The Dynamics of Default Reasoning

B. van Linder W. van der Hoek J.-J. Ch. Meyer
Utrecht University
Department of Computer Science
P.O. Box 80.089
3508 TB Utrecht
The Netherlands
Email: bernd@cs.ruu.nl

Abstract

In this paper we look at default reasoning from a dynamic, agent-oriented, point of view. Using the KARO-architecture that we previously defined ([15, 16, 24, 25]), we introduce actions that model the (attempted) jumping to conclusions that is a fundamental part of reasoning by default. Application of such an action consists of three parts. First it is checked whether the formula that the agent tries to jump to is a default, thereafter it is checked whether the default formula can consistently be incorporated by the agent, and if this is the case the formula is included in the agent's beliefs. We define the ability and opportunity of agents to apply these actions, and the states of affairs after application. To formalize formulae being defaults, we introduce the modality of common possibility. This modality is related to, but not reducible to, the notions of common knowledge and 'everybody knows'-knowledge. To model the qualitative difference that exists between hard, factual knowledge and beliefs derived by default, we employ different modalities to represent these concepts, thus combining knowledge, beliefs, and defaults. Based on the concepts used to model the default reasoning of agents, we look into the dynamics of the supernormal fragment of default logic. We show in particular that by sequences of jumps to conclusions agents can end up with extensions in the sense of default logic of their belief.

Content Areas: Reasoning about Action, Knowledge Representation, Nonmonotonic Reasoning.

1. Introduction

The formalization of the behaviour of rational agents is a topic of continuing interest in Artificial Intelligence. Since the pioneering work of Moore ([35, 36]) in which knowledge and actions are considered, research has concentrated on belief and knowledge, and the problems of these two mental attitudes have received intense investigation (for a survey see [8, 34]). In recent years, research in AI has been dealing with other aspects of the agents' behaviour, including commitment, desire, intention, and obligation ([4, 18, 43, 45]).

In [15] we defined a framework in which the *knowledge* and *abilities* of agents, as well as the *opportunities* for and the *results* of their actions are formalized. In this so called KARO-architecture[1] it can for instance be modelled that an agent knows that some action

[1]The term KARO, for Knowledge – Abilities – Results – Opportunities, is chosen in analogy with the BDI-architectures of Rao & Georgeff ([39, 40, 41]).

is a *correct* plan to achieve some goal since it knows that performing the action will lead to the goal, and that it knows that an action is a *feasible* plan since the agent knows of its ability to perform the action. In subsequent research we extended our framework with nondeterministic actions ([16]), epistemic tests ([25]), and communicative actions ([24]).

The main aim of this paper is to formalize a form of default reasoning by rational agents in our framework. The capacity to reason by default is very important when modelling rational agents: in many situations agents lack certain knowledge concerning the situation at hand, but nevertheless have to take decisions. The formalization that we present here has two remarkable features. The first is the attention we pay to the *dynamic* part of default reasoning: agents may execute certain actions that model the jumping to conclusions that is a major part of the reasoning by default. The second remarkable feature is the use of the new modality of *common possibility* to define defaults. The idea underlying this definition is that defaults are formulae that by virtue of their plausibility are used to fill in gaps in the knowledge of an agent. A formula is considered to be plausible, and hence considered to be a default, if it is a common possibility. We argue on the basis of some lemmas that common possibility indeed models a useful kind of defaults.

Although we combine both features mentioned above in the framework presented in this paper, we would like to stress that these are by no means dependent on each other. The dynamic part depends in no way on the notion of common possibility to formalize defaults: any representation of defaults can be used in combination with the actions that constitute the dynamic part. And the notion of common possibility is also interesting in a purely epistemic context, without any reference to dynamics.

The combination of different modalities of knowledge and belief has been investigated by several authors (for example [14, 22, 23]), and the same can be said for the combination of knowledge and defaults ([6]). Nevertheless, the system that we end up with is probably the first one that combines *knowledge*, *beliefs* and furthermore treats *defaults* the way that we do. Furthermore our system offers a formalization of default reasoning from an *agent-oriented*, *semantic* point of view. However, the most important contribution of this paper is that it firmly places default reasoning in the *dynamic context* where, in our opinion, it belongs.

1.1. Organization of the paper

The rest of the paper is organized as follows.

In Sect. 2 we (re)introduce some of our ideas on knowledge, abilities, opportunities, and results. Furthermore the notion of epistemic updates is explained, and the formal definitions of the KARO-architecture are given. Sect. 3 deals with our approach towards default reasoning. We formally define defaults and the beliefs of agents in 3.2. In 3.3 the consistency checks necessary to ensure correctness of the jumps to conclusions are introduced, and in 3.4 the definitions of the action that models an attempt to jump to a conclusion are given. In 3.5 we look into the agents' abilities with respect to the jumping to conclusions actions. Some remarks on the notion of default plans and the part they play in the reasoning of agents concerning the correctness and feasibility of their plans conclude Sect. 3. In Sect. 4 we look into the dynamics of supernormal default logic. We show in particular that successive jumps to conclusions can be used by agents to end up with extensions in the sense of default logic of their original belief set. In Sect. 5 we summarize and discuss options for further research.

2. Knowledge, abilities, opportunities, and results

Formalizing *knowledge* has been a subject of research both in analytical philosophy and in AI for quite some time (cf. [8, 13, 34, 35]). In representing knowledge we follow, both from a syntactical and a semantic point of view, the approach common in epistemic logic: the formula $\mathbf{K}_i\varphi$ denotes the fact that agent i knows φ, and the semantics of knowledge is given by means of a possible worlds semantics.

An important aspect of any investigation of (human) action, is the relation that exists between *ability* and *opportunity*. In order to successfully complete an action, both the opportunity and the ability to perform the action are necessary. Although these notions are interconnected, they are surely not identical (cf. [20]). The abilities of agents can be seen as comprising mental and physical powers, moral capacities, and human and physical possibility. The opportunity to perform actions is best described by the notion of circumstantial possibility. A nice example that illustrates the difference between ability and opportunity is that of a lion in a zoo (cf. [7]): although the lion will never have the opportunity to eat a zebra, it certainly has the ability to do so. We propose that in order to make the behaviour of rational agents, like for instance robots, as realistic as possible, abilities and opportunities need also be distinguished in AI environments. In AI two approaches are common when it comes down to abilities. The first is to declare abilities fixed, and allow only actions of which the agent is able (cf. [43]). The major disadvantage of this approach is that it cannot be modelled that agents acquire abilities or that they loose abilities, even though these are rather natural notions: rational agents can acquire abilities by learning, and by performing ability-destructive actions (cf. [15]) agents may loose abilities[2]. The second approach is to somehow reduce abilities to opportunities (cf. for instance [45]). The major disadvantage of this approach is that one loses the possibility to reason about actions for which opportunity but no ability exist and *vice versa*, which provides for a considerable loss of flexibility compared to our approach. We formalize the abilities of agents via the $\mathbf{A}_i$ operator; the formula $\mathbf{A}_i\alpha$ denotes the fact that agent i has the ability to do α.

In defining the *result* of an action, we follow the ideas of [48], in which the state of affairs brought about by execution of the action is defined to be its result.

When using the definitions of opportunities and results as given above, the framework of (propositional) dynamic logic provides an excellent means to formalize these notions. If we use events $\mathrm{do}_i(\alpha)$ to refer to the performance of the action α by the agent i, the formula $\langle \mathrm{do}_i(\alpha)\rangle\varphi$ represents the fact that the agent i has the opportunity to do α and that doing α leads to φ. The formula $[\mathrm{do}_i(\alpha)]\varphi$ is noncommittal about the opportunity of the agent to do α but states that should the opportunity arise, only states of affairs satisfying φ would result. Besides the possibility to formalize both opportunities and results when using dynamic logic, another advantage lies in the compatibility of epistemic and dynamic logic from a semantic point of view: the possible world semantics can be used to provide meaning both to epistemic and dynamic notions.

[2] A typical real-life example of an ability-destructive action is the stinging of a bee: bees have a one-time ability to sting; when the opportunity to sting comes around a second time it cannot be taken.

2.1. Informative actions

When formalizing the behaviour of rational agents, the treatment of *informative* actions should receive due attention, not only since these actions frequently occur in real life, but also because of the importance of these actions for planning. The distinguishing feature of informative actions is that their effect is to *extend the information* that an agent has about the current situation. Typical examples of informative actions are sensing and observing, but also the jumping to conclusions which is an important part of default reasoning.

In [25] we show how the informative action that consists of testing (or observing) the real world can be formalized in our system; in [24] we deal with the transfer of knowledge through communication. The semantics of both actions is based on the use of *epistemic updates*, a notion we introduced in [25]. In this paper we use the analogous notion of *doxastic updates* to model the extension of beliefs that follows the application of defaults. The idea behind both these updates is the following. An informative action produces information concerning some proposition φ for the agent that executes the action. As such, this action puts an end to any uncertainties concerning φ that the agent may have, hereby reducing the number of epistemic (doxastic) alternatives that it has. Given this intuitive meaning of epistemic and doxastic updates, it is obvious that these actions cause transitions between pairs (Model, State), thus generalizing the usual actions from dynamic logic that cause inter-state transitions within a model. The actual formalization of epistemic updates in models that obey an **S5** axiomatization for knowledge, these are the models we considered in [24, 25], is based on the idea that an update of the knowledge of an agent i in a state s with a formula φ divides the original epistemic equivalence class of i in s into two new equivalence classes: one containing the epistemic alternatives that support φ, the other containing those that do not support φ (cf. [25]). In this way it is ensured that after the update, agent i *knows whether* φ in the state s.

The formalization of doxastic updates, and their use in the modelling of reasoning by default, is the subject of Sect. 3.

2.2. The KARO-architecture

Next the definitions of the KARO-architecture are given. The system that we define here is a modified version of the one given in [25].

2.2.1. The syntax

In addition to natural numbers representing the rational agents whose behaviour we are formalizing, we introduce a constant e in our language. The intuitive reading of this constant could either be *external environment/observer*, or *supervisor*. The idea is that this external environment performs some actions that are not for the agents to perform, but that appear in the implementation of actions which are. A typical example of such an action is the epistemic update: this update is used in the implementation of the epistemic test, but is not an action that the agent may perform. These special 'low-level' actions are not visible for the agents: when they reason about an action α in the implementation of which such a low-level action occurs, then α appears as an indivisible unit to the agent. Since these low-level actions are used only in determining results and opportunities of the agents' actions, it is of no use to discuss the ability of the external environment with regard to these actions.

2.1. DEFINITION. Let some finite set $\{1, \ldots, n\}$ of agents, and some denumerable sets Π of propositional symbols, and At of atomic actions be given. The language $\mathcal{L}$ and the class of actions Ac are defined by mutual induction as follows.

1. $\mathcal{L}$ is the smallest superset of Π such that
   - if $\varphi, \psi \in \mathcal{L}$ then $\neg\varphi, \varphi \vee \psi \in \mathcal{L}$,
   - if $i \in \mathcal{A}$, $\alpha \in Ac$ and $\varphi \in \mathcal{L}$ then $\mathbf{K}_i\varphi, \langle \mathrm{do}_i(\alpha)\rangle\varphi, \mathbf{A}_i\alpha \in \mathcal{L}$.

   The elements of $\mathcal{L}$ are called *formulae*. The purely propositional fragment of $\mathcal{L}$ is denoted by $\mathcal{L}_0$.
2. Ac is the smallest superset of At such that
   - if $\varphi \in \mathcal{L}$ then $\mathtt{confirm}\ \varphi \in Ac$,
   - if $\alpha_1 \in Ac$ and $\alpha_2 \in Ac$ then $\alpha_1; \alpha_2 \in Ac$,
   - if $\varphi \in \mathcal{L}$ and $\alpha_1, \alpha_2 \in Ac$ then $\mathtt{if}\ \varphi\ \mathtt{then}\ \alpha_1\ \mathtt{else}\ \alpha_2\ \mathtt{fi} \in Ac$,
   - if $\varphi \in \mathcal{L}$ and $\alpha_1 \in Ac$ then $\mathtt{while}\ \varphi\ \mathtt{do}\ \alpha_1\ \mathtt{od} \in Ac$,

   The elements of Ac are called *actions*.

