in S$ that is inconsistent with a P_1 after a transformation CWA. Recall that a consistent set S of literals is an answer set for a disjunctive program without negation as a failure if and only if it is a minimal set closed under this program. And given that a CWA transformation removes potential syntactic incompatibilities, then $S \cup P_1$ is consistent, from the perspective of Ag_1.

Proof (Restricting atoms with constraints). To prove this we need to show that by adding a constraint as a product of trying to accept an intention of mental model, *i.e.* the constrained atom added to a program, the resulting program under an answer set evaluation will not be altered. In other words, let $\bot \leftarrow x$ be a constraint (e.g., an external intention), then when it is added to the set of intentions S of the other agent, then the AS evaluation remains equal. We can assume that $\texttt{AS}(S) = \{T\}$ is a consistent set, then by adding the constraint $S \cup \{\bot \leftarrow x\}$ we have that $\texttt{AS}(S \cup \{\bot \leftarrow x) = \{T'\}$, so we have to prove that T and T' are the same. It is straightforward to see that AS implies the elimination of rules that have empty heads, among others under the Gelfond-Lifschitz transformation [56], so $\nexists x \in T'$ implies that $T = T'$.

Appendix B. Computational complexity theory in logic programming

In this section, we present a background and the associated definitions for the analysis of the computational complexity of logic programming. We follow the notation introduced in [30,31,99].

B.1. Complexity classes

In computational complexity theory, a Turing Machine, informally, is a hypothetical device that has the ability to make correct guesses and describes an abstract machine that manipulates symbols on a strip of tape according to a table of rules. Formally, a *deterministic Turing machine* (DTM) is defined as a quadruple (S, Σ, δ, s_0), where S is a finite set of *states*, Σ is a finite alphabet of *symbols*, δ is a *transition function*, and $s_0 \in S$ is the *initial state*. The machine takes successive steps of computation according to δ. There are three additional states halt, yes, and no that are not in S. Assume that a DTM is in a state $s \in S$ and the cursor points to the symbol $\delta \in \Sigma$ on the tape. When any of the states halt, yes, or no is reached, DTM halts. We say that DTM *accepts* an *input I* if DTM halts in yes. Similarly, we say that DTM rejects the input in the case of halting in no. If a halt is reached, we say that the output of DTM on I is *computed*. This output, denoted by DTM(I), is defined as the string contained in the initial segment of the tape which ends before the first blank [31]. In contrast to a DTM, a *nondeterministic Turing Machine* (NDTM) defined as a quadruple (S, Σ, Δ, s_0), where S, Σ, s_0 are the same as a DTM. However, the possible operations of this machine are described by Δ, which is no longer a function, but is given by the expression:

$$\Delta \subseteq (S \times \Sigma) \times (S \cup \{\texttt{halt}, \texttt{yes}, \texttt{no}\}) \times \Sigma \times \{-1, 0, +1\}$$

The *time* expended by a DTM named T on an input I is defined as the number of steps taken by T on I from the start to halting. If T does not halt on I, the time is considered to be infinite. For an NDTM T, we define the time expended by T on I as 1, if T does not accept I, and otherwise as the minimum over the number of steps in any accepting computation of T. The *space* required by a DTM on I is the number of cells visited by the cursor during the computation on I. In the case of an NDTM, the space is defined as 1, if such machine does not accept I, and otherwise as the minimum number of cells visited on the tape over all accepting computations [31]. In a DTM or an NDTM named T, let f be a function from the positive integers to themselves. We say that T halts in time $O(f(n))$ if there exist positive integers c and n_0 such that the time expended by T on any input of length n is not greater than $cf(n)$ for all $n \geq n_0$. Likewise, we say that T halts *within space* $O(f(n))$ if the space required by T on any input of length n is not greater than $cf(n)$ for all $n \geq n_0$, where c and n_0 are positive integers. If T halts within space $O(n^d)$, where d is a positive integer, then we call T a *polynomial-space* DTM or NDTM [31]. Different languages can be defined based on these definitions; for space complexity, we can have:

$$TIME(f(n)) = L | L is decided by some DTM in time O(f(n)), \tag{2}$$

$$NTIME(f(n)) = L | L is decided by some NDTM in time O(f(n)), \tag{3}$$

$$SPACE(f(n)) = L | L is decided by some DTM within space O(f(n)), \tag{4}$$

$$NSPACE(f(n)) = L | L is decided by some NDTM within space O(f(n)) \tag{5}$$

Complexity classes of most interest are not classes corresponding to particular functions but their unions, and then some complexity classes can be derived from these languages [31]:

