Modelling Agents Roles in the Epistemic Logic L-DINF

Stefania Costantini[1,3], Andrea Formisano[2,3,*] and Valentina Pitoni[1]

[1] Università di L'Aquila (DISIM), Via Vetoio, 1, L'Aquila, 67100, Italy
[2] Università di Udine (DMIF), Via delle Scienze, 206, Udine, 33100, Italy
[3] INdAM — GNCS, Piazzale Aldo Moro, 5, Roma, 00185, Italy

Abstract

In this paper, we further advance a line of work aimed to formally model via epistemic logic (aspects of) the group dynamics of cooperative agents. In fact, we have previously proposed and here extend a particular logical framework (the Logic of "Inferable" *L-DINF*), where a group of cooperative agents can jointly perform actions. I.e., at least one agent of the group can perform the action, either with the approval of the group or on behalf of the group. In this paper, we introduce agents' *roles* within a group. We choose to model roles in terms of the actions that each agent is enabled by its group to perform. We extend the semantics and the proof of strong completeness of our logic, and we show the usefulness of the new extension via a significant example.

Keywords

Epistemic Logic, Multi Agent System, Cooperation and Roles

1. Introduction

This paper falls within a research effort whose overall objective is to devise a comprehensive framework based upon epistemic logic, so as to allows a designer to formalize and formally verify agents and Multi-Agent Systems (MAS). We have been particularly interested in modelling the capability to construct and execute joint plans within a group of agents. However, such a logical framework will really be useful if it will be immersed (either fully or in parts) into a real agent-oriented programming language.[1] To this aim, we have taken all along into particular account the connection between theory and practice, so as to make our logic actually usable by a system's designers.

Cooperation among agents in a MAS allow agents to achieve better and faster results, and it is often the case that a group can fulfil objectives that are out of reach for the single agent. Often, each participating agent is not able to solve a whole problem or to reach an overall goal by itself, but can only cope with a small subproblem/subgoal for which it has the required competence. The overall result/goal is accomplished by means of cooperation with other agents. This is the motivation that led us to develop logics where agents belong to groups, and it is possible for agents to reason about what their group of agents can do and what they themselves are able to do or prefer to do (in terms of actions to perform) and which cost they are able to pay for the execution of a costly action, whereas however, in case of insufficient budget, an agent can be supported by its group.

In this paper, we introduce *roles* that agents may assume within the group (concerning which actions they are both able and enabled to perform). I.e., within a group, an action can be performed only by agents which are allowed by the group to do so (supposedly, because they have the right competences).

This paper continues, in fact, a long-lasting line of work aimed to formally model via epistemic logic (aspects of) the group dynamics of cooperative agents via the Logic of "Inferable" *L-DINF* (first introduced in [4]). As mentioned, in past work we have taken into consideration actions' *cost* [5], and the preferences that each agent can have for what concerns performing each action [6].

The key assumption underlying our approach is that, although logic has proved a good tool to express the semantics underlying (aspects of) agent-oriented programming languages, in order to foster a practical adoption there are, at least, the following requirements. (i) It is important to keep the complexity of the logic low enough to be practically manageable. (ii) It is important to ensure *modularity*, as it allows programmers to better organize the definition of the application at hand, and allows an agent-systems' definition to be more flexible and customizable. Notice, moreover, that modularity can be an advantage for explainability, in the sense of making the explanation itself modular. (iii) It is important not to overload syntax, as a cumbersome syntax can discourage practical adoption.

So, our approach tries to join the rigour of logic and a attention to practical aspects. Thus, we allow a designer

NMR 2022: 20th International Workshop on Non-Monotonic Reasoning, August 07–09, 2022, Haifa, Israel

[*] Corresponding author.

stefania.costantini@univaq.it (S. Costantini); andrea.formisano@uniud.it (A. Formisano); valentina.pitoni@univaq.it (V. Pitoni)

0000-0002-5686-6124 (S. Costantini); 0000-0002-6755-9314 (A. Formisano); 0000-0002-4245-4073 (V. Pitoni)

[1] Note that several agent-oriented programming languages and systems exist, where, since our approach is logic-based, we are interested in those of them which are based upon computational logic (cf., e.g., [1, 2, 3] for recent surveys on such languages), and thus endowed (at least in principle) with a logical semantics.

to define in a separate way at the semantic level which actions are allowed for each agent to perform at each stage, with which degree of preference and, now, taking the agent's *role* within the group into account. So far in fact, the specification was missing about which actions an agent is allowed to perform: in practical situations in fact, it will hardly be the case that every agent can perform every action, meaning being "able to perform" and "allowed to perform". This is in fact the new feature that we introduce here.

For an in-depth discussion on the relationship of logic *L-DINF* with related work, the reader may refer to [5]. One may notice that the logic presented here has no explicit time. We tackled however the issue of time in previous work, discussed in [7, 8, 9].

The paper is organized as follows. In Sect. 2 we introduce syntax and semantics of *L-DINF*, together with an axiomatization of the proposed logical system.[2] In Sect. 3 we discuss an example of application of the new logic. In Sect. 4 we present our definition of canonical model of an *L-DINF* theory. Finally, in Sect. 5 we conclude.

2. Logical framework

The logic *L-DINF* consists of a static component and a dynamic one. The former, called *L-INF*, is a logic of explicit beliefs and background knowledge. The dynamic component, called *L-DINF*, extends the static one with dynamic operators capturing the consequences of the agents' inferential actions on their explicit beliefs as well as a dynamic operator capturing what an agent can conclude by performing inferential actions in its repertoire.

2.1. Syntax

Let $Atm = \{p, q, \ldots\}$ be a countable set of atomic propositions. By $Prop$ we denote the set of all propositional formulas, i.e. the set of all Boolean formulas built out of the set of atomic propositions Atm. A set Atm_A represents the physical actions that an agent can perform, including "active sensing" actions (e.g., "let's check whether it rains", "let's measure the temperature"). The language of *L-DINF*, denoted by $\mathcal{L}_{\textit{L-DINF}}$, is defined as follows:

$$\begin{array}{rcl}
\varphi, \psi & ::= & p \mid \neg\varphi \mid \varphi \wedge \psi \mid \mathbf{B}_i\,\varphi \mid \mathbf{K}_i\,\varphi \mid \\
& & do_i(\phi_A) \mid do_i^P(\phi_A) \mid can\_do_i(\phi_A) \mid \\
& & do_G(\phi_A) \mid do_G^P(\phi_A) \mid can\_do_G(\phi_A) \mid \\
& & intend_i(\phi_A) \mid intend_G(\phi_A) \mid \\
& & exec_i(\alpha) \mid exec_G(\alpha) \mid [G:\alpha]\,\varphi \mid \\
& & pref\_do_i(\phi_A, d) \mid pref\_do_G(i, \phi_A) \\
\alpha & ::= & \vdash(\varphi,\psi) \mid \cap(\varphi,\psi) \mid \downarrow(\varphi, \psi) \mid \dashv(\varphi, \psi)
\end{array}$$

where p ranges over Atm, $i \in Agt$, $\phi_A \in Atm_A$, and $d \in \mathbb{N}$. (Other Boolean operators are defined from $\neg$ and $\wedge$ in the standard manner. Moreover, for simplicity, whenever $G = \{i\}$ we will write i as subscript in place of $\{i\}$.) The language of *inferential actions* of type α is denoted by $\mathcal{L}_{\mathsf{ACT}}$. The static part *L-INF* of *L-DINF*, includes only those formulas not having sub-formulas of type α, namely, no inferential operation are admitted.