The language $\mathcal{L}'$ and the class of actions Ac', formalizing the external environment part of our language, are defined as follows:

- $\mathcal{L}'$ is the set $\{\langle \mathrm{do}_e(\alpha')\rangle\varphi \mid \varphi \in \mathcal{L} \text{ and } \alpha' \in Ac'\}$ where
- Ac' is the set $\{\mathtt{ep\_update}\ (\vartheta, j) \mid \vartheta \in \mathcal{L}, j \in \mathcal{A}\}$

The constructs $\wedge$, $\rightarrow$, $\leftrightarrow$, **tt** and **ff** are defined as usual. Other additional constructs are introduced by definitional abbreviation:

$\mathbf{M}_i\varphi$	is	$\neg\mathbf{K}_i\neg\varphi$
$[\mathrm{do}_x(\alpha)]\varphi$	is	$\neg\langle \mathrm{do}_x(\alpha)\rangle\neg\varphi$ where $x \in \mathcal{A} \cup \{e\}$
$\mathtt{if}\ \varphi\ \mathtt{then}\ \alpha_1\ \mathtt{fi}$	is	$\mathtt{if}\ \varphi\ \mathtt{then}\ \alpha_1\ \mathtt{else\ skip\ fi}$
$\mathtt{skip}$	is	$\mathtt{confirm}\ \mathbf{tt}$
$\mathtt{fail}$	is	$\mathtt{confirm}\ \mathbf{ff}$
α^0	is	$\mathtt{skip}$
α^{n+1}	is	$\alpha; \alpha^n$

2.2.2. The semantics

2.2. DEFINITION. The class M of Kripke models contains all tuples $\mathcal{M} =< \mathcal{S}, \pi, \mathrm{R}, \mathtt{r}, \mathtt{c} >$ such that

1. $\mathcal{S}$ is a set of possible worlds, or states.
2. $\pi : \Pi \times \mathcal{S} \rightarrow \mathbf{bool}$ is a total function that assigns a truth value to propositional symbols in possible worlds.
3. $\mathrm{R} : \mathcal{A} \rightarrow \wp(\mathcal{S} \times \mathcal{S})$ is a function that yields the epistemic accessibility relations for a given agent. Since we assume to deal with **S5** models, it is demanded that $\mathrm{R}(i)$ is an equivalence relation for all i. For reasons of practical convenience we define $[s]_{\mathrm{R}(i)} \stackrel{\mathrm{def}}{=} \{s' \in \mathcal{S} \mid (s, s') \in \mathrm{R}(i)\}$ to be the $\mathrm{R}(i)$ equivalence class of s.
4. $\mathtt{r} : \mathcal{A} \times At \rightarrow \mathcal{S} \rightarrow \wp(\mathcal{S})$ is such that $\mathtt{r}(i, a)(s)$ yields the (possibly empty) state transition in s caused by the event $\mathrm{do}_i(a)$. This function is such that for all atomic actions a it holds that $|\mathtt{r}(i, a)(s)| \leq 1$ for all i and s, i.e., these events are *deterministic*.
5. $\mathtt{c} : \mathcal{A} \times At \rightarrow \mathcal{S} \rightarrow \mathbf{bool}$ is the capability function such that $\mathtt{c}(i, a)(s)$ indicates whether the agent i is capable of performing the action a in s.

2.3. DEFINITION. Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathtt{r}, \mathtt{c} \rangle$ be some Kripke model. The set $\mathcal{M}_{\subseteq}$ of *epistemic sub-models* of $\mathcal{M}$ is defined by:

$$\mathcal{M}_{\subseteq} = \{< \mathcal{S}, \pi, \mathrm{R}', \mathtt{r}, \mathtt{c} > \in \mathsf{M} \mid \forall i \in \mathcal{A}[\mathrm{R}'(i) \subseteq \mathrm{R}(i)]\}$$

2.4. DEFINITION. Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathtt{r}, \mathtt{c} \rangle$ be some Kripke model, let $\mathcal{S}' \subseteq \mathcal{S}$, and φ be some formula.

- $\mathrm{Cl}_{\mathrm{eq}}(\mathcal{S}') \stackrel{\mathrm{def}}{=} \mathcal{S}' \times \mathcal{S}'$.
- $[\![\varphi]\!] \stackrel{\mathrm{def}}{=} \{s \in \mathcal{S} \mid \mathcal{M}, s \models \varphi\}$.

2.5. DEFINITION. Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathtt{r}, \mathtt{c} \rangle$ be some Kripke model from M. For propositional symbols, negated formulae and disjunctions, $\mathcal{M}, s \models \varphi$ is inductively defined as usual. For the other clauses $\mathcal{M}, s \models \varphi$ is defined as follows:

$$\begin{array}{ll}
\mathcal{M}, s \models \mathbf{K}_i\varphi & \Leftrightarrow \forall s' \in \mathcal{S}[(s,s') \in \mathrm{R}(i) \Rightarrow \mathcal{M}, s' \models \varphi] \\
\mathcal{M}, s \models \langle \mathrm{do}_i(\alpha) \rangle \varphi & \Leftrightarrow \exists \mathcal{M}', s'[\mathcal{M}', s' \in \mathtt{r}(i,\alpha)(\mathcal{M}, s)\,\&\,\mathcal{M}', s' \models \varphi] \\
\mathcal{M}, s \models \mathbf{A}_i\alpha & \Leftrightarrow \mathtt{c}(i,\alpha)(\mathcal{M}, s) = \mathbf{1}
\end{array}$$

where $\mathtt{r}$ and $\mathtt{c}$ are defined by:

$$\begin{array}{ll}
\mathtt{r} & : \ (\mathcal{A} \times Ac) \cup (\{e\} \times Ac') \to (\mathcal{M}_{\subseteq} \times \mathcal{S}) \cup \mathcal{S} \to \wp(\mathcal{M}_{\subseteq} \times \mathcal{S}) \\
\mathtt{r}(i,a)(\mathcal{M}', s) & = \mathcal{M}', \mathtt{r}(i,a)(s) \\
\mathtt{r}(i, \mathtt{confirm}\ \varphi)(\mathcal{M}', s) & = \{(\mathcal{M}', s)\} \text{ if } \mathcal{M}', s \models \varphi \\
 & \quad \emptyset \ \text{ otherwise} \\
\mathtt{r}(i, \alpha_1; \alpha_2)(\mathcal{M}', s) & = \mathtt{r}(i,\alpha_2)(\mathtt{r}(i,\alpha_1)(\mathcal{M}', s)) \\
\mathtt{r}(i, \mathtt{if}\ \varphi\ \mathtt{then}\ \alpha_1 & = \mathtt{r}(i,\alpha_1)(\mathcal{M}', s) \text{ if } \mathcal{M}', s \models \varphi \\
\quad \mathtt{else}\ \alpha_2\ \mathtt{fi})(\mathcal{M}', s) & \quad \mathtt{r}(i,\alpha_2)(\mathcal{M}', s) \text{ otherwise} \\
\mathtt{r}(i, \mathtt{while}\ \varphi\ \mathtt{do}\ \alpha_1\ \mathtt{od})(\mathcal{M}', s) & = \{(\mathcal{M}'', s') \mid \exists k \in \mathbb{N} \exists \mathcal{M}_0, s_0 \ldots \exists \mathcal{M}_k, s_k \in \mathcal{S} \\
 & \quad [\mathcal{M}_0, s_0 = \mathcal{M}', s\,\&\,\mathcal{M}_k, s_k = \mathcal{M}'', s'\,\&\,\forall j < k \\
 & \qquad [\mathcal{M}_{j+1}, s_{j+1} \in \mathtt{r}(i, \mathtt{confirm}\ \varphi; \alpha_1)(\mathcal{M}_j, s_j)] \\
 & \qquad \&\,\mathcal{M}'', s' \models \neg\varphi]\}
\end{array}$$

and

$$\begin{array}{ll}
\mathtt{r}(e, \mathtt{ep_update}\ (\vartheta, i))(\mathcal{M}', s) & = \mathcal{M}'', s \text{ where} \\
\mathcal{M}'' = \langle \mathcal{S}'', \pi'', \mathrm{R}'', \mathrm{B}'', \mathtt{r}'', \mathtt{c}'' \rangle \in \mathcal{M}_{\subseteq} \text{ with} & \\
\quad \mathcal{S}'' = \mathcal{S}', \pi'' = \pi', \mathtt{r}'' = \mathtt{r}', \mathtt{c}'' = \mathtt{c}' \text{ and} & \\
\quad \mathrm{R}''(i') = \mathrm{R}'(i') \text{ for } i' \neq i, & \\
\quad \mathrm{R}''(i) = (\mathrm{R}'(i) \setminus \mathrm{Cl}_{\mathrm{eq}}([s]_{\mathrm{R}'(i)})) \cup (\mathrm{Cl}_{\mathrm{eq}}([s]_{\mathrm{R}'(i)}) \cap [\![\vartheta]\!]) \cup (\mathrm{Cl}_{\mathrm{eq}}([s]_{\mathrm{R}'(i)}) \cap [\![\neg\vartheta]\!]) & \\
\text{where } \mathtt{r}(i,\alpha)(\emptyset) & = \emptyset
\end{array}$$

and

$$\begin{array}{ll}
\mathtt{c} & : \ \mathcal{A} \times Ac \to (\mathcal{M}_{\subseteq} \times \mathcal{S}) \cup \mathcal{S} \to \mathbf{bool} \\
\mathtt{c}(i,a)(\mathcal{M}', s) & = \mathtt{c}(i,a)(s) \\
\mathtt{c}(i, \mathtt{confirm}\ \varphi)(\mathcal{M}', s) & = \mathbf{1} \text{ if } \mathcal{M}', s \models \varphi \\
 & \quad \mathbf{0} \text{ otherwise} \\
\mathtt{c}(i, \alpha_1; \alpha_2)(\mathcal{M}', s) & = \mathtt{c}(i,\alpha_1)(\mathcal{M}', s)\,\&\,\mathtt{c}(i,\alpha_2)(\mathtt{r}(i,\alpha_1)(\mathcal{M}', s)) \\
\mathtt{c}(i, \mathtt{if}\ \varphi\ \mathtt{then}\ \alpha_1 & = \mathtt{c}(i, \mathtt{confirm}\ \varphi; \alpha_1)(\mathcal{M}', s) \text{ or} \\
\quad \mathtt{else}\ \alpha_2\ \mathtt{fi})(\mathcal{M}', s) & \quad \mathtt{c}(i, \mathtt{confirm}\ \neg\varphi; \alpha_2)(\mathcal{M}', s) \\
\mathtt{c}(i, \mathtt{while}\ \varphi\ \mathtt{do}\ \alpha_1\ \mathtt{od})(\mathcal{M}', s) & = \mathbf{1} \text{ if } \exists k \in \mathbb{N}[\mathtt{c}(i, (\mathtt{confirm}\ \varphi; \alpha_1)^k; \\
 & \qquad \mathtt{confirm}\ \neg\varphi)(\mathcal{M}', s) = 1] \\
 & = \mathbf{0} \text{ otherwise} \\
\text{where } \mathtt{c}(i,\alpha)(\emptyset) & = \mathbf{1}.
\end{array}$$

Satisfiability and validity are defined as usual.

2.6. REMARK. Note that no definition of the ability for the epistemic update function is given, since this action is only used in the implementation of other actions and is not available to agents for execution.

The definition of $\mathtt{r}(e, \mathtt{ep\_update}\ (\vartheta, j))$ is based on the intuition given above; the definition of the function $\mathtt{r}$ for the other actions is the standard one for the class of deterministic while programs (see for instance [9]).

With regard to the abilities of agents, the motivation for the choices made in Def. 2.5 is the following. An agent is capable of performing a sequential composition $\alpha_1; \alpha_2$ iff it is capable of performing α_1 and it is capable of executing α_2 after it has performed α_1. The definition of $\mathtt{c}(i, \mathtt{confirm}\ \varphi)(s)$ is based on the idea that an agent is able to get confirmation for a formula φ if and only if this formula holds. Note that the definitions of $\mathtt{r}(i, \mathtt{confirm}\,\varphi)$ and $\mathtt{c}(i, \mathtt{confirm}\,\varphi)$ express an interesting relation between circumstances, opportunities, and abilities: in circumstances such that φ holds, the agents both have the opportunity and the ability to confirm φ. An agent is capable of performing a conditional composition, if it is able to either get confirmation for the condition and thereafter perform the then-part, or it is able to confirm the negation of the condition and perform the else-part afterwards. Lastly, an agent is capable of performing a repetitive composition $\mathtt{while}\ \varphi\ \mathtt{do}\ \alpha_1\ \mathtt{od}$ iff it is able to perform the action $(\mathtt{confirm}\ \varphi; \alpha_1)^k; \mathtt{confirm}\ \neg\varphi$ for some $k \in \mathbb{N}$.