$$P = \cup_{d>0} TIME(n^d), \tag{7}$$

$$NP = \cup_{d>0} NTIME(n^d), \tag{8}$$

$$EXPTIME = \cup_{d>0} TIME(2^{n^d}), \tag{9}$$

$$NEXPTIME = \cup_{d>0} NTIME(2^{n^d}), \tag{10}$$

$$PSPACE = \cup_{d>0} SPACE(n^d), \tag{11}$$

$$EXPSPACE = \cup_{d>0} SPACE(2^{n^d}) \tag{12}$$

$$L = SPACE(log\ n) \tag{13}$$

$$NL = NSPACE(log\ n) \tag{14}$$

For every language L in σ', let L denote its *complementary class*, that is, the set $\sigma'^* \setminus L$, then co-$C = \{L | L \in C\}$ where C is any complementary class [31]. The polynomial space (and time) class also has different sub-classes. However, they are established in terms of *oracle Turing machines*; informally, an oracle Turing extends the capabilities of DTM (or NDTM) by providing it with a *black box* (the oracle) that can instantly solve some computational problem. The oracle is typically represented as an additional tape and states allowing the machine to *query* the oracle and receive an answer. The *polynomial hierarchy* consists of classes δ_i^p, σ_i^p, and Π_i^p defined by the following equalities [31]:

$$\delta_0^p = \delta_i^p = \delta_i^p = P \tag{15}$$

$$\delta_{i+1}^p = P^{\sigma_i^p}, \tag{16}$$

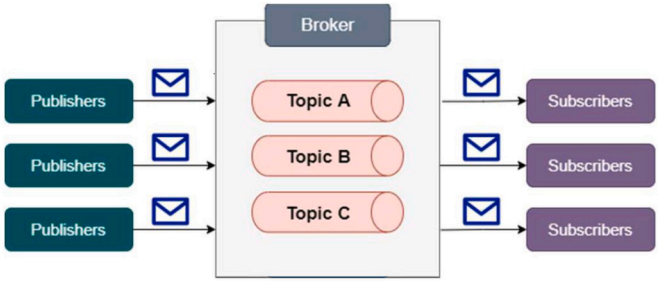


Fig. 7. Typical architecture of a publish-subscribe middleware (broker) allowing the subscription of specific topics. Adapted from [77].

$$\sigma^{p}_{i+1} = NP^{\sigma^{p}_{i}}, \tag{17}$$

$$\Pi^{p}_{i+1} = co - \sigma^{p}_{i+1} \tag{18}$$

for all $i \geq 0$.

B.2. Alternatives to reduce the computational cost of BDI-like control loops

Algorithm 1 follows a well-established control loop sequence where the specific internal functions can be implemented in different ways, leading to various computational performances. In the social robot s literature, practitioners have proposed several heuristics to decrease the computational cost of mechanisms such as generating answer sets. However, other specific functions require tailoring their implementations to decrease the time response or the computational requirements of the software. In this section, we present heuristics for decreasing the computational cost of some functions of Algorithm 1 and implementation alternatives for others. We limit our attention to those related to the core of this paper, leaving the computational cost analysis and their heuristics of functions such as **cooperate**() (Line 18) and **plan**() (Line 20) out of the goal of the paper, and because they require background information of the specific environment where a heuristic will be implemented.

This is not an exhaustive list of approaches, but it can be seen as a guide for practitioners that need to implement our framework.

- **getFacts**() (Line 9). In Algorithm 1 for obtaining facts from other agents or the environment, the function getFacts() can be reduced from a computational complexity theory perspective using several heuristics; we highlight in the following some potential alternatives that have been presented in the social robot s literature.
  - *Publish-subscribe paradigm*: publish-subscribe architectures are commonly used in the social robots literature, specifically for agents with no cognitive architectures, for example in agent-based wireless sensor networks [21,36], mobile agents [67,78], robot management [91], etc. A publish-subscribe paradigm in our algorithm can be implemented using a *middleware* platform where all the communications among agents and the environment can be managed. In these architectures, agents can be *publishers* or/and *subscribers* of a specific *topic*, for example, a health-oriented agent or robot should obtain information related to the health condition of a person from *information channels* or other agents that are publishing data on that specific topic, see Fig. 7 as an example of such paradigm. In this setting, by using this paradigm, Algorithm 1 will obtain only related facts.
  - *Logic-based representation language for communication protocols*: when two formal (logic-based) agents communicate with each other, they need to use specific *protocols* to establish the legality of their *utterances* that are specified in terms of their mental states. Different formal specification of protocols has been proposed in the literature (*e.g.*, [45,92]), and partially extended in the well-established FIPA[2] protocol. Using such formal mechanisms for communication protocols, our Algorithm could only obtain information from channels that use such protocols.
- **update**() (Line 10). Updates of logic programs is an active research line that has a long history since the early definition of what a social robot should be (see review [79]). In Algorithm 1, the update() function behaves as a program update, in terms of [70,71,131] where is typically defined as an operation that brings a knowledge base up to date when the world described by its changes, whereas a *revision* is typically described as an operation that deals with incorporating *new better knowledge* about a world that did not change [79]. In this sense, several heuristics can be implemented to decrease the computational cost of update() considering the specification language and the semantics. Different update and belief *operators* may work to implement update(); however, the syntactic approach that we suggested may limit the use of some semantics. A major group of semantics use the *causal rejection principle* [3,80] that says: "a rule should be rejected when a more recent rule directly contradicts it". Our Algorithm uses this principle to establish potential intentions contradictions. Several semantics have been proposed that follow this principle, with different complexity levels that depend on the specification language and the type of update. While some follow the *belief update tradition* and construct an updated program given the original program and its update, others only assign a set of stable models to a pair or sequence of programs where each represents an update of the preceding ones [79]. We cannot provide a detailed account of every update semantics here. However, we suggest to the reader a review by Leite et al. in [79] for a formal introduction to those semantics. Therefore, as a practical *rule of thumb* to select a heuristic to decrease the computational cost of update(), the experimenter should consider the need for *rich* language specifications, *e.g.,* using logical disjunction, preferences, strong/weak negation, etc., which impact in the semantics complexity and the associated computational cost.
- **wish**() (Line 10). In goal-based reasoning agents, forming a new goal implies the creation of a new goal and initiating the so-called *Goal Lifecycle* [1]. In the formal argumentation literature, several mechanisms for the generation of non-conflicting desires/goals have been proposed considering classical logic [6,5], and also (extended) logic programming [59]. Heuristics for decreasing the computational costs of such argumentation-based mechanisms are linked to selecting suitable argumentation semantics. However, as it is well-known, such selection is impacted by several aspects, such as the richness of the underlying language.
- **intend**() (Line 12). In our approach, the generation of intentions is given by an answer-set process. The notion of an answer set for *extended logic programs* is a generalization of the concept of a stable model introduced in [56]; as for the complexity, there is no increase for extended logic programs over logic programs under the stable model semantics [30].

