Quantificational Modal Operators and Their Semantics*

Stefano Borgo

Laboratory for Applied Ontology
Institute for Cognitive Sciences and Technology
Via alla Cascata 56/C
38100 Povo (Trento), Italy

borgo@loa-cnr.it

Abstract

We study a system of modal logic for representing and reasoning in multi-agent systems. Building on dynamic action logic and Henkin quantifiers, we introduce a class of operators that present important features for capturing concurrency, independence, collaboration, and co-ordination between agents. The main goal of the paper is to introduce the formal semantics of these operators and to show how they can model different types of agents. This yields a way to directly compare a variety of phenomena in multi-agent systems. Some examples are given.

1 Introduction

For about 20 years we have witnessed an increasing interest in the formal study of phenomena comprising several entities which present independent and autonomous behavior like software agents, human beings, biological entities, social organizations, robots and stock markets [We$_1$99].

In this research area, the issue of (true) concurrency has special interest since it puts at the center phenomena where several entities act simultaneously perhaps affecting each other. This issue is coupled with the need to formalize group collaboration as well as group dynamics. Another challenge is the formalization of the independence of an agent from the others and from the groups of which it is a member. Modelling the choices an agent makes or can make depending on its "internal" status, its beliefs about the external world, its knowledge about (and relationship with) other entities/agents is at the center of many representation problems.

The usual logical machinery often requires the coexistence of logical operators (dynamic, temporal, epistemic, and deontic) in one and the same

* The author has been supported by the projects TOCAI.IT (MIUR), MOSTRO and LIFOA (Provincia Autonoma di Trento).

Johan van Benthem, Dov Gabbay, Benedikt Löwe (eds.). *Interactive Logic. Proceedings of the 7th Augustus de Morgan Workshop, London.* Texts in Logic and Games 1, Amsterdam University Press 2007, pp. 49–69.

language [vHVe$_2$02, vHWo$_3$03]. This strategy is not satisfactory because of the complexity of the logical systems thus obtained [Be$_0$+02, vHWo$_3$03]. Furthermore, these logics are hard to compare to the point that the uniformity of the very phenomena at stake is lost in the different formalizations. An example is given by logics like **ATL** [Al$_2$He$_2$Ku$_1$02], **CL**, **ECL** [Pa$_7$02], **ATEL** [vHWo$_3$02], and **STIT** logic [Ho$_6$01, Br$_2$He$_3$Tr05] which deal with roughly the same type of scenario [Go$_2$Ja$_2$04, Wö04].

Our work focuses on the formalization of multi-agent systems with particular emphasis on concurrency, independence, collaboration, and coordination issues. Our ideal scenario comprises a fixed set of agents that individually or in group, isolately or co-ordinated, take actions simultaneously and in this way determine the transitions between states of the system. The main goal of the paper is to show that the language we have developed has several natural interpretations which allow us to capture different types of agents while maintaining the very same syntax. The novelty is given by a new type of operators that combines modality with quantification. For this reason, these operators are called *quantificational modal operators*.

In this approach, we take a general perspective and do not limit our work to a specific notion of agent (and so we are not going to give one). Note that, for presentation purposes, we often describe the agents as having some degree of rationality. This is not necessary but it helps in conveying the meaning of the operators. Also, in this paper we do not discuss proof-theoretical properties of the resulting logics. These interpretations correspond to quite different axiomatic systems and here we lack the space for their analysis. The interested reader can find in [Bo$_1$05b] an axiomatic presentation of one of these systems. Finally, note that in this study we shall always stick to two-valued semantics.

Structure of the paper. Section 2 first introduces the propositional fragment of the logic by defining the constant modal operators and, secondly, extends them with free variables. In Section 3, we present the full language by introducing the quantificational modal operators and then study alternative interpretations for one-column operators. In Section 4 we briefly discuss the extension of these semantics to multi-column operators. The following section looks at a couple of examples. Finally, Section 6 relates our work to other logical approaches and adds some final remark.

2 Basic modalities for MAS

The modalities we want to study can be seen as an extension of *Dynamic Action Logic*, that is, the application of Dynamic Logic (**DL**) [Ha$_1$Ko$_6$Ti$_1$00] to model actions. Similarly to **DL**, our basic operators, called *constant modal operators*, are modalities indexed by constant identifiers (denoting actions). However, these operators differ from those of **DL** in two aspects:

syntactically they require several constant identifiers to individuate even the simplest modalities, and semantically they are not associated to a unique interpretation.

In a system of two agents, say $\mathcal{A}_1$ and $\mathcal{A}_2$, our modal operators have the shape of a $2 \times n$ matrix ($n > 0$) where the first row lists the (constants denoting the) actions performed by agent $\mathcal{A}_1$, in the order of their execution, and the second row lists the actions performed by agent $\mathcal{A}_2$. For instance the expression $\left[\begin{smallmatrix} c_1 & c_3 \\ c_2 & c_4 \end{smallmatrix}\right]$ is a modality corresponding to the state transition identified by the concurrent execution of action c_1 (by agent $\mathcal{A}_1$) and c_2 (by agent $\mathcal{A}_2$) followed by the concurrent execution of action c_3 (by agent $\mathcal{A}_1$) and c_4 (by agent $\mathcal{A}_2$). Each entry of the matrix denotes an action and the combination of these actions characterises the meaning of the modal operator by identifying, in the usual Kripke style semantics, the accessibility relation associated with that operator.

More generally, an operator in the shape of a $k \times n$ matrix is a modality for a system with k agents. It is always assumed that the number of rows in the operators matches the number of agents in the system (as a consequence all the operators in a language have the same number of rows). Also, each agent is associated to the same row in all operators.