2.2.3. Modelling tests and communication via epistemic updates

To give an impression as to how epistemic (and later on doxastic) updates are used in modelling informative actions, we look globally at epistemic tests (cf. [25]) and communication (cf. [24]).

By performing an epistemic test, which roughly corresponds to an observation, for some formula φ, agents aim at acquiring knowledge whether φ. If φ is the case in the state where the test is performed the agent acquires the knowledge that φ, otherwise it knows $\neg\varphi$ after testing. To capture the intuitive idea of observations, it is demanded that tests can only be performed for propositional formulae. The actual implementation using epistemic updates is defined as follows.

$$\mathtt{r}(i, \mathtt{test}\ \varphi)(\mathcal{M}, s) = \mathtt{r}(e, \mathtt{ep\_update}\ (\varphi, i)))(\mathcal{M}, s)$$

We showed that formalizing tests in this way results in an intuitively acceptable framework (cf. [25]).

With regard to communication we defined a send action that roughly behaves as follows: if some agent i requests a formula φ from agent j, and it is both the case that j knows φ and that i trusts j on φ, then j transfers its knowledge on φ to agent i. The implementation of the send action is informally defined as follows.

- $\mathtt{r}(j, \mathtt{send}(i, \varphi))(\mathcal{M}, s) = \mathtt{r}(e, \mathtt{ep\_update}\ (\varphi, i))(\mathcal{M}, s)$
  if $\mathcal{M}, s \models \mathbf{K}_i\varphi$ and i trusts j on φ and has made a request for φ.

The doxastic updates, which we use to model *belief* updates instead of knowledge updates, are based on the same ideas as the epistemic updates presented here.

2.2.4. Some remarks on completeness, decidability, and complexity

In [25] we presented a complete axiomatization of the basic logic system without informative actions. One of the main features of this axiomatization is the presence of *infinitary* rules used to capture the infinite character of the repetitive composition. Due to these infinitary rules, decidability cannot be proved in a standard way (cf. [19]). Although we expect the basic logic system without informative actions to be decidable, proving this fact remains an open problem, and the same can be said for the complete system containing informative actions.

The basic logic without informative actions can be seen as the join of an **S5** logic for n agents joined with a deterministic propositional dynamic logic for n agents, and with an ability component added to it. The satisfiability problem for an **S5** logic for $n \geq 2$ agents is known to be PSPACE complete ([11]), and satisfiability in (agentless) propositional dynamic logic with deterministic while-loops is also known to be PSPACE complete ([12]). Since joining logics only enlarges complexity (cf. [44]), the satisfiability problem for the complete system including informative actions is at least PSPACE complete.

2.2.5. A comparison with other architectures

Several other formalizations of multi-agent frameworks have been proposed in the literature on AI. Here we briefly mention some of these formalizations, and point out differences with our KARO-architecture.

The first of these systems was probably the one defined by Moore. Moore was not only the first to introduce the logic of knowledge into AI, but was also the first to combine a logic of knowledge and action ([35, 36]). His main concern was the study of knowledge-preconditions for actions - the question what agents need to know in order to be able to perform some action. Moore defines a notion of ability in terms of 'know how': the knowledge of how to perform an action is defined to denote the ability to perform the action. Moore uses quantifying in in a first-order framework to define that an agent knows what an action is. In our opinion the main drawback of this notion of ability lies in the fact that it cannot account for agents that are able to perform actions without knowing how to perform the action. This property is for instance typical for human(oid) agents: these agents continually perform actions (walking, talking etc.) without knowing how they do this. The developments in research on epistemic and dynamic logic that followed Moore's work, make this first system look a little bit out of date now: the notation seems unnecessarily complex, and the system is altogether rather difficult to understand.

In more recent years, the system of Cohen & Levesque ([4]) has been very influential. Syntactically, their system is a many-sorted, quantified, multi-modal logic with equality, containing four primary modalities: one representing the beliefs of an agent, one representing the agents' goals, one modality indicating that some action will happen next, and one indicating that some action has just happened. The concepts of intention and commitment are defined in terms of these primary modalities. Semantically, Cohen & Levesque use a possible worlds model. Some obvious differences exist between our approach and the one of Cohen & Levesque. First of all, the primary modalities of their system differ completely from ours: whereas we consider knowledge, abilities, results, and opportunities, Cohen & Levesque deal with belief, goals, and a sort of future and past notion of combined ability and opportunity. It is furthermore the case that their system has a '*temporal* logic' flavour

to it, whereas ours is typically '*dynamic* logic' of nature.

The system of Rao & Georgeff ([39, 40, 41]) also deals with formalizing the behaviour of rational agents. Their framework is based on three primitive modalities: beliefs, desires, and intentions. Semantically their formalism is based on a branching model of time, in which belief-, desire- and intention-accessible worlds are themselves branching time structures. The emphasis of their approach lies within formalizing the revision of intentions, beliefs, and goals. The system of Rao & Georgeff differs at least in two obvious aspects. Firstly, their choice of fundamental notions differs from ours. Secondly, again the underlying framework is typically a temporal logic one, and not a dynamic logic one.

Although not really a logic-based formalization, the research on agent-oriented programming as initiated by Shoham ([43]) also deals with the behaviour of rational agents. Shoham's framework is meant to provide a new programming paradigm, based on a societal view of computation. A logical system with clear syntax and semantics should underlay any AOP system. The logical system that Shoham defines in [43] contains three primary concepts, namely belief, obligation (or commitment) and capability. Due to the absence of actions in Shoham's language, the notion of capability that Shoham uses is completely different from ours. However it bears some resemblance to the notion of feasibility and correctness that we formalized in our Can-predicate (see [15]): where we state that some agent has a plan that is known to be correct and feasible to achieve a given goal φ, Shoham would say that this agent is capable of (achieving) φ. In the work of Thomas ([45]) a logical system is given to provide for a formal foundation of a part of the agent-oriented programming language. The basic notions that Thomas considers are beliefs, capabilities, plans and intentions. Although capability is claimed to be primitive, it is in fact defined as future opportunity. Semantically Thomas uses a possible worlds semantics based on an underlying branching-time structure, in which various atomic actions provide for the branching. Although Thomas' framework resembles ours in her choice of primitive modalities, there are some important differences. First of all, as mentioned above Thomas' notion of ability differs from ours, and is not really primitive. Secondly, only atomic actions are considered, whereas our framework basically deals with the compositional behaviour of complex actions. Lastly, although actions are explicitly present in the language, the whole system still has a temporal logic, and not a dynamic logic, flavour to it.

Summarizing, two major differences between our approach and other well known approaches can be pointed out. Firstly, none of the other systems mentioned above deals with abilities as a really primitive notion. We treat ability as a first-class citizen, and propose that the advantages of doing so in terms of an enhanced flexibility and expressiveness, visible for instance in the formalization of learning and forgetting, are substantial. A second remarkable difference lies in the fact that we use dynamic logic as the basic underlying system whereas the systems mentioned above (with the exception of Moore's system) use some sort of temporal logic for this goal. In our opinion a logic of action should be able to express both results and opportunities of actions, which was our main reason to choose for a dynamic and not for a temporal logic. The main advantage of using dynamic logic, however, is that our state transitions bear less weight than those in a temporal logic framework. Transitions in a dynamic framework do not fix a number of possible futures, but are used to reason about results and opportunities. As such, the incorporation of informative actions seems to be much more straightforward in a dynamic logic based framework, than in a temporal logic based one.

3. Defaults and dynamics

As mentioned in Sect. 1, the capacity to reason by default is very important when modelling rational agents. In default reasoning, reliable yet fallible conclusions are derived on the basis of the presence of certain information and the absence of other information. A natural way to look at the kind of derivations that are common in default reasoning is to regard these as 'jumps to conclusions'. Whereas in classical logic derivations can be considered to 'walk to a conclusion' by taking one well-argued, secure step at a time, in default reasoning one jumps to conclusions that one in general cannot walk to.

Usually agents know certain formulae to be true, other formulae to be false, and are uncertain with respect to still other formulae. The agent may try to fill in the gaps in its knowledge by trying to jump to certain conclusions. These attempted jumps are modelled as explicit actions, `try_jump` φ for 'try to jump to φ', in our dynamic framework. The execution of such a `try_jump` φ action consists of three stages. In the first stage it is checked whether the formula φ is a suitable candidate to jump to. The formulae that are suitable candidates for a given agent in a given state are called its *defaults*. These formulae intuitively correspond to *supernormal* defaults –these are prerequisite-free, normal defaults– in Reiter's default logic ([42]) and *possible hypotheses* in Poole's system ([38]). If the formula that the agent is trying to jump to is a default, the second stage of execution follows; if the formula is not a default, the jump fails. In the second stage of execution it is checked whether the default can consistently be adopted by the agent to fill in some gap in its knowledge (due to the extreme flexibility of our framework the actual consistency check is more elaborate; exact details can be found in 3.3). If the default cannot be adopted consistently, the jump reduces to the empty action skip; otherwise the default is actually adopted, which constitutes the third stage of execution.

It is important to notice that in our opinion formulae derived by default are of a different nature than facts *known* to be true . In this we agree with several other formalizations of default reasoning, like for instance NML3 ([6]), EDL ([31, 32, 33]) and DIL ([26, 27, 37]). To accommodate for this qualitative difference between known facts and beliefs, we introduce a new modality, representing the beliefs (typically acquired by default) that an agent has. The implementation of the `try_jump` action is such that formulae derived by default are included in the agent's *beliefs*, and not in its knowledge.

3.1. Formalizing defaults: defaults as common possibilities

In the intuitive ideas expressed above, an important part is played by the defaults, the formulae that may be jumped to. An obvious approach towards formalizing these defaults would be to extend the Kripke models $\mathcal{M}$ with some additional δ function, that yields for each agent in each state of $\mathcal{M}$ the formulae that are candidates for jumping. In this way each agent is equipped with its predefined set of defaults. Although this works out properly, we would like to propose another approach towards the formalization of defaults. We want to formalize defaults by using concepts already present in our KARO-architecture. In this way the process of reasoning by default is formalized in its entirety within our KARO- architecture. In order to do this we need a modal (epistemic) translation of defaults that somehow does justice to the empirical character of defaults. If one looks at some of the modal (epistemic) translations of defaults that have been proposed in the literature ([21, 28, 29, 30, 46]), it turns out that for supernormal defaults $\mathbf{tt} : \varphi/\varphi$ in an **S5**

framework all of these translations amount to either the formula $\mathbf{M}_i\varphi \rightarrow \varphi$ or $\mathbf{M}_i\varphi \rightarrow \mathbf{K}_i\varphi$. These epistemic translations stem from the usual, static, account of default reasoning and are therefore not completely suitable for our goals. Intuitively, our notion of defaults corresponds to the premises of both the implications given above, whereas the conclusions of these implications are (possibly) brought about by execution of the `try_jump` φ action. Hence the obvious thing to do would be to consider the premises of the implications $\mathbf{M}_i\varphi \rightarrow \varphi$ and $\mathbf{M}_i\varphi \rightarrow \mathbf{K}_i\varphi$, viz. the formula $\mathbf{M}\varphi$, as a candidate to represent our kind of defaults. In that way agents would consider each formula that they do not know to be false to be a default. However in our opinion this formalization would do not justice to the empirical character of defaults. More in particular, the idea of defaults being rooted in *common* sense, which is generally accepted to form the roots of defaults, is not visible when formalizing defaults as ordinary epistemic possibilities. In our multi-agent KARO-architecture *common* sense is related to the knowledge and lack of knowledge of *all* agents. To capture this idea of defaults as determined by the (lack of) knowledge of all agents, we propose the modality of *common possibility*. The intuitive interpretation of some formula φ being a common possibility is that it is considered possible by all agents, i.e., none of the agents knows φ to be false. If one considers epistemic possibility to correspond to the natural language 'might' (cf. [47]), common possibility could be interpreted as a *'uniformly supported might'*: if φ is a common possibility, then each agent accepts that φ might be the case. A real-life example of such a uniformly supported might is for instance the statement 'there is a block of 99 consecutive nines in the decimals of π', since nowhere in the assembled knowledge of all humans is there any fact contradicting this statement. For the same reason the statement 'there is no block of 99 consecutive nines in the decimals of π' is a uniformly supported might. Also both '$P = NP$' and '$P \neq NP$' are uniformly supported mights. A non-mathematical example is that, assuming that Ronald Reagan -being one of a group of human agents- was aware of his own knowledge, exactly one of the statements 'Reagan was aware of Iran-Contra gate' and 'Reagan was not aware of Iran-Contra gate' is a uniformly supported might. It is not possible that both are, since Reagan has knowledge contradicting one of these statements. In this case the agents that do not know which one of these statements is a uniformly supported might are not able to jump to any conclusion concerning Reagan's awareness (see also 3.5). In these examples common possibilities/uniformly supported mights have an *optimistic* flavour to them: agents that jump to these formulae seem a little hazardous. In fact however these agents are not that bold at all: there is no other agent that could tell them that their jumps are made for incorrect reasons. So not only does the jumping agent itself not have any knowledge contradicting the default, but also is it not possible to acquire this kind of knowledge through communication with other agents.