The expression $intend_i(\phi_A)$ indicates the intention of agent i to perform the physical action ϕ_A in the sense of the BDI agent model [10]. This intention can be part of an agent's knowledge base from the beginning, or it can be derived later. We do not cope with the formalization of BDI, for which the reader may refer, e.g., to [11]. So, we treat intentions rather informally, assuming also that $intend_G(\phi_A)$ holds whenever all agents in group G intend to perform the action ϕ_A.

The formula $do_i(\phi_A)$ indicates *actual execution* of the action ϕ_A by the agent i. This fact is automatically recorded by the new belief $do_i^P(\phi_A)$ (postfix "P" standing for "past" action). By precise choice, do_i and do_i^P (and similarly do_G and do_G^P) are not axiomatized. In fact, we assume they are realized in a way that is unknown at the logical level. Hence, the axiomatization concerns only the relationship between doing and being enabled to do.

The expressions $can\_do_i(\phi_A)$ and $pref\_do_i(\phi_A, d)$ are closely related to $do_i(\phi_A)$. In fact, $can\_do_i(\phi_A)$ is to be seen as an enabling condition, indicating that agent i is enabled to execute action ϕ_A, while instead $pref\_do_i(\phi_A, d)$ indicates the level d of preference/willingness of agent i to perform that action. $pref\_do_G(i, \phi_A)$ indicates that agent i exhibits the *maximum level* of preference on performing action ϕ_A within all members of its group G. Notice that, if a group of agents intends to perform an action ϕ_A, this will entail that the entire group intends to do ϕ_A, that will be enabled to be actually executed only if at least one agent $i \in G$ can do it, i.e., it can derive $can\_do_i(\phi_A)$.

Formulas of the form $\mathbf{B}_i\,\varphi$ represent beliefs of agent i, while those of the form $\mathbf{K}_i\,\varphi$ express background knowledge. Explicit beliefs, i.e., facts and rules acquired via perceptions during an agent's operation are kept in the *working memory* of the agent. Unlike explicit beliefs, an agent's background knowledge is assumed to satisfy *omniscience* principles, such as closure under conjunction and known implication, and closure under logical consequence, and introspection. In fact, $\mathbf{K}_i$ is actually the well-known S5 modal operator often used to model/represent knowledge. The fact that background knowledge is closed under logical consequence is justified because we conceive it as a kind of stable reliable *knowledge base*, or *long-term memory*. We assume the background knowledge to include: facts (formulas) known by the agent from the beginning, and facts the agent has later decided to store in its long-term memory (by means of some decision

[2] Note that the part on syntax is reported almost literally from previous work, as the enhancements presented here lead to modifications of the semantics only.

mechanism not treated here) after having processed them in its working memory. We therefore assume background knowledge to be irrevocable, in the sense of being stable over time.

A formula of the form $[G : \alpha]\, \varphi$, with $G \in 2^{Agt}$, and where α must be an inferential action, states that "φ holds after action α has been performed by at least one of the agents in G, and all agents in G have common knowledge about this fact".

Remark 1. *If an inferential action is performed by an agent $i \in G$, the others agents belonging to the same group G have full visibility of this action and, therefore, as we suppose agents to be cooperative, it is as if they had performed the action themselves.*

Borrowing from [12], we distinguish four types of inferential actions α which allow us to capture some of the dynamic properties of explicit beliefs and background knowledge: $\downarrow(\varphi, \psi)$, $\cap(\varphi,\psi)$, $\dashv(\varphi, \psi)$, and $\vdash(\varphi,\psi)$, These actions characterize the basic operations of forming explicit beliefs via inference:

$\downarrow(\varphi, \psi)$: this inferential action infers ψ from φ in case φ is believed and, according to agent's background knowledge, ψ is a logical consequence of φ. If the execution succeeds, the agent starts believing ψ.

$\cap(\varphi,\psi)$: this action closes the explicit beliefs φ and ψ under conjunction. I.e., $\varphi \wedge \psi$ is deduced from φ and ψ.

$\dashv(\varphi, \psi)$: this inferential action performs a simple form of "belief revision". It removes ψ from the working memory in case φ is believed and, according to agent's background knowledge, $\neg\psi$ is logical consequence of φ. Both ψ and φ are required to be ground atoms.

$\vdash(\varphi,\psi)$: this inferential action adds ψ to the working memory in case φ is believed and, according to agent's working memory, ψ is logical consequence of φ. Notice that, unlike $\cap(\varphi,\psi)$, this action operates directly on the working memory without retrieving anything from the background knowledge.

Formulas of the forms $exec_i(\alpha)$ and $exec_G(\alpha)$ express executability of inferential actions either by agent i, or by a group G of agents (which is a consequence of any of the group members being able to execute the action). It has to be read as: "α is an inferential action that agent i (resp. an agent in G) can perform".

Remark 2. *In the mental actions $\vdash(\varphi,\psi)$ and $\downarrow(\varphi, \psi)$, the formula ψ which is inferred and asserted as a new belief can be $can\_do_i(\phi_A)$ or $do_i(\phi_A)$, which denote the possibility of execution or actual execution of physical action ϕ_A. In fact, we assume that when inferring $do_i(\phi_A)$ (from $can\_do_i(\phi_A)$ and possibly other conditions) then the action is actually executed, and the corresponding belief $do_i^P(\phi_A)$ is asserted, possibly augmented with a time-stamp. Actions are supposed to succeed by default; in case of failure, a corresponding failure event will be perceived by the agent. The do_i^P beliefs constitute a* history *of the agent's operation, so they might be useful for the agent to reason about its own past behaviour, and/or, importantly, they may be useful to provide* explanations *to human users.*

Remark 3. *Explainability in our approach can be directly obtained from proofs. Let us assume for simplicity that inferential actions can be represented in infix form as $\varphi_n\ OP\ \varphi_{n+1}$. Also, $exec_i(\alpha)$ means that the mental action α is executable by agent i and it is indeed executed. If, for instance, the user wants an explanation of why the physical action ϕ_A has been performed, the system can respond by exhibiting the proof that has lead to ϕ_A, put in the explicit form:*
$(exec_i(\varphi_1 OP_1\ \varphi_2) \wedge \ldots\ \wedge exec_i(\varphi_{n-1} OP_n\ \varphi_n) \wedge$
$exec_i(\varphi_n OP_n\ can\_do_i(\phi_A)) \wedge intend_i(\phi_A)) \vdash do_i(\phi_A)$
where each OP_i is one of the (mental) actions discussed above. The proof can possibly be translated into natural language, and declined either top-down or bottom-up.

As said in the Introduction, we model agents which, to execute an action, may have to pay a cost, so they must have a consistent budget available. Our agents, moreover, are entitled to perform only those physical actions that they conclude they can do. In our approach, an action can be executed by a group of agents if at least one agent in the group can do, and the group has the necessary budget available, sharing the cost according to some policy. Being our agent cooperative, among the agents that are able to do some physical action, one is selected (with any deterministic rule) among those which best prefer to perform that action. We assume that agents are aware of and agree with the cost-sharing policy.