Theorem 2 *([10]). Given a propositional extended logic program P, deciding whether P has an answer set is NP-complete, and extended propositional logic programming is co-NP-complete.*

In this setting, the `intended` function can be hardly decreased its computational complexity if the Gelfond-Lifschitz definition is followed. However, some answer-set solvers (*e.g.* DLV,[3] and the Potassco tools,[4] among others) have implemented different heuristics to improve the performance of answer-sets calculations.

[2] Foundation for Intelligent Physical Agents (FIPA). Communicative Act Library Specification, 2002. http://www.fipa.org/specs/fipa00037/.

[3] DLV System Web page https://www.dlvsystem.it/dlvsite/dlv/.

[4] Potassco Web page https://potassco.org/.

In the literature have been introduced different heuristics, *e.g.* domain-specific in [40,54], improvements using parallel optimization [15,53], enhance ASP encoding [55], look-back heuristics [57,94], machine learning-based solver configurations [65,89], among others. See reviews [31,49].

– **repairCooperation**() (Line 16). In our approach, the three proposed configurations for modifying a program require a *search mechanism* to find potential intention conflicts, then the suggested modifications can be performed. In the literature, several heuristics have been proposed to improve the performance of such types of searches. However, the heuristic depends on the type of program and its extension. When stratified programs are used, stratified tree search algorithms can be used (see [81]), and when programs are large other different heuristics can be applied, such as Monte-Carlo [22] and trial-based heuristics [72].

References

[1] D.W. Aha, Goal reasoning: foundations, emerging applications, and prospects, AIM Mag. 39 (2) (Jul 2018) 3–24, https://doi.org/10.1609/aimag.v39i2.2800.

[2] C.E. Alchourrón, P. Gärdenfors, D. Makinson, On the logic of theory change: partial meet contraction and revision functions, J. Symb. Log. 50 (2) (1985) 510–530.

[3] J.J. Alferes, J.A. Leite, L.M. Pereira, H. Przymusinska, T.C. Przymusinski, Dynamic updates of non-monotonic knowledge bases, J. Log. Program. 45 (1–3) (2000) 43–70.

[4] J.J. Alferes, L.M. Pereira, Update-programms can update programs, in: Non-Monotonic Extensions of Logic Programming: Second International Workshop, NMELP'96 Bad Honnef, Germany, September 5–6, 1996 Selected Papers 2, Springer, 1997, pp. 110–131.

[5] L. Amgoud, C. Devred, M.C. Lagasquie-Schiex, A constrained argumentation system for practical reasoning, in: Argumentation in Multi-Agent Systems, Springer, 2009, pp. 37–56.

[6] L. Amgoud, S. Kaci, On the generation of bipolar goals in argumentation-based negotiation, in: International Workshop on Argumentation in Multi-Agent Systems, Springer, 2004, pp. 192–207.

[7] D. Ancona, V. Mascardi, Coo-bdi: extending the bdi model with cooperativity, in: Declarative Agent Languages and Technologies, 2004, pp. 109–134.

[8] P.E. Baxter, J. de Greeff, T. Belpaeme, Cognitive architecture for human–robot interaction: towards behavioural alignment, Biol. Inspir. Cognit. Archit. 6 (2013) 30–39.

[9] H. Beck, M. Dao-Tran, T. Eiter, Answer update for rule-based stream reasoning, in: IJCAI, Citeseer, 2015, pp. 2741–2747.

[10] R. Ben-Eliyahu, R. Dechter, Propositional semantics for disjunctive logic programs, Ann. Math. Artif. Intell. 12 (1994) 53–87.

[11] N.O. Bernsen, L. Dybkjær, H. Dybkjær, Wizard of Oz Simulation, Springer London, London, 1998, pp. 127–160.

[12] S. Bødker, Through the interface-a human activity approach to user interface design, DAIMI Report Series 16 (224) (1987).