We now state this formally:

Let PropId be a non-empty countable set, the set of *proposition identifiers*. Let ActId (disjoint from PropId) be a non-empty countable set whose elements are called *action identifiers*. These are the individual constants of the language. Following standard modal logic, complex formulas are generated inductively from proposition identifiers through the connectives of implication ($\rightarrow$) and negation ($\neg$), and the modal operators described below. As usual, we shall make use of the standard conventions for $\wedge, \vee, \leftrightarrow$.

Fix an integer $k \geq 1$ which, informally, is the number of agents in the system. A *constant modality marker*[1] for k is a $k \times n$-matrix ($n \geq 1$)

$$M = \begin{matrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{k1} & a_{k2} & \cdots & a_{kn} \end{matrix}$$

where $a_{ij} \in \mathrm{ActId}$ (a_{ij}, a_{mn} not necessarily distinct).

A *constant modal operator* for k is an expression $[M]$ where M is a constant modality marker for k.

The set of k-formulas (formulas for short) is the smallest set F_k satisfying:

I) PropId $\subseteq F_k$ (the elements of PropId are called atomic formulas),

II) If φ and ψ are in F_k, then so are $\neg\varphi$ and $\varphi \rightarrow \psi$,

[1] Elsewhere we have been calling these *constant modality identifiers.*

III) If $[M]$ is a constant modal operator for k and φ is in F_k, then $[M]\varphi$ is in F_k also.

The semantics is as follows:

Fix a set Act of actions, we call *k-action* an expression in the shape of a $k \times n$ matrix ($n \geq 1$) over Act. A *k-agent Kripke frame* is a triple $\mathcal{K} = \langle W, \mathrm{Act}; R\rangle$ where W is a non-empty set *(the set of states)*, Act is a non-empty set *(the set of actions)*, and R is a function mapping k-actions (over Act) of size $k \times 1$ to binary relations on W, $R\begin{pmatrix}\alpha_1\\ \vdots\\ \alpha_k\end{pmatrix} \subseteq W \times W$.

A *k-agent Kripke structure* is a tuple $\mathcal{M} = \langle W, \mathrm{Act}; R, [\![\cdot]\!]\rangle$ where $\langle W, \mathrm{Act}; R\rangle$ is a k-agent Kripke frame and $[\![\cdot]\!]$ is a function *(the valuation function)* such that $[\![p]\!] \subseteq W$ for $p \in \mathrm{PropId}$ and $[\![a]\!] \in \mathrm{Act}$ for $a \in \mathrm{ActId}$.

Let us write $\mathcal{A}_1$ for the first agent, $\ldots$, $\mathcal{A}_k$ for the kth agent. If agent $\mathcal{A}_1$ performs the action denoted by a_1, agent $\mathcal{A}_2$ the action denoted by $a_2, \ldots,$ agent $\mathcal{A}_k$ the action denoted by a_k, we write $\begin{bmatrix}a_1\\ a_2\\ \vdots\\ a_k\end{bmatrix}$ for the modal operator describing the evolution of the system which is determined by the concurrent execution of actions $[\![a_1]\!], \ldots, [\![a_k]\!]$ by agents $\mathcal{A}_1, \ldots, \mathcal{A}_k$, respectively. That is, the interpretation of $\begin{bmatrix}a_1\\ a_2\\ \vdots\\ a_k\end{bmatrix}$ is k-action $\begin{matrix}[\![a_1]\!]\\ [\![a_2]\!]\\ \vdots\\ [\![a_k]\!]\end{matrix}$.

Function $[\![\cdot]\!]$ is extended inductively to multi-column operators in the language as follows: if $[A]$ is a multi-column operator obtained by juxtaposition of constant modality markers B and C (i.e. $[A] = [BC]$), then we put $R([\![A]\!]) = R([\![B]\!]) \circ R([\![C]\!])$. More formally,

$$\left[\!\!\left[\begin{matrix}a_{11}\\ a_{21}\\ \vdots\\ a_{k1}\end{matrix}\right]\!\!\right] =_{\mathrm{def}} \begin{matrix}[\![a_1]\!]\\ [\![a_2]\!]\\ \vdots\\ [\![a_k]\!]\end{matrix} \quad ; \quad \left[\!\!\left[\begin{matrix}a_{11} & a_{12} & \cdots & a_{1n}\\ a_{21} & a_{22} & \cdots & a_{2n}\\ \vdots & \vdots & & \vdots\\ a_{k1} & a_{k2} & \cdots & a_{kn}\end{matrix}\right]\!\!\right] =_{\mathrm{def}} \left[\!\!\left[\begin{matrix}a_{11}\\ a_{21}\\ \vdots\\ a_{k1}\end{matrix}\right]\!\!\right] \left[\!\!\left[\begin{matrix}a_{12}\\ a_{22}\\ \vdots\\ a_{k2}\end{matrix}\right]\!\!\right] \ldots \left[\!\!\left[\begin{matrix}a_{1n}\\ a_{2n}\\ \vdots\\ a_{kn}\end{matrix}\right]\!\!\right]$$

Note that we write $[\![M]\!]$ instead of $[\![[M]]\!]$.

