Erkenntnis manuscript No.
(will be inserted by the editor)

The Logic of Imagination Acts

A Formal System for the Dynamics of Imaginary Worlds

Joan Casas-Roma · Antonia Huertas ·
M. Elena Rodríguez

Received: 23 August 2018 / Accepted: 27 May 2019

Abstract Imagination has received a great deal of attention in different fields such as psychology, philosophy and the cognitive sciences, in which some works provide a detailed account of the mechanisms involved in the creation and elaboration of imaginary worlds. Although imagination has also been formalized using different logical systems, none of them captures those dynamic mechanisms. In this work, we take inspiration from the Common Frame for Imagination Acts, that identifies the different processes involved in the creation of imaginary worlds, and we use it to define a dynamic formal system called the Logic of Imagination Acts. We build our logic by using a possible-worlds semantics, together with a new set of static and dynamic modal operators. The role of the new dynamic operators is to call different algorithms that encode how the formal model is expanded in order to capture the different mechanisms involved in the creation and development of imaginary worlds. We provide the

This work is funded by EC FP7 grant 621403 (ERA Chair: Games Research Opportunities), by the Project "Hybrid Intensional Logic" (ref. FFI2013-47126-P) given by the Spanish MINECO, the project 2018-2020: Traducciones, lógicas combinadas, descripciones, lógica intensiva, teoría de tipos, lógica híbrida, identidad, lógica y educación (ref. FFI2017-82554), given by the Spanish MICINN, and a doctoral grant from the Universitat Oberta de Catalunya (UOC).

This is a post-peer-review, pre-copyedit version of an article published in Erkenntnis. The final authenticated version is available online at: https://doi.org/10.1007/s10670-019-00136-z

J. Casas-Roma (*corresponding author*)
The Metamakers Institute,
Falmouth University (Falmouth, UK)
Tel.: +34 660 581 622
E-mail: joan.casasroma@falmouth.ac.uk

A. Huertas, M. E. Rodríguez
Department of Computer Science, Multimedia and Telecommunication,
Universitat Oberta de Catalunya (Barcelona, Spain).

definitions of the language, the semantics and the algorithms, together with an example that shows how the model is expanded. By the end, we discuss some interesting features of our system, and we point out to possible lines of future work.

Keywords imagination · imaginary worlds · modal logic · dynamic logic · algorithms

1 Introduction and Motivations

Imagining is something we use everyday in our lives, and in a wide variety of ways: when planning our next move in a chess game, when picturing how we could decorate our new room, or even when listening to a story-teller, our mind creates, develops and evaluates imaginary worlds aimed to guide our actions, update our beliefs, or simply entertain us. Imagination has received a great deal of attention by philosophers, cognitive scientists and psychologists (as it can be seen in works like [24], [10], [19], or [13], among others). Its interest within the studies of the mind is beyond any doubt, and its relation to other mechanisms of the mind, such as emotions, behavior, desires and beliefs, makes imagination particularly interesting in many different areas.

In the literature, there are many works that study the relation of imagination with respect to other mental attitudes, such as knowledge (as in [14]), beliefs or desires, or how imagination affects our decision-making abilities (like [18]), or our emotions (as in [22]); some of these works also provide insights on how the mechanisms of creating and developing new imaginary worlds work (like [20], [26] or [15]). These mechanisms are used to create new representations of alternative worlds, which were not in our mind before, and which will be discarded once we have achieved our goal with respect to them.

The main goal of the present paper is to define a dynamic system that allows to capture, using formal models based on modal logic, how imaginary worlds are created and developed. Although there are different logical systems that already deal with imagination, none of them focus on its dynamic aspect. The theoretical background of our proposal is based on the so-called Common Frame for Imagination Acts, which identifies the mechanisms that are used in order to create and develop new imaginary worlds. In order to capture the intuitions behind such theory, we define a formal syntax and semantics that, guided by a set of algorithms, allow to model the dynamics of imaginary worlds. The system we define provides a detailed account of how such dynamics work in a formal setting, and thus provides a tool that can be used to study those dynamics in greater detail in a formal, algorithmic setting.

After introducing the main motivations in the current section, we briefly review the underlying theories and existing logics of imagination in Section 2, and we point out how there are no dynamic logics of imagination. Throughout Section 3, we define the syntax, models, algorithms and semantics of the Logic of Imagination Acts, and we provide an example of the logic in use in Section

4. Finally, we conclude with a discussion of the pros and cons of our proposal in Section 5, and point out to some interesting lines of future work.

2 Theories and Logics of Imagination

In order to account for the dynamics of how imaginary worlds are created and developed we will focus on the *Common Frame for Imagination Acts*, introduced in [8], and which is mainly based on the theories of Nichols and Stich in [20], Williamson in [26] and Langland-Hassan in [15]. The Common Frame for Imagination Acts analyses how the previous works identify certain mechanisms involved in the formation of imaginary worlds and fine-grains those mechanisms into four different processes, called the *Initialization*, the *Description*, the *Default Evolution* and the *Unscripted Additions*. Although we provide more details on those processes in Section 3.3, they can be briefly summarized as follows:

1. The Initialization process takes an initial premise and creates a brand-new set of imaginary worlds (which can be formed by one or more worlds), shaped according to the conditions specified by such initial premise. We use the term *imaginary scenario* to refer to the set of imaginary worlds that result from the execution of one imagination processes.
2. The Description uses certain rules to infer what other static details would be the case in the imaginary scenario, based on what is already the case in there. The emphasis on the "static" means that this process is responsible for elaborating the state of affairs of an imaginary world in which nothing has happened yet.
3. The Default Evolution is also used to infer what else would be the case in the imaginary scenario but, conversely to the previous process, the Default Evolution does take into account a certain action happening, and thus changing the imaginary world in some way.
4. The Unscripted Additions correspond to the voluntary addition of new premises to already existing imaginary worlds. The particularity of those newly added premises is that they do not follow any particular rule, but rather a desire of the agent to alter the imaginary world in such and such way.

Throughout the rest of the present work we take the Common Frame for Imagination Acts as our underlying theory, and so our formal system will aim to represent and model the dynamics captured by the previous processes.

When considering formal systems, few authors have ventured into the uncharted seas of logic and imagination. David Lewis, in [16] define a logic to account for counterfactual reasoning by using a system of spheres and a modal operator that moves the evaluation point to counterfactual worlds. Later, in [21], Niiniluoto formalizes imagination as a propositional attitude and discusses some of its properties. Costa-Leite, in [9], goes one step beyond and formalizes the distinction between "imagination", "conception" and "possibility" through following the intuitions of Descartes and Hume. Wansing brings beliefs into the picture in [25] and uses neighborhood semantics and STIT mechanics to account for agentive imagination. Through various works like [2], [4] and [3], Berto formalizes conceivability in both a paraconsistent and a classical setting, and introduces the mechanics of "aboutness", which determine what is relevant for the agent to import conceiving an alternative world.

Even though these works highlight very interesting features of imagination, they all represent imagination in *static*, pre-determined scenarios, like snapshots of a specific moment. Although Wansing's approach goes one step beyond and takes into account the agentive character of imagination, it still works in predefined, tree-like structures: the agent can be seen as "choosing" what to imagine, indeed, but these choices are already contained in the initial model of the situation.

Our approach amends this and captures something that has been overlooked in previous works: imagination is, in essence, dynamic. When we imagine, we create and unfold worlds that are not real, but which nevertheless are governed by a certain set of rules or mechanisms, as identified by the previously mentioned theories of imagination. Even though a first approach to a dynamic logic of imagination acts is defined in [7], the resulting system ends up being too shallow, and the dynamics of imagination end up being rather simplified. In the next section, we define the Logic of Imagination Acts as a formal system accounting for the dynamics of imagination, which is the main contribution of the present work. The main aim of our proposal is to define a set of algorithms that can be executed at any time over a formal model and which will compute how this model is expanded by further developing an imaginary world, following the mechanisms described in the Common Frame for Imagination Acts. Due to this, our proposal is based on a possible-world semantics, uses a modal logical language, and adds a set of dynamic algorithms that detail how the model must be expanded, whenever they are executed.

3 The Logic of Imagination Acts

Throughout this section we introduce the definitions needed in order to define the Logic of Imagination Acts. Nevertheless, we will need to refer to certain elements of the system before explicitly defining them in a formal way.

Our main goal in this contribution is to provide a first approach to the dynamic mechanisms involved in the creation and development of imaginary worlds. Due to this, we choose to define our logic on a propositional language in order to focus on the dynamic aspects of the way imaginary worlds are added and expanded, while avoiding serious technical challenges that first-order modal systems face with regards to their domains, as it can be seen in works such as [11]. In this sense, we follow the approach of [25] in interpreting expressions such as "Alice imagines a unicorn", as the propositional expression "Alice imagines that there is a unicorn". As we point out in Section 5, a natural direction of further work is to explore how this logic, and the algorithms defined in it, could be expanded into a first-order setting.

3.1 Syntax

The language of the Logic of Imagination Acts is formed by a countably infinite set of *atomic formulas*, called ATOM, and represented by the lowercase letters p, q, and so on. There is also a countably infinite set of *nominals* (taken from hybrid logic), called NOM and represented by the lowercase letters i, j, and such. Besides, we have a countably infinite set of *atomic actions* (or simply *actions*), called ACT, and represented by the Greek letters α, β, and so on; note that these actions will only be used, in our language, to sign a special modal operator that we introduce in the further lines.

We use the standard propositional operators $\neg, \wedge, \vee, \rightarrow$ (standing for "negation", "conjunction", "disjunction" and "material implication", respectively); besides, we also use the hybrid[1] operator @. We use bracket symbols (, [,),] as usual (and we omit them when the context is clear).

We introduce four new dynamic operators, and four new static ones. Intuitively, each dynamic operator is responsible for calling one of the four algorithms (which will be defined in Section 3.3), and each dynamic operator has a corresponding static operator, used to evaluate the transitions created by the related execution of the algorithm. The new dynamic operators are $\mathrm{Init}(\delta)$, $\mathrm{Descr}(\zeta)$, $\mathrm{Evo}(\alpha)$ and $\mathrm{Add}(\delta)$, which are related to the static operators $\langle\delta\rangle^I$, $\langle\zeta\rangle^D$, $\langle\alpha\rangle^E$ and $\langle\delta\rangle^A$, respectively. Aside form this, we also introduce a new static modal operator $\langle\mathrm{Img}\rangle$ standing for a general, process-independent imagination operator.