3.2. Defining common possibilities and beliefs

The common possibility operator $\mathbf{N}$ is defined in accordance with the intuition given in the previous section.

3.1. DEFINITION. For all formulae φ, the formula $\mathbf{N}\varphi$, for nobody knows not φ, is defined by:

$$\mathbf{N}\varphi \stackrel{\text{def}}{=} \mathbf{M}_1\varphi \wedge \ldots \wedge \mathbf{M}_n\varphi$$

3.2. REMARK. Note that for single-agent frameworks common possibility is just epistemic possibility. Except when explicitly stated otherwise it is assumed throughout this paper that we deal with genuine multi-agent situations.

3.3. REMARK. Although the common possibility operator is intuitively related to the modalities of 'everybody knows' and common knowledge (cf. [8, 34]), it is in fact an independent modality that cannot be reduced to one of these other modalities.

The formalization of belief, typically acquired by default, as we present it, combines two approaches from the literature, viz. [22] and [31]. Following [22], and thereby also [13], we consider the main difference between knowledge and belief to be the veridicality of knowledge. That is, if an agent *knows* φ, then φ must be true, but when an agent *beliefs* φ, φ need not be the case.

Syntactically, our approach is based on the one of [22].

3.4. DEFINITION. The language $\mathcal{L}$ as given in Def. 2.1 is extended with the operators $\mathbf{B}_i$ for all agents i. The formula $\mathbf{B}_i\varphi$ represents the belief of the agent i in φ. Formulae without occurrences of any $\mathbf{B}_i$ operator are called *B-free*, *B-objective* or simply *objective*.

The semantics of the belief operator as we give it, is conceptually different, yet essentially identical, to the one given in [22]. Instead of an accessibility relation, in [22] denoted by T, we use a set of designated worlds that together constitute the body of belief of the agent. Defining the semantics like this has two advantages as compared with the definitions of [22]. The first is merely an intuitive one: by using a set, the worlds that define the beliefs of an agent have become more tangible. One can see at a single glance how the beliefs of the agent are determined, without examining a possibly complex accessibility relation. The second advantage is of a more technical nature. As we will see in Sect. 3.4, the technique of doxastic updates can more easily be applied when belief is interpreted using sets instead of accessibility relations.

3.5. DEFINITION. The Kripke models as given in Def. 2.2 are extended with a function $\mathrm{B} : \mathcal{A} \times \mathcal{S}/\mathrm{R} \to \wp(\mathcal{S})$, such that for all agents i, i' and sets $[s]_{\mathrm{R}(i)}$ it holds that:

- $\mathrm{B}(i, [s]_{\mathrm{R}(i')})$ is undefined if $i \neq i'$.
- $\mathrm{B}(i, [s]_{\mathrm{R}(i)}) \subseteq [s]_{\mathrm{R}(i)}$,
- $\mathrm{B}(i, [s]_{\mathrm{R}(i)}) \neq \emptyset$.

Truth of belief formulae in states of a model is defined as follows:

$$\mathcal{M}, s \models \mathbf{B}_i\varphi \Leftrightarrow \forall s' \in \mathrm{B}(i, [s]_{\mathrm{R}(i)})[\mathcal{M}, s' \models \varphi]$$

When defining the semantics of belief as done in 3.5, we still obtain the desirable properties of the notion of belief of [22]. The following lemma states that the set of designated (belief) worlds defines a doxastic accessibility relation in the sense of [22] and *vice versa*.

3.6. LEMMA. *Let* $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}, \mathtt{r}, \mathtt{c} \rangle$ *be a Kripke model.*

- *Let the semantics for the* $\mathbf{B}_i$ *operator be as in Def. 3.5. Let for all agents* $i \in \mathcal{A}$ *the relation* $\mathrm{T}(i) \subseteq \mathcal{S} \times \mathcal{S}$ *be defined by:*

  $$\mathrm{T}(i) = \{(t, u) \in \mathcal{S} \times \mathcal{S} \mid \exists s \in \mathcal{S}[t \in [s]_{\mathrm{R}(i)} \,\&\, u \in \mathrm{B}(i, [s]_{\mathrm{R}(i)})]\}$$

  Then for all agents $i \in \mathcal{A}$*:*

1. $\mathrm{T}(i)$ *is serial*
2. $\mathrm{T}(i) \subseteq \mathrm{R}(i)$
3. *for all* $s,t,u \in \mathcal{S}$*: if* $(s,t) \in \mathrm{R}(i)$ *and* $(t,u) \in \mathrm{T}(i)$*, then* $(s,u) \in \mathrm{T}(i)$

- *Let the semantics for the* $\mathbf{B}_i$ *operator be given by a serial relation* $\mathrm{T}(i)$ *such that* $\mathrm{T}(i) \subseteq \mathrm{R}(i)$ *and for all* $s,t \in \mathcal{S}$*, if* $(s,t) \in \mathrm{R}(i)$ *and* $(t,u) \in \mathrm{T}(i)$*, then* $(s,u) \in \mathrm{T}(i)$*. Define for all agents* i *and for all* $[s]_{\mathrm{R}(i)}$*,* $\mathrm{B}(i,[s]_{\mathrm{R}(i)}) = \{s' \in \mathcal{S} \mid (s,s') \in \mathrm{T}(i)\}$*. Then:*
  1. $\mathrm{B}(i,[s]_{\mathrm{R}(i)}) \neq \emptyset$
  2. $\mathrm{B}(i,[s]_{\mathrm{R}(i)}) \subseteq [s]_{\mathrm{R}(i)}$

3.2.1. Knowledge, belief, and common possibilities

In this paragraph we look more deeply into the epistemic and doxastic notions that we introduced thus far. In particular we establish some relations that exists between these notions.

As already remarked in Def. 2.2, knowledge obeys an **S5** axiomatization, this in accordance with common practice in AI (cf. [8, 22, 34]).

The belief operator $\mathbf{B}_i$ satisfies a **KD45** axiomatization (this according to the Chellas qualification given in [3]). This means in particular that agents do not hold false beliefs, and they have both positive and negative introspection on their beliefs.

3.7. Lemma. *For all agents* i *and formulae* φ *and* ψ *we have:*

- $\models \mathbf{B}_i(\varphi \to \psi) \to (\mathbf{B}_i\varphi \to \mathbf{B}_i\psi)$
- $\models \neg\mathbf{B}_i\mathbf{ff}$
- $\models \mathbf{B}_i\varphi \to \mathbf{B}_i\mathbf{B}_i\varphi$
- $\models \neg\mathbf{B}_i\varphi \to \mathbf{B}_i\neg\mathbf{B}_i\varphi$
- $\models \varphi \Rightarrow \models \mathbf{B}_i\varphi$

The relation between knowledge and belief is as in the system of [22]. It is in particular the case that knowledge is stronger than belief, that the belief in knowledge is equivalent with the knowledge itself, and that knowledge of belief and belief are equivalent notions.

3.8. Lemma. *For all agents* i *and formulae* φ *and* ψ *we have:*

- $\models \mathbf{K}_i\varphi \to \mathbf{B}_i\varphi$
- $\models \mathbf{K}_i\neg\varphi \to \neg\mathbf{B}_i\varphi$

3.9. Definition ([14]). A formula φ is *i-doxastic sequenced* if there is a formula ψ, and operators $\mathbf{X}_1,\ldots,\mathbf{X}_m \in \{\mathbf{K}_i,\mathbf{B}_i,\neg\mathbf{K}_i,\neg\mathbf{B}_i\}$ and $m > 0$ such that $\varphi = \mathbf{X}_1\ldots\mathbf{X}_m\psi$.

3.10. Lemma ([14]). *Let* φ *be an i-doxastic sequenced formula. Then:*

- $\models \mathbf{K}_i\varphi \leftrightarrow \varphi$
- $\models \mathbf{B}_i\varphi \leftrightarrow \varphi$

3.11. Remark. Besides the desirable properties expressed in Lemma 3.8 and Lemma 3.10, our system also has some of the undesirable properties of the system of [22]. In particular, we inherit the property that the desirable axiom $\mathbf{B}_i\varphi \to \mathbf{B}_i\mathbf{K}_i\varphi$ cannot be added to our system without collapsing knowledge and belief. In [14] it is investigated how the underlying axiomatizations for knowledge and belief could be modified in order to allow

the axiom $\mathbf{B}_i\varphi \to \mathbf{B}_i\mathbf{K}_i\varphi$ without causing a collapse. It turns out that in order to allow for this axiom, other desirable axioms have to be discarded. For instance, removing negative introspection for knowledge would allow $\mathbf{B}_i\varphi \to \mathbf{B}_i\mathbf{K}_i\varphi$ to be included as an axiom. We are not convinced that presence of this axiom outweighs any of the losses that it causes, and therefore decide to use the 'standard' system of Kraus & Lehmann.

Not surprisingly, our common possibility operator $\mathbf{N}$ shares some of the properties of the epistemic possibility operator $\mathbf{M}_i$. In particular, $\mathbf{N}$ satisfies the dual **KT4** axiomatization, but satisfies only one direction of the dual **5** axiom. However, whereas the epistemic possibility operator satisfies the (in general undesired) axiom of weak belief $\mathbf{M}_i(\varphi \vee \psi) \leftrightarrow (\mathbf{M}_i\varphi \vee \mathbf{M}_i\psi)$ (cf. [17]), the common possibility operator satisfies only the right-to-left implication.

3.12. Lemma. *For all formulae φ and ψ we have:*

1. $\models \varphi \to \mathbf{N}\varphi$
2. $\models \mathbf{N}\varphi \vee \mathbf{N}\neg\varphi$
3. $\models \mathbf{N}\mathbf{N}\varphi \to \mathbf{N}\varphi$
4. $\models \neg\mathbf{N}\varphi \to \mathbf{N}\neg\mathbf{N}\varphi$
5. $\not\models \mathbf{N}\neg\mathbf{N}\varphi \to \neg\mathbf{N}\varphi$
6. $\not\models \mathbf{N}(\varphi \vee \psi) \to (\mathbf{N}\varphi \vee \mathbf{N}\psi)$
7. $\models \mathbf{N}\varphi \to \mathbf{N}(\varphi \vee \psi)$
8. $\models \varphi \to \psi \Rightarrow \models \mathbf{N}\varphi \to \mathbf{N}\psi$
9. $\not\models \mathbf{N}\varphi \wedge \mathbf{N}(\varphi \to \psi) \to \mathbf{N}\psi$
10. $\not\models \mathbf{N}\varphi \wedge \mathbf{N}\psi \to \mathbf{N}(\varphi \wedge \psi)$
11. $\not\models \mathbf{N}\varphi \to \neg\mathbf{N}\neg\varphi$

3.13. Remark. The properties formalized in Lemma 3.12 indicate that $\mathbf{N}\varphi$ is an acceptable candidate to represent defaults. Clause 1 states that true formulae are defaults. This is a consequence of the reflexivity of the epistemic accessibility relation that ensures veridicality of knowledge. For each agent considers all formulae that hold in the 'current' state to be possible, and hence all these formulae are uniformly supported mights. Clause 2 indicates that in principle all gaps in the agent's knowledge are 'fillable', i.e., for all formulae φ either φ or $\neg\varphi$ is a default. Possibly the most important and remarkable property of the common possibility operator with regard to its usability to represent defaults is given by clause 6. This clause indicates that disjunctive defaults are not necessarily trivialized, that is, these disjunctions are not necessarily reduced to their disjuncts. This property is very important for the expressive power of our framework. Consider for instance the situation of a lottery with $m \gg 1$ players. Then it is not the case that player 1 wins by default, and neither is this the case for any of the players 2 to $m \Leftrightarrow 1$. But it is also the case that by default one of these $m \Leftrightarrow 1$ players actually does win. Since $\{\neg\mathbf{N}w_1, \ldots, \neg\mathbf{N}w_{m-1}, \mathbf{N}(w_1 \vee \ldots \vee w_{m-1})\}$ is satisfiable, this aspect of the lottery can be formalized in our framework. Note that this situation cannot be formalized by taking ordinary (single-agent) epistemic possibility instead of common possibility, since $\mathbf{M}_i(\varphi \vee \psi) \leftrightarrow \mathbf{M}_i\varphi \vee \mathbf{M}_i\psi$ is a valid formula. Clauses 10 and 11 show that the Nixon-diamond can be represented. That is, it is possible to represent that it is a default that Nixon was a pacifist and that it is a default that he was a non-pacifist, even though it is not a default that he was a walking contradiction.