The truth value of a formula is defined inductively:

1. Let $p \in \mathrm{PropId}$, then $\mathcal{M}, s \models p$ if $s \in [\![p]\!]$
2. $\mathcal{M}, s \models \neg\varphi$ if $\mathcal{M}, s \not\models \varphi$
3. $\mathcal{M}, s \models \varphi \rightarrow \psi$ if $\mathcal{M}, s \not\models \varphi$ or $\mathcal{M}, s \models \psi$
4. $\mathcal{M}, s \models [A]\varphi$ if $\mathcal{M}, t \models \varphi$ for all $t \in W$ such that $(s, t) \in R([\![A]\!])$

A *k-agent Kripke model* for a set of formulas Σ in the language is a k-agent Kripke structure $\mathcal{M}$ such that all formulas $\varphi \in \Sigma$ hold in all states of $\mathcal{M}$ (i.e., are *valid* in $\mathcal{M}$).

As anticipated, the language here presented modifies the basic fragment of **DL**. However, the major novelty, we believe, lies in the change of perspective it pushes for: we need k action identifiers (or multiples of k) to describe the evolution of the whole system since only the combination of all *concurrent actions* can provide this information. Also, this formalism provides a more acceptable notion of action since the outcome of the execution of $\alpha \in \mathrm{Act}$ by an agent is not determined by α alone.

Now we extend the constant operators by allowing the occurrence of free variables. This extension is a first step to introduce quantificational operators, a move we shall motivate in next section.

Fix a new set $\mathrm{Var} = \{x, y, z, \ldots\}$ of variables. Let $\Im$ be an environment function from the set Var to the set Act and let a *modality marker* be any $k \times n$ matrix defined as before but this time with condition $a_{i,j} \in \mathrm{ActId} \cup \mathrm{Var}$ (for all relevant indices i, j). The extension of the set of k-formulas F_k to include these modalities is trivial. Their interpretation requires the new function $\Im$, that is, now relation $\models$ is defined over the triple $\mathcal{M}, s, \Im$. For instance, clause 4. becomes: $\mathcal{M}, s, \Im \models [M]\varphi$ if $\mathcal{M}, t, \Im \models \varphi$ for all $t \in W$ such that $(s,t) \in R(\llbracket M \rrbracket)$ where $\llbracket x \rrbracket = \Im(x)$ when $x \in \mathrm{Var}$. The remaining clauses are analogous to 1–3 above.

3 Quantificational modal operators

Our next goal is the introduction of quantificational modalities in the logic of Section 2. The basic idea is to enrich modality markers with a form of generalized quantifiers introduced by Henkin in [He$_1$61].

Henkin quantifiers are matrices of standard quantifier prefixes[2] such as

$$\begin{pmatrix} \forall x_1 & \exists x_2 & \exists x_3 \\ \exists y_1 & \forall y_2 & \exists y_3 \end{pmatrix}. \tag{1.1}$$

Syntactically these are unary operators, that is, if (H) is a Henkin quantifier and φ is a formula, then $(H)\varphi$ is a formula as well. There is no restriction on the number or position of the quantifiers $\forall$ and $\exists$ in the matrix but no variable may occur more than once.

It is well known that Henkin quantifiers are more expressive than standard quantifiers [Kr$_2$Mo$_4$95] and we take advantage of their strength to ensure row independence in the modalities (which, in turn, models the independence of the agents from each other). Consider a modality with

[2] We follow common practice and use the term ‘Henkin quantifiers’ although these are, properly said, Henkin prefixes. Also, note that the matrix form can be relaxed as we do in formula (1.3) below.

constants and free variables (as described at the end of Section 2), say

$$\begin{bmatrix} x_1 & a & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} p_0. \tag{1.2}$$

The occurrence of a free variable in entry $A(i,j)$ of the matrix suggests that the jth action that agent $\mathcal{A}_i$ is going to perform has not been fixed. One can leave it undetermined meaning that this action depends on the model's environment function. Alternatively, one may want the agent herself to decide which action to perform. In the latter situation, we want to model whether the agent decides in favor or against the realization of p_0 since p_0 is implicitly 'proposed as a goal' by that modal formula. For this reason, we combine modal operators and (a version of) Henkin quantifiers as in the following expression

$$\begin{pmatrix} & & \exists x_3 \\ \exists y_1 & \forall y_2 & \exists y_3 \end{pmatrix} \begin{bmatrix} x_1 & a & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} p_0 \tag{1.3}$$

where, of course, each agent is associated to the same row in both the Henkin quantifier and the modality. In this expression, a free variable x_h in position (i,j) indicates that the agent $\mathcal{A}_i$ at step j performs the action $\Im(x_h)$, i.e., the action determined by the model's environment function. A constant c in position (i,j) indicates that the agent $\mathcal{A}_i$ at step j performs the action $[\![c]\!]$. Finally, a quantified variable x_h in position (i,j) indicates that the agent $\mathcal{A}_i$ chooses what to perform at step j and the specific quantifier marks the attitude of that agent (at this time-step) toward formula p_0. More precisely, if $\exists x_h$ occurs, at time-step j agent $\mathcal{A}_i$ chooses an action with the intention of making p_0 true. Instead, if $\forall x_h$ occurs, the same agent chooses an action randomly.[3]

Here we propose a restriction of this language that consists in merging Henkin quantifiers and modality markers into a unique operator, called *quantificational modal operator*. Thus, instead of formula (1.3) we write

$$\begin{bmatrix} x_1 & a & \exists x_3 \\ \exists y_1 & \forall y_2 & \exists y_3 \end{bmatrix} p_0. \tag{1.4}$$

Following the above discussion, we shall say that p_0 is a goal for agent $\mathcal{A}_i$ at time j if an existentially quantified variable occurs at position (i,j) of the modality, and that p_0 is not a goal (at that time for that agent) if an universally quantified variable occurs instead.