The set FORM of well-formed formulas of the language are inductively defined as follows:

$$\begin{gathered} i \mid p \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \varphi \rightarrow \psi \mid @_i\varphi \mid \langle\mathrm{Img}\rangle\varphi \mid \\ \mathrm{Init}(\delta) \mid \mathrm{Descr}(\zeta) \mid \mathrm{Evo}(\alpha) \mid \mathrm{Add}(\delta) \mid \\ \langle\delta\rangle^I\varphi \mid \langle\zeta\rangle^D\varphi \mid \langle\alpha\rangle^E\varphi \mid \langle\delta\rangle^A\varphi \end{gathered}$$

where $i \in \mathrm{NOM}$, $p \in \mathrm{ATOM}$, $\{\varphi, \psi\} \subseteq \mathrm{FORM}$, $\delta \in \mathrm{FORM}^*$, $\zeta \in \mathrm{FACT}$ and $\alpha \in \mathrm{ACT}$; we explain in the following lines what these special sets of formulas are[2]. We distinguish certain particular sets of formulas that are aimed to be used only by certain operators:

- FORM^* corresponds to the propositional fragment of FORM; we typically refer to elements of FORM^* by using δ, γ and such. Therefore, a formula $\delta \in \mathrm{FORM}^*$ can be of the following form:
  - p (for $p \in \mathrm{ATOM}$)

[1] In a nutshell, hybrid logic allows to uniquely identify possible worlds through a set of *nominals* NOM. Then, a formula $@_i\varphi$ expresses that "at world i, it is the case that φ". The addition of such operator to our logic can greatly increase its potential by allowing to express what is the case in different stages of an imagination act. For a thorough introduction to hybrid logic, we refer to [5], or [6].

[2] The way we define the set of actions ACT is inspired by the way Propositional Dynamic Logic, or PDL, defines a set of *atomic programs* Π_0. In PDL (see [12]), these programs are also used to sign a modal operator, just as we do in our case; nevertheless, PDL also defines a set of operators over programs, which can be used to combine them in different ways.

  - $\neg\varphi$ (for $\varphi \in$ FORM*)
  - $\varphi \vee \psi$ (for $\{\varphi, \psi\} \subseteq$ FORM*)
  - $\varphi \wedge \psi$ (for $\{\varphi, \psi\} \subseteq$ FORM*)
  - $\varphi \to \psi$ (for $\{\varphi, \psi\} \subseteq$ FORM*)
- FACT is a particular subset of FORM, and it corresponds to the set of formulas aimed to represent the *factual rules* in which the agent believes, and which represent how certain conditions could give rise to certain consequences; we typically refer to elements of FACT by using ζ, ζ_1, ζ_2 and so on. We require every formula $\zeta \in$ FACT to be of the following form[3]:
  - $\varphi\langle\rightarrow\rangle\psi$ (for $\{\varphi, \psi\} \subseteq$ FORM*)
- SCRIPT is another particular subset of FORM, and it corresponds to the set of formulas aimed to represent the *scripts* in which the agent believes, and which represent what consequences, given certain conditions, an action or event will trigger in an imaginary world; we typically refer to elements of SCRIPT by using ξ_1, ξ_2 and such. We require every formula $\xi \in$ SCRIPT to be of one of the following forms, where $[\rightarrow]^\alpha \equiv \neg\langle\rightarrow\rangle^\alpha\neg$ as usual in modal logic:
  - $\varphi\langle\rightarrow\rangle^\alpha\psi$ (for $\{\varphi, \psi\} \subseteq$ FORM* and $\alpha \in$ ACT)
  - $\varphi[\rightarrow]^\alpha\psi$ (for $\{\varphi, \psi\} \subseteq$ FORM* and $\alpha \in$ ACT)

We introduce two symbols $\top, \bot$ to refer to *truth* and *falsity*, respectively, and we define them as follows (for $p \in$ ATOM):

$$\top \equiv p \vee \neg p$$
$$\bot \equiv p \wedge \neg p$$

Furthermore, and in order to distinguish particular formulas belonging to the sets FACT and SCRIPT, we introduce two new symbols that will be used to encode a combination of two different operators as follows (for some $\zeta \in$ FACT and $\alpha \in$ ACT):

$$\varphi\langle\rightarrow\rangle\psi \equiv \varphi \rightarrow \langle\zeta\rangle^D\psi$$
$$\varphi\langle\rightarrow\rangle^\alpha\psi \equiv \varphi \rightarrow \langle\alpha\rangle^E\psi$$

The intuitive reading of the dynamic operators is the following: formula Init(δ) is read as "the agent creates a new imaginary world using the initial premise δ"; formula Descr(ζ) is read as "the agent elaborates on the static details of an imaginary world by using the factual rule ζ"; formula Evo(α) is read as "the agent elaborates on the consequences that action α would have in the imaginary world"; lastly, formula Add(δ) is read as "the agent adds a new premise δ into the imaginary world".

The associated static operators can be intuitively interpreted as follows: formula $\langle\delta\rangle^I\varphi$ is interpreted as "after an execution of the Initialization process,

[3] In a nutshell, we only allow the antecedent and the consequent to belong to the propositional fragment of the language because we are interested in seeing how imaginary worlds are created and developed: modal and hybrid operators convey information about other worlds and the relations between different worlds. Leaving aside the technical complications that this would involve, imagining *about* other worlds or their relations falls outside the scope of the current goal of this work.

with initial premise δ, the agent imagines a world in which φ is the case". The rest of the corresponding static formulas are read similarly. Operator $\langle \mathrm{Img} \rangle$ is intuitively interpreted as a kind of "wildcard" imagination operator concerning any of the mechanisms involved in the elaboration of imaginary worlds. As such, a formula $\langle \mathrm{Img} \rangle \varphi$ is interpreted as "through some process of imagination, the agent imagines a world where φ holds", or, for short, "the agent imagines a world where φ holds".

Although the agent's beliefs about the real world play a very important role in imaginary worlds, we define the Logic of Imagination Acts without using an explicit doxastic operator. The reasons for doing so is because our current goal is to focus on the dynamics described by the four algorithms that will capture the processes of the Common Frame for Imagination Acts: adding more technical complexity to the setting could easily deviate us from our goal, and so we have decided to omit an explicit formal representation of beliefs for now. However, in the next section we explain how the way we define our models still allows us to implicitly account for beliefs up to the extend that we need them for the dynamic mechanisms captured in the logic.

3.2 The Models for Imagination Acts

We define a *Model for Imagination Acts* as a structure $\mathcal{M} = \langle W, R_I, R_D, R_E, R_A, V, N \rangle$, where:

- W is a non-empty set of elements called *possible-worlds* or *states of affairs*. We use the lowercase letters $w, v, u, \dots$ to refer to the elements of W.
- $R_I \subseteq W \times W \times \mathrm{FORM}^*$ is a ternary relation called the *initialization relation*. Intuitively, an element (w, v, δ) captures how, through the Initialization process, and by using an initial premise δ and taking w as the *world of reference*, a new imaginary world v is created. We use triplets of the form $(w, v, \delta), (u, z, \gamma), \dots$ to refer to elements of R_I.
- $R_D \subseteq W \times W \times \mathrm{FACT}$ is a ternary relation called the *description relation*. Intuitively, an element (w, v, ζ) captures how, through the Description process, and by using a factual rule $\zeta \in \mathrm{FACT}$ and taking w as the *world of reference*, an imaginary world v resulting from the application of ζ is created. We use triplets of the form $(w, v, \zeta_1), (u, z, \zeta_2), \dots$ to refer to elements of R_D.
- $R_E \subseteq W \times W \times \mathrm{ACT}$ is a ternary relation called the *evolution relation*. Intuitively, an element (w, v, α) captures how, through the Default Evolution process, by performing (or imagining to perform) an action $\alpha \in \mathrm{ACT}$, and by taking w as the *world of reference*, an imaginary world v is created as a result of action α taking place. We use triplets of the form $(w, v, \alpha), (u, z, \beta), \dots$ to refer to elements of R_E.
- $R_A \subseteq W \times W \times \mathrm{FORM}^*$ is a ternary relation called the *addition relation*. Intuitively, an element (w, v, δ) captures how, through the Unscripted Addition process, using a premise δ, and by taking w as the *world of*

reference, an imaginary world v is created. We use triplets of the form $(w, v, \delta), (u, z, \gamma), \ldots$ to refer to elements of R_A.

- V : ATOM $\rightarrow \mathcal{P}(W)$ is a function from atomic formulas of the language to subsets of the power set of W, called the *valuation function.* Intuitively, it keeps track of which atomic formulas are true at which subset of possible worlds.
- N : NOM $\rightarrow W$ is an exhaustive function setting, for each element of NOM, a possible world in W, and called the *nominal function.* Intuitively, it specifies which nominal is used to identify each world.

Even though neither the language nor the models explicitly account for beliefs, we argue that our logic implicitly model beliefs in the following way:

- Any of the initial real possible worlds account for a state of affairs the agent believes to be possible. As we have no doxastic ordering, we consider that the agent believes them all to be equally plausible.
- The sets of formulas FACT and SCRIPT capture the rules the agent believes in, regarding the way imaginary worlds can be developed.

Following these conventions, we can focus on the dynamics of imagination acts without adding more technical complexity, but still being able to refer to the agent's beliefs when needed. As we point out in Section 5, extending the system with an explicit doxastic relation is a rather interesting line of future work. We do not elaborate on this point for reasons of space, but, as we already anticipate in the aforementioned section, effectively integrating doxastic models into our setting would present some major technical challenges that fall outside the scope of our present goal.