We have not introduced costs and budget, feasibility of actions and willingness to perform them, *in the language* for two reasons: to keep the complexity of the logic reasonable, and to make such features customizable in a modular way.[3] So, as seen below, costs and budget are coped with at the semantic level which easily allows modular modification, for instance to define modalities of cost sharing are different from the one shown here, where group members share an action cost in equal parts. For

[3] We intend to use this logic in practice, to formalize memory in DALI agents, where DALI is a logic-based agent-oriented programming language [13, 14, 15]. So, computational effectiveness and modularity are crucial. Assuming that agents share the cost is reasonable when agents share resources, or cooperate to a common goal, as discussed, e.g., in [16, 17].

brevity we introduce a single budget function, and thus, implicitly, a single resource to be spent. Several budget functions, each one concerning a different resource, might be plainly defined.

2.2. Semantics

Definition 2.1 introduces the notion of *L-INF model*, which is then used to introduce semantics of the static fragment of the logic.

Notice that many relevant aspects of an agent's behaviour are specified in the definition of *L-INF model*, including which mental and physical actions an agent can perform, which is the cost of an action and which is the budget that the agent has available, which is the preference degree of the agent to perform each action. This choice has the advantages of keeping the complexity of the logic under control, and of making these aspects modularly modifiable. In this paper, we introduce new function H that, for each agent i belonging to a group, enables the agent to perform a certain set of actions, so, in this way, it specifies the *role* of i within the group.

As before let Agt be the set of agents.

Definition 2.1. *A model is a tuple $M = (W, N, \mathcal{R}, E, B, C, A, H, P, V)$ where:*

- *W is a set of worlds (or situations);*
- *$\mathcal{R} = \{R_i\}_{i \in Agt}$ is a collection of equivalence relations on W: $R_i \subseteq W \times W$ for each $i \in Agt$;*
- *$N : Agt \times W \longrightarrow 2^{2^W}$ is a neighborhood function such that, for each $i \in Agt$, each $w, v \in W$, and each $X \subseteq W$ these conditions hold:*

  (C1) *if $X \in N(i, w)$ then $X \subseteq \{v \in W \mid wR_iv\}$,*
  (C2) *if wR_iv then $N(i, w) = N(i, v)$;*

- *$E : Agt \times W \longrightarrow 2^{\mathcal{L}_{\mathsf{ACT}}}$ is an executability function of mental actions such that, for each $i \in Agt$ and $w, v \in W$, it holds that:*

  (D1) *if wR_iv then $E(i, w) = E(i, v)$;*

- *$B : Agt \times W \longrightarrow \mathbb{N}$ is a budget function such that, for each $i \in Agt$ and $w, v \in W$, the following holds*

  (E1) *if wR_iv then $B(i, w) = B(i, v)$;*

- *$C : Agt \times \mathcal{L}_{\mathsf{ACT}} \times W \longrightarrow \mathbb{N}$ is a cost function such that, for each $i \in Agt$, $\alpha \in \mathcal{L}_{\mathsf{ACT}}$, and $w, v \in W$, it holds that:*

  (F1) *if wR_iv then $C(i, \alpha, w) = C(i, \alpha, v)$;*

- *$A : Agt \times W \longrightarrow 2^{Atm_A}$ is an executability function for physical actions such that, for each $i \in Agt$ and $w, v \in W$, it holds that:*

  (G1) *if wR_iv then $A(i, w) = A(i, v)$;*

- *$H : Agt \times W \longrightarrow 2^{Atm_A}$ is an enabling function for physical actions such that, for each $i \in Agt$ and $w, v \in W$, it holds that:*

  (G2) *if wR_iv then $H(i, w) = H(i, v)$;*

- *$P : Agt \times W \times Atm_A \longrightarrow \mathbb{N}$ is a preference function for physical actions ϕ_A such that, for each $i \in Agt$ and $w, v \in W$, it holds that:*

  (H1) *if wR_iv then $P(i, w, \phi_A) = P(i, v, \phi_A)$;*

- *$V : W \longrightarrow 2^{Atm}$ is a valuation function.*

To simplify the notation, let $R_i(w)$ denote the set $\{v \in W \mid wR_iv\}$, for $w \in W$. The set $R_i(w)$ identifies the situations that agent i considers possible at world w. It is the *epistemic state* of agent i at w. In cognitive terms, it can be conceived as the set of all situations that agent i *can retrieve* from its long-term memory and reason about.

While $R_i(w)$ concerns background knowledge, $N(i, w)$ is the set of all facts that agent i explicitly believes at world w, a fact being identified with a set of worlds. Hence, if $X \in N(i, w)$ then, the agent i has the fact X under the focus of its attention and believes it. $N(i, w)$ is the explicit *belief set* of agent i at world w.

The executability of inferential actions is determined by the function E. For an agent i, $E(i, w)$ is the set of inferential actions that agent i can execute at world w. The value $B(i, w)$ is the budget the agent has available to perform inferential actions. Similarly, the value $C(i, \alpha, w)$ is the cost to be paid by agent i to execute the inferential action α in the world w. The executability of physical actions is determined by the function A. For an agent i, $A(i, w)$ is the set of physical actions that agent i can execute at world w. $H(i, w)$ instead is the set of physical actions that agent i is enabled by its group to perform. Which means, H defines the *role* of an agent in its group, via the actions that it is allowed to execute.

Agent's preference on executability of physical actions is determined by the function P. For an agent i, and a physical action ϕ_A, $P(i, w, \phi_A)$ is an integer value d indicating the degree of willingness of agent i to execute ϕ_A at world w.

Constraint (**C1**) imposes that agent i can have explicit in its mind only facts which are compatible with its current epistemic state. Moreover, according to constraint (**C2**), if a world v is compatible with the epistemic state of agent i at world w, then agent i should have the same explicit beliefs at w and v. In other words, if two situations are equivalent as concerns background knowledge, then they cannot be distinguished through the explicit belief set. This aspect of the semantics can be extended in future work to allow agents make plausible assumptions. Analogous properties are imposed by constraints (**D1**), (**E1**), and (**F1**). Namely, (**D1**) imposes that agent i always knows which inferential actions it can perform and those it cannot. (**E1**) states that agent i always knows the available budget in a world (potentially needed to perform actions). (**F1**) determines that agent i always knows how much it costs to perform an inferential action. (**G1**) and (**H1**) determine that an agent i always knows which physical actions it can perform and those it cannot, and with which degree of willingness, where (**G2**) specifies that an agent also knows whether its group gives it the permission to execute a certain action or not, i.e., if that action pertains to its *role* in the group.

Truth values of *L-DINF* formulas are inductively defined as follows.