Definition 3.1. A *quantificational modality marker* is a $k \times n$ matrix with each entry containing an action identifier, a variable, or a quantified variable

[3] Informally, she chooses according to her goals. The occurrence of '$\forall$' tells us that p_0 is not a goal for this agent when choosing at this position in the matrix. Thus, from the local perspective given by the formula one can think that the agent chooses randomly at time-step j. A view that further justifies our adoption of the symbols $\forall$ and $\exists$.

provided a variable occurs at most once in a marker. A *quantificational* (*modal*) *operator* is an expression $[M]$ where M is a quantificational modality marker.

We write QOP for the set of quantificational modal operators, OP for the quantificational operators where no variable occurs quantified.

The set F_k of k-formulas is defined as in Section 2 but in clause III) we now use the larger class of quantificational operators. The scope of the modal operator is the formula to which it applies. The scope of a quantifier in a modal operator is the scope of the modal operator itself.[4] Note that we inherit from the Henkin quantifiers the general proviso on variable occurrence in quantificational modality markers (and equivalently in quantificational modal operators).

It remains to discuss the semantics of this language. Below, we investigate alternative interpretations for the quantificational operators starting with one column modalities. We anticipate, at the informal level, that the interpretation in a structure $\mathcal{M} = \langle W, \mathrm{Act}; R, [\![\cdot]\!]\rangle$ of a quantificational operator, say $\begin{bmatrix} x_1 & a & \exists x_3 \\ \exists y_1 & \forall y_2 & \exists y_3 \end{bmatrix}$, takes two steps. In the first step one interprets formula $\begin{pmatrix} & & \exists x_3 \\ \exists y_1 & \forall y_2 & \exists y_3 \end{pmatrix} \varphi$ with $\varphi = \begin{bmatrix} x_1 & a & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} p_0$. The second step amounts to the evaluation of the formula $\begin{bmatrix} x_1 & a & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix} p_0$ obtained by using the values chosen at the first step for the interpretation of the bound variables. The precise formulation in different cases (all restricted to one-column operators) is given below. Note also that we restrict our examples to two-agent systems. However, the argument is easily generalized to an arbitrary number of agents.

3.1 Risk-averse co-ordinated agents

We begin with an interpretation that follows from the classical meaning of the quantifiers $\exists$ and $\forall$. This view relies on what actions exist without considering agents' strategies or capacities.

Fix a structure $\mathcal{M}$ for two agents, say $\mathcal{A}_1$ and $\mathcal{A}_2$, and let $[\![b]\!] = \beta$. First, we look at $\begin{bmatrix} \exists x \\ b \end{bmatrix} p_0$. This formula holds at a state s if and only if there exists an action α such that p_0 is true at all states t with $(s,t) \in R\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$. Such a formula is read: "there exists an action α such that after agent $\mathcal{A}_1$ executes α and (concurrently) agent $\mathcal{A}_2$ executes β, p_0 holds". Similarly, formula $\begin{bmatrix} \forall x \\ b \end{bmatrix} p_0$ corresponds to: "for any action α, after agent $\mathcal{A}_1$ executes it and

[4] It follows from the previous discussion that a quantified variable in the modal operator stands for a quantifier prefix (bounding the occurrences of the variable in the scope of the modality) and for a bound occurrence of that very variable (whose value is needed in this position to interpret the modal operator itself).

(concurrently) agent $\mathcal{A}_2$ executes β, p_0 holds" since it is true at a state s if and only if for all actions α, p_0 is true at all states t with $(s,t) \in R\begin{pmatrix}\alpha\\\beta\end{pmatrix}$.

If we assume that the agents form a coalition or, more generally, are coordinated whenever they both have p_0 as goal, we have that $\begin{bmatrix}\exists x\\\exists y\end{bmatrix} p_0$ is true if there exist actions α and β (not necessarily distinct) such that p_0 holds in all states reachable through $\begin{pmatrix}\alpha\\\beta\end{pmatrix}$. The meaning of formula $\begin{bmatrix}\forall x\\\forall y\end{bmatrix} p_0$ is now obvious: it is true if p_0 is true in *any* state reachable from s via *any* transition.

An important issue arises when considering operators where both quantifiers occur as, for instance, in formula $\begin{bmatrix}\forall x\\\exists y\end{bmatrix} p_0$. To establish the truth value of this formula at a given state we have two options. One can verify that a value β for y exists such that p_0 is true in all states reachable through $\begin{pmatrix}\alpha\\\beta\end{pmatrix}$ for any α. An alternative is to state the formula true if for every action α, there exists β such that p_0 is true in all states t with $(s,t) \in R\begin{pmatrix}\alpha\\\beta\end{pmatrix}$.

By embracing the first interpretation, one extends the semantics of Section 2 with the following clause for quantificational *one-column* operators (for multi-column operators further issues must be addressed, see Section 4):

5_1. Let $[X]$ be a quantificational operator with existentially quantified variables $x_1,\dots,x_r$ and universally quantified variables $y_1,\dots,y_s$ $(r,s \geq 0)$. Then $\mathcal{M},s,\Im \models [X]\varphi$ if: there exist $\alpha_1,\dots,\alpha_r \in \mathrm{Act}$ such that for all $\beta_1,\dots,\beta_s \in \mathrm{Act}$, if Γ is the k-action obtained by substituting α_i for $\exists x_i$, β_j for $\forall y_j$, and $[\![c_h]\!]$ for each action identifier or free variable c_h in $[X]$ (for all relevant indices i,j,h), then for all $(s,t) \in R(\Gamma)$, $\mathcal{M},t,\Im^* \models \varphi$ where $\Im^*(x_i) = \alpha_i$, $\Im^*(y_j) = \beta_j$, and $\Im^*(z) = \Im(z)$.