3.3 The Algorithms

We now proceed to define four different algorithms, each one accounting for one of the four distinct processes involved in the creation and development of imaginary worlds. During the execution of any of these four algorithms, a Model for Imagination Acts $\mathcal{M}$ is expanded into its *expanded model* $\mathcal{M}^+$. We refer to any of the elements of $\mathcal{M}^+$ as the *expanded element* (with its corresponding name), and we identify them as $\mathcal{M}^+ = \langle W^+, R_I^+, R_D^+, R_E^+, R_A^+, V^+, N^+ \rangle$. Note, however, that not each algorithm will expand every element of $\mathcal{M}$; nevertheless, we will still talk about the expanded version of such elements, when referring to them either during, or just after the execution of one of such algorithms[4].

Even though each algorithm has its own particularities, there are certain processes within them that are very similar, both in the way they work and in their outcome. We provide a complete specification for each algorithm, but we will assign names or labels to some of those internal processes. The

[4] Each one of the four algorithms we define in the following pages is executed upon a model $\mathcal{M}$. On the first step of their execution, there are certain initial conditions that each algorithm has to check; if any of those conditions is not fulfilled, the algorithm does not expand model $\mathcal{M}$ in any way; in that case, we consider that $\mathcal{M}^+ = \mathcal{M}$.

reason for doing so is to help the reader to identify, beforehand, what these processes do, and identify that they stand for similar processes appearing in the other algorithms. Note that, even if they intend to do the same thing, two similar processes in two different algorithms might still be technically different: they may refer to different parts of the model, or they may alter them in different ways. The core part of each algorithm, however, is quite different in the way it works and the rules it follows to modify the model. An exception would be the cases of the `Init` and the `Add` algorithm, which work almost in the same way, but which requires different parameters; in particular, the `Init` algorithm builds up on a real possible world, while the `Add` algorithm requires an imaginary possible world. For the sake of clarity, nevertheless, we still provide an independent definition for each one of them.

It is important to keep in mind that the execution of the algorithms is not required to follow any specific order. If there is yet no imaginary world created, we do need to start with an execution of the algorithm responsible for handling the Initialization of the scenario, but, after this, the agent may choose to elaborate on the imaginary worlds by using any of the other three algorithms, without the need to follow any specific order.

3.3.1 The `Init` Algorithm

This algorithm corresponds to the Initialization process, and it handles the creation of new imaginary worlds from scratch, based on a certain world of reference representing the state of affairs of the real world, as believed by the agent, and by using a certain initial premise determining the content of the imagining to be initiated. As the authors claim in [8], the effect that the so-called world of reference have upon these brand-new imaginary worlds is captured by importing all those facts (i.e.; atomic propositions) that are true in the real world of reference[5], as long as they are consistent with the initial premise δ used to create the new imaginary worlds. The whole process is depicted in Figure 1.

The `InitAlg` computes the initial act of imagining a certain premise $\delta \in$ FORM* while taking a real possible world w^R as the world of reference, which corresponds to the following call:

$$\texttt{InitAlg}(\delta, w^R)$$

The algorithm follows these steps:

1. *Check initial conditions*: If world w^R is *not* a real possible world (that is, if there exists some v such that $(v, w^R, \#)$ is in either R_I, R_D, R_E, or R_A, being $\#$ one of the corresponding formulas required by each accessibility relation), do nothing. Similarly, if δ is contradictory (that is, if $\delta \equiv \bot$), do nothing.

[5] Roughly speaking, this process aims to account for a *ceteris paribus* effect in the new imaginary world. As argued in [4], conceived and imagined worlds are usually governed by such effect.

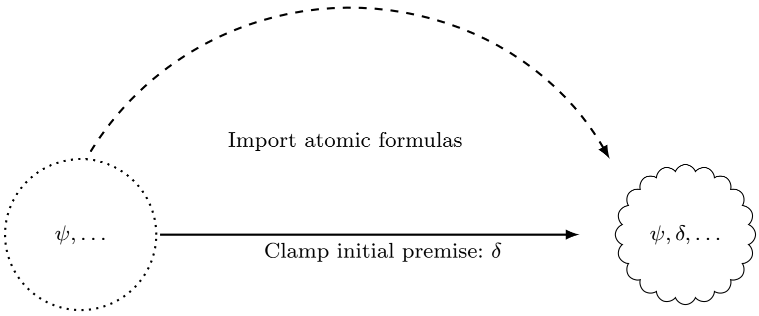

Fig. 1 Initiating the imagining using an initial premise.

2. *Compute DNF*: In order to handle the formula in an efficient way, we compute the *Disjunctive Normal Form* (DNF) of δ, to which we refer as $\mathrm{DNF}(\delta) = \delta_1 \vee \ldots \vee \delta_n$.
3. *Create imaginary worlds*: Create new imaginary possible worlds $w_1, \ldots, w_n$ for each clause $\delta_1, \ldots, \delta_n$ in $\mathrm{DNF}(\delta)$. This defines the *expanded set of possible worlds* as follows:
$$W^+ = W \cup \{w_1, \ldots, w_n\}$$
4. *Expand accessibility relation*: Once the new possible worlds have been created, the `InitAlg` must create the new initialization relations expressing that, by imagining formula δ and taking w^R as the world of reference, the agent creates new imaginary worlds $w_1, \ldots, w_n$. This defines the *expanded set of initialization relations* as follows:
$$R_I^+ = R_I \cup \Big(\bigcup_{i=1 \ldots n} \{(w^R, w_i, \delta)\} \Big)$$
5. *Expand nominal structure*: Now, the `InitAlg` must add a set of new nominals to refer to the newly created imaginary worlds. This defines both the *expanded set of nominals*, by adding one new nominal k_i for each new possible world w_i, and the *expanded nominal function*, which is a functional extension of N relating the new pairs of nominals and possible worlds:
$$\mathrm{NOM}^+ = \mathrm{NOM} \cup \{k_1, \ldots, k_n\}$$
$$N^+ = N \cup \Big(\bigcup_{i=1 \ldots n} \{(k_i, w_i)\} \Big)$$
6. *Expand valuation function*: Last but not least, the `InitAlg` must expand the valuation function to account for what is the case in the new imaginary possible worlds. In order to do so, the algorithm must account for both the atoms that are present in each δ_i, and also for the atoms that are true in the world of reference w^R and which should be imported to the new

imaginary worlds, provided they do not appear in δ_i; this is so because any atom appearing in δ_i would have been given priority, with respect to the atoms holding at w^R. Therefore, the definition of the *expanded valuation function* involves two different phases:

(a) *Clamp new atoms*: Firstly, the `InitAlg` must set the new valuation functions according to the atoms p appearing in δ_i, for each new imaginary world w_i:

$$V_1^+(p) = V(p) \cup \Big(\bigcup_i \{w_i \mid p \text{ is a positive literal appearing in } \delta_i\}\Big)$$

(b) *Import existing atoms*: Then, it must import all the atoms holding at world w^R, provided they do not appear in δ_i, for each new imaginary possible world w_i:

$$V^+(p) = V_1^+(p) \cup \Big(\bigcup_i \{w_i \mid w^R \in V_1^+(p) \text{ and } p \text{ is not a literal of } \delta_i\}\Big)$$

7. The `InitAlg` has finished its execution: a new set of imaginary possible worlds satisfying δ has been created, and these worlds are now accessible through the initialization relation R_I from the world of reference w^R.

3.3.2 The `Descr` Algorithm

In order to capture the *Description* process, the `DescrAlg` will make use of the so-called *factual rules*, which are represented in our language by formulas belonging to FACT. In brief, factual rules are a kind of implication-based formula capturing a sort of hypothetical conditional of the form "if φ was the case, then ψ would also be the case"; as we saw in Section 3.1, this notion of hypothetical condition is captured using operator $\langle\rightarrow\rangle$. This kind of rules are used in the Description process to express that if the imaginary world fulfills a certain condition, then it could also fulfill a certain outcome as well. It is important to note that, even though the `Init` process may have created more than one new imaginary world, the execution of the `Descr` process is focused on one of such imaginary worlds. In other words, the agent do not simultaneously elaborate the static details of all the possible imaginary worlds that have been created, but rather chooses to elaborate the state of affairs represented in one of them; this also holds for the `Evo` and `Add` processes as well. Nevertheless, as the algorithms can be executed at any time, and by choosing any admitted world as the world of reference, it is possible for the agent to also elaborate the details of a different imaginary world, although this requires a different execution of the relevant algorithm.

This elaboration of the imaginary world, nevertheless, must be formally handled in a very specific way. We want our formal models to be able to keep track of the different imaginative processes the agent uses, and to show how they affect the elaboration of the imaginary worlds. Therefore, if we were to represent the application of the rule "if it was raining, I would be carrying and umbrella" by adding the fact "I am carrying an umbrella" to the already existing imaginary world satisfying the fact "it is raining", we would lose track, in the formal model, of *how* that world became such in which I am *also* carrying an umbrella.

Taking these considerations, we want the Description process to be represented, in our formal models, as an accessibility relation linking two *different* worlds: the world of reference, where the antecedent φ of the factual rule expressing that "if φ was the case, then ψ would also be the case" holds, and a different imaginary world, similar to the world of reference in everything, except by the fact that ψ also holds at that new world as a result of applying that specific factual rule. In other words, these formulas (and the way they are processed by our logical system) represents a kind of *Modus Ponens* rule in which the antecedent is evaluated at the world of reference, but the consequent affects a different accessible world[6].

The execution of the algorithm representing the Description process, therefore, will need a world of reference w^R and a certain formula $\varphi\langle\rightarrow\rangle\psi \in$ FACT. Then, the algorithm will need to check whether the antecedent φ is true at w^R and, if it is, create a new imaginary world v, accessible from w^R, and in which ψ holds; the rest of atomic formulas determining the state of affairs of the new world v will be taken from w^R. In other words, the only changes that v will have with respect to w^R are those changes needed to make ψ true at v. Figure 2 represents how the Description process would work.