[13] O. Boissier, R.H. Bordini, J.F. Hübner, A. Ricci, A. Santi, Multi-agent oriented programming with jacamo, Sci. Comput. Program. 78 (6) (2013) 747–761.

[14] O. Boissier, J.F. Hübner, A. Ricci, The jacamo framework, in: Social Coordination Frameworks for Social Technical Systems, Springer, 2016, pp. 125–151.

[15] J. Bomanson, M. Gebser, T. Janhunen, Rewriting optimization statements in answer-set programs, in: Technical Communications of the 32nd International Conference on Logic Programming (ICLP 2016), Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[16] M. Bratman, Intention, Plans, and Practical Reason, Harvard University Press, 1987.

[17] M.E. Bratman, Shared intention, Ethics 104 (1) (1993) 97–113.

[18] M.E. Bratman, A desire of one's own, J. Philos. 100 (5) (2003) 221–242.

[19] C. Breazeal, Designing Sociable Machines, vol. 3, chap. 4, Springer, 2002, pp. 149–156.

[20] C. Breazeal, Toward sociable robots, Robot. Auton. Syst. 42 (3) (2003) 167–175, socially Interactive Robots.

[21] O. Caicedo, E. De La Cruz, G. Taimal, Middleware de seguridad para el interworking wlan-ims, Revista Facultad de Ingeniería Universidad de Antioquia 56 (2010) 193–202.

[22] G. Chaslot, S. De Jong, J.T. Saito, J. Uiterwijk, Monte-Carlo tree search in production management problems, in: Proceedings of the 18th BeNeLux Conference on Artificial Intelligence, vol. 9198, 2006.

[23] O.J. Clark, S. Grogan, J. Cole, N. Ray, How might avatar appearance influence health-related outcomes? A systematic review and narrative meta-review, https://doi.org/10.31234/osf.io/j3675, May 2019, psyarxiv.com/j3675.

[24] P. Cohen, Foundations of collaborative task-oriented dialogue: what's in a slot?, in: Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, Association for Computational Linguistics, Stockholm, Sweden, Sep 2019, pp. 198–209.

[25] P.R. Cohen, L. Galescu, A Planning-Based Explainable Collaborative Dialogue System, 2023.

[26] P.R. Cohen, H.J. Levesque, I.A. Smith, On team formation, Synth. Libr. (1997) 87–114.

[27] E. Coronado, T. Kiyokawa, G.A.G. Ricardez, I.G. Ramirez-Alpizar, G. Venture, N. Yamanobe, Evaluating quality in human-robot interaction: a systematic search and classification of performance and human-centered factors, measures and metrics towards an industry 5.0, J. Manuf. Syst. 63 (2022) 392–410.

[28] J.L. Crowley, J. Coutaz, J. Grosinger, J. Vazquez-Salceda, C. Angulo, A. Sanfeliu, L. Iocchi, A.G. Cohn, A hierarchical framework for collaborative artificial intelligence, IEEE Pervasive Comput. (2022) 1–10, https://doi.org/10.1109/MPRV.2022.3208321.

[29] P. Damacharla, A.Y. Javaid, J.J. Gallimore, V.K. Devabhaktuni, Common metrics to benchmark human-machine teams (hmt): a review, IEEE Access 6 (2018) 38637–38655.

[30] E. Dantsin, T. Eiter, G. Gottlob, A. Voronkov, Complexity and expressive power of logic programming, in: Proceedings of Computational Complexity. Twelfth Annual IEEE Conference, IEEE, 1997, pp. 82–101.

[31] E. Dantsin, T. Eiter, G. Gottlob, A. Voronkov, Complexity and expressive power of logic programming, ACM Comput. Surv. 33 (3) (2001) 374–425.

[32] K. Dautenhahn, Socially intelligent robots: dimensions of human–robot interaction. Philosophical transactions of the royal society B, Biol. Sci. 362 (1480) (2007) 679–704.

[33] D. David, P. Thérouanne, I. Milhabet, The acceptability of social robots: a scoping review of the recent literature, Comput. Hum. Behav. 107419 (2022).

[34] J. Delgrande, P. Peppas, S. Woltran, Agm-style belief revision of logic programs under answer set semantics, in: Logic Programming and Nonmonotonic Reasoning: 12th International Conference, LPNMR 2013, Corunna, Spain, September 15-19, 2013, in: Proceedings, vol. 12, Springer, 2013, pp. 264–276.

[35] J. Delgrande, T. Schaub, H. Tompits, S. Woltran, A model-theoretic approach to belief change in answer set programming, ACM Trans. Comput. Log. 14 (2) (Jun 2013) 1–46.

[36] F.C. Delicato, P.F. Pires, A.Y. Zomaya, Middleware platforms: state of the art, new issues, and future trends, in: The Art of Wireless Sensor Networks: Volume 1: Fundamentals, 2014, pp. 645–674.

[37] F. Dignum, B. Dunin-Keplicz, R. Verbrugge, Creating collective intention through dialogue, Log. J. IGPL 9 (2) (2001) 289–304.

[38] J. Dix, A classification theory of semantics of normal logic programs: I. Strong properties, Fundam. Inform. 22 (3) (1995) 227–255.