Some of the aspects of the relations between knowledge, belief and common possibility are formalized in the following lemma.

3.14. LEMMA. *For all agents i and formulae φ and ψ we have:*

1. $\models \mathbf{K}_i\neg\varphi \rightarrow \neg\mathbf{N}\varphi$
2. $\not\models \mathbf{B}_i\neg\varphi \rightarrow \neg\mathbf{N}\varphi$
3. $\models \mathbf{N}\mathbf{K}_i\varphi \leftrightarrow \mathbf{K}_i\varphi$
4. $\models \mathbf{N}\mathbf{B}_i\varphi \leftrightarrow \mathbf{B}_i\varphi$
5. $\models \mathbf{N}\neg\mathbf{K}_i\varphi \leftrightarrow \neg\mathbf{K}_i\varphi$
6. $\models \mathbf{N}\neg\mathbf{B}_i\varphi \leftrightarrow \neg\mathbf{B}_i\varphi$
7. $\models \mathbf{K}_i\mathbf{N}\varphi \rightarrow \mathbf{N}\varphi$
8. $\not\models \mathbf{N}\varphi \rightarrow \mathbf{K}_i\mathbf{N}\varphi$
9. $\not\models \mathbf{B}_i\mathbf{N}\varphi \rightarrow \mathbf{N}\varphi$
10. $\not\models \mathbf{N}\varphi \rightarrow \mathbf{B}_i\mathbf{N}\varphi$
11. $\models \mathbf{B}_i\mathbf{N}\varphi \rightarrow \neg\mathbf{K}_i\neg\varphi$

3.15. REMARK. The clauses 1 and 2 of Lemma 3.14 nicely emphasize the ontological difference between knowledge and belief: it is possible that an agent considers a formula to be a default although it beliefs the negation of the formula, but if the agent knows the negation of the formula, the formula will not be considered a default. The clauses 3 to 6 are related to the introspective properties that the agents have both for knowledge and belief. The last clause states, when read in its contrapositive form, that formulae known to be false are not believed to be defaults; this seems to be highly desirable a property for a *rational* agent.

3.3. Checking consistency

In [6], Doherty formulates two general conditions under which an agent should be permitted to adopt by default a formula as one of its beliefs. The first of these conditions basically states that the formula must be a default: it should be plausible according to some criteria. We have formalized this demand for plausibility by introducing the notion of common possibility. Here we look into the second condition: the consistency of the formula that is to be believed with the other beliefs that the agent already has.

In our framework, two different consistency checks need to be performed. The first of these is the obvious one, in which it is checked that the formula that is to be believed is possible given the beliefs of the agent. That is, for a formula φ it is verified that $\neg\mathbf{B}_i\neg\varphi$ holds. This is a straightforward consistency check of the kind that is employed in consistency-based formalizations of default reasoning (cf. [28]). The second consistency check that is to be performed is due to the expressiveness of our framework in which defaults are not necessarily propositional formulae, but may contain all sorts of operators. Now when allowing non-propositional formulae, it is possible that an update with a formula that can consistently be assumed, does still not result in the agent actually believing the formula. The following example, a variant of an example given in [24], illustrates this point.

3.16. EXAMPLE. Let $i \in \mathcal{A}$ be some agent. Consider the model $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}, \mathtt{r}, \mathtt{c}\rangle$, with $\mathcal{S} = \{s_0, s_1\}$, $\pi(p, s_0) = \mathbf{1}, \pi(p, s_1) = \mathbf{0}$, $\mathrm{R}(i) = \mathcal{S}^2, \mathrm{B}(i, [s]_{\mathrm{R}(i)}) = [s]_{\mathrm{R}(i)}$, and $\mathtt{r}$ and $\mathtt{c}$ are arbitrary. In this model the following statements are true:

- $\mathcal{M}, s_0 \models p \wedge \neg \mathbf{B}_i p$
- $\mathcal{M}, s_1 \not\models p \wedge \neg \mathbf{B}_i p$
- $\mathcal{M}, s_0 \not\models \mathbf{B}_i \neg (p \wedge \neg \mathbf{B}_i p)$

Although the formula $\varphi \stackrel{\text{def}}{=} p \wedge \neg \mathbf{B}_i p$ is consistent with the beliefs of the agent in $\mathcal{M}$, it is not possible to incorporate this formula in the beliefs of the agent by performing a doxastic update. For a doxastic update in s_0 with φ results in a model $\mathcal{M}'$ such that $\mathrm{B}'(i, [s]_{\mathrm{R}(i)}) = \{s_0\}$. For this model it holds that $\mathcal{M}', s_0 \not\models \mathbf{B}_i \varphi$.

We solve the problem observed in example 3.16 in a rather pragmatic way: it is simply checked whether a doxastic update indeed results in the formula being believed. If this is not the case, the formula does not pass this second consistency check and is therefore not adopted by the agent as one of its beliefs. The presence of dynamic operators in our language allows for a concise representation of this second consistency check. Formulae that pass this second check are called *default honest*, in analogy with the notion of honesty as defined by Halpern and Moses ([10]). A typical class of default honest formulae is that given by the language $\mathcal{L}_0$ of all propositional formulae; typical default dishonest formulae are those that are inconsistent in the Kraus & Lehmann axiomatization of knowledge and belief ([22]).

3.4. Jumping to conclusions

Having dealt with the formalization of both defaults and the beliefs derived by default, and the appropriate checks for consistency, we are now able to define the `try_jump` action that formalizes the three-stage process of jumping to conclusions. Besides the `try_jump` action, we introduce two new low-level actions that are used to model the different stages in the execution of the `try_jump` action.

3.17. Definition. The class Ac of actions (and hence the language $\mathcal{L}$) as defined in 2.1 is extended as follows:

- if $i \in \mathcal{A}$ and φ is a formula then `try_jump` $\varphi \in Ac$

The class of actions Ac' is extended with the sets $\{$`dox_update` $(\varphi, j)\}$, representing the low-level action that performs *doxastic* updates, and $\{$`bel_update` $(\varphi, j)\}$ representing belief updates. For both sets φ is an element of $\mathcal{L}$ and $j \in \mathcal{A}$.

3.18. Definition. Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}, \mathtt{r}, \mathtt{c} \rangle$ be a Kripke model, let $i \in \mathcal{A}$ be some agent, and let φ be some formula. The function `r` is for the action `try_jump` , and the meta-actions `dox_update` and `bel_update` defined as follows.

- $\mathtt{r}(i,$ `try_jump` $\varphi)(\mathcal{M}, s) = \mathtt{r}(e,$ `bel_update` $(\varphi, i))(\mathcal{M}, s)$ if $\mathcal{M}, s \models \mathbf{N}\varphi$,
- $\mathtt{r}(i,$ `try_jump` $\varphi)(\mathcal{M}, s) = \emptyset$ if $\mathcal{M}, s \not\models \mathbf{N}\varphi$.
- $\mathtt{r}(e,$ `bel_update` $(\varphi, i))(\mathcal{M}, s) = \mathtt{r}(e,$ `dox_update` $(\varphi, i))(\mathcal{M}, s)$ if $\mathcal{M}, s \models \neg \mathbf{B}_i \neg \varphi \wedge \langle \mathrm{do}_e($`dox_update` $(\varphi, i)) \rangle \mathbf{B}_i \varphi$,
- $\mathtt{r}(e,$ `bel_update` $(\varphi, i))(\mathcal{M}, s) = \mathcal{M}, s$ otherwise.
- $\mathtt{r}(e,$ `dox_update` $(\varphi, i))(\mathcal{M}, s) = \emptyset$ if $\mathcal{M}, s \models \mathbf{B}_i \neg \varphi$.
- $\mathtt{r}(e,$ `dox_update` $(\varphi, i))(\mathcal{M}, s) = \mathcal{M}', s$, where $\mathcal{M}' =< \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}', \mathtt{r}, \mathtt{c} >$ with $\mathrm{B}'(i, [s]_{\mathrm{R}(i)}) = \mathrm{B}(i, [s]_{\mathrm{R}(i)}) \cap [\![\varphi]\!]$ if $\mathcal{M}, s \not\models \mathbf{B}_i \neg \varphi$.

3.19. REMARK. In the first two clauses of Def. 3.18 a case distinction is made on φ being a default: in the case where φ is not a default, i.e., $\mathcal{M}, s \not\models \mathrm{N}\varphi$, the attempt to jump to φ fails. In the case where φ is a default, it is checked whether φ can consistently be adopted by the agent, i.e., it is checked whether $\mathcal{M}, s \models \neg\mathbf{B}_i\neg\varphi \wedge \langle \mathrm{do}_e(\texttt{dox\_update}\ (\varphi, i))\rangle\mathbf{B}_i\varphi$; dependent on whether this consistency check is passed a belief update is performed. Note the different treatment of the case where φ is not a default and the one in which φ cannot consistently be assumed: in the first case the `try_jump` action reduces to `fail`, in the second case the action reduces to `skip`. This corresponds to the intuition that trying to jump to a non-jumpable formula results in a fail, whereas trying to jump to a jumpable formula that is unacceptable due to the context, does not change anything. From a technical point of view the reduction to `skip` in the second case is related to the treatment of normal defaults with inconsistent consequents in default logic: these defaults do not cause the resulting extension to be inconsistent -which would correspond to a reduction to `fail` in our framework- but are simply neglected -which corresponds to the reduction to `skip`.

The formal counterpart of the notion of default honesty (informally introduced in 3.3), as it is used in Def. 3.18 is as follows.

3.20. DEFINITION. Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}, \mathtt{r}, \mathtt{c}\rangle$ be a Kripke model, let $i \in \mathcal{A}$ be some agent, and let φ be some formula. The formula φ is *default honest* for i in $\mathcal{M}, s$ iff

$$\mathcal{M}, s \models [\mathrm{do}_e(\texttt{dox\_update}\ (\varphi, i)]\mathbf{B}_i\varphi$$

An important class of default honest formulae is that of purely propositional formulae. The following lemma, the proof of which is straightforward, formalizes this property.

3.21. LEMMA. *For all agents i and all purely propositional formulae $\varphi \in \mathcal{L}_0$ it holds that*

$$\models [\mathrm{do}_e(\texttt{dox\_update}\ (\varphi, i)]\mathbf{B}_i\varphi$$

or, equivalently

$$\models \neg\mathbf{B}_i\neg\varphi \rightarrow \langle \mathrm{do}_e(\texttt{dox\_update}\ (\varphi, i)\rangle\mathbf{B}_i\varphi$$

The models that result from applying defaults are well-defined in the sense that in particular the belief function B satisfies the demands it should meet.

3.22. LEMMA. *For all Kripke models $\mathcal{M}$ and $s \in \mathcal{S}$, for formulae φ we have:*

- *if* $\mathtt{r}(i, \texttt{try\_jump}\ \varphi)(\mathcal{M}, s) = \mathcal{M}', s$ *then $\mathcal{M}'$ is a well-defined Kripke model.*

An important aspect of informative actions, whether they produce knowledge or belief, is that execution of these actions causes *minimal change* to the model under consideration. That is, besides the knowledge and belief fluents, these actions should not affect any other aspects of the model. This is also the case for our apply default action, provided that the default that is applied indeed is a default, i.e., it is a plausible conclusion.