A call to the `DescrAlg` needs a factual rule $\zeta \in$ FACT and an imaginary possible world as the world of reference w^R, as follows:

$$\texttt{DescrAlg}(\zeta, w^R)$$

The algorithm follows these steps:

1. *Check initial conditions*: If world w^R is a real world (that is, there is no v such that $(v, w^R, \#)$ is in neither R_I, R_D, R_E, nor R_A, being $\#$ one of the corresponding formulas required by each accessibility relation), do nothing.
2. *Compute DNF*: Formula $\zeta \in$ FACT is required to be of the form $\zeta = \varphi\langle\rightarrow\rangle\psi$ (where $\{\varphi, \psi\} \subseteq$ FORM*). Now, the `DescrAlg` must check whether the antecedent of such formula is true at the world of reference and, if it is, then it must elaborate on the description of the imaginary world by clamping its consequent to a new imaginary world (or several new ones, depending

[6] It is worth noting how this kind of hypothetical conditionals are somewhat similar to the kind of formulas David Lewis is interested in [16]. Nevertheless, Lewis' use is different from ours: in his work, Lewis evaluates a formula of the kind $\varphi \Box\!\!\rightarrow \psi$ at a world aimed to represent the real world, and the operator $\Box\!\!\rightarrow$ moves the whole evaluation point to an accessible counterfactual world in order to assess whether the conditional $\varphi \rightarrow \psi$ holds in there. His way of evaluating hypothetical conditionals, therefore, is by moving the whole conditional to an alternative world. Our way of understanding them, however, will be to asses whether the antecedent φ holds in the *current* world of evaluation, and if so, then create a *new* world fulfilling ψ and defined by taking the current one as the reference. Our understanding of this kind of conditionals, therefore, will be used to determine the way a certain imaginary world could change, given the information provided by the specific formula being evaluated.

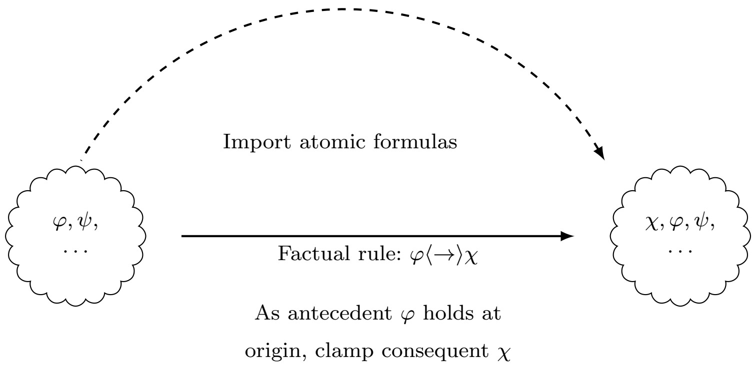

Fig. 2 Follow a factual rule to elaborate on the static details.

on the form of the consequent). If $\mathcal{M}, w^R \vDash \varphi$ (the satisfiability symbol is introduced later on in Section 3.4, but in this case it expresses that the antecedent of ζ holds at the world of reference w^R), then the `DescrAlg` must compute $\mathrm{DNF}(\psi) = \psi_1 \vee \ldots \vee \psi_n$ (that is, the DNF of the consequent of ζ).

3. *Create imaginary worlds*: The algorithm must create n new imaginary possible worlds $w_1, \ldots, w_n$, one for each clause in $\psi_1 \vee \ldots \vee \psi_n$. This defines the *expanded set of possible worlds* as follows:

$$W^+ = W \cup \{w_1, \ldots, w_n\}$$

4. *Expand accessibility relation*: Create new R_D relations linking the world of reference w^R with each new world $w_1, \ldots, w_n$ created in the previous step. This defines the *expanded set of description relations* as follows:

$$R_D^+ = R_D \cup \Big(\bigcup_{i=1 \ldots n} \{(w^R, w_i, \zeta)\} \Big)$$

5. *Expand nominal structure*: The `DescrAlg` must now add a set of new nominals to refer to the newly created imaginary worlds. Again, this defines both the *expanded set of nominals* and the *expanded nominal function*:

$$\mathrm{NOM}^+ = \mathrm{NOM} \cup \{k_1, \ldots, k_n\}$$

$$N^+ = N \cup \Big(\bigcup_{i=1 \ldots n} \{(k_i, w_i)\} \Big)$$

6. *Expand valuation function*: Once more, the algorithm must account for both the atoms that are present in each ψ_i, and also for the atoms that are true in the world of reference w^R and which should be imported to the new imaginary worlds, provided they do not appear in ψ_i. Again, the definition of the *expanded valuation function* involves two different phases:

(a) *Clamp new atoms*: Firstly, the `DescrAlg` must set the new valuation functions according to the atoms p appearing in ψ_i, for each new imaginary possible world w_i:

$$V_1^+(p) = V(p) \cup \Big(\bigcup_i \{w_i \mid p \text{ is a positive literal appearing in } \psi_i\}\Big)$$

(b) *Import existing atoms*: Then, it must import all the atoms that are true at world w^R, provided they do not appear in ψ_i, for each new imaginary possible world w_i:

$$V^+(p) = V_1^+(p) \cup \Big(\bigcup_i \{w_i \mid w^R \in V_1^+(p) \text{ and } p \text{ is not a literal of } \psi_i\}\Big)$$

7. The `DescrAlg` has finished its execution: a new set of imaginary possible worlds has been created as a result of the agent following a factual rule ζ describing what else could be the case in a particular imaginary world fulfilling certain conditions. Moreover, these new imaginary worlds are accessible through the description relation R_D from the world of reference w^R.

3.3.3 The `Evo` Algorithm

The main difference with respect to the previous `DescrAlg` is that the `EvoAlg` involves a certain *action* or *event* to happen. In this case, the algorithm will need to use formulas belonging to the set SCRIPT, which have the form $\varphi\langle\rightarrow\rangle^\alpha\psi$, and which encode something like "if φ was the case, and action α happened, then ψ would also be the case". In this case, we also want to capture the sense of "modal conditional" by evaluating whether the antecedent of a script holds, at the world of reference, and, if it does so, then clamp the consequent of such script into a new accessible imaginary world. Note that, in this case, though, we cannot use a"simple modal conditional", as we did in the previous case, but instead a "signed modal conditional", which would depend on a certain action α.

The way of executing the `EvoAlg`, though, is different from the `DescrAlg`. Whereas in the previous case our agent picked up a specific factual rule and used it to elaborate on the scenario, we do not want our agent, in this case, to imagine that a single script affects the scenario, but rather to imagine that a certain action α takes place, and then infer *every* consequence that α would carry with it: this represents a major change in the way the algorithms for the Description and the Default Evolution processes will work.

The motivations behind this decision are based on the fact that, although we can choose to elaborate the static details of an imaginary world in a step-by-step manner, when elaborating on the consequences of certain events or actions, we are not as selective as with the static details. Consider an imagining about a tea-party in which the kettle is burning hot (and so is the tea inside of it) and in which we are really thirsty; now, say that we imagine to drink the whole cup in a single gulp. Would we typically imagine that, by drinking the whole cup of burning-hot tea, we *just* become satiated, without taking into account which other consequences would follow from drinking the burning-hot tea? Would we not imagine as well that we would burn our mouths, by drinking the burning-hot tea in a single gulp?

Taking this into account, we claim that, while the static elaboration of an imaginary scenario can follow particular factual rules, one at a time, the dynamic elaboration of it is must be based on an action occurring, and must take into account all the consequences of such action; that is, it must take into account all the scripts detailing the consequences of a certain action α, as depicted in Figure 3. Note that, in the Figure, we have associated, to each script and each of its consequents, a numerical super-index in order to facilitate seeing what consequences each script bears with it.

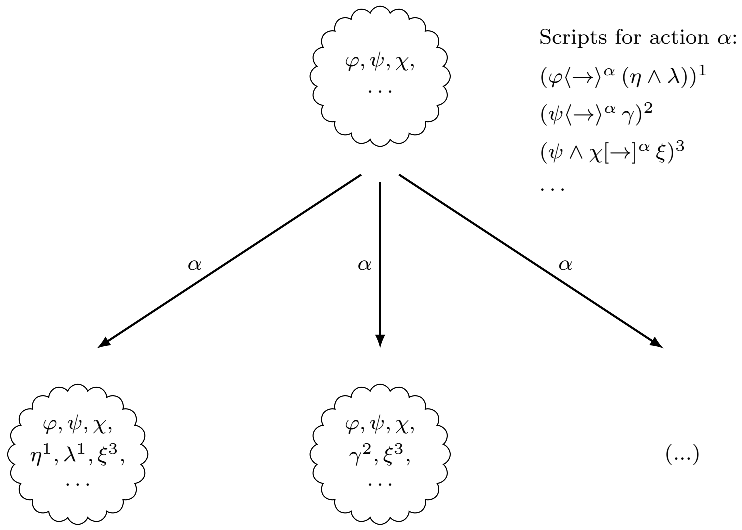

Fig. 3 The scripts determine all the possible consequences of an action.

Following the considerations pointed out in the previous paragraphs, the argument needed for this algorithm is not a script belonging to SCRIPT, but rather an action $\alpha \in$ ACT; however, and as we explain in brief, the algorithm then uses the scripts $\xi \in$ SCRIPT in order to develop the imaginary scenario.

It is worth noting that, during the development of the algorithm, the elements of the expanded model will be defined recursively over a series of loops. Due to this, there is a convention that, for the sake of simplicity, we introduce in our notation. In particular, and due to the fact that the algorithm involves looping over possibly many formulas, the expansions of the corresponding elements $W^{+}, R_{E}^{+}, V^{+}, \ldots$ are begin accumulated at each loop. Therefore, we assume that, whenever we refer to any element $W, R_{E}, V, \ldots$ we are referring to the most “updated” version of that element, in the sense of already including whatever has been added to it in the previous loop[7].