Given a model $M = (W, N, \mathcal{R}, E, B, C, A, H, P, V)$, $i \in Agt$, $G \subseteq Agt$, $w \in W$, and a formula $\varphi \in \mathcal{L}_{\textit{L-INF}}$, we introduce the following shorthand notation:

$$\|\varphi\|^M_{i,w} = \{v \in W : wR_iv \text{ and } M, v \models \varphi\}$$

whenever $M, v \models \varphi$ is well-defined (see below). Then, we set:

(t1) $M, w \models p$ iff $p \in V(w)$
(t2) $M, w \models exec_i(\alpha)$ iff $\alpha \in E(i, w)$
(t3) $M, w \models exec_G(\alpha)$ iff $\exists i \in G$ with $\alpha \in E(i, w)$
(t4) $M, w \models can\_do_i(\phi_A)$ iff $\phi_A \in A(i, w) \cap H(i, w)$
(t5) $M, w \models can\_do_G(\phi_A)$ iff $\exists i \in G$
with $\phi_A \in A(i, w) \cap H(i, w)$
(t6) $M, w \models pref\_do_i(\phi_A, d)$ iff $\phi_A \in A(i, w)$ and
$P(i, w, \phi_A) = d$
(t7) $M, w \models pref\_do_G(i, \phi_A)$ iff
$M, w \models pref\_do_i(\phi_A, d)$
for $d = \max\{P(j, w, \phi_A) \mid$
$j \in G \wedge \phi_A \in A(j, w) \cap H(j, w)\}$
(t8) $M, w \models \neg\varphi$ iff $M, w \not\models \varphi$
(t9) $M, w \models \varphi \wedge \psi$ iff $M, w \models \varphi$ and $M, w \models \psi$
(t10) $M, w \models \mathbf{B}_i\, \varphi$ iff $||\varphi||^M_{i,w} \in N(i, w)$
(t11) $M, w \models \mathbf{K}_i\, \varphi$ iff $M, v \models \varphi$ for all $v \in R_i(w)$

As seen above, a physical action can be performed by a group of agents if at least one agent of the group can do it, and the level of preference for performing this action is set to the maximum among those of the agents enabled to do this action. For any inferential action α performed by any agent i, we set:

$$M, w \models [G : \alpha]\varphi \text{ iff } M^{[G:\alpha]}, w \models \varphi$$

With $M^{[G:\alpha]} = \langle W, N^{[G:\alpha]}, \mathcal{R}, E, B^{[G:\alpha]}, C, A, H, P, V\rangle$. Such model $M^{[G:\alpha]}$ represents the fact that the execution of an inferential action α affects the sets of beliefs of agent i and modifies the available budget. Such operation can add new beliefs by direct perception, by means of one inference step, or as a conjunction of previous beliefs. Hence, when introducing new beliefs (i.e., performing mental actions), the neighborhood must be extended accordingly.

The following property $enabled_w(G, \alpha)$ (for a world w, an action α and a group of agents G) concerns a key aspect in the definition of the logic. Specifically, it states when an inferential action is enabled, i.e., under which conditions, and by which agent(s), an action may be performed:

$$enabled_w(G, \alpha) : \;\; \exists j \in G\, (\alpha \in E(j, w) \wedge \frac{C(j,\alpha,w)}{|G|} \leq \min_{h \in G} B(h, w)).$$

In the above particular formulation (that is not fixed, but can be customized to the specific application domain) if at least an agent can perform it; and if the "payment" due by each agent, obtained by dividing the action's cost equally among all agents of the group, is within each agent's available budget. In case more than one agent in G can execute an action, we implicitly assume the agent j performing the action to be the one corresponding to the lowest possible cost. Namely, j is such that $C(j, \alpha, w) = \min_{h \in G} C(h, \alpha, w)$. This definition reflects a parsimony criterion reasonably adoptable by cooperative agents sharing a crucial resource such as, e.g., energy or money. Other choices might be viable, so variations of this logic can be easily defined simply by devising some other enabling condition and, possibly, introducing differences in neighborhood update. Notice that the definition of the enabling function basically specifies the "concrete responsibility" that agents take while concurring with their own resources to actions' execution. Also, in case of specification of various resources, different corresponding enabling functions might be defined.

Our contribution to modularity is that functions A, P and H, i.e., executability of physical actions, preference level of an agent about performing each action, and permission concerning which actions to actually perform, are not meant to be built-in. Rather, they can be defined via separate sub-theories, possibly defined using different logics, or, in a practical approach, even via pieces of code. This approach can be extended to function C, i.e., the cost of mental actions instead of being fixed may in principle vary, and be computed upon need.

2.3. Belief Update

In the logic defined so far, updating an agent's beliefs accounts to modify the neighborhood of the present world. The updated neighborhood $N^{[G:\alpha]}$ resulting from execution of a mental action α is specified as follows.

- If α is $\downarrow\!(\psi, \chi)$, then, for each $i \in Agt$ and $w \in W$,
$$N^{[G:\downarrow(\psi,\chi)]}(i, w) = N(i, w) \cup \{||\chi||^M_{i,w}\}$$
if $i \in G$ and $enabled_w(G, \downarrow\!(\psi, \chi))$ and $M, w \models \mathbf{B}_i\psi \wedge \mathbf{K}_i(\psi \rightarrow \chi)$. Otherwise, the neighborhood does not change (i.e., $N^{[G:\downarrow(\psi,\chi)]}(i, w) = N(i, w)$).
- If α is $\cap(\psi,\chi)$, then, for each $i \in Agt$ and $w \in W$,
$$N^{[G:\cap(\psi,\chi)]}(i, w) = N(i, w) \cup \{||\psi \wedge \chi||^M_{i,w}\}$$
if $i \in G$ and $enabled_w(G, \cap(\psi,\chi))$ and $M, w \models \mathbf{B}_i\psi \wedge \mathbf{B}_i\chi$. Otherwise, the neighborhood remains unchanged (i.e., $N^{[G:\cap(\psi,\chi)]}(i, w) = N(i, w)$).

- If α is $\dashv(\psi,\chi)$, then, for each $i \in Agt$ and $w \in W$,

$$N^{[G:\dashv(\psi,\chi)]}(i,w) = N(i,w) \setminus \{||\chi||^M_{i,w}\}$$

if $i{\in}G$ and $enabled_w(G,\dashv(\psi,\chi))$ and $M,w \models \mathbf{B}_i\psi \wedge \mathbf{K}_i(\psi \rightarrow \neg\chi)$. Otherwise, the neighborhood does not change (i.e., $N^{[G:\dashv(\psi,\chi)]}(i,w) = N(i,w)$).

- If α is $\vdash(\psi,\chi)$, then, for each $i \in Agt$ and $w \in W$,

$$N^{[G:\vdash(\psi,\chi)]}(i,w) = N(i,w) \cup \{||\chi||^M_{i,w}\}$$

if $i{\in}G$ and $enabled_w(G,\vdash(\psi,\chi))$ and $M,w \models \mathbf{B}_i\psi \wedge \mathbf{B}_i(\psi \rightarrow \chi)$. Otherwise, the neighborhood remains unchanged: $N^{[G:\vdash(\psi,\chi)]}(i,w) = N(i,w)$.

Notice that, after an inferential action α has been performed by an agent $j \in G$, all agents $i \in G$ see the same update in the neighborhood. Conversely, for any agent $h \notin G$ the neighborhood remains unchanged (i.e., $N^{[G:\alpha]}(h,w) = N(h,w)$). However, even for agents in G, the neighborhood remains unchanged if the required preconditions, on explicit beliefs, knowledge, and budget, do not hold (and hence the action is not executed).

Notice also that we might devise variations of the logic by making different decisions about neighborhood update to implement, for instance, partial visibility within a group.