3.23. LEMMA. *Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}, \mathtt{r}, \mathtt{c}\rangle$ be some Kripke model. Let $s \in \mathcal{S}$, $i \in \mathcal{A}$ and φ some formula. Let $\mathcal{M}' =< \mathcal{S}', \pi', \mathrm{R}', \mathrm{B}', \mathtt{r}', \mathtt{c}' >$ and $\mathcal{M}', s' = \mathtt{r}(i, \texttt{try\_jump}\ \varphi)(\mathcal{M}, s)$. Then:*

1. *If* $\mathcal{M}, s \not\models \neg\mathbf{B}_i\neg\varphi \wedge \langle \mathrm{do}_e(\mathtt{dox\_update}\ (\varphi, i))\rangle\mathbf{B}_i\varphi$ *then* $\mathcal{M}', s' = \mathcal{M}, s$.
2. *If* $\mathcal{M}, s \models \neg\mathbf{B}_i\neg\varphi \wedge \langle \mathrm{do}_e(\mathtt{dox\_update}\ (\varphi, i))\rangle\mathbf{B}_i\varphi$ *then:*
   - $\mathcal{S}' = \mathcal{S}, \pi' = \pi, \mathrm{R}' = \mathrm{R}, \mathtt{r}' = \mathtt{r}, \mathtt{c}' = \mathtt{c}$ *and* $s' = s$,
   - $\lambda x.\mathrm{B}'(j, x) = \lambda x.\mathrm{B}(j, x)$ *for* $j \neq i$
   - *Let* $\mathcal{M}'' =< \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}'', \mathtt{r}, \mathtt{c} >$ *be a Kripke model such that*
     - $\mathrm{B}''(i', [s]_{\mathrm{R}(i')}) = \mathrm{B}(i', [s]_{\mathrm{R}(i')})$ *for* $i' \neq i$
     - $\mathrm{B}''(i, [s]_{\mathrm{R}(i)}) \subseteq \mathrm{B}(i, [s]_{\mathrm{R}(i)})$

     Then if $\mathcal{M}'', s \models \mathbf{B}_i\varphi$ *then* $\mathrm{B}''(i, [s]_{\mathrm{R}(i)}) \subseteq \mathrm{B}'(i, [s]_{\mathrm{R}(i)})$.

3.24. Remark. Intuitively, Lemma 3.23 could be read as stating that a jump to a *default* causes minimal change to the model. Attempted jumps to non-defaults are doomed to fail, thereby causing a more or less 'maximal' change.

3.25. Lemma. *For all agents* i *and* j*, and for all formulae* φ *and* ψ *we have:*

1. $\models \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\psi \rightarrow [\mathrm{do}_i(\mathtt{try\_jump}\ \varphi)]\psi$
2. $\models \mathbf{N}\varphi \leftrightarrow \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{tt}$
3. $\models \varphi \rightarrow \psi \Rightarrow \models \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{tt} \rightarrow \langle \mathrm{do}_i(\mathtt{try\_jump}\ \psi)\rangle\mathbf{tt}$
4. $\models \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{tt} \leftrightarrow \langle \mathrm{do}_j(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{tt}$
5. $\not\models \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{B}_i\varphi \rightarrow \langle \mathrm{do}_j(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{B}_j\varphi$
6. $\models \mathbf{N}\varphi \wedge \mathbf{B}_i\neg\varphi \rightarrow (\psi \leftrightarrow \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\psi)$
7. *For B-free formulae* φ*:*
   $\models \mathbf{N}\varphi \wedge \neg\mathbf{B}_i\neg\varphi \rightarrow \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{B}_i\varphi$
8. *For B-free formulae* ψ*, and for all formulae* φ*:*
   $\models \mathbf{B}_i\psi \rightarrow [\mathrm{do}_i(\mathtt{try\_jump}\ \varphi)]\mathbf{B}_i\psi$

3.26. Remark. Lemma 3.25 shows that the formal equivalents of the intuitive ideas that we presented are indeed brought about by our definitions. The first clause of Lemma 3.25 states that the event consisting of the attempt to jump to a conclusion is a deterministic one. Clause 2 formalizes the idea that being a default sets up the opportunity for jumps. In clause 4 it is formalized that agents have equal opportunities with respect to attempted jumps to conclusions. Basically this is a consequence of the fact that defaults are rooted in *common* sense, and are therefore equal for all agents. Although agents have equal opportunities, clause 5 states that the attempted jumps to conclusions may work out differently for different agents. This is as one would expect, since the result of such a jump depends on the beliefs of the jumping agent. Clause 6 states that the event that consists of the jump to an unacceptable default, i.e., a default that is inconsistent with the beliefs of the applying agent, reduces to the empty action `skip`; nothing changes as the result of a jump to such a default. Clause 7 formalizes that a jump to an acceptable default results in the acquisition of belief in the default. The last clause states that by attempted jumps to conclusions agents retain their set of objective beliefs: no belief revision of objective formulae takes place.

3.27. Remark. As already mentioned in Sect. 1, the definitions constituting the dynamic part of default reasoning are in no way dependent on the modality of common possibility to represent defaults. In fact arbitrary default rules $\varphi : \psi/\vartheta$ *à la* Reiter ([42]) can be 'dynamized' in the KARO-architecture in the following way. Assume some Kripke model $\mathcal{M}$ is given together with a function δ that yields for each agent in each state of the

model the set of default rules that the agent has at its disposal. Instead of jumping to default conclusions, agents apply default rules $\varphi : \psi/\vartheta$ as follows. Based on some modal translation of the default rule, for instance the translation $\mathbf{K}_i\varphi \wedge \mathbf{M}_i\psi \rightarrow \mathbf{K}_i\vartheta$ as proposed by Konolige ([21]), it is checked whether the precondition part of the default rule, in this case $\mathbf{K}_i\varphi \wedge \mathbf{M}_i\psi$, is satisfied. If this is the case the consequent part of the default rule, i.e., $\mathbf{K}_i\vartheta$ is brought about by performing the appropriate epistemic update. In this way agents can reason with general default rules. Further investigation into these matters is left for a future paper.

3.5. The ability to jump

In the previous (sub)sections, we dealt with the formalization of the opportunity for and the result of the action that consists of an attempted jump to a conclusion. Here we look at the *ability* of agents to make these jumps.

For 'mental' actions, like testing (observing) and communicating, the abilities of agents are closely related to their (lack of) knowledge. This observation seems to hold *a fortiori* for the action that consists of jumping to a default conclusion. For when testing and communicating, at least some interaction takes place, either with the real world in case of testing, or with other agents when communicating, whereas jumping to conclusions is a strictly mental, agent-internal, activity. Therefore it seems obvious that the ability of an agent to make a jump to a default conclusion depends on the mental state of the agent only. In our opinion an intuitively acceptable formalization of the agents' abilities is to state that an agent is able to jump to only those formulae that it *knows* to be defaults. So agents have to *know* their defaults in order to be *able* to use them. This intuitive idea leads to the following formal definition.

3.28. Definition. Let $\mathcal{M}$ be some Kripke model, let s be some state in $\mathcal{M}$, $i \in \mathcal{A}$ some agent, and φ some formula. The capability function $\mathtt{c}$ is for the action `try_jump` defined by

$$\mathtt{c}(i, \mathtt{try\_jump}\ \varphi)(\mathcal{M}, s) = \mathbf{1} \Leftrightarrow \mathcal{M}, s \models \mathbf{K}_i\mathbf{N}\varphi$$

The following lemma formalizes some properties that result from Def. 3.28.

3.29. Lemma. *For all agents i, j and for all formulae φ and ψ we have:*

1. $\not\models \mathbf{A}_i\mathtt{try\_jump}\ \varphi \leftrightarrow \mathbf{A}_j\mathtt{try\_jump}\ \varphi$
2. $\models \mathbf{A}_i\mathtt{try\_jump}\ \varphi \leftrightarrow \mathbf{K}_i\mathbf{A}_i\mathtt{try\_jump}\ \varphi$
3. $\models \mathbf{A}_i\mathtt{try\_jump}\ \varphi \rightarrow \langle \mathrm{do}_i(\mathtt{try\_jump}\ \varphi)\rangle\mathbf{tt}$

3.30. Remark. The first clause of Lemma 3.29 is an important one. It states that agents do not necessarily have equal abilities with regard to the jumping to default conclusions. Hence even though all agents have equal *opportunities* to jump (cf. clause 4 of Lemma 3.25), with respect to their *abilities* some agents are more equal than others. Through its abilities each agent can be equipped with its personal set of defaults, namely those defaults that the agent is aware of. Note that this nicely corresponds to the intuition: since I know that birds typically fly I may conclude that Tweety flies when hearing of some bird Tweety. But although pteroydactyls do fly by default, people who are not aware of this fact will not be able to conclude that Tweety flies upon hearing that it is a pterodactyl. Clause 2

states that agents know of their abilities to apply defaults. The last clause states that the `try_jump` action is **A**-*realizable* ([15]), i.e., the *ability* to perform the action implies the *opportunity* to do so. This property seems to be typical for actions without physical (circumstantial) prerequisites, of which the application of defaults obviously is a prototypical example.

3.6. The belief in correct and feasible plans

Inspired by the concepts introduced in [36], we introduced in [15] the Can-predicate and the Cannot-predicate. Intuitively these predicates formalize the knowledge and the reasoning of agents regarding the (in)correctness and (in)feasibility of their plans to achieve certain goals. The definition of these predicates is based on the idea that an agent i knows that action α is a *correct* plan to achieve φ iff it knows that $\langle \mathrm{do}_i(\alpha)\rangle\varphi$ holds. Agent i knows that α is a *feasible* plan for φ iff it knows that it is able to do α, i.e., $\mathbf{K}_i\mathbf{A}_i\alpha$ holds. This intuition is formalized in the definition of the Can-predicate as we give it. We also defined a Cannot-predicate, which has as its intended meaning that the agent knows that it cannot reach some goal φ by performing some action α, since it knows that either the action does not lead to the desired goal or it is not capable of performing the action, i.e., the agent knows that the action is either an incorrect or an infeasible plan.

3.31. Definition. The Can-predicate and the Cannot-predicate are defined as follows.

- $\mathbf{Can}_i(\alpha,\varphi) \equiv \mathbf{K}_i(\langle \mathrm{do}_i(\alpha)\rangle\varphi \wedge \mathbf{A}_i\alpha)$.
- $\mathbf{Cannot}_i(\alpha,\varphi) \equiv \mathbf{K}_i(\neg\langle \mathrm{do}_i(\alpha)\rangle\varphi \vee \neg\mathbf{A}_i\alpha)$.

Situations where agents have complete knowledge are however not very common when modelling real-life situations. In general agents hardly ever have complete knowledge concerning the correctness and feasibility of their plans. Still these agents have to decide on the usability of their actions to achieve certain goals. At this point a special kind of defaults, so called *default plans* come into play. Default plans are actions that an agent can adopt by default in cases where it cannot adopt an action on the basis of its knowledge. For example 'Usually when I feel tired, I take a nap and feel better afterwards.' is a default plan, and so is 'If you're stuck with your research, it is usually a good idea to go and read something'. Formally, default plans are defined as follows.

3.32. Definition. The action α is a default plan with respect to φ for agent i in state s of model $\mathcal{M}$ iff $\mathcal{M}, s \models \mathbf{N}(\langle \mathrm{do}_i(\alpha)\rangle\varphi \wedge \mathbf{A}_i\alpha)$.

The result of jumping to a default plan α is a state of affairs in which the agent *believes* α to be a correct and feasible plan to achieve the goal. As we did for plans known to be correct and feasible, we introduce predicates to model plans believed to be (in)correct and/or (in)feasible.

3.33. Definition. The Could-predicate and the Couldnot-predicate are defined as follows.

- $\mathbf{Could}_i(\alpha,\varphi) \stackrel{\text{def}}{=} \mathbf{B}_i(\langle \mathrm{do}_i(\alpha)\rangle\varphi \wedge \mathbf{A}_i\alpha)$.
- $\mathbf{Couldnot}_i(\alpha,\varphi) \stackrel{\text{def}}{=} \mathbf{B}_i(\neg\langle \mathrm{do}_i(\alpha)\rangle\varphi \vee \neg\mathbf{A}_i\alpha)$.

Note that both these predicates are indeed weakened forms of the corresponding predicates given in Def. 3.31. Note furthermore that due to the fact that beliefs may not be inconsistent, agents do not hold actions both correct and feasible and incorrect or infeasible at the same time.

3.34. LEMMA. *For all agents i, actions α and formulae φ we have*

- $\models \mathbf{Can}_i(\alpha,\varphi) \rightarrow \mathbf{Could}_i(\alpha,\varphi)$.
- $\models \mathbf{Cannot}_i(\alpha,\varphi) \rightarrow \mathbf{Couldnot}_i(\alpha,\varphi)$.
- $\models \mathbf{Could}_i(\alpha,\varphi) \rightarrow \neg\mathbf{Couldnot}_i(\alpha,\varphi)$.