Thus, the `EvoAlg` needs an action $\alpha \in \mathrm{ACT}$, which the agent would imagine happening, and an imaginary world as the world of reference w^R, as follows:

$$\texttt{EvoAlg}(\alpha, w^R)$$

Now, the algorithm must follow these steps:

1. *Check initial conditions*: If world w^R is a real world, do nothing; if α does not appear in any formula within SCRIPT, do nothing (it would mean that the agent has no beliefs at all about the consequences of action α).
2. *Create queue of scripts*: Each formula $\xi \in$ SCRIPT is either of the form $\varphi\langle\rightarrow\rangle^\alpha\psi$ or of the form $\varphi[\rightarrow]^\alpha\psi$, for some $\alpha \in \mathrm{ACT}$. Create a queue S^α of formulas sorted as follows[8]:
   – Firstly, look for all the diamond-formulas in SCRIPT which are about α, and add them to the queue S^α while prioritizing the ones with the least complex antecedent (that is, the ones whose antecedent has less atomic formulas); for example, a formula $p\langle\rightarrow\rangle^\alpha$... has more priority than a formula $p \vee q\langle\rightarrow\rangle^\alpha$.... Formulas with the same antecedent complexity are sorted sequentially.
   – Secondly, look for all box-formulas in SCRIPT that are about α, and add them to the queue while prioritizing as well the ones with the least complex antecedent.
3. *Loop through the scripts*: This loops form the central part of this algorithm. The loop starts by evaluating the first formula $\xi_1 \in S^\alpha$, and keeps looping until it has evaluated every script in the queue[9].

3(a) *Evaluate diamond-formula*: If the current script $\xi \in S^\alpha$ being evaluated is of the form $\varphi\langle\rightarrow\rangle^\alpha\psi$, and if its antecedent φ holds at world w^R (that is, if $\mathcal{M}, w^R \vDash \varphi$), do:
   i. *Compute DNF*: Compute the DNF of the consequent, $\mathrm{DNF}(\psi) = \psi_1 \vee \ldots \vee \psi_n$.
   ii. *Create imaginary worlds*: Create n new imaginary worlds $w_1, \ldots, w_n$, one for each $\psi_1 \vee \ldots \vee \psi_n$. This defines the expanded set of possible

[7] We can draw a parallelism with the way variables are usually handled in programming languages. In there, it is typical to override the value of a variable by using its own value; for instance, one can increase the value of an integer index i by saying $i = i + 1$.

[8] In a nutshell, we process $\langle\rightarrow\rangle^\alpha$ formulas first to give priority to $[\rightarrow]^\alpha$ formulas. Last evaluated formulas could override something already added by previous formulas, and we claim that those scripts detailing *necessary* consequences of α should have priority over scripts detailing *possible* consequences of it.

[9] The idea behind this loop is that, conversely to what happened with the Description process (in which the agent elaborated the scenario step by step by picking a single factual rule each time), in this case the agent imagines performing an action. Therefore, the agent must check for *all* the consequences of such action, which are described (according to their preconditions) by the formulas in SCRIPT; as we have already argued by the beginning of the current section, the reason for doing so is that one cannot imagine that she performs an action, and then that only *some* of its consequences happen.

worlds as follows:

$$W^+ = W \cup \{w_1, \dots, w_n\}$$

iii. *Expand accessibility relation*: Create new R_E relations from w^R to each new imaginary world w_i, signed with action α:

$$R_E^+ = R_E \cup \Big(\bigcup_{i=1\dots n} \{(w^R, w_i, \alpha)\} \Big)$$

iv. *Check nominal structure*: Similarly to what happened in the other algorithms, the `EvoAlg` must now add a set of new nominals to refer to the newly created imaginary worlds. This defines both the *expanded set of nominals*, by adding one new nominal k_i for each new possible world w_i created during the current execution of the present algorithm, and the *expanded nominal function*, which is a functional extension of N relating the new pairs of nominals and possible worlds:

$$\mathrm{NOM}^+ = \mathrm{NOM} \cup \{k_1, \dots, k_n\}$$

$$N^+ = N \cup \Big(\bigcup_{i=1\dots n} \{(k_i, w_i)\} \Big)$$

v. *Expand valuation function - Clamp new atoms*: Set the valuation of each new imaginary world w_i according to the consequences of the corresponding clause ψ_i in DNF(ψ); note that, in this case, we do *not* yet import the atomic formulas of the world of reference, as we first need to keep evaluating the consequences of action α according to the other formulas in S^α: we do this because box-formulas may also affect the atomic valuation of the new imaginary worlds being created right now, and this must be prioritized over importing atomic formulas from w^R:

$$V^+(p) = V(p) \cup \Big(\bigcup_i \{w_i \mid p \text{ is a positive literal appearing in } \psi_i\} \Big)$$

3(b) *Evaluate box-formulas*: If the current script $\xi \in S^\alpha$ is of the form $\varphi[\to]^\alpha\psi$, and if its antecedent φ holds at the world of reference w^R (that is, if $\mathcal{M}, w^R \vDash \varphi$), do:

i. *Compute DNF*: Compute the DNF of the consequent, DNF(ψ) $= \psi_1 \vee \ldots \vee \psi_m$.

ii. *Loop over already existing worlds*: When evaluating box-formulas within the script queue, there are some considerations that are worth clarifying. Whereas diamond-formulas express that certain outcomes *could* follow from certain antecedent, box-formulas state that certain outcomes *must* follow. Therefore, box-formulas should also take into account those new imaginary worlds that have already been created when evaluating diamond-formulas, and apply their corresponding consequences to them as well.

Thus, if there exists at least one world w_i such that $(w^R, w_i, \alpha) \in R_E$ (that is, if at least one new imaginary world has been already created during the current execution of this algorithm), then each possible consequence ψ_j in DNF(ψ) must be handled while taking into account those already existing imaginary worlds. In order to handle this, the algorithm must loop over the already existing possible worlds w_i created during the current execution of the algorithm, and, for each w_i, do:

A. *Create new imaginary worlds*: For the already existing imaginary world w_i being considered, which we will call w_{i_m} throughout the current loop, the algorithm must create $m-1$ new imaginary possible worlds w_{i_j}, for $j = 1, \ldots, (m-1)$, and being m determined by DNF$(\psi) = \psi_1 \vee \ldots \vee \psi_m$; this defines the expanded set of possible worlds as follows[10]:

$$W^+ = W \cup \{w_{i_1}, \ldots, w_{i_{(m-1)}}\}$$

B. *Expand accessibility relation*: Create, for each new possible imaginary world, new R_E relations as follows:

$$R_E^+ = R_E \cup \Big(\bigcup_{j=1\ldots(m-1)} \{(w^R, w_{i_j}, \alpha)\} \Big)$$

C. *Expand nominal structure*: Create and associate new nominals to these new imaginary worlds. This defines both the expanded set of nominals, and the expanded nominal function as follows:

$$\mathrm{NOM}^+ = \mathrm{NOM} \cup \{k_{i_1}, \ldots, k_{i_{(m-1)}}\}$$

$$N^+ = N \cup \Big(\bigcup_{j=1\ldots(m-1)} \{(k_{i_j}, w_{i_j})\} \Big)$$

D. *Expand valuation function - Clamp new atoms*: Set the valuation of each new imaginary world w_{i_j} according to the consequences of the corresponding clause ψ_j in DNF(ψ). Note that, in the previous step, we have created $m-1$ new imaginary worlds; this results from the fact that, before evaluating the current box-formula, we already had at least one existing imaginary world created in a previous step of the current execution of

[10] In order to clarify the intuitions behind this step, suppose that the algorithm is evaluating a box-formula that represents m alternative outcomes: by following the way our other algorithms have been working, we should create m new possible imaginary worlds to account for each one of those outcomes. However, if there already exists an imaginary possible world (possibly as a result of evaluating a diamond-formula), then one of such outcomes should be represented in the world that already exists —for, otherwise, if we created m new imaginary worlds, aside from the already existing one, we would have $m+1$ new possible worlds to account for just m alternative outcomes represented in the current box-formula. Therefore, when evaluating a box-formula upon an already existing imaginary world, we only have to create $m-1$ new worlds.

this algorithm (probably while evaluating a diamond-formula). Now, as $\text{DNF}(\psi) = \psi_1 \vee \ldots \vee \psi_m$, and as we have used index $j = 1, \ldots, (m-1)$ for the new imaginary worlds created in this loop, we must associate the world w_{i_m}, which was the one that already existed when entering the current loop, with the clause ψ_m of $\text{DNF}(\psi)$, and associate the rest of the newly created imaginary worlds w_{i_j} with the rest of the clauses in $\text{DNF}(\psi_j)$, for $j = 1, \ldots, (m-1)$. The clamping of new atoms, in this case, must proceed in three different steps or phases.

Firstly, as the newly created worlds w_{i_j} are meant to be copies of the original diamond-world w_{i_m}, the algorithm needs to ensure that those worlds w_{i_j} satisfy the same atomic propositions as w_{i_m}: in other words, it should add every world w_{i_j} to the valuation function of any atom satisfied by w_{i_m} as follows:

$$V^+_{copy}(p) = V(p) \cup \Big(\bigcup_{j=1\ldots(m-1)} \{w_{i_j} \mid w_{i_m} \in V(p)\} \Big)$$

Secondly, the algorithm must proceed and clamp the positive atoms in each clause φ_j to their corresponding imaginary world w_{i_j}, as usual:

$$V^+_{add}(p) = V^+_{copy}(p) \cup \Big(\bigcup_{j=1\ldots m} \{w_{i_j} \mid p \text{ is a positive literal in } \psi_j\} \Big)$$

Thirdly, it could be the case that the new box-formula being evaluated forces certain atomic formulas to be false at a certain imaginary world. In order to account for that, we follow the "inverse" of the process we have been following when clamping new atoms: this time, we look for any atom appearing as a *negative* literal in the corresponding clause φ_j and, if the imaginary world w_{i_j} appears in the valuation function of such atom, we *remove* it (in order to force that atom to be false in that world):

$$V^+_1(p) = V^+_{add}(p) \setminus \Big(\bigcup_{j=1\ldots m} \{w_{i_j} \mid p \text{ is a negative literal in } \psi_j\} \Big)$$

iii. *Create witness world*[11]: Conversely, if there are no new imaginary worlds created during the current execution of the algorithm (because, for instance, there are no diamond-formulas in the current set of scripts), then the box-formula being evaluated must create a

[11] The relation between this step and step 3(b).ii could be understood as an "if ... else" statement, in terms of programming languages. Namely, the algorithm first checks whether there already exists any imaginary world and, if it does, goes through the corresponding "if" branch; otherwise (or "else"), if there are no already existing imaginary worlds, the algorithm must go through this current branch. Note, therefore, that both branches are *never* going to be executed for the same script, but rather just one of the two branches.

so-called “witness-world” to account for the consequences described by it[12].