Since each agent in G has to contribute to cover the costs of execution by consuming part of its available budget, an update of the budget function is needed. For an action α we assume that $j \in G$ executes α. Hence, for each $i \in Agt$ and each $w \in W$, we set

$$B^{[G:\alpha]}(i,w) = B(i,w) - C(j,\alpha,w)/|G|,$$

if $i{\in}G$ and $enabled_w(G,\alpha)$ and, depending on α,

$$\begin{array}{ll}
M,w \models \mathbf{B}_i\psi \wedge \mathbf{K}_i(\psi \rightarrow \chi) & \text{if } \alpha \text{ is } \downarrow(\psi,\chi), \text{ or} \\
M,w \models \mathbf{B}_i\psi \wedge \mathbf{B}_i\chi & \text{if } \alpha \text{ is } \cap(\psi,\chi)), \text{ or} \\
M,w \models \mathbf{B}_i\psi \wedge \mathbf{K}_i(\psi \rightarrow \neg\chi) & \text{if } \alpha \text{ is } \dashv(\psi,\chi), \text{ or} \\
M,w \models \mathbf{B}_i\psi \wedge \mathbf{B}_i(\psi \rightarrow \chi) & \text{if } \alpha \text{ is } \vdash(\psi,\chi).
\end{array}$$

Otherwise, $B^{[G:\alpha]}(i,w){=}B(i,w)$, i.e., the budget is preserved.

We write $\models_{\textit{L-DINF}} \varphi$ to denote that $M,w \models \varphi$ holds for all worlds w of every model M.

We introduce below relevant consequences of our formalization. For lack of space we omit the proof, that can be developed analogously to what done in [5]. As consequence of previous definitions, for any set of agents G and each agent $i \in G$, we have the following:

$\models_{\textit{L-INF}} (\mathbf{K}_i(\varphi \rightarrow \psi)) \wedge \mathbf{B}_i\,\varphi) \rightarrow [G : \downarrow(\varphi,\psi)]\,\mathbf{B}_i\,\psi.$

Namely, if the agent i has φ among beliefs and $\mathbf{K}_i(\varphi \rightarrow \psi)$ in its background knowledge, then as a consequence of the action $\downarrow(\varphi,\psi)$ the agent i and its group G start believing ψ.

$\models_{\textit{L-INF}} (\mathbf{K}_i(\varphi \rightarrow \neg\psi)) \wedge \mathbf{B}_i\,\varphi) \rightarrow [G : \dashv(\varphi,\psi)]\,\neg\mathbf{B}_i\,\psi.$

Namely, if the agent i has φ among beliefs and $\mathbf{K}_i(\varphi \rightarrow \neg\psi)$ in its background knowledge (for φ,ψ ground atoms), then, as a consequence of the action $\downarrow(\varphi,\psi)$ the agent i and its group G stop believing ψ.

$\models_{\textit{L-INF}} (\mathbf{B}_i\varphi \wedge \mathbf{B}_i\psi) \rightarrow [G : \cap(\varphi,\psi)]\mathbf{B}_i(\varphi \wedge \psi).$

Namely, if the agent i has φ and ψ as beliefs, then as a consequence of the action $\cap(\varphi,\psi)$ the agent i and its group G start believing $\varphi \wedge \psi$.

$\models_{\textit{L-INF}} (\mathbf{B}_i(\varphi \rightarrow \psi)) \wedge \mathbf{B}_i\,\varphi) \rightarrow [G : \vdash(\varphi,\psi)]\,\mathbf{B}_i,\psi.$

Namely, if the agent i has φ among its beliefs and $\mathbf{B}_i(\varphi \rightarrow \psi)$ in its working memory, then as a consequence of the action $\vdash(\varphi,\psi)$ the agent i and its group G start believing ψ.

2.4. Axiomatization

Below we introduce the axiomatization of our logic. The *L-INF* and *L-DINF* axioms and inference rules are the following, together with the usual axioms of propositional logic (where $G \subseteq Agt$ and $i \in Agt$):

1. $(\mathbf{K}_i\,\varphi \wedge \mathbf{K}_i(\varphi \rightarrow \psi)) \rightarrow \mathbf{K}_i\,\psi$;
2. $\mathbf{K}_i\,\varphi \rightarrow \varphi$;
3. $\neg\mathbf{K}_i(\varphi \wedge \neg\varphi)$;
4. $\mathbf{K}_i\,\varphi \rightarrow \mathbf{K}_i\,\mathbf{K}_i\,\varphi$;
5. $\neg\mathbf{K}_i\,\varphi \rightarrow \mathbf{K}_i\,\neg\mathbf{K}_i\,\varphi$;
6. $\mathbf{B}_i\,\varphi \wedge \mathbf{K}_i\,(\varphi \leftrightarrow \psi) \rightarrow \mathbf{B}_i\,\psi$;
7. $\mathbf{B}_i\,\varphi \rightarrow \mathbf{K}_i\,\mathbf{B}_i\,\varphi$;
8. $\dfrac{\varphi}{\mathbf{K}_i\,\varphi}$;
9. $[G:\alpha]p \leftrightarrow p$;
10. $[G:\alpha]\neg\varphi \leftrightarrow \neg[G:\alpha]\varphi$;
11. $exec_G(\alpha) \rightarrow \mathbf{K}_i\,(exec_G(\alpha))$;
12. $[G:\alpha](\varphi \wedge \psi) \leftrightarrow [G:\alpha]\varphi \wedge [G:\alpha]\psi$;
13. $[G:\alpha]\mathbf{K}_i\,\varphi \leftrightarrow \mathbf{K}_i\,([G:\alpha]\varphi)$;
14. $[G:\downarrow(\varphi,\psi)]\mathbf{B}_i\,\chi \leftrightarrow \mathbf{B}_i\,([G:\downarrow(\varphi,\psi)]\chi) \vee \big((\mathbf{B}_i\,\varphi \wedge \mathbf{K}_i\,(\varphi \rightarrow \psi)) \wedge \mathbf{K}_i\,([G:\downarrow(\varphi,\psi)]\chi \leftrightarrow \psi)\big)$;
15. $[G:\cap(\varphi,\psi)]\mathbf{B}_i\,\chi \leftrightarrow \mathbf{B}_i\,([G:\cap(\varphi,\psi)]\chi) \vee \big((\mathbf{B}_i\,\varphi \wedge \mathbf{B}_i\,\psi) \wedge \mathbf{K}_i\,[G:\cap(\varphi,\psi)]\chi \leftrightarrow (\varphi \wedge \psi)\big)$;
16. $[G:\vdash(\varphi,\psi)]\mathbf{B}_i\,\chi \leftrightarrow \mathbf{B}_i\,([G:\vdash(\varphi,\psi)]\chi) \vee \big((\mathbf{B}_i\,\varphi \wedge \mathbf{B}_i\,(\varphi \rightarrow \psi)) \wedge \mathbf{K}_i\,([G:\vdash(\varphi,\psi)]\chi \leftrightarrow \psi)\big)$;
17. $[G:\dashv(\varphi,\psi)]\neg\mathbf{B}_i\,\chi \leftrightarrow \mathbf{B}_i\,([G:\dashv(\varphi,\psi)]\chi) \vee \big((\mathbf{B}_i\,\varphi \wedge \mathbf{K}_i\,(\varphi \rightarrow \neg\psi)) \wedge \mathbf{K}_i\,([G:\dashv(\varphi,\psi)]\chi \leftrightarrow \psi)\big)$;
18. $intend_G(\phi_A) \leftrightarrow \forall i \in G\; intend_i(\phi_A)$;
19. $do_G(\phi_A) \rightarrow can\_do_G(\phi_A)$;
20. $do_i(\phi_A) \rightarrow can\_do_i(\phi_A) \wedge pref\_do_G(i,\phi_A)$;

21. $\frac{\psi \leftrightarrow \chi}{\varphi \leftrightarrow \varphi[\psi/\chi]}$.