[39] J. Dix, A classification theory of semantics of normal logic programs: II. Weak properties, Fundam. Inform. 22 (3) (1995) 257–288.

[40] C. Dodaro, P. Gasteiger, N. Leone, B. Musitsch, F. Ricca, K. Shchekotykhin, Combining answer set programming and domain heuristics for solving hard industrial problems (application paper), Theory Pract. Log. Program. 16 (5–6) (Sep 2016) 653–669, https://doi.org/10.1017/S1471068416000284.

[41] P.M. Dung, An argumentation-theoretic foundation for logic programming, J. Log. Program. 22 (2) (1995) 151–177.

[42] W. Dvorák, P.E. Dunne, Computational problems in formal argumentation and their complexity, J. Appl. Logics 4 (8) (2017) 2557–2622.

[43] A. Dyoub, S. Costantini, G.D. Gasperis, Answer set programming and agents, Knowl. Eng. Rev. 33 (2018) e19.

[44] T. Eiter, G. Gottlob, H. Mannila, Disjunctive datalog, ACM Trans. Database Syst. 22 (3) (1997) 364–418.

[45] U. Endriss, N. Maudet, F. Sadri, F. Toni, Logic-based agent communication protocols, in: Advances in Agent Communication: International Workshop on Agent Communication Languages, ACL 2003, Melbourne, Australia, July 14, 2003. Revised and Invited Papers, Springer, 2004, pp. 91–107.

[46] Y. Engeström, R. Miettinen, R.L. Punamäki, Activity theory and individual and social transformation, in: Perspectives on Activity Theory, in: Learning in Doing: Social, Cognitive and Computational Perspectives, Cambridge University Press, 1999, pp. 19–38.

[47] C. Esterwood, L.P. Robert, A systematic review of human and robot personality in health care human-robot interaction, Front. Robt. AI 8 (2021) 748246.

[48] W. Faber, G. Pfeifer, N. Leone, Semantics and complexity of recursive aggregates in answer set programming, Artif. Intell. 175 (1) (Jan 2011) 278–298.

[49] A. Falkner, G. Friedrich, K. Schekotihin, R. Taupe, E.C. Teppan, Industrial applications of answer set programming, Künstl. Intell. 32 (2) (Aug 2018) 165–176, https://doi.org/10.1007/s13218-018-0548-6.

[50] M.R. Fellows, M.A. Langston, On search decision and the efficiency of polynomial-time algorithms, in: Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing, 1989, pp. 501–512.

[51] T. Fong, I. Nourbakhsh, K. Dautenhahn, A survey of socially interactive robots, Robot. Auton. Syst. 42 (3) (2003) 143–166.

[52] W.D. Freeman, D.K. Sanghavi, M.S. Sarab, M.S. Kindred, E.M. Dieck, S.M. Brown, T. Szambelan, J. Doty, B. Ball, H.M. Felix, J.C. Dove, J.M. Mallea, C. Soares, L.V. Simon, Robotics in simulated COVID-19 patient room for health care worker effector tasks: preliminary, feasibility experiments. Mayo clinic proceedings: innovations, Qual. Outcomes 5 (1) (Feb 2021) 161–170.

[53] M. Gebser, R. Kaminski, T. Schaub, Complex optimization in answer set programming, Theory Pract. Log. Program. 11 (4–5) (2011) 821–839.