Therefore, if there exists no world w_i such that $(w^R, w_i, \alpha) \in R_E$ (which would be either because there are no diamond-formulas referring to action α, or because w^R does not fulfill the antecedents of such formulas), then do the following:

A. *Create imaginary worlds*: Create m new imaginary possible worlds, one for each $\psi_1 \vee \ldots \vee \psi_m$. This defines the expanded set of possible worlds as follows:

$$W^+ = W \cup \{w_1, \ldots, w_m\}$$

B. *Expand accessibility relation*: Create new R_E accessibility relations from the world of reference w^R to these new imaginary worlds as follows:

$$R_E^+ = R_E \cup \Big(\bigcup_{j=1\ldots m} \{(w^R, w_i, \alpha)\} \Big)$$

C. *Expand nominal structure*: Create and associate new nominals to these new imaginary worlds. This defines both the expanded set of nominals, and the expanded nominal function as follows:

$$\mathrm{NOM}^+ = \mathrm{NOM} \cup \{k_1, \ldots, k_m\}$$

$$N^+ = N \cup \Big(\bigcup_{i=1\ldots m} \{(k_i, w_i)\} \Big)$$

D. *Expand valuation function - Clamp new atoms*: Set the valuation of each new imaginary world w_i according to the consequences of the corresponding clause ψ_i in DNF(ψ):

$$V_1^+(p) = V(p) \cup \Big(\bigcup_i \{w_i \mid p \text{ is a positive literal in } \psi_i\} \Big)$$

iv. *Expand valuation function - Import existing atoms*: After creating new worlds if necessary, the `EvoAlg` must import all the atoms that are true at the world of reference w^R, provided they do not appear in ψ_i, for each new imaginary possible world w_i. Note that, even in the case of imaginary worlds created by using diamond-formulas,

[12] It is worth mentioning that, in modal logic, the box operator $\Box$ has a sort of “vacuous” or “trivial” truth-condition: namely, if a world w has no accessibility relations at all, then every formula of the form $\Box\varphi$ would be vacuously true in there; as there are no worlds accessible from w, then *every* world accessible from w satisfies φ. That being said, we do not want our `EvoAlg` to conform to this fact, when evaluating a box-formula. One may argue that, if no diamond-formula about α has been previously evaluated, and so no new worlds have been created, then every box-formula about α could be true without the need of creating any world at all as a consequence of the agent imagining it. Nevertheless, this is not the way we want these formulas to behave, regarding the development of imaginary worlds, and so we still require our `EvoAlg` to create, at least, one witness world for a box-formula, in case there exist none yet.

this must be done after having evaluated the box-formulas, as they may require to clamp further atoms in those worlds; importing existing atoms *before* evaluating box-formulas would probably import atomic formulas that would need to be removed afterwards as a requirement of the box-formulas being processed. The last step of the expanded valuation function, therefore, is as follows:

$$V^+(p) = V_1^+(p) \cup \Big(\bigcup_i \{w_i \mid w^R \in V_1^+(p),\, p \text{ not a literal of } \psi_i\}\Big)$$

4. The `EvoAlg` has finished its execution: a new set of imaginary worlds have been created as a result of evaluating the consequences of a certain action or event α, according to the scripts believed by the agent, and which detail what would likely happen in the imaginary world, if α took place in there.

3.3.4 The `Add` *Algorithm*

The `AddAlg` is responsible for computing the voluntary addition of new premises into the imaginary scenario, and it is closely related to the `InitAlg`. Whereas in the Initialization the agent creates a new imaginary scenario by taking reality as the reference, in this case the agent decides to clamp a new premise into an already existing imaginary scenario. In this case, we go back to the agent choosing to clamp an arbitrary formula into the imagining, as we do not need to restrict ourselves to any limited set of formulas that could be used, as we did in the previous two algorithms. Figure 4 depicts how this process works.

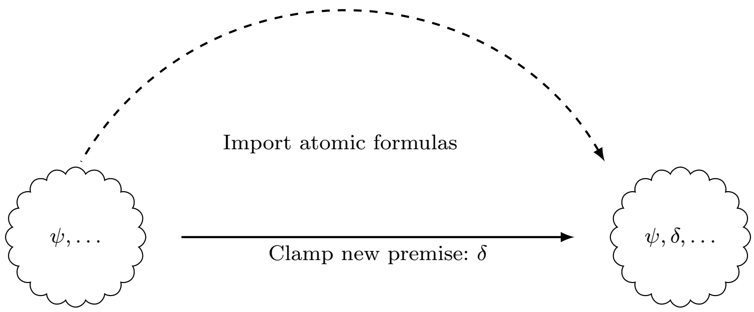

Fig. 4 Adding new premises into the imagining.

Note that, although this algorithm is almost identical to the `InitAlg`, there is still an important difference that justifies that we have defined two distinct algorithms: whereas the `InitAlg` creates new imaginary worlds by using a *real world* as its referent, the `AddAlg` creates new imaginary worlds, but by adding a new premise into an *already existing* imaginary world. Consequently, we can say that the range of possible worlds available to both algorithms is complementary, in the sense that the `InitAlg` can only use real possible worlds, and the `AddAlg` can only use imaginary possible worlds.

A call to the `AddAlg` needs a certain formula $\delta \in \text{FORM}^*$ and an imaginary possible world as the world of reference w^R, as follows:

$$\texttt{AddAlg}(\delta, w^R)$$

The algorithm follows these steps:

1. *Check initial conditions*: If world w^R is a real possible world, do nothing. Similarly, if δ is contradictory (that is, if $\delta \equiv \bot$), do nothing.
2. *Compute DNF*: In order to handle the formula in an efficient way, we compute the *Disjunctive Normal Form* (DNF) of δ, to which we refer as $\text{DNF}(\delta) = \delta_1 \vee \ldots \vee \delta_n$.
3. *Create imaginary worlds*: Create a new imaginary possible world $w_1, \ldots, w_n$ for each clause $\delta_1, \ldots, \delta_n$ in $\text{DNF}(\delta)$. This defines the *expanded set of possible worlds* as follows:
$$W^+ = W \cup \{w_1, \ldots, w_n\}$$
4. *Expand accessibility relation*: Once the new possible worlds have been created, the `AddAlg` must create the new addition relations expressing that, by imagining formula δ at the world of reference w^R, the agent has crated new imaginary worlds $w_1, \ldots, w_n$. This defines the *expanded set of addition relations* as follows:
$$R_A^+ = R_A \cup \Big(\bigcup_{i=1\ldots n} \{(w^R, w_i, \delta)\} \Big)$$
5. *Expand nominal structure*: Now, the `AddAlg` must add a set of new nominals to refer to the newly created imaginary worlds. This defines both the *expanded set of nominals*, by adding one new nominal k_i for each new possible world w_i, and the *expanded nominal function*, which is an extension of N:
$$\text{NOM}^+ = \text{NOM} \cup \{k_1, \ldots, k_n\}$$
$$N^+ = N \cup \Big(\bigcup_{i=1\ldots n} \{(k_i, w_i)\} \Big)$$
6. *Expand valuation function*: The `AddAlg` must expand the valuation function to account for the new imaginary possible worlds. In order to do so, the algorithm must account for both the atoms that are present in each δ_i, and also for the atoms that are true in the world of reference w^R and which should be imported to the new imaginary worlds, provided they do not appear in δ_i. Therefore, the definition of the *expanded valuation function* involves two different phases:
   (a) *Clamp new atoms*: The `AddAlg` must set the new valuation functions according to the atoms p appearing in δ_i, for each new imaginary possible world w_i:
$$V_1^+(p) = V(p) \cup \Big(\bigcup_i \{w_i \mid p \text{ is a positive literal appearing in } \delta_i\} \Big)$$

(b) *Import existing atoms*: Then, it must import all the atoms that are true at world w^R, provided they do not appear in δ_i, for each new imaginary possible world w_i:

$$V^+(p) = V_1^+(p) \cup \Big(\bigcup_i \{w_i \mid w^R \in V_1^+(p) \text{ and } p \text{ is not a literal of } \delta_i\}\Big)$$

7. The `AddAlg` has finished its execution: a new set of imaginary possible worlds satisfying δ has been created, and these worlds are now accessible through the addition relation R_A from the world of reference w^R.

3.4 Semantics

Having presented the four algorithms of the Logic of Imagination Acts, we define its semantics as follows, for a Model for Imagination Acts $\mathcal{M}$, a possible world $w \in W$, and being $p \in$ ATOM, $i \in$ NOM, $\{\varphi, \psi\} \subseteq$ FORM, $\delta \in$ FORM*, $\zeta \in$ FACT, and $\alpha \in$ ACT:

Propositional formulas:

$\mathcal{M}, w \vDash p$	iff	$w \in V(p)$
$\mathcal{M}, w \vDash \neg\varphi$	iff	$\mathcal{M}, w \nvDash \varphi$
$\mathcal{M}, w \vDash \varphi \wedge \psi$	iff	$\mathcal{M}, w \vDash \varphi$ and $\mathcal{M}, w \vDash \psi$
$\mathcal{M}, w \vDash \varphi \vee \psi$	iff	$\mathcal{M}, w \vDash \varphi$ or $\mathcal{M}, w \vDash \psi$
$\mathcal{M}, w \vDash \varphi \to \psi$	iff	$\mathcal{M}, w \vDash \neg\varphi$ or $\mathcal{M}, w \vDash \psi$

Hybrid and modal formulas:

$\mathcal{M}, w \vDash i$	iff	$N(i) = w$ and, for every $v \in W$, if $\mathcal{M}, v \vDash i$, then $v = w$
$\mathcal{M}, w \vDash @_i\varphi$	iff	there exists a world $v \in W$ such that $N(i) = v$ and $\mathcal{M}, v \vDash \varphi$
$\mathcal{M}, w \vDash \langle\text{Img}\rangle\varphi$	iff	there exists a world $v \in W$ such that either $(w, v, \delta) \in R_I$, $(w, v, \zeta) \in R_D$, $(w, v, \alpha) \in R_E$ or $(w, v, \delta) \in R_A$ and $\mathcal{M}, v \vDash \varphi$