We write *L-DINF* $\vdash \varphi$ to denote that φ is a theorem of *L-DINF*. It is easy to verify that the above axiomatization is sound for the class of *L-INF* models, namely, all axioms are valid and inference rules preserve validity. In particular, soundness of axioms 14–17 immediately follows from the semantics of $[G{:}\alpha]\varphi$, for each inferential action α, as previously defined.

Notice that, by abuse of notation, we have axiomatized the special predicates concerning intention and action enabling. Axioms 18–20 concern in fact physical actions, stating that: what is intended by a group of agents is intended by them all; and, neither an agent nor a group of agents can do what it is not enabled to do. Such axioms are not enforced by the semantics, but are supposed to be enforced by a designer's/programmer's encoding of parts of an agent's behaviour. In fact, axiom 18 enforces agents in a group to be cooperative. Axioms 19 and 20 ensure that agents will attempt to perform actions only if their preconditions are satisfied, i.e., if they can do them.

We do not handle such properties in the semantics as done, e.g., in dynamic logic, because we want agents' definition to be independent of the practical aspect, so we explicitly intend to introduce flexibility in the definition of such parts.

3. Problem Specification and Inference: An example

In this section, we propose an example to explain the usefulness of the new extension. For the sake of simplicity of illustration and of brevity, the example is in "skeletal" form.

Consider a group of four agents, who are the crew of an ambulance, including a driver, two nurses, and a medical doctor. The driver is the only one enabled to drive the ambulance. The nurses are enabled to perform a number of tasks, such as, e.g., administer a pain reliever, or clean, disinfect and bandage a wound, measure vital signs. It is however the task of a doctor to make a diagnosis, to prescribe medications, to order, perform, and interpret diagnostic tests, and to perform complex medical procedures.

Imagine that the hospital received notice of a car accident with an injured person. Then, it will inform the group of the fact that a patient needs help (how exactly is not treated here, because this depends on how the multi-agent system is implemented, but a message exchange will presumably suffice). The group will reason, and devise the intention/goal $\mathbf{K}_i(\mathit{intend}_G(\mathit{rescue\_patient}))$.

Among the physical actions that agents in the group can perform are for instance the following:

$\mathit{diagnose\_patient}$
$\mathit{administer\_urgent\_treatment}$
$\mathit{measure\_vital\_signs}$
$\mathit{pneumothorax\_aspiration}$
$\mathit{local\_anesthesia}$
$\mathit{bandage\_wounds}$
$\mathit{drive\_to\_patient}$
$\mathit{drive\_to\_hospital}$.

The group will now be required to perform a planning activity. Assume that, as a result of the planning phase, the knowledge base of each agent i contains the following rule, that specifies how to reach the intended goal in terms of actions to perform and sub-goals to achieve:

$$\begin{aligned}\mathbf{K}_i\big(&\mathit{intend}_G(\mathit{rescue\_patient}) \rightarrow \\ &\mathit{intend}_G(\mathit{drive\_to\_patient}) \wedge \\ &\mathit{intend}_G(\mathit{diagnose\_patient}) \wedge \\ &\mathit{intend}_G(\mathit{stabilize\_patient}) \wedge \\ &\mathit{intend}_G(\mathit{drive\_to\_hospital})\big).\end{aligned}$$

By axiom 18 listed in previous section, every agent will also have the specialized rule (for $i \leq 4$)

$$\begin{aligned}\mathbf{K}_i\big(&\mathit{intend}_G(\mathit{rescue\_patient}) \rightarrow \\ &\mathit{intend}_i(\mathit{drive\_to\_patient}) \wedge \\ &\mathit{intend}_i(\mathit{diagnose\_patient}) \wedge \\ &\mathit{intend}_i(\mathit{stabilize\_patient}) \wedge \\ &\mathit{intend}_i(\mathit{drive\_to\_hospital})\big).\end{aligned}$$

Then, the following is entailed for each of the agents:

$$\begin{aligned}&\mathbf{K}_i\big(\mathit{intend}_i(\mathit{rescue\_patient}) \rightarrow \\ &\qquad \mathit{intend}_i(\mathit{drive\_to\_patient})\big) \\ &\mathbf{K}_i\big(\mathit{intend}_i(\mathit{rescue\_patient}) \rightarrow \\ &\qquad \mathit{intend}_i(\mathit{diagnose\_patient})\big) \\ &\mathbf{K}_i\big(\mathit{intend}_i(\mathit{rescue\_patient}) \rightarrow \\ &\qquad \mathit{intend}_i(\mathit{stabilize\_patient})\big) \\ &\mathbf{K}_i\big(\mathit{intend}_i(\mathit{rescue\_patient}) \rightarrow \\ &\qquad \mathit{intend}_i(\mathit{drive\_to\_hospital})\big).\end{aligned}$$

While driving to the patient and then back to the hospital are actions, $\mathit{intend}_G(\mathit{stabilize\_patient})$ is a goal.

Assume now that the knowledge base of each agent i contains also the following general rules, stating that the group is available to perform each of the necessary actions. Which agent will in particular perform each action ϕ_A?

According to items (t4) and (t7) in the definition of truth values, for *L-DINF* formulas, this agent will be chosen as the one which best prefers to perform this action, among those that can do it. Formally, in the present situation, $\mathit{pref\_do}_G(i, \phi_A)$ returns the agent i in the group with the highest degree of preference on performing A, and $\mathit{can\_do}_G(\phi\_A)$ is true if there is some agent i in the group which is able and allowed to perform ϕ_A, i.e., $\phi_A \in A(i,w) \wedge \phi_A \in H(i,w)$.

$$\begin{aligned}\mathbf{K}_i\big(&\mathit{intend}_G(\mathit{drive\_to\_patient}) \wedge \\ &\mathit{can\_do}_G(\mathit{drive\_to\_patient}) \wedge \\ &\mathit{pref\_do}_G(i, \mathit{drive\_to\_patient}) \\ &\qquad \rightarrow \mathit{do}_G(\mathit{drive\_to\_patient})\big)\end{aligned}$$

$$\mathbf{K}_i\big(intend_G(diagnose\_patient)\wedge can\_do_G(diagnose\_patient)\wedge pref\_do_G(i, diagnose\_patient) \rightarrow do_G(diagnose\_patient)\big)$$

$$\mathbf{K}_i\big(intend_G(drive\_to\_hospital)\wedge can\_do_G(drive\_to\_hospital)\wedge pref\_do_G(i, drive\_to\_hospital) \rightarrow do_G(drive\_to\_hospital)\big).$$

As before, by axiom 18 such rules can be specialized to each single agent.

$$\mathbf{K}_i\big(intend_i(drive\_to\_patient)\wedge can\_do_i(drive\_to\_patient)\wedge pref\_do_i(i, drive\_to\_patient) \rightarrow do_G(drive\_to\_patient)\big)$$
$$\mathbf{K}_i\big(intend_i(diagnose\_patient)\wedge can\_do_i(diagnose\_patient)\wedge pref\_do_i(i, diagnose\_patient) \rightarrow do_i(diagnose\_patient)\big)$$
$$\mathbf{K}_i\big(intend_i(drive\_to\_hospital)\wedge can\_do_i(drive\_to\_hospital)\wedge pref\_do_i(i, drive\_to\_hospital) \rightarrow do_i(drive\_to\_hospital)\big)$$

So, for each action ϕ_A required by the plan, there will be some agent (let us assume for simplicity only one), for which $do_i(\phi_A))$ will be concluded. In our case, the agent driver will conclude $do_i(drive\_to\_patient))$ and $do_i(drive\_to\_hospital))$; the agent doctor will conclude $do_i(stabilize\_patient))$.