[54] M. Gebser, B. Kaufmann, J. Romero, R. Otero, T. Schaub, P. Wanko, Domain-specific heuristics in answer set programming, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 27, 2013, pp. 350–356.
[55] M. Gebser, T. Schaub, S. Thiele, GrinGo: a new grounder for answer set programming, in: Logic Programming and Nonmonotonic Reasoning, Springer, Berlin, Germany, 2007, pp. 266–271.
[56] M. Gelfond, V. Lifschitz, Classical negation in logic programs and disjunctive databases, New Gener. Comput. 9 (3-4) (1991) 365–385.
[57] E. Goldberg, Y. Novikov, Berkmin: a fast and robust sat-solver, Discrete Appl. Math. 155 (12) (2007) 1549–1561.
[58] E. Guerrero, P. Kalmi, Gamification strategies: a characterization using formal argumentation theory, SN Comput. Sci. 3 (4) (2022) 1–19.
[59] E. Guerrero, H. Lindgren, Practical Reasoning About Complex Activities, SpringerLink (Jun 2017) 82–94.
[60] E. Guerrero, H. Lindgren, Practical reasoning about complex activities, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 10349, 2017, pp. 82–94.
[61] E. Guerrero, H. Lindgren, Practical reasoning about complex activities, in: International Conference on Practical Applications of Agents and Multi-Agent Systems, Springer, 2017, pp. 82–94.
[62] E. Guerrero, T. Vartiainen, P. Kalmi, What if gamified software is fully proactive? Towards autonomy-related design principles, in: Personalizing Persuasive Technologies Workshop, vol. 3153, CEUR-WS, 2022, pp. 1–7.
[63] P. Gustavsson, M. Holm, A. Syberfeldt, L. Wang, Human-robot collaboration–towards new metrics for selection of communication technologies, Proc. CIRP 72 (2018) 123–128.
[64] S. Honig, T. Oron-Gilad, Understanding and resolving failures in human-robot interaction: literature review and model development, Front. Psychol. 9 (2018) 21.
[65] H. Hoos, M. Lindauer, T. Schaub, claspfolio 2: advances in algorithm selection for answer set programming, Theory Pract. Log. Program. 14 (4–5) (2014) 569–585.
[66] K. Inoue, C. Sakama, Equivalence of logic programs under updates, in: Logics in Artificial Intelligence: 9th European Conference, JELIA 2004, Lisbon, Portugal, September 27-30, 2004. Proceedings 9, Springer, 2004, pp. 174–186.
[67] M. Ionescu, I. Marsic, Stateful publish-subscribe for mobile environments, in: WMASH '04: Proceedings of the 2nd ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots, Association for Computing Machinery, New York, NY, USA, Oct 2004, pp. 21–28.
[68] N.R. Jennings, Commitments and conventions: the foundation of coordination in multi-agent systems, Knowl. Eng. Rev. 8 (3) (1993) 223–250.
[69] R. Johnsonbaugh, M. Schaefer, Algorithms, vol. 2, Pearson Education, 2004.
[70] H. Katsuno, A.O. Mendelzon, Updating a knowledge base and revising it, in: Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference (KR91), Morgan Kaufmann Pub, 1991, p. 387.
[71] A.M. Keller, M.W. Wilkins, On the use of an extended relational model to handle changing incomplete information, IEEE Trans. Softw. Eng. 7 (1985) 620–633.
[72] T. Keller, M. Helmert, Trial-based heuristic tree search for finite horizon MDPs, ICAPS 23 (Jun 2013) 135–143, https://doi.org/10.1609/icaps.v23i1.13557.
[73] D. Kinny, M. Georgeff, J. Hendler, Experiments in optimal sensing for situated agents, in: Proceedings of the Second Pacific Rim International Conference on Artificial Intelligence, 1992, pp. 1176–1182.
[74] D. Kinny, E. Sonenberg, M. Ljungberg, G. Tidhar, A. Rao, E. Werner, Planned team activity, in: C. Castelfranchi, E. Werner (Eds.), European Workshop on Modelling Autonomous Agents in a Multi-Agent World, Springer, 1994, pp. 227–256.
[75] B. Kitchenham, Procedures for Performing Systematic Reviews, vol. 33, Keele University, Keele, UK, 2004, pp. 1–26.
[76] C.U. Krägeloh, J. Bharatharaj, S.K. Sasthan Kutty, P.R. Nirmala, L. Huang, Questionnaires to measure acceptability of social robots: a critical review, Robotics 8 (4) (Oct 2019) 88, https://doi.org/10.3390/robotics8040088.
[77] S. Kul, A. Sayar, A survey of publish/subscribe middleware systems for microservice communication, in: 2021 5th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), IEEE, 2021, pp. 781–785.
[78] P. Leitão, S. Karnouskos, L. Ribeiro, J. Lee, T. Strasser, A.W. Colombo, Smart agents in industrial cyber–physical systems, Proc. IEEE 104 (5) (Mar 2016) 1086–1101, https://doi.org/10.1109/JPROC.2016.2521931.
[79] J. Leite, M. Slota, A brief history of updates of answer-set programs, Theory Pract. Log. Program. 23 (1) (2023) 57–110.
[80] J.A. Leite, L.M. Pereira, Generalizing updates: from models to programs, in: Logic Programming and Knowledge Representation: Third International Workshop, LPKR'97 Port Jefferson, New York, USA, October 17, 1997, Selected Papers 3, Springer, New York, USA, 1998, pp. 224–246.
[81] L.H. Lelis, S. Zilles, R.C. Holte, Stratified tree search: a novel suboptimal heuristic search algorithm, in: Proceedings of the 2013 International Conference on Autonomous Agents and Multi-Agent Systems, 2013, pp. 555–562.
[82] N. Leone, G. Pfeifer, W. Faber, T. Eiter, G. Gottlob, S. Perri, F. Scarcello, The dlv system for knowledge representation and reasoning, ACM Trans. Comput. Log. 7 (3) (2006) 499–562.
[83] A.N. Leontiev, Activity, Consciousness, and Personality, Prentice-Hall, Englewood Cliffs, N.J., 1978.