This semantic clause puts strong constrains on the (one-column) modal operators to the point that it suffices to enrich the basic language of Section 2 with the quantifiers of standard first-order logic to eliminate the need of quantificational modalities. Indeed, clause 5_1 corresponds to the interpretation obtained by replacing quantified variables in the operator with a sequence of standard quantifiers as shown, in a two-agent system, by the following function τ_1:

$$p \xmapsto{\tau_1} p \quad \text{(for } p \text{ atomic)} \tag{1_1}$$

$$\neg\varphi \xmapsto{\tau_1} \neg\tau_1(\varphi) \tag{2_1}$$

$$\varphi \to \psi \xmapsto{\tau_1} \tau_1(\varphi) \to \tau_1(\psi) \tag{3_1}$$

$$\begin{bmatrix} \forall x \\ \forall y \end{bmatrix} \varphi \xmapsto{\tau_1} \forall x, y \begin{bmatrix} x \\ y \end{bmatrix} \tau_1(\varphi) \tag{4_1}$$

$$\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} \varphi \xmapsto{\tau_1} \exists x \forall y \begin{bmatrix} x \\ y \end{bmatrix} \tau_1(\varphi) \tag{5_1}$$

$$\begin{bmatrix} \forall x \\ \exists y \end{bmatrix} \varphi \xmapsto{\tau_1} \exists y \forall x \begin{bmatrix} x \\ y \end{bmatrix} \tau_1(\varphi) \tag{6_1}$$

$$\begin{bmatrix} \exists x \\ \exists y \end{bmatrix} \varphi \xmapsto{\tau_1} \exists x, y \begin{bmatrix} x \\ y \end{bmatrix} \tau_1(\varphi) \tag{7_1}$$

We dub the agents satisfying clause 5_1 the *risk-averse co-ordinated agents*: "risk-averse" because they choose independently of others' decisions as τ_1 makes clear in (5_1) and (6_1). They are "co-ordinated" because, whenever they have a common goal, if possible they execute actions that combined allow them to reach that goal: case (7_1). Indeed, if agents $\mathcal{A}_1$ and $\mathcal{A}_2$ aim at making a formula true, then they behave like a coalition.

Note that the following formula-schema holds for clause 5_1:

$$\begin{bmatrix} \vdots \\ \forall x \\ \vdots \end{bmatrix} p_0 \to \begin{bmatrix} \vdots \\ \exists x \\ \vdots \end{bmatrix} p_0.$$

3.2 Isolated agents

Let us go back to formula $\begin{bmatrix} \exists x \\ b \end{bmatrix} p_0$. We now want to interpret this formula as saying that at a state s agent $\mathcal{A}_1$ *can* choose an action α such that p_0 is true at t for all $(s,t) \in R\begin{pmatrix} \alpha \\ \beta \end{pmatrix}$. That is, this time formula $\begin{bmatrix} \exists x \\ b \end{bmatrix} p_0$ corresponds to reading "agent $\mathcal{A}_1$ can choose an action such that after agent $\mathcal{A}_1$ executes it and (concurrently) agent $\mathcal{A}_2$ executes β, p_0 holds".

Regarding formula $\begin{bmatrix} \forall x \\ b \end{bmatrix} p_0$, informally here we read it as follows: "no matter the action that agent $\mathcal{A}_1$ can choose, after agent $\mathcal{A}_1$ executes it and (concurrently) agent $\mathcal{A}_2$ executes β, p_0 holds". Since we do not put restrictions on the actions an agent can choose or execute, all the actions are possible and must be considered to evaluate the truth value of this formula, i.e., we end up with the following reading: "after agent $\mathcal{A}_1$ executes some action and (concurrently) agent $\mathcal{A}_2$ executes β, p_0 holds".

The meaning of $\begin{bmatrix} \forall x \\ \forall y \end{bmatrix} p_0$ is quite natural at this point: "no matter which action agent $\mathcal{A}_1$ executes and (concurrently) which action agent $\mathcal{A}_2$ executes, p_0 holds in the reached states". That is, with the above assumptions, the notion of 'choice' does not affect the meaning of '$\forall$'. For operators where both quantifiers occur, consider first $\begin{bmatrix} \forall x \\ \exists y \end{bmatrix} p_0$. Here if the agents choose independently (not knowing each other's doing), for the formula to be true the second agent has to find an action β such that for all actions α and all $(s,t) \in R\begin{pmatrix}\alpha\\ \beta\end{pmatrix}$, p_0 holds at t. Analogously, for $\begin{bmatrix}\exists x\\ \forall y\end{bmatrix} p_0$.

Finally, formula $\begin{bmatrix}\exists x\\ \exists y\end{bmatrix} p_0$ is true if the agents can choose actions, say α and β (possibly the same), such that p_0 is true at any state t such that $(s,t) \in R\begin{pmatrix}\alpha\\ \beta\end{pmatrix}$, i.e. "for all choices α made by $\mathcal{A}_1$ and all choices β made by $\mathcal{A}_2$, after $\mathcal{A}_1$ has executed α and $\mathcal{A}_2$ has (concurrently) executed β, p_0 holds". From our reading, we know that $\mathcal{A}_1$ and $\mathcal{A}_2$ have p_0 as (common) goal and, similarly to Section 3.1, we may further establish that they co-operate (or not). However, now we are not in a position to provide a formal definition yet. It remains to be explained what it means that the very agents 'can choose'.

For the time being, let us assume that

1) all elements in the quantificational modal operators (in particular, all action identifiers) are known to all agents

2) all agents choose independently and without communicating (in particular, they do not co-operate nor co-ordinate)

The goal is to extend the semantics of Section 2 to formulas with quantificational operators in such a way that these assumptions are captured.