For reasons of clarity we introduce the abbreviation $\mathit{Consistent}_i(\varphi) \stackrel{\text{def}}{=} \neg\mathbf{B}_i\neg\varphi \wedge \langle \mathrm{do}_e(\texttt{dox\_update}\ (\varphi,i))\rangle\mathbf{B}_i\varphi$ to represent the fact that φ can pass the two stage consistency check. The abbreviation $\mathit{Consistent\_Default\_Plan}_i(\alpha,\varphi) \stackrel{\text{def}}{=} \mathbf{N}(\langle \mathrm{do}_i(\alpha)\rangle\varphi \wedge \mathbf{A}_i\alpha) \wedge \mathit{Consistent}_i(\langle \mathrm{do}_i(\alpha)\rangle\varphi \wedge \mathbf{A}_i\alpha)$ denotes that both α is a default plan for φ and that it is possible to make the jump to belief in α being a correct and feasible plan for φ.

3.35. LEMMA. *For all agents i, actions α and formulae φ we have*

- $\models \mathit{Consistent\_Default\_Plan}_i(\alpha,\varphi) \Rightarrow$
  $\models \mathbf{Can}_i(\texttt{try\_jump}\ (\langle \mathrm{do}_i(\alpha)\rangle\varphi \wedge \mathbf{A}_i\alpha), \mathbf{Could}_i(\alpha,\varphi))$

Intuitively, Lemma 3.35 states that agents know that they can come to the belief in correctness and feasibility of default plans. In a sense this lemma states that agents can *convince* themselves of the usability of default plans. Special attention should be paid to the menagerie of modal operators and predicates occurring in Lemma 3.35: the knowledge operator $\mathbf{K}_i$, the *Consistent_Default_Plan* predicate, the Can- and Could-predicate, the opportunity and result formula $\langle \mathrm{do}_i(\alpha)\rangle\varphi$ and the ability operator $\mathbf{A}_i$.

4. The dynamics of Supernormal Default Logic

The most prominent consistency-based approach to default reasoning is Reiter's default logic ([1, 2, 28, 42]). In default logic, default statements are formalized by special inference rules $\varphi : \psi/\vartheta$, where φ, ψ and ϑ are formulae of the language. In the context of default logic these inference rules are called *defaults*. The formula φ is called the *prerequisite* of the default, ψ is the *justification*, and ϑ is the *consequent*. The intuitive interpretation of such a default is that if φ is known (believed), and ψ is consistent with the set of knowledge (belief), then the conclusion ϑ may be inferred. A default $\varphi\ :\ \psi\ /\ \vartheta$ is called *seminormal* if ψ logically implies ϑ; it is called *normal* if ψ and ϑ are identical, and *supernormal* if it is normal and has an empty prerequisite. In this section we give a dynamic interpretation of the supernormal fragment of default logic, using the concepts introduced in the previous sections. More specific, we show that our agents may construct *extensions* (in the sense of default logic) of their set of beliefs by successive jumps to conclusions.

Since we restrict ourselves to the supernormal fragment of default logic, the formal definitions that we give are tailored at this fragment. As such these definitions are considerably less complex than those for general default logic.

4.1. DEFINITION. A default theory Δ is a pair (W, D), where W is a set of propositional formulae representing the (propositional) beliefs of an agent and D is a set of formulae, representing supernormal defaults.

The amount of belief induced by a default theory is given by the *extension* of the default theory. In the following definition, which was a theorem in [5], 'Th' denotes the closure operator of classical propositional logic.

4.2. DEFINITION. Let $\Delta = (W, D)$ be a supernormal default theory. For all sets of formulae S such that

- $S \subseteq D$
- $W \cup S$ is consistent
- $W \cup S \cup \{\varphi\}$ is inconsistent for all $\varphi \in D \setminus S$,

$\mathrm{Th}(W \cup S)$ is an extension of Δ.

It turns out that the framework defined in the previous sections provides a sort of dynamic interpretation of supernormal default logic. More in particular, consider an agent in some state s of a given model $\mathcal{M}$. For any given finite set D of defaults for the agent in $\mathcal{M}, s$, it holds that any sequence of successive attempted jumps to the defaults from D results in a default logic extension of the belief set of the agent given D. The belief set of an agent, given a model and a state in the model, is given by the following definition.

4.3. DEFINITION. Let $\mathcal{M} = \langle \mathcal{S}, \pi, \mathrm{R}, \mathrm{B}, \mathtt{r}, \mathtt{c} \rangle$ be some Kripke model with $s \in \mathcal{S}$, and let $i \in \mathcal{A}$.

- The belief set of i in $\mathcal{M}, s$, notation $\mathcal{B}(i, \mathcal{M}, s)$, is defined by:
$$\mathcal{B}(i, \mathcal{M}, s) \stackrel{\text{def}}{=} \{\varphi \in \mathcal{L}_0 \mid \mathcal{M}, s \models \mathbf{B}_i\varphi\}$$

4.4. THEOREM. *Let $\mathcal{M}$ be some Kripke model, let $s \in \mathcal{M}$, and let i be some agent. Define $W = \mathcal{B}(i, \mathcal{M}, s)$ and let $D = \{\varphi_1, \ldots, \varphi_m\} \subseteq \{\varphi \in \mathcal{L}_0 \mid \mathcal{M}, s \models \mathbf{N}\varphi\}$. For all permutations $(\psi_1, \ldots, \psi_m)$ of $(\varphi_1, \ldots, \varphi_m)$, $\{\varphi \in \mathcal{L}_0 \mid \mathcal{M}, s \models [\mathrm{do}_i(\mathtt{try\_jump}\ \psi_1; \ldots, \mathtt{try\_jump}\ \psi_m)]\mathbf{B}_i\varphi\}$ is an extension of (W, D).*

PROOF OF THEOREM 4.4: We prove the theorem by proving three lemmas and a corollary.

4.5. LEMMA. *For all $1 \leq j, k \leq m$:*

- $\mathtt{r}(i, \mathtt{try\_jump}\ \psi_1; \ldots; \mathtt{try\_jump}\ \psi_j)(\mathcal{M}, s) \neq \emptyset$
- $\mathtt{r}(i, \mathtt{try\_jump}\ \psi_1; \ldots; \mathtt{try\_jump}\ \psi_j)(\mathcal{M}, s) \models \mathbf{N}\psi_k$

PROOF OF LEMMA 4.5: We start by showing both clauses for $j = 1$. Note that $\mathcal{M}, s \models \mathbf{N}\psi_k$, for all $1 \leq k \leq m$, by definition of D. Then by inspection of Def. 3.18 we have $\mathtt{r}(i, \mathtt{try\_jump}\ \psi_1)(\mathcal{M}, s) = \mathtt{r}(e, \mathtt{bel\_update}\ (\psi_1, i))(\mathcal{M}, s)$. We distinguish two cases:

1. $\mathcal{M}, s \models \neg\mathbf{B}_i\neg\psi_1 \wedge \langle \mathrm{do}_e(\mathtt{dox\_update}\ (\psi_1, i)\rangle \mathbf{B}_i\psi_1$. Then $\mathtt{r}(e, \mathtt{bel\_update}\ (\psi_1, i)(\mathcal{M}, s) = \mathtt{r}(e, \mathtt{dox\_update}\ (\psi_1, i)(\mathcal{M}, s)$. By definition of $\mathtt{r}$ for the $\mathtt{dox\_update}$ action it follows that $\mathtt{r}(e, \mathtt{dox\_update}\ (\psi_1, i)(\mathcal{M}, s) \neq \emptyset$, and thus $\mathtt{r}(i, \mathtt{try\_jump}\ \psi_1)(\mathcal{M}, s) \neq \emptyset$.
2. $\mathcal{M}, s \not\models \neg\mathbf{B}_i\neg\psi_1 \wedge \langle \mathrm{do}_e(\mathtt{dox\_update}\ (\psi_1, i)\rangle \mathbf{B}_i\psi_1$. Then $\mathtt{r}(e, \mathtt{bel\_update}\ (\psi_1, i)(\mathcal{M}, s) = \mathcal{M}, s$, by definition of $\mathtt{r}$ for $\mathtt{bel\_update}$.

In both cases $\mathtt{r}(i, \mathtt{bel\_update}\ (\psi_1, i))(\mathcal{M}, s) \neq \emptyset$, and hence, since also $\mathcal{M}, s \models \mathbf{N}\varphi$, $\mathtt{r}(i, \mathtt{try\_jump}\ \psi_1)(\mathcal{M}, s) \neq \emptyset$. Let $\mathcal{M}', s = \mathtt{r}(i, \mathtt{bel\_update}\ (\psi_1, i))(\mathcal{M}, s)$. By inspection of Def. 3.18, it is clear that $\mathcal{M}', s \models \mathbf{N}\psi_k$, for all $1 \leq k \leq m$. Intuitively this is also clear since jumps to conclusions only affect *belief* fluents, and defaults are dependent on *knowledge* fluents only. The arguments given for ψ_1 can straightforwardly be repeated for ψ_2 to ψ_m, which suffices to conclude the lemma.
⊠

4.6. LEMMA. *Let for all $0 \leq j \leq m \Leftrightarrow 1$:*

- $\mathcal{M}_0, s = \mathcal{M}, s$
- $\mathcal{M}_{j+1}, s = \mathtt{r}(i, \mathtt{try\_jump}\ \psi_{j+1})(\mathcal{M}_j, s)$

Note that by Lemma 4.5 these models are all well-defined. Now for all $0 \leq j \leq m \Leftrightarrow 1$*:*

- $\neg\psi_{j+1} \in \mathcal{B}(i, \mathcal{M}_j, s) \Rightarrow \mathcal{B}(i, \mathcal{M}_{j+1}, s) = \mathcal{B}(i, \mathcal{M}_j, s)$
- $\neg\psi_{j+1} \notin \mathcal{B}(i, \mathcal{M}_j, s) \Rightarrow \mathcal{B}(i, \mathcal{M}_{j+1}, s) = \mathrm{Th}(\mathcal{B}(i, \mathcal{M}_j, s) \cup \{\psi_{j+1}\})$

Proof of lemma 4.6: We show both clauses.

- The first clause is straightforward: since $\mathcal{M}_j, s \models \mathbf{N}\psi_{j+1}$, it follows by Def. 3.18 that if $\neg\psi_{j+1} \in \mathcal{B}(i, \mathcal{M}_j, s)$, i.e., $\mathcal{M}_j, s \models \mathbf{B}_i\neg\psi_{j+1}$, then $\mathcal{M}_{j+1}, s = \mathcal{M}_j, s$. Hence trivially $\mathcal{B}(i, \mathcal{M}_{j+1}, s) = \mathcal{B}(i, \mathcal{M}_j, s)$.
- From Lemma 4.5 it follows that $\mathcal{M}_j, s \models \mathbf{N}\psi_{k+1}$, for all $0 \leq j < m$, $1 \leq k \leq m$. From this and using that purely propositional formulae are default honest (cf. Lemma 3.21), one sees by inspecting Def. 3.18 that from $\neg\psi_{j+1} \notin \mathcal{B}(i, \mathcal{M}_j, s)$, it follows that $\mathtt{r}(i, \mathtt{try\_jump}\ \psi_{j+1})(\mathcal{M}_j, s) = \mathtt{r}(e, \mathtt{dox\_update}\ (\psi_{j+1}, i))(\mathcal{M}_j, s)$. We prove that $\mathcal{B}(i, \mathcal{M}_{j+1}, s) = \mathrm{Th}(\mathcal{B}(i, \mathcal{M}_j, s) \cup \{\psi_{j+1}\})$ by proving that both sets are subsets of each other.
  - '$\supseteq$' For propositional formulae ψ we have $\models \mathbf{B}_i\psi \rightarrow [\mathrm{do}_i(\mathtt{try\_jump}\ \varphi)]\mathbf{B}_i\psi$ by clause 8 of Lemma 3.25, hence $\mathcal{B}(i, \mathcal{M}_j, s) \subseteq \mathcal{B}(i, \mathcal{M}_{j+1}, s)$. From clauses 1 and 7 of Lemma 3.25 it follows that $\psi_{j+1} \in \mathcal{B}(i, \mathcal{M}_{j+1}, s)$. Hence $\mathcal{B}(i, \mathcal{M}_j, s) \cup \{\psi_{j+1}\} \subseteq \mathcal{B}(i, \mathcal{M}_{j+1}, s)$ and hence $\mathrm{Th}(\mathcal{B}(i, \mathcal{M}_j s) \cup \{\psi_{j+1}\}) \subseteq \mathrm{Th}(\mathcal{B}(i, \mathcal{M}_{j+1}, s)) = \mathcal{B}(i, \mathcal{M}_{j+1}, s)$.
  - '$\subseteq$' Assume that $\varphi \in \mathcal{B}(i, \mathcal{M}_{j+1}, s)$. By definition of `dox_update`, it follows that φ holds in all the states in $\mathrm{B}_j(i, [s]_{\mathrm{R}_j(i)})$ that satisfy ψ_{j+1}. But then $\psi_{j+1} \rightarrow \varphi$ holds in all the states from $\mathrm{B}_j(i, [s]_{\mathrm{R}_j(i)})$, and hence $\mathcal{M}_j, s \models \mathbf{B}_i(\psi_{j+1} \rightarrow \varphi)$. Then $(\psi_{j+1} \rightarrow \varphi) \in \mathcal{B}(i, \mathcal{M}_j, s)$, and thus $\varphi \in \mathrm{Th}(\mathcal{B}(i, \mathcal{M}, s) \cup \{\psi_{j+1}\})$.