Dynamic imagination formulas:

$\mathcal{M}, w \vDash \text{Init}(\delta)$	iff	δ is not contradictory ($\delta \not\equiv \bot$) and either there already exists $v \in W$ s.t. $(w, v, \delta) \in R_I$ or, after executing `InitAlg`(δ, w), $\mathcal{M}$ is expanded into $\mathcal{M}^+$
$\mathcal{M}, w \vDash \text{Descr}(\zeta)$	iff	either there already exists $v \in W$ such that $(w, v, \zeta) \in R_D$ or, after executing `DescrAlg`(ζ, w), $\mathcal{M}$ is expanded into $\mathcal{M}^+$
$\mathcal{M}, w \vDash \text{Evo}(\alpha)$	iff	either there already exists $v \in W$ such that $(w, v, \alpha) \in R_E$ or, after executing `EvoAlg`(α, w), $\mathcal{M}$ is expanded into $\mathcal{M}^+$
$\mathcal{M}, w \vDash \text{Add}(\delta)$	iff	δ is not contradictory ($\delta \not\equiv \bot$) and either there already exists $v \in W$ s.t. $(w, v, \delta) \in R_A$ or, after

executing $\texttt{AddAlg}(\delta, w)$, $\mathcal{M}$ is expanded into $\mathcal{M}^+$

Static imagination formulas:

$\mathcal{M}, w \vDash \langle\delta\rangle^I\varphi$	iff	there is some $v \in W$ s.t. $(w, v, \delta) \in R_I$ and it is the case that $\mathcal{M}, v \vDash \varphi$ and $\mathcal{M}, v \vDash \delta$
$\mathcal{M}, w \vDash \langle\zeta\rangle^D\varphi$	iff	there is some $v \in W$ s.t. $(w, v, \zeta) \in R_D$ and it is the case that $\mathcal{M}, v \vDash \varphi$
$\mathcal{M}, w \vDash \langle\alpha\rangle^E\varphi$	iff	there is some $v \in W$ s.t. $(w, v, \alpha) \in R_E$ and it is the case that $\mathcal{M}, v \vDash \varphi$
$\mathcal{M}, w \vDash \langle\delta\rangle^A\varphi$	iff	there is some $v \in W$ s.t. $(w, v, \delta) \in R_A$ and it is the case that $\mathcal{M}, v \vDash \varphi$ and $\mathcal{M}, v \vDash \delta$

Recall that the expanded model $\mathcal{M}^+$ is computed by any execution of either the `InitAlg`, the `DescrAlg`, the `EvoAlg` or the `AddAlg`; elements W^+, R_I^+, R_D^+, R_E and R_A^+ belong to the expanded model $\mathcal{M}^+$, and are also computed by the execution of the previous algorithms.

4 An Example

In this section we provide a brief example of a model in which all four algorithms have been sequentially executed. Although we do not provide a step-by-step elaboration of each execution due to reasons of space, we briefly highlight what has happened in each execution. The whole model can be seen in Figure 5, where we highlighted in bold font the atomic propositions that had been clamped in each world by each algorithm execution. Note that, although we do not explicitly specify so in this example, the formulas used by the `DescrAlg` and the `EvoAlg` must belong to the sets FACT and SCRIPT in this particular example.

The initial model was such that only included world w from Figure 5, and in which the following algorithms were executed in the following order:

1. $\texttt{InitAlg}(\neg p \rightarrow (r \wedge s), \text{w})$: When computing the DNF of premise $\delta = \neg p \Rightarrow (r \wedge s)$, we get $p \vee (r \wedge s)$, which means that the algorithm must create two new imaginary worlds, v_1 and v_2, satisfying p and $r \wedge s$, respectively. As we can see in the example, once these atomic propositions have been clamped, only those atomic propositions that hold at w and do *not* contradict the already clamped ones are imported in the new imaginary worlds.
2. $\texttt{DescrAlg}((r \wedge s)\langle\rightarrow\rangle(\neg q \vee t \vee (q \wedge \neg s)), v_2)$: In the next step, the agent (or we as modelers accounting for the agent's choices) decides to elaborate on the static details of imaginary world v_2 by using a certain factual rule ζ. First, the `DescrAlg` must check whether the antecedent of ζ, which is $(r \wedge s)$, holds at v_2, and it does indeed; now, it must compute the DNF of the antecedent of ζ, which already is $\neg q \vee t \vee (q \wedge \neg s)$, create three new imaginary worlds, u_1, u_2 and u_3, and clamp the corresponding clause into them. After doing so, it must import those atomic propositions holding

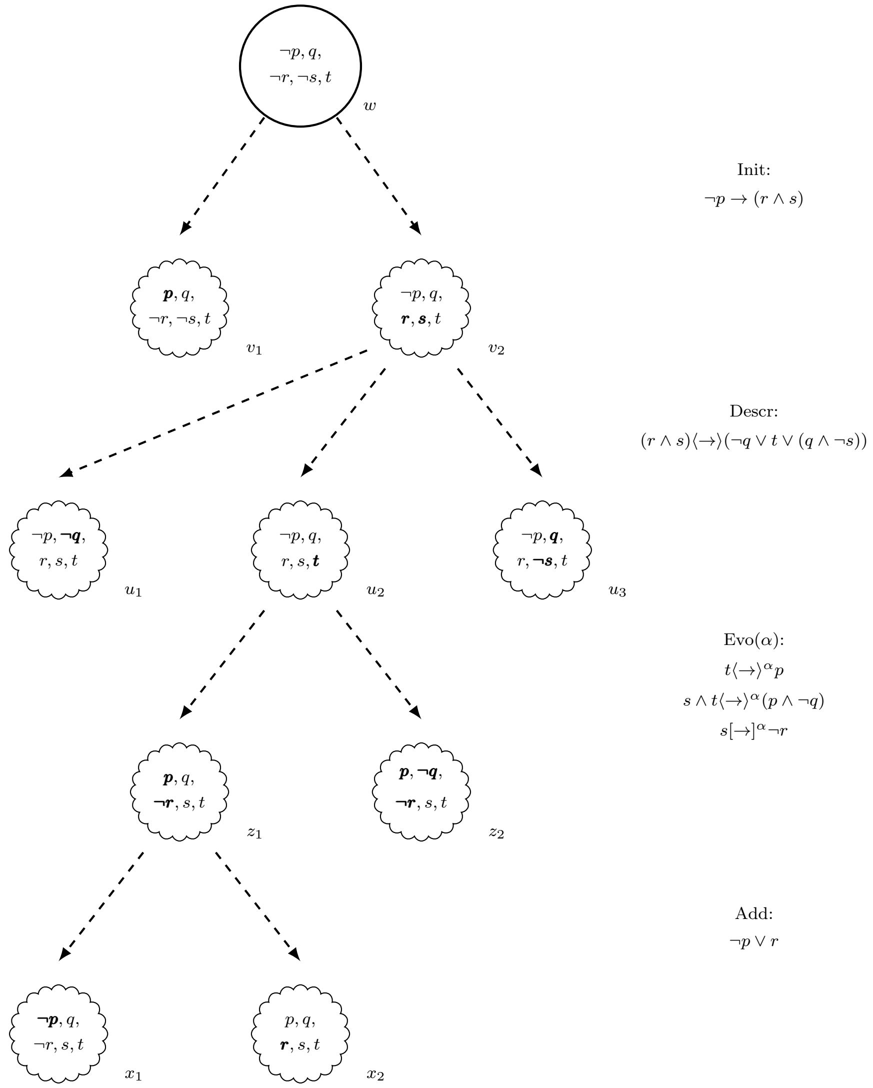

Fig. 5 The model for the example, with the execution of each algorithm.

at v_2 that do not contradict with the ones already clamped in each new imaginary world.

3. `EvoAlg`(α, u_2): In this case, recall that the `EvoAlg` takes an action α as the argument and then it loops over every script whose main operator is either $\langle\to\rangle^{\alpha}$ or $[\to]^{\alpha}$. The relevant scripts can be seen in Figure 5. The first script processed in this step is $t\langle\to\rangle^{\alpha}p$; the `EvoAlg` checks whether the antecedent t holds at the world of reference u_2 and, as it does, creates a new imaginary world z_1 where it clamps p. Note that the `EvoAlg` does not

yet import any atomic propositions from the world of reference, as it must loop over all the available scripts first. Therefore, the algorithm now moves on to process script $s \wedge t\langle\rightarrow\rangle^{\alpha}(p \wedge \neg q)$; as the antecedent $s \wedge t$ holds at u_2, the algorithm creates a new world z_2 and clamps the consequent $p \wedge \neg q$ into it. Now, the algorithm must process a box-formula $s[\rightarrow]^{\alpha}\neg r$; as the antecedent s holds at the world of reference u_2, the algorithm must loop over every already existing imaginary world created in this execution of `EvoAlg` and clamp the consequent $\neg r$ into them. At this point, as there are no more scripts about α, the `EvoAlg` can import those atomic propositions holding at u_2 that do not contradict with what have been clamped in the new imaginary worlds z_1 and z_2.

4. `AddAlg`$(\neg p \vee r, z_1)$: Similarly to what happened in step 1, the algorithm computes the DNF of δ, which is already $\neg p \vee r$, and thus creates two new worlds, x_1 and x_2, where it clamps the corresponding formulas $\neg p$ and r, respectively. Once this has been done, the `AddAlg` imports those atomic propositions from z_1 that do not contradict with the already clamped ones.

The previous paragraphs highlight how each algorithm has been executed in the model of the example, and thus account for how the new imaginary worlds have been created. Recall that, although we have just executed each algorithm once, and we did so in an ordered fashion, our system allows the modeler to execute any algorithm at any time (except the `InitAlg`, which is required to be executed before any other algorithm in order to create the first layer of imaginary worlds).

The model shows how the content of the agent's imaginings changes through the execution of different processes. As a result of the first execution of the `Init` process, the agent creates a certain set of imaginary worlds that account for the initial premise, but which, aside from that, resemble the real world believed by the agent in every other way through importing those believed atoms that do not contradict with the new clamped ones. However, the execution of further mechanisms of imagination, represented by each step in the model expansion, provides additional specifications to the imagining, be them in the form of factual rules, scripts or additional premises, that, in turn, modify the state of affairs of the imaginary worlds to accommodate such constrains.