First, we fix a new function $\mathfrak{C}$, called *choice function*. The intent is that $\mathfrak{C}$ codifies the behavior of the agents by providing the choices the agents make. Function $\mathfrak{C}$ takes as arguments: (1) the modal operator, (2) its scope formula and (3) the variable x, which implicitly gives the agent's index i.[5] Also, since the choices of the agents may depend on their knowledge about the state of the system and other agents' actions, we furnish $\mathfrak{C}$ with two further arguments: (4) the actual state w and (5) subsets of $\{1,\ldots,k\} \times \mathrm{Var} \times \mathrm{Act}$. The reason for this last argument will be discussed in Section 3.3. On input $([M], \varphi, x, w, K)$, with $K \subset \{1,\ldots,k\} \times \mathrm{Var} \times \mathrm{Act}$, $\mathfrak{C}$ returns pairs $(x, \alpha) \in \mathrm{Var} \times \mathrm{Act}$.[6] Thus, given a variable x in row i of a formula φ, $\mathfrak{C}$ provides the agent $\mathcal{A}_i$'s choice(s) for this variable taking into account other information like the actual state. (Since argument (5) is not relevant in this section, for the time being we take $K = \varnothing$.)

[5] More generally (as will be seen later), $\mathfrak{C}$ takes sets of variables as third argument thus we should write $\{x\}$.

[6] For each x occurring in the third argument, $\mathfrak{C}$ returns one or more pairs (x, α) with $\alpha \in \mathrm{Act}$. Admittedly, one may want to generalize the choice function even further to capture some special case. However, the one we have introduced here suffices for a large class of multi-agent systems.

5_2. Let $[X]$ be a quantificational operator with existentially quantified variables $x_1, \ldots, x_r$ and with universally quantified variables $y_1, \ldots, y_s$ $(r, s \geq 0)$. Then, $\mathcal{M}, s, \Im \models [X]\varphi$ if: for all $\alpha_1, \ldots, \alpha_r \in$ Act such that $\alpha_i \in \mathfrak{C}([X], \varphi, x_i, s, \varnothing)$ and for all $\beta_1, \ldots, \beta_s \in$ Act, if Γ is the k-action obtained by substituting α_i for $\exists x_i$, β_j for $\forall y_j$, and $[\![c_h]\!]$ for each action identifier or free variable c_h in $[X]$ (for all relevant indices i, j, h), then for all $(s, t) \in R(\Gamma)$, $\mathcal{M}, t, \Im^* \models \varphi$ where $\Im^*(x_i) = \alpha_i$, $\Im^*(y_j) = \beta_j$, and $\Im^*(c_h) = \Im(c_h)$.

Let $\vec{\alpha} = \alpha_1, \ldots, \alpha_r$ and $\vec{\alpha}' = \alpha'_1, \ldots, \alpha'_r$ be two r-tuples in Act. A *fusion* of $\vec{\alpha}$ and $\vec{\alpha}'$ is a r-tuple $\vec{\alpha}'' = \alpha''_1, \ldots, \alpha''_r$, where $\alpha''_i \in \{\alpha_i, \alpha'_i\}$.

Proposition 3.2. If both $\vec{\alpha}$ and $\vec{\alpha}'$ satisfy the condition in clause 5_2, then any fusion of $\vec{\alpha}, \vec{\alpha}'$ satisfies it as well.

Informally, this property shows that the agents described by clause 5_2 cannot communicate and, consequently, we say that the agents described by this clause are *isolated.*

Unlike in the previous section, in the semantics given by 5_2 formulas

$$\begin{bmatrix} \vdots \\ \exists x \\ \vdots \\ \forall y \\ \vdots \end{bmatrix} p_0 \to \begin{bmatrix} \vdots \\ \exists x \\ \vdots \\ \exists y \\ \vdots \end{bmatrix} p_0\ ; \quad \begin{bmatrix} \vdots \\ \exists x \\ \vdots \\ a \\ \vdots \end{bmatrix} p_0 \to \begin{bmatrix} \vdots \\ \exists x \\ \vdots \\ \exists y \\ \vdots \end{bmatrix} p_0 \tag{1.5}$$

are not valid since the choice function $\mathfrak{C}$ may be sensitive to the occurrences of quantifiers in $[X]$. This result is motivated by the following scenarios.

Two people, $\mathcal{R}_1$ and $\mathcal{R}_2$, are celebrating some achievement. $\mathcal{R}_1$ brought a cake for the occasion. We write p_0 for "the cake is sliced."

In the first scenario, $\mathcal{R}_2$ does not care about cutting the cake and this is known to $\mathcal{R}_1$. A formula that correctly models this situation contains a quantificational operator with an existential quantifier in the first row (the row associated to $\mathcal{R}_1$). The different attitude of $\mathcal{R}_2$ is described by the occurrence of an universal quantifier in the second row. The formula is: $\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} p_0$. Things change if $\mathcal{R}_2$ also wants the cake to be cut. This second scenario is described by the formula $\begin{bmatrix} \exists x \\ \exists y \end{bmatrix} p_0$.