  From '$\supseteq$' and '$\subseteq$' we conclude the second clause of Lemma 4.6.

⊠

4.7. Corollary. *For all* $0 \leq j \leq m$, $\mathcal{B}(i, \mathcal{M}_j, s) \subseteq \mathcal{B}(i, \mathcal{M}_m, s)$.

Proof of corollary 4.7: Straightforward from Lemma 4.6.

⊠

Note that since $\mathcal{M}_m$ is a well-defined Kripke model and since beliefs are not inconsistent we have that $\mathcal{B}(i, \mathcal{M}_m, s)$ is consistent.

4.8. Lemma. *Let fired*(D) *be the set* $\mathcal{B}(i, \mathcal{M}_m, s) \cap D$*. Then we have for all* $\psi \in D \setminus$ *fired*(D), $\neg\psi \in \mathcal{B}(i, \mathcal{M}_m, s)$.

Proof of lemma 4.8: Let $\psi \in D \setminus$ *fired*(D). Since $\psi \in D$, $\psi = \psi_k$ for some $1 \leq k \leq m$. Since $\psi \notin \mathcal{B}(i, \mathcal{M}_m, s)$, it follows by Corollary 4.7 that $\psi \notin \mathcal{B}(i, \mathcal{M}_k, s)$. By contraposition on the second clause of Lemma 4.6 it follows that $\neg\psi \in \mathcal{B}(i, \mathcal{M}_{k-1}, s)$, and again by Corollary 4.7 it follows that $\neg\psi \in \mathcal{B}(i, \mathcal{M}_m, s)$.

⊠

Summarizing, the set *fired*(D) meets the following three demands:

- *fired*$(D) \subseteq D$
- $W \cup$ *fired*(D) is consistent

- for all $\varphi \in D \setminus \mathit{fired}(D)$, $W \cup \mathit{fired}(D) \cup \{\varphi\}$ is inconsistent.

According to Def. 4.2, $\mathrm{Th}(W \cup \mathit{fired}(D))$ is an extension of W.
⊠

5. Discussion

In this paper we semantically investigated default reasoning from a dynamic, agent-oriented point of view. In order to do this we defined actions that model the reasoning by default of an agent. Execution of an action `try_jump` φ consists of three stages: first it is checked whether φ indeed is a default, i.e., it is checked whether φ is *plausible*, thereafter it is checked whether φ can *consistently* be included in the beliefs of the agent, and if this is the case the beliefs of the agent are *updated* accordingly. To model plausibility a new modality, viz. that of common possibility, is introduced. Due to the fact that we allow updates with arbitrary formulae, a strengthening of the usual consistency check of consistency-based default reasoning is necessary. The eventual belief update that follows application of the `try_jump` action is formalized using doxastic updates. Having introduced beliefs derived by default, we defined the Could-predicate and the Couldnot-predicate which formalize the idea of actions being believed to be (in)correct and/or (un)feasible plans. We furthermore defined the notion of default plans, which are actions that an agent may adopt by default in cases where it cannot decide on the basis of its knowledge which action to take.

Armed with the concepts defined to model the default reasoning of agents, we looked into the dynamics of supernormal default logic. We proved that successive jumps to conclusions can be used to construct default logic extensions of the belief sets of agents, thus gaining additional proof of the intuitive acceptability of our framework.

Future research concerning the topics introduced in this paper, will mainly be focussed on the concept of belief revision. By allowing agents to discard some of their beliefs (possibly acquired by default), a flexible framework would result that models both default reasoning and belief revision.

Acknowledgements

This research is partially supported by ESPRIT III BRA project No.6156 'DRUMS II', ESPRIT BRWG project No.8319 'MODELAGE', and the Vrije Universiteit Amsterdam; the third author is furthermore partially supported by the Katholieke Universiteit Nijmegen. Thanks are due to Thomas Arts and Theo Huibers for their very helpful criticism and comments on a draft version of this paper.

References

[1] P. Besnard. *An Introduction to Default Logic.* Springer-Verlag, Berlin-Heidelberg, 1989.

[2] G. Brewka. *Nonmonotonic Reasoning: Logical Foundations of Commonsense.* Cambridge University Press, Cambridge, 1991.

[3] B.F. Chellas. *Modal Logic. An Introduction.* Cambridge University Press, Cambridge, 1980.

[4] P.R. Cohen and H.J. Levesque. Intention is choice with commitment. *Artificial Intelligence*, 42:213–261, 1990.

[5] J. Dix. Default theories of Poole-type and a method for constructing cumulative versions of default logic. In B. Neumann, editor, *Proceedings of the 10th European Conference on Artificial Intelligence (ECAI'92)*, pages 289–293. John Wiley & Sons, 1992.

[6] P. Doherty. *NML3 - A Non-Monotonic Formalism with Explicit Defaults*. PhD thesis, Department of Computer and Information Science, Linköping University, Sweden, October 1991.

[7] D. Elgesem. *Action Theory and Modal Logic*. PhD thesis, Institute for Philosophy, University of Oslo, Oslo, Norway, 1993.

[8] R. Fagin, J.Y. Halpern, Y. Moses, and M.Y. Vardi. *Reasoning about Knowledge*. MIT Press, Cambridge MA, 1994. To appear.

[9] J. Halpern and J. Reif. The propositional dynamic logic of deterministic, well-structured programs. *Theoretical Computer Science*, 27:127–165, 1983.

[10] J.Y. Halpern and Y. Moses. Towards a theory of knowledge and ignorance. In *Proc. AAAI Workshop on Non-monotonic Logic*, pages 125–143, 1984. Reprinted in *Logics and Models of Concurrent Systems*, (ed., K. Apt), Springer-Verlag, Berlin/New York, pp. 459–476, 1985.

[11] J.Y. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artifical Intelligence*, 54:319–379, 1992.

[12] D. Harel. Dynamic logic. In D.M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 2, chapter 10, pages 497–604. D. Reidel, Dordrecht, 1984.

[13] J. Hintikka. *Knowledge and Belief*. Cornell University Press, Ithaca, NY, 1962.

[14] W. van der Hoek. Systems for knowledge and beliefs. *Journal of Logic and Computation*, 3(2):173–195, 1993.

[15] W. van der Hoek, B. van Linder, and J.-J. Ch. Meyer. A logic of capabilities. In A. Nerode and Yu. V. Matiyasevich, editors, *Proceedings of the Third International Symposium on the Logical Foundations of Computer Science (LFCS'94)*, volume 813 of *Lecture Notes in Computer Science*, pages 366–378. Springer Verlag, 1994.

[16] W. van der Hoek, B. van Linder, and J.-J. Ch. Meyer. Unravelling nondeterminism: On having the ability to choose (extended abstract). In P. Jorrand and V. Sgurev, editors, *Proceedings of the Sixth International Conference on Artificial Intelligence: Methodology, Systems, Applications (AIMSA'94)*, pages 163–172. World Scientific, 1994.

[17] W. van der Hoek and J.-J. Ch. Meyer. Possible logics for belief. *Logique & Analyse*, 127–128:177–194, 1989.

[18] J. Horty and Y. Shoham, Program Chairs. Reasoning about mental states: Formal theories & applications. Technical Report SS-93-05, AAAI Press, 1993. Papers from the 1993 Spring Symposium.

[19] G.E. Hughes and M.J. Cresswell. *A Companion to Modal Logic*. Methuen & Co. Ltd., London, 1984.

[20] A. Kenny. *Will, Freedom and Power*. Basil Blackwell, Oxford, 1975.

[21] K. Konolige. On the relation between default and autoepistemic logic. *Artificial Intelligence*, 35:343–382, 1988.

[22] S. Kraus and D. Lehmann. Knowledge, belief and time. *Theoretical Computer Science*, 58:155–174, 1988.

[23] H. Levesque. A logic of implicit and explicit belief. In *Proceedings of the National Conference on Artificial Intelligence*, pages 198–202, 1984.

[24] B. van Linder, W. van der Hoek, and J.-J. Ch. Meyer. Communicating rational agents. In B. Nebel and L. Dreschler-Fischer, editors, *KI-94: Advances in Artificial Intelligence*, volume 861 of *Lecture Notes in Computer Science (subseries LNAI)*, pages 202–213. Springer Verlag, 1994.

[25] B. van Linder, W. van der Hoek, and J.-J. Ch. Meyer. Tests as epistemic updates. In A.G. Cohn, editor, *Proceedings of the 11th European Conference on Artificial Intelligence (ECAI'94)*, pages 331–335. John Wiley & Sons, 1994.

[26] B. van Linder and E. Hubbers. Default ionic logic, its syntax and semantics – part I. Technical Report IR–296, Vrije Universiteit Amsterdam, August 1992.

[27] B. van Linder and E. Hubbers. Default ionic logic, its syntax and semantics – part II. Technical Report IR–297, Vrije Universiteit Amsterdam, August 1992.

[28] V.W. Marek and M. Truszczyński. *Nonmonotonic Logic*. Springer-Verlag, 1993.

[29] W. Marek and M. Truszczyński. Modal logic for default reasoning. *Annals of Mathematics and Artificial Intelligence*, 1:275–302, 1990.

[30] D. McDermot and J. Doyle. Non-monotonic logic I. *Artificial Intelligence*, 13:41–72, 1980.

[31] J.-J. Ch. Meyer and W. van der Hoek. A modal logic for nonmonotonic reasoning. In W. van der Hoek, J.-J. Ch. Meyer, Y.H. Tan, and C. Witteveen, editors, *Non-Monotonic Reasoning and Partial Semantics*, pages 37–77. Ellis Horwood, Chichester, 1992.

[32] J.-J. Ch. Meyer and W. van der Hoek. A default logic based on epistemic states. In M. Clarke, R. Kruse, and S. Moral, editors, *Proceedings of ESQUARU 1993*, volume 747 of *Lecture Notes in Computer Science*, pages 265–273. Springer Verlag, 1993.

[33] J.-J. Ch. Meyer and W. van der Hoek. An epistemic logic for defeasible reasoning using a meta-level architecture methaphor. Technical Report IR-329, Vrije Universiteit Amsterdam, August 1993.

[34] J.-J. Ch. Meyer and W. van der Hoek. *Epistemic Logic for AI and Computer Science.* Cambridge University Press, 1994. To appear.

[35] R.C. Moore. Reasoning about knowledge and action. Technical Report 191, SRI International, 1980.

[36] R.C. Moore. A formal theory of knowledge and action. Technical Report 320, SRI International, 1984.

[37] A. Nait-Abdallah. The Logic of Partial Information. To appear, 1994.

[38] D. Poole. A logical framework for default reasoning. *Artificial Intelligence*, 36:27–47, 1988.

[39] A.S. Rao and M.P. Georgeff. Asymmetry thesis and side-effect problems in linear time and branching time intention logics. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI91)*, pages 498–504, 1991.

[40] A.S. Rao and M.P. Georgeff. Modeling rational agents within a BDI-architecture. In J. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning*, pages 473–484, San Mateo CA, 1991. Mogan Kaufmann.

[41] A.S. Rao and M.P. Georgeff. A model-theoretic approach to the verification of situated reasoning systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI'93)*, pages 318–324, 1993.

[42] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13:81–132, 1980.

[43] Y. Shoham. Agent-oriented programming. *Artificial Intelligence*, 60:51–92, 1993.

[44] E. Spaan. *Complexity of Modal Logics.* PhD thesis, Universiteit van Amsterdam, 1993.

[45] S.R. Thomas. *PLACA, An Agent Oriented Programming Language.* PhD thesis, Department of Computer Science, Stanford University, Stanford CA, September 1993. Appeared as technical report STAN-CS-93-1487.

[46] M. Truszczyński. Modal interpretations of default logic. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 393–398, 1991.

[47] F. Veltman. Defaults in update semantics. Technical Report LP-91-02, ILLC, University of Amsterdam, 1991.

[48] G.H. von Wright. *Norm and Action.* Routledge & Kegan Paul, London, 1963.