As previously stated, when an agent derives $do_i(\phi_A)$ for any physical action ϕ_A, the action is supposed to have been performed via some kind of *semantic attachment* which links the agent to the external environment.

Since $intend_G(stabilize\_patient)$ is not an action but a sub-goal, the group will have to devise a plan to achieve it. This will imply sensing actions and forms of reasoning not shown here. Assume that the diagnosis has been pneumothorax, and that the patient has also some wounds which are bleeding. Upon completion of the planning phase, the knowledge base of each agent i contains the following rule, that specifies how to reach the intended goal in terms of actions to perform:

$$\mathbf{K}_i\big(intend_G(stabilize\_patient) \rightarrow intend_G(measure\_vital\_signs)\wedge intend_G(local\_anesthesia)\wedge intend_G(bandage\_wounds)\wedge intend_G(pneumothorax\_aspiration)\big).$$

As before, these rules will be instantiated and elaborated by the single agents, and there will be some agent who will finally perform each action. Specifically, the doctor will be the one to perform pneumothorax aspiration, and the nurses (according to their competences and their preferences) will measure vital signs, administer local anesthesia and bandage the wounds. The new function H, in a sensitive domain such as healthcare, guarantees that each procedure is administered by one who is capable to (function A) but also enabled (function H), and so can take responsibility for the action.

An interesting point concerns derogation, i.e., for instance, life or death situations where, unfortunately, noone who is enabled to perform some urgently needed action is available; in such situations perhaps, anyone who is capable to perform this action might perform it. For instance, a nurse, in absence of a doctor, might attempt urgent pneumothorax aspiration.

From such perspective, semantics could be modified as follows:

(t4') $M, w \models able\_do_i(\phi_A)$ iff $\phi_A \in A(i, w)$

(t4") $M, w \models enabled\_do_i(\phi_A)$ iff $\phi_A \in A(i, w) \cap H(i, w)$

(t4-new) $M, w \models can\_do_i(\phi_A)$ iff $(\phi_A \in A(i, w) \cap H(i, w)) \vee (\phi_A \in A(i, w) \wedge \nexists j{\in}G : \phi_A \in A(j, w) \cap H(j, w))$

(t5-new) $M, w \models can\_do_G(\phi_A)$ iff $\exists\, i \in G$ s.t. $M, w \models can\_do_i(\phi_A)$

4. Canonical Model and Strong Completeness

In this section, we introduce the notion of canonical model of our logic, and we outline the proof of strong completeness w.r.t. the proposed class of models (by means of a standard canonical-model argument). As before, let Agt be a set of agents.

Definition 4.1. *A* canonical *L-INF model is a tuple* $M_c = \langle W_c, N_c, \mathcal{R}_c, E_c, B_c, C_c, A_c, H_c, P_c, V_c\rangle$ *where:*

- W_c *is the set of all maximal consistent subsets of* $\mathcal{L}_{\textit{L-INF}}$*;*
- $\mathcal{R}_c = \{R_{c,i}\}_{i\in Agt}$ *is a collection of equivalence relations on* W_c *such that, for every* $i \in Agt$ *and* $w, v \in W_c$*,* $wR_{c,i}v$ *if and only if (for all* φ*,* $\mathbf{K}_i\,\varphi \in w$ *implies* $\varphi \in v$*);*
- *For* $w \in W_c$*,* $\varphi \in \mathcal{L}_{\textit{L-INF}}$ *let* $A_\varphi(i, w) = \{v \in R_{c,i}(w) \mid \varphi \in v\}$*. Then, we put* $N_c(i, w) = \{A_\varphi(i, w) \mid \mathbf{B}_i\,\varphi \in w\}$*;*
- $E_c : Agt \times W_c \longrightarrow 2^{\mathcal{L}_{\mathsf{ACT}}}$ *is such that, for each* $i \in Agt$ *and* $w, v \in W_c$*, if* $wR_{c,i}v$ *then* $E_c(i, w) = E_c(i, v)$*;*
- $B_c : Agt \times W_c \longrightarrow \mathbb{N}$ *is such that, for each* $i \in Agt$ *and* $w, v \in W_c$*, if* $wR_{c,i}v$ *then* $B_c(i, w) = B_c(i, v)$*;*
- $C_c : Agt \times \mathcal{L}_{\mathsf{ACT}} \times W_c \longrightarrow \mathbb{N}$ *is such that, for each* $i \in Agt$*,* $\alpha \in \mathcal{L}_{\mathsf{ACT}}$*, and* $w, v \in W_c$*, if* $wR_{c,i}v$ *then* $C_c(i, \alpha, w) = C_c(i, \alpha, v)$*;*

- $A_c : Agt \times W_c \longrightarrow 2^{Atm_A}$ *is such that, for each* $i \in Agt$ *and* $w, v \in W_c$, *if* $wR_{c,i}v$ *then* $A_c(i, w) = A_c(i, v)$*;*
- $H_c : Agt \times W_c \longrightarrow 2^{Atm_A}$ *is such that, for each* $i \in Agt$ *and* $w, v \in W_c$, *if* $wR_{c,i}v$ *then* $H_c(i, w) = H_c(i, v)$*;*
- $P_c : Agt \times W_c \times Atm_A \longrightarrow \mathbb{N}$ *is such that, for each* $i \in Agt$ *and* $w, v \in W$, *if* $wR_{c,i}v$ *then* $P_c(i, w, \phi_A) = P_c(i, v, \phi_A)$*;*
- $V_c : W_c \longrightarrow 2^{Atm}$ *is such that* $V_c(w) = Atm \cap w$*.*

Note that, analogously to what done before, $R_{c,i}(w)$ denotes the set $\{v \in W_c \mid wR_{c,i}v\}$, for each $i \in Agt$. It is easy to verify that M_c is an *L-INF* model as defined in Def. 2.1, since, it satisfies conditions (**C1**),(**C2**),(**D1**),(**E1**),(**F1**),(**G1**),(**G2**),(**H1**). Hence, it models the axioms and the inference rules 1–17 and 21 introduced before (while, as mentioned in Section 2.4, axioms 18–20 are assumed to be enforced by the specification of agents behaviour). Consequently, the following properties hold too. Let $w \in W_c$, then:

- given $\varphi \in \mathcal{L}_{\textit{L-INF}}$, it holds that $\mathbf{K}_i\, \varphi \in w$ if and only if $\forall v \in W_c$ such that $wR_{c,i}v$ we have $\varphi \in v$;
- for $\varphi \in \mathcal{L}_{\textit{L-INF}}$, if $\mathbf{B}_i \varphi \in w$ and $wR_{c,i}v$ then $\mathbf{B}_i \varphi \in v$.