This is, in some way, similar to the way an agent updates the set of possible worlds in epistemic and doxastic logic as a result of receiving new information. The main difference with respect to those systems is that, whereas gaining knowledge is often accounted by reducing the number of epistemically plausible worlds, and updating the agent's doxastic preferences is usually represented by modifying the accessibility relations, the update of information in imaginary worlds is represented by expanding and creating new possible worlds. This aims to account for the fact that, when elaborating on an imagining, the agent is not necessarily concerned with the already existing imaginings she may be entertaining, but rather with "opening" new branches within the imagining. Furthermore, this approach allows us, as modelers, to trace each and every one of the steps performed by the agent within a whole imagination episode;

it allows us to see all the alternative imaginary worlds that have been created in previous steps, as well as which particular worlds the agent has chosen to focus on at each step.

5 Conclusions and Future Work

Probably the most interesting feature of the Logic of Imagination Acts is its dynamic modularity, which amounts to dynamic versatility. An act of imagination is seen as a sequence of possibly many executions of different processes, each of them being part of the act of imagination as a whole, but each one of them distinct enough to account for a particular way in which the imagining can be elaborated. In this setting, we can see within a particular act of imagination and we can identify and split it into the different processes that take part in it. The level of detail this approach gives us when modeling the formation and elaboration of imaginary worlds allows us to study the dynamics of imagination in depth.

Being able to develop imagination acts in a modular, non-brute-forced way, allows us to represent acts of imagination that can create and develop imaginary worlds in a wide variety of different ways. We do not define an exhaustive algorithm detailing each and every possible alternative in an act of imagination, as if it was a game of chess being analyzed by a computer in order to check each and every possible move. Instead, we now aim to define a system that captures the way human beings elaborate on their imagining by partially exploring one or another possibility, and leaving many options aside.

There are, nevertheless, certain shortcomings in our approach. Regarding the richness of the logical language, we believe that having an explicit ways of considering beliefs would be a very interesting addition. Even though we have already argued, by the end of Section 3.2, how we implicitly interpret beliefs in this logic, the lack of an explicit operator accounting for the "believe" attitude limits the interaction our logic can have with this other mental attitude. Our proposal provides great insights regarding how imaginary worlds are created and develop, but if we expanded our logic in order to include explicit representation for beliefs, we could then explore how those imaginary worlds would potentially affect the agent's beliefs. For instance, if we had a way of explicitly representing the agent's beliefs in the real worlds, we could study how one further algorithm, corresponding to the "realization" of an agent who, through performing a thought experiment, learns something new, could then update her beliefs about the real world. In such case, in would also be interesting to compare our approach to other systems able to deal with belief revision or internal reasoning steps, such as some applications of Dynamic Epistemic Logic (see [17] for an overview) and other works devoted to represents the dynamics of awareness and realization (like in [23]).

We can already forsee in advance, however, how explicitly accounting for beliefs would pose some challenging questions. If the doxastic relation has some kind of ordering, such as in the case of the plausibility models defined, for instance, in [1], the "believed worlds" that would have to be taken into account for the algorithms when importing beliefs would always need to be ones that are actually believed by the agent –in other words, the most plausible ones, or "top-worlds". In terms of the dynamics of imagination, therefore, this would render all the "non-top worlds" effectively useless for our purposes. An interesting line of research, though, would be how to allow the agent to imagine that her beliefs are actually different, which would require duplicating the whole doxastic model in an imaginary setting, but modifying the doxastic relations appropriately; this would be a first step towards allowing the agent to imagine, first, that she beliefs something different than she actually does, and then to allow her to create new imaginary worlds based on her new "imaginary beliefs".

Regarding other possible lines of future work, it would be interesting to study how our logic could be adapted into a first-order setting. Even though we have chosen to stick to a propositional setting in order to focus on the dynamics of the imagination mechanisms, expanding our work to account for individuals would greatly increase its expressive power. With this, it would be possible to represent how an agent imagines something being something else, for instance in pretense play –say, a banana being a telephone. Nevertheless, and as our imaginary worlds are created dynamically via the execution of the algorithms, this would surely pose some challenges with regards to the issues of constant and varying domains present in first-order modal systems, as identified in [11].

Furthermore, we would like to consider how we could add, to the current system, the mechanisms needed to allow imagination processes to affect not only the states of affairs represented by imaginary worlds, but also the sets of rules and scripts holding in there. Currently, imaginary possible worlds can be indeed different to the real possible worlds, but they are different in terms of what is the case in there; the factual rules and the scripts believed by the agent, nevertheless, are constant throughout the model. However, imagination should be able to alter that as well. It is true that I can imagine that *things* are different, but I can also imagine that the *rules* governing the world are also different. In order to account for that, we would need to add still one further layer to our formal models and associate, to each possible world, a particular set of factual rules and scripts believed in there. This way, the real possible worlds would account for the factual rules and scripts that the agent actually believes about the real world, but we could also create an imaginary world in which such rules and scripts were different.

Due to the nature of how imaginary worlds are gradually developed, it would also be an interesting contribution to see how our algorithms would accommodate a *paracomplete* setting[13]. In a nutshell, a paracomplete setting would allow our possible worlds to have certain truth-value gaps, with respect to certain atomic formulas; this, in turn, would allow us to determine, for each

[13] Some logical systems defined by Berto in [2] and [4] allow for both paracomplete and paraconsistent worlds.

possible world and atomic formula, whether the formula is true in there, false in there, or simply undetermined or unspecified. When using this setting, we could avoid having to import *every* possible atomic formula from the world of reference, and thus we could just allow the new imaginary worlds to be developed in a truly step-by-step way, filling up only those details that are specified by the corresponding premise, factual rule or script. We think that it would be interesting to see how our layer of imagination algorithms could be adapted into such setting, and thus it is an interesting topic for future work. As a way of complementing this paracomplete setting, we could also consider adding *aboutness*[14] into the system to import only those atomic formulas that were related somehow to the details being clamped into the new imaginary worlds.

References

1. A. Baltag and S. Smets. A qualitative theory of dynamic interactive belief revision. *Logic and the foundations of game and decision theory (LOFT 7)*, 3:9–58, 2008.
2. F. Berto. On conceiving the inconsistent. In *Proceedings of the Aristotelian Society*, volume 114, pages 103–121. Wiley Online Library, 2014.
3. F. Berto. Aboutness in imagination. *Philosophical Studies*, 175(8):1871–1886, 2017.
4. F. Berto. Impossible worlds and the logic of imagination. *Erkenntnis*, 82(6):1277–1297, 2017.
5. P. Blackburn. Representation, reasoning, and relational structures: a hybrid logic manifesto. *Logic Journal of the IGPL*, 8(3):339–365, 2000.
6. P. Blackburn and J. Seligman. Hybrid languages. *Journal of Logic, Language and Information*, 4(3):251–272, 1995.
7. J. Casas-Roma, A. Huertas, and M. E. Rodríguez. An analysis of imagination acts. In *Research Workshop on Hybrid Intensional Logic*, 2017. 10th - 11th November, University of Salamanca (Oral communication).
8. J. Casas-Roma, A. Huertas, and M. E. Rodríguez. Towards a shared frame for imaginative episodes. In *Fourth Philosophy of Language and Mind Conference*, 2017. 21st - 23rd September, Ruhr University Bochum (Oral communication).
9. A. Costa-Leite. Logical properties of imagination. *Abstracta*, 6(1):103–116, 2010.
10. G. Currie and I. Ravenscroft. *Recreative Minds: Imagination in Philosophy and Psychology.* Oxford University Press, 2002.
11. M. Fitting and R. L. Mendelsohn. *First-order modal logic*, volume 277. Springer Science & Business Media, 2012.
12. D. Harel. Dynamic logic. In D. M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume II: Extensions of Classical Logic, chapter 10, pages 497–604. D. Reidel Publishing Company, 1984.
13. A. Kind, editor. *The Routledge Handbook of Philosophy of Imagination.* Routledge, Taylor & Francis Group, 2016.
14. A. Kind and P. Kung, editors. *Knowledge Through Imagination.* Oxford University Press, 2016.
15. P. Langland-Hassan. On choosing what to imagine. In A. Kind and P. Kung, editors, *Knowledge Through Imagination*, chapter 2, pages 61–84. Oxford University Press, 2016.
16. D. Lewis. *Counterfactuals.* Blackwell Publishing, 1973.
17. L. S. Moss. Dynamic epistemic logic. In H. Van Ditmarsch, J. Y. Halpern, W. van der Hoek, and B. Kooi, editors, *Handbook of Epistemic Logic*, chapter 6, pages 261–312. College Publications, 2015.
18. B. Nanay. The role of imagination in decisionmaking. *Mind & Language*, 31(1):127–143, 2016.

[14] Also considered by Berto in [3].

19. S. Nichols, editor. *The architecture of the imagination: New essays on pretence, possibility, and fiction*. Clarendon Press / Oxford University Press, 2006.
20. S. Nichols and S. P. Stich. A cognitive theory of pretense. *Cognition*, 74(2):115–147, 2000.
21. I. Niiniluoto. Imagination and fiction. *Journal of Semantics*, 4(3):209–222, 1985.
22. T. Schroeder and C. Matheson. Imagination and emotion. In S. Nichols, editor, *The Architecture of the Imagination: New Essays on Pretence, Possibility and Fiction*, chapter 2, pages 19–40. Clarendon Press, Oxford, 2006.
23. F. R. Velázquez-Quesada. *Small steps in dynamics of information*. Institute for Logic, Language and Computation, 2011.
24. K. L. Walton. *Mimesis as make-believe: On the foundations of the representational arts*. Harvard University Press, 1990.
25. H. Wansing. Remarks on the logic of imagination. a step towards understanding doxastic control through imagination. *Synthese*, 194(8):2843–2861, 2017.
26. T. Williamson. Knowing by imagining. In A. Kind and P. Kung, editors, *Knowledge Through Imagination*, chapter 4, pages 113–123. Oxford University Press, 2016.