We now use these scenarios to prove that formula $\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} p_0 \to \begin{bmatrix} \exists x \\ \exists y \end{bmatrix} p_0$ fails. In the antecedent $\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} p_0$, $\mathcal{R}_1$ has the goal of getting the cake cut. Since the other agent does not have such a goal, $\mathcal{R}_1$ chooses to execute

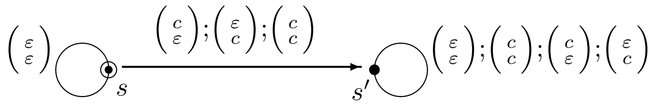

FIGURE 1. The 'cake slicing' frame (The actual state is double circled.)

the action 'cut the cake' which ensures the satisfaction of p_0 in the next state. In the consequent, $\mathscr{A}_1$ and $\mathscr{A}_2$ have the same goal and this is common knowledge because of point 1) of page 58. Since they both have the same goal, $\mathscr{A}_1$ chooses not to cut the cake to let $\mathscr{A}_2$ do the honors. For the same reason, $\mathscr{A}_2$ decides not to cut the cake. Since nobody performs the action of cutting the cake (recall this is a single time-step with concurrent actions) the consequent formula turns out to be false. (Clearly, one can reformulate the example using an action like 'write in memory slot #321' which fails whenever two agents try to perform it at the same time.)

Let us see how function $\mathfrak{C}$ looks for these agents.

The attitude of both agents can be described by the informal rule "cut the cake unless somebody else is willing to do it". In a structure as depicted in Figure 1 where p_0 is false at s and true at s', and the possible actions are c, ε (c stands for "cut the cake" and ε for "do nothing"), function $\mathfrak{C}$ is given by

- $\mathfrak{C}\left(\begin{bmatrix} \exists x \\ \forall y \end{bmatrix}, p_0, x, s, \varnothing\right) = \{(x, c)\}$
  (agent $\mathscr{A}_1$, who has to decide the value of x, chooses c since agent $\mathscr{A}_2$ is not committed to get the cake cut as recognizable by the universal quantifier in the second row);
- $\mathfrak{C}\left(\begin{bmatrix} \exists x \\ \forall y \end{bmatrix}, p_0, y, s, \varnothing\right) = \{(y, c), (y, \varepsilon)\}$
  (agent $\mathscr{A}_2$ may perform any action since p_0 is not her goal);
- $\mathfrak{C}\left(\begin{bmatrix} \exists x \\ \exists y \end{bmatrix}, p_0, x, s, \varnothing\right) = \{(x, \varepsilon)\}$
  (agent $\mathscr{A}_1$ decides not to cut the cake: $\mathscr{A}_2$ is going to ensure it since it is her goal as recognizable by the existential quantifier in the second row);
- $\mathfrak{C}\left(\begin{bmatrix} \exists x \\ \exists y \end{bmatrix}, p_0, y, s, \varnothing\right) = \{(y, \varepsilon)\}$
  (agent $\mathscr{A}_2$ decides not to cut the cake: $\mathscr{A}_1$ is going to ensure it since it is her goal as recognizable by the existential quantifier in the first row).

Function $\mathfrak{C}$ provides a new parameter in the interpretation of the quantificational operators. Clause 5_2 is thus a schema that matches a variety of k-agent systems depending on the parameter $\mathfrak{C}$ and relationship $\models$ should have $\mathfrak{C}$ as index. Note that, due to the complexity of behaviors that function $\mathfrak{C}$ may encode, it is not possible to discharge quantificational operators in this semantics. The analogous of function τ_1 of Section 3.1 that is faithful for 5_2 may not exist.

Clause 5_2 is more general than 5_1. In particular, $\mathfrak{C}$ can formally capture cooperation. The collaboration among the agents is obtained by taking the whole set of variables of the collaborating agents as third argument for $\mathfrak{C}$. That is, $\mathfrak{C}$ provides the variable instantiations for all the agents at once by outputting a set $\{(x_1, \alpha_1), \ldots, (x_r, \alpha_r)\}$. We do not discuss the import of assumptions 1) and 2) further. Instead, below an alternative semantics is given since it sheds light on another issue.

3.3 Optimistic co-ordinated agents

In the discussion of Section 3.1, we mentioned two interpretations for the operators where both universal and existential quantifiers occur. One leads to clause 5_1. The other interpretation is rendered by a different function, here called τ_3, whose definition follows from that of τ_1 provided cases (5_1) and (6_1) are substituted by

$$\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} \varphi \xmapsto{\tau_3} \forall y \exists x \begin{bmatrix} x \\ y \end{bmatrix} \tau_3(\varphi) \tag{5_3}$$

$$\begin{bmatrix} \forall x \\ \exists y \end{bmatrix} \varphi \xmapsto{\tau_3} \forall x \exists y \begin{bmatrix} x \\ y \end{bmatrix} \tau_3(\varphi), \tag{6_3}$$

respectively.

Here is the semantic clause that matches function τ_3:

5_3. Let $[X]$ be a quantificational operator with existentially quantified variables $x_1, \ldots, x_r$ and universally quantified variables $y_1, \ldots, y_s$ $(r, s \geq 0)$. Then $\mathcal{M}, s, \Im \models [X]\varphi$ if: for all $\beta_1, \ldots, \beta_s \in \mathrm{Act}$, there exist $\alpha_1, \ldots, \alpha_r \in \mathrm{Act}$ such that if Γ is the k-action obtained by substituting α_i for $\exists x_i$, β_j for $\forall y_j$, and $[\![c_h]\!]$ for each action identifier or free variable c_h in $[X]$ (for all relevant indices i, j, h), then for all $(s, t) \in R(\Gamma)$, $\mathcal{M}, t, \Im^* \models \varphi$ where $\Im^*(x_i) = \alpha_i$, $\Im^*(y_j) = \beta_j$, and $\Im^*(z) = \Im(z)$.

This clause captures the simple *possibility* for a given formula to be true. For instance, consider the paper/scissor/stone game. If p_0 stands for "$\mathcal{A}_1$ wins", then formula $\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} p_0$ is valid according to this latter semantics and tells us that one has always a chance to win a play of this game. However, the same formula is false for 5_1 since that clause requires the existence of a winning strategy for each possible play.