[84] S.V. Lev, Thought and word, in: Though and Language, MIT Press, 2012, chap. 7.
[85] V. Lifschitz, On the declarative semantics of logic programs with negation, in: Foundations of Deductive Databases and Logic Programming, Morgan Kaufmann Publishers Inc., 1988, pp. 177–192.
[86] V. Lifschitz, Answer set planning, in: Logic Programming and Nonmonotonic Reasoning: 5th International Conference, LPNMR'99 El Paso, Texas, USA, December 2–4, 1999 Proceedings 5, Springer, 1999, pp. 373–374.
[87] V. Lifschitz, D. Pearce, A. Valverde, Strongly equivalent logic programs, ACM Trans. Comput. Log. 2 (4) (2001) 526–541.
[88] A. Malchanau, V. Petukhova, H. Bunt, Towards integration of cognitive models in dialogue management: designing the virtual negotiation coach application, Dialogue Discourse 9 (2) (2018) 35–79.
[89] M. Maratea, L. Pulina, F. Ricca, The multi-engine asp solver me-asp, in: Logics in Artificial Intelligence: 13th European Conference, JELIA 2012, Toulouse, France, September 26-28, 2012. Proceedings, Springer, 2012, pp. 484–487.
[90] M. Marge, A.I. Rudnicky, Miscommunication detection and recovery in situated human-robot dialogue, ACM Trans. Interact. Intell. Syst. 9 (1) (2019).
[91] M. Matteucci, Publish/subscribe middleware for robotics: requirements and state of the art, Tech. Report N 2003.3, 2003.
[92] P. McBurney, R.M. Van Eijk, S. Parsons, L. Amgoud, A dialogue game protocol for agent purchase negotiations, Auton. Agents Multi-Agent Syst. 7 (3) (Nov 2003) 235–273, https://doi.org/10.1023/A:1024787301515.
[93] M. McTear, Conversational modelling for chatbots: current approaches and future directions, in: Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung, 2018, pp. 175–185.
[94] M.W. Moskewicz, C.F. Madigan, Y. Zhao, L. Zhang, S. Malik, Chaff: engineering an efficient sat solver, in: Proceedings of the 38th Annual Design Automation Conference, 2001, pp. 530–535.
[95] R.R. Murphy, D. Schreckenghost, Survey of metrics for human-robot interaction, in: 2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI), IEEE, 2013, pp. 197–198.
[96] A. Nowak, P. Lukowicz, P. Horodecki, Assessing artificial intelligence for humanity: will ai be the our biggest ever advance? Or the biggest threat [opinion], IEEE Technol. Soc. Mag. 37 (4) (2018) 26–34, https://doi.org/10.1109/MTS.2018.2876105.
[97] A.R. Panisson, S. Sarkadi, P. McBurney, S. Parsons, R.H. Bordini, On the formal semantics of theory of mind in agent communication, in: M. Lujak (Ed.), Agreement Technologies - 6th International Conference, AT 2018, Bergen, Norway, December 6-7, 2018, Revised Selected Papers, in: Lecture Notes in Computer Science, vol. 11327, Springer, 2018, pp. 18–32.
[98] P. Panzarasa, N.R. Jennings, T.J. Norman, Formalizing collaborative decision-making and practical reasoning in multi-agent systems, J. Log. Comput. 12 (1) (2002) 55–117.
[99] C.H. Papadimitriou, Computational complexity, in: Encyclopedia of Computer Science, John Wiley and Sons Ltd., GBR, 2003, pp. 260–265.
[100] L. Per, Troubles with Mutualities: Towards a Dialogical Theory of Misunderstanding and Miscommunication, chap. 8, Cambridge University Press, UK, 1995, pp. 176–212.
[101] M. Persiani, T. Hellström, The mirror agent model: a bayesian architecture for interpretable agent behavior, in: Explainable and Transparent AI and Multi-Agent Systems: 4th International Workshop, EXTRAAMAS 2022, Virtual Event, May 9–10, 2022, Revised Selected Papers, Springer, 2022, pp. 111–123.
[102] M. Persiani, M. Tewari, Mediating joint intention with a dialogue management system, in: 1st International Workshop on New Foundations for Human-Centered AI, Virtual (Santiago de Compostela, Spain), September 4, 2020, RWTH Aachen University, 2020, pp. 79–82.
[103] P. Quaresma, J.G. Lopes, A logic programming framework for the abductive inference of intentions in cooperative dialogues, in: F. Pfenning (Ed.), Logic Programming and Automated Reasoning, Springer, Berlin, Heidelberg, 1994, pp. 189–199.
[104] R. Rampin, V. Rampin, Taguette: open-source qualitative data analysis, J. Open Sour. Softw. 6 (68) (2021) 3522, https://doi.org/10.21105/joss.03522.
[105] A.S. Rao, M.P. Georgeff, An abstract architecture for rational agents, KR 92 (1992) 439–449.
[106] A.S. Rao, M.P. Georgeff, et al., Bdi agents: from theory to practice, in: ICMAS, vol. 95, 1995, pp. 312–319.
[107] L.D. Riek, Wizard of oz studies in hri: a systematic review and new reporting guidelines, J. Human-Robot Interact. 1 (1) (2012) 119–136, https://doi.org/10.5898/JHRI.1.1.Riek.
[108] B. Sahindal, Detecting Conversational Failures in Task-Oriented Human-Robot Interactions, Master's thesis, KTH, EECS, 2020.
[109] C. Sakama, K. Inoue, An abductive framework for computing knowledge base updates, Theory Pract. Log. Program. 3 (6) (2003) 671–715, https://doi.org/10.1017/S1471068403001716.
[110] C. Sakama, K. Inoue, Coordination in answer set programming, ACM Trans. Comput. Log. 9 (2) (2008) 1–30.
[111] C. Sakama, T.C. Son, Interacting answer sets, in: International Workshop on Computational Logic in Multi-Agent Systems, Springer, 2009, pp. 122–140.
[112] Ş. Sarkadi, A.R. Panisson, R.H. Bordini, P. McBurney, S. Parsons, Towards an approach for modelling uncertain theory of mind in multi-agent systems, in: M. Lujak (Ed.), Agreement Technologies, Springer International Publishing, Cham, 2019, pp. 3–17.