Thus, $R_{c,i}$-related worlds have the same knowledge and N_c-related worlds have the same beliefs, i.e. there can be $R_{c,i}$-related worlds with different beliefs.

By proceeding similarly to what done in [12], we obtain the proof of strong completeness. For lack of space, we list the main theorems but omit lemmas and proofs, that we have however developed analogously to what done in previous work [5].

Theorem 4.1. *L-INF is strongly complete for the class of L-INF models.*

Theorem 4.2. *L-DINF is strongly complete for the class of L-INF models.*

5. Conclusions

In this paper, we discussed the last advances of a line of work concerning how to exploit a logical formulation for a formal description of the cooperative activities of groups of agents. In past work, we had introduced beliefs about physical actions concerning whether they could, are, or have been executed, preferences in performing actions, single agent's and group's intentions. So far however, a limitation was missing about which actions an agent is allowed to perform; in practical situations, in fact, it will hardly be the case that every agent can perform every action.

We have listed some useful properties of the extended logic, that we have indeed proved, mainly strong completeness. Since the extension is small and modular, the complexity of the extended logic has no reason to be higher than that of the original *L-DINF*. In future work, we mean to extend our logic in the direction of describing multiple groups of agents and their interactions. We also mean to introduce in *L-DINF*, based on our past work, an explicit notion of time and time intervals.

References

[1] R. H. Bordini, L. Braubach, M. Dastani, A. E. F. Seghrouchni, J. J. Gómez-Sanz, J. Leite, G. M. P. O'Hare, A. Pokahr, A. Ricci, A survey of programming languages and platforms for multi-agent systems, Informatica (Slovenia) 30 (2006) 33–44.

[2] R. Calegari, G. Ciatto, V. Mascardi, A. Omicini, Logic-based technologies for multi-agent systems: a systematic literature review, Auton. Agents Multi Agent Syst. 35 (2021) 1. doi:10.1007/s10458-020-09478-3.

[3] A. Garro, M. Mühlhäuser, A. Tundis, M. Baldoni, C. Baroglio, F. Bergenti, P. Torroni, Intelligent agents: Multi-agent systems, in: S. Ranganathan, M. Gribskov, K. Nakai, C. Schönbach (Eds.), Encyclopedia of Bioinformatics and Computational Biology - Volume 1, Elsevier, 2019, pp. 315–320. doi:10.1016/b978-0-12-809633-8.20328-2.

[4] S. Costantini, V. Pitoni, Towards a logic of "inferable" for self-aware transparent logical agents, in: C. Musto, D. Magazzeni, S. Ruggieri, G. Semeraro (Eds.), Proc. XAI.it@AIxIA 2020, volume 2742 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2020, pp. 68–79. URL: http://ceur-ws.org/Vol-2742/paper6.pdf.

[5] S. Costantini, A. Formisano, V. Pitoni, An epistemic logic for multi-agent systems with budget and costs, in: W. Faber, G. Friedrich, M. Gebser, M. Morak (Eds.), Logics in Artificial Intelligence - 17th European Conference, JELIA 2021, Proceedings, volume 12678 of *LNCS*, Springer, 2021, pp. 101–115. doi:10.1007/978-3-030-75775-5_8.

[6] S. Costantini, A. Formisano, V. Pitoni, An epistemic logic for modular development of multi-agent systems, in: N. Alechina, M. Baldoni, B. Logan (Eds.), EMAS 2021, Revised Selected papers, volume 13190 of *LNCS*, Springer, 2022, pp. 72–91. doi:10.1007/978-3-030-97457-2_5.

[7] S. Costantini, A. Formisano, V. Pitoni, Timed memory in resource-bounded agents, in: C. Ghidini, B. Magnini, A. Passerini, P. Traverso (Eds.), AI*IA 2018 - Advances in Artificial Intelligence - XVIIth International Conference of the Italian Association for Artificial Intelligence, Proceedings, volume 11298 of *LNCS*, Springer, 2018, pp. 15–29.

[8] S. Costantini, V. Pitoni, Memory management in resource-bounded agents, in: M. Alviano, G. Greco, F. Scarcello (Eds.), AI*IA 2019 - Advances in Artificial Intelligence - XVIIIth International Conference of the Italian Association for Artificial Intelligence, volume 11946 of *LNCS*, Springer, 2019, pp. 46–58.

[9] V. Pitoni, S. Costantini, A temporal module for logical frameworks, in: B. Bogaerts, E. Erdem, P. Fodor, A. Formisano, G. Ianni, D. Inclezan, G. Vidal, A. Villanueva, M. De Vos, F. Yang (Eds.), Proc. of ICLP 2019 (Tech. Comm.), volume 306 of *EPTCS*, 2019, pp. 340–346.

[10] A. S. Rao, M. Georgeff, Modeling rational agents within a BDI architecture, in: Proc. of the Second Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'91), Morgan Kaufmann, 1991, pp. 473–484.

[11] H. V. Ditmarsch, J. Y. Halpern, W. V. D. Hoek, B. Kooi, Handbook of Epistemic Logic, College Publications, 2015. Editors.

[12] P. Balbiani, D. F. Duque, E. Lorini, A logical theory of belief dynamics for resource-bounded agents, in: Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems, AAMAS 2016, ACM, 2016, pp. 644–652.

[13] S. Costantini, A. Tocchio, A logic programming language for multi-agent systems, in: S. Flesca, S. Greco, N. Leone, G. Ianni (Eds.), Proc. of JELIA-02, volume 2424 of *LNAI*, Springer, 2002, pp. 1–13. doi:10.1007/3-540-45757-7_1.

[14] S. Costantini, A. Tocchio, The DALI logic programming agent-oriented language, in: J. J. Alferes, J. A. Leite (Eds.), Proc. of JELIA-04, volume 3229 of *LNAI*, Springer, 2004, pp. 685–688. doi:10.1007/978-3-540-30227-8_57.

[15] G. De Gasperis, S. Costantini, G. Nazzicone, DALI multi agent systems framework, doi 10.5281/zenodo.11042, DALI GitHub Software Repository, 2014. DALI: github.com/AAAI-DISIM-UnivAQ/DALI.

[16] S. Costantini, G. De Gasperis, Flexible goal-directed agents' behavior via DALI mass and ASP modules, in: 2018 AAAI Spring Symposia, Stanford University, Palo Alto, California, USA, March 26-28, 2018, AAAI Press, 2018.

[17] S. Costantini, G. De Gasperis, G. Nazzicone, DALI for cognitive robotics: Principles and prototype implementation, in: Y. Lierler, W. Taha (Eds.), Practical Aspects of Declarative Languages - 19th Int. Symp. PADL 2017, Proceedings, volume 10137 of *LNCS*, Springer, 2017, pp. 152–162. doi:10.1007/978-3-319-51676-9_10.