As before, one can restate clause 5_3 using function $\mathfrak{C}$ explicitly. In this case the fifth argument of $\mathfrak{C}$ is crucial. This argument provides extra knowledge that the agents have while making their choices. In clause 5_3, the agents choosing for the existentially quantified variables are aware of the choices made for the universally quantified variables even though they are done concurrently. This information is provided by set $K = \bigcup_h \{(h, y_1, \beta_1), (h, y_2, \beta_2), \ldots, (h, y_s, \beta_s)\}$ where h ranges over the indeces of the agents associated to a universally quantified variable (such a generality allows us to directly extend the function to multi-column operators). Co-ordination is ensured by providing the whole set of variables $x_1, \ldots, x_r$ as third argument as we have seen in the reconstruction of 5_1 within clause 5_2.

We dub the agents satisfying 5_3 the *optimistic co-ordinated agents.*

We conclude this section with an observation. The difference between clause 5_1 and 5_3 corresponds to the difference between α-ability (effectiveness) and β-ability applied to coalitions (cf. [vHWo$_3$05] and the references therein). It is fairly easy to rewrite schema 5_2 and the conditions on parameter $\mathfrak{C}$ to capture β-ability.

4 Knowing the past, reasoning about the future

Consider a constant two-column[7] operator $\left[\begin{smallmatrix} c_1 & c_3 \\ c_2 & c_4 \end{smallmatrix}\right]$, call it $[C]$. From the definition of the valuation function and clause 4 of Section 2, multi-column constant operators split into simpler operators without loss of information. The following formula is valid

$$\begin{bmatrix} c_1 & c_3 \\ c_2 & c_4 \end{bmatrix} \varphi \equiv \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \begin{bmatrix} c_3 \\ c_4 \end{bmatrix} \varphi.$$

This equivalence does not hold for quantificational operators though, i.e. in general

$$[M_1 M_2]\varphi \not\equiv [M_1][M_2]\varphi \quad (M_1, M_2 \in \mathrm{QOP}). \tag{1.6}$$

One reason is that the order of instantiation of quantified variables may change in the two formulas. For instance, evaluating formula $\left[\begin{smallmatrix} \exists x_1 & \forall x_2 \\ \forall y_1 & \exists y_2 \end{smallmatrix}\right] \varphi$ with clause 5_1 we instantiate first variables x_1, y_2 and only later x_2 and y_1. The instantiation order for the same clause becomes x_1, y_1, y_2, x_2 when we

[7] We give examples using two-column operators. The generalization to n-column operators is often straightforward although some care is needed.

consider formula $\begin{bmatrix} \exists x_1 \\ \forall y_1 \end{bmatrix} \begin{bmatrix} \forall x_2 \\ \exists y_2 \end{bmatrix} \varphi$. This is not so for other semantic alternatives and one is free to adopt or reject a constraint like (1.6) by selecting an appropriate semantics.

It should be clear by now that to establish the truth value of a formula where the constant operator $[C]$ occurs, it is necessary to consider all the action identifiers (and their positions) occurring in $[C]$. For instance, knowing that c_1, c_3 are the actions executed by agent $\mathcal{A}_1$ (in that order) does not suffice to know which states are reachable.

Informally, the formula $\begin{bmatrix} c_1 & \exists x \\ c_2 & c_4 \end{bmatrix} p_0$ means: "first, agent $\mathcal{A}_1$ executes c_1 and (concurrently) agent $\mathcal{A}_2$ executes c_2, then agent $\mathcal{A}_1$ chooses and executes an action and (concurrently) agent $\mathcal{A}_2$ executes c_4". In the light of the previous section, one can interpret the existential quantifier in different ways. The set of choices for x will depend on what agent $\mathcal{A}_1$ knows about the formula itself and in particular about operator $\begin{bmatrix} c_1 & \exists x \\ c_2 & c_4 \end{bmatrix}$. For if she is aware of the presence of c_1, c_2, c_4 and of their positions, she can use the semantic clauses to verify if there is an action that executed after c_1, forces the system to states satisfying p_0. Agent $\mathcal{A}_1$ might rely on default rules (or preferences) when she lacks some information about the components of the operator.

To establish the truth value of the formula, it is important to state what agent $\mathcal{A}_1$ knows (or does not know) about the operator. Several options are possible. For instance, assuming perfect recall, one can assume that agent $\mathcal{A}_1$ is aware that c_1 is in position (1,1) of the modality marker since she has just executed that action. If $\mathcal{A}_1$ and $\mathcal{A}_2$ are *totally isolated* agents, then one can assume that agent $\mathcal{A}_1$ has no information about what $\mathcal{A}_2$ has done earlier, that is, she has no knowledge on the content of position (2,1) of the operator. Analogously for $\mathcal{A}_2$. If $\mathcal{A}_1$ and $\mathcal{A}_2$ are *isolated* but can observe each other's doings, at entry $(1, 2)$ agent $\mathcal{A}_1$ knows that c_2 is in position (2,1). For the simple reason that $\mathcal{A}_1$ and $\mathcal{A}_2$ act concurrently, agent $\mathcal{A}_1$ does know what $\mathcal{A}_2$ is going to execute as second action only if they are co-ordinating or action c_4 is public knowledge.[8] We conclude pointing out that also the quantifiers occurring in the operator may be hidden. After all, an agent may be aware or unaware of the *changes of attitude* in the other agents at different times including, perhaps, her own (past or future) changes.

[8] Most of these features are captured using the fifth argument of function $\mathfrak