[113] N. Schütte, B. Mac Namee, J. Kelleher, Robot perception errors and human resolution strategies in situated human–robot dialogue, Adv. Robot. 31 (5) (2017) 243–257.
[114] J.R. Searle, Collective intentions and actions, in: Intentions in Communication, 1990, p. 401.
[115] J.R. Searle, F. Kiefer, M. Bierwisch, et al., Speech Act Theory and Pragmatics, vol. 10, Springer, 1980.
[116] G.I. Simari, S.D. Parsons, Rational decision making in autonomous agents, in: VI Workshop de Investigadores en Ciencias de la Computación, 2004.
[117] Y.A. Solangi, Z.A. Solangi, S. Aarain, A. Abro, G.A. Mallah, A. Shah, Review on natural language processing (NLP) and its toolkits for opinion mining and sentiment analysis, in: 2018 IEEE 5th International Conference on Engineering Technologies and Applied Sciences (ICETAS), 2018, pp. 1–4.
[118] J. Ruiz-del Solar, M. Salazar, V. Vargas-Araya, U. Campodonico, N. Marticorena, G. Pais, R. Salas, P. Alfessi, V.C. Rojas, J. Urrutia, Mental and emotional health care for COVID-19 patients: employing pudu, a telepresence robot, IEEE Robot. Autom. Mag. 28 (1) (Jan 2021) 82–89, https://doi.org/10.1109/MRA.2020.3044906.
[119] L. Steels, Personal dynamic memories are necessary to deal with meaning and understanding in human-centric ai, in: NeHuAI@ ECAI, 2020, pp. 11–16.
[120] M. Tewari, H. Lindgren, Younger and older adults' perceptions on role, behavior, goal and recovery strategies for managing breakdown situations in human-robot dialogues, in: Proceedings of the 9th International Conference on Human-Agent Interaction, HAI '21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 433–437.
[121] M. Tewari, H. Lindgren, Expecting, understanding, relating, and interacting-older, middle-aged and younger adults' perspectives on breakdown situations in human–robot dialogues, Front. Robot. AI 9 (2022).
[122] M. Tewari, M. Persiani, Towards we-intentional human-robot interaction using theory of mind and hierarchical task network, in: The 5th International Conference on Computer-Human Interaction Research and Applications (CHIRA 2021), Online, October 28-29, 2021, Sitepress Digital Library, 2021, pp. 291–299.
[123] L. Tian, S. Oviatt, A taxonomy of social errors in human-robot interaction, J. Hum.-Robot Interact. 10 (2) (2021) 32.
[124] S.C. Tran, E. Pontelli, M. Balduccini, T. Schaub, Answer set planning: a survey, Theory Pract. Log. Program. 23 (1) (Jan 2023) 226–298.
[125] R. Tuomela, We-intentions revisited, Philos. Stud. 125 (3) (2005) 327–369.
[126] R. Tuomela, Joint intention, we-mode and i-mode, Midwest Studi. Philos. 30 (1) (2006) 35–58.
[127] R. Tuomela, K. Miller, We-intentions, Philos. Stud. 53 (3) (1988) 367–389.
[128] R. Tuomela, K. Miller, We-intentions, Social Ontology in the Making (2003) 69.
[129] D. Walton, E.C.W. Krabbe, Commitment in Dialogue: Basic Concepts of Interpersonal Reasoning, State University of New York Press, 1995.
[130] S. Whelan, K. Murphy, E. Barrett, C. Krusche, A. Santorelli, D. Casey, Factors affecting the acceptability of social robots by older adults including people with dementia or cognitive impairment: a literature review, Int. J. Soc. Robot. 10 (5) (Nov 2018) 643–668, https://doi.org/10.1007/s12369-018-0471-x.
[131] M. Winslett, Updating Logical Databases, Cambridge Tracts in Theoretical Computer Science, Cambridge University Press, 1990.
[132] M. Wooldridge, S. Parsons, G. Goos, J. Hartmanis, J. van Leeuwen, Intention Reconsideration Reconsidered, vol. 1555, Springer, Berlin, Heidelberg, 1999, pp. 63–79.
[133] G. Yang, H. Lv, Z. Zhang, L. Yang, J. Deng, S. You, J. Du, H. Yang, Keep healthcare workers safe: application of teleoperated robot in isolation ward for covid-19 prevention and control, Chin. J. Mech. Eng. 33 (1) (2020) 1–4.
[134] A. Zacharaki, I. Kostavelis, A. Gasteratos, I. Dokas, Safety bounds in human robot interaction: a survey, Saf. Sci. 127 (2020) 104667.
[135] L. Zhao, J. Qian, L. Chang, G. Cai, Using asp for knowledge management with user authorization, Data Knowl. Eng. 69 (8) (2010) 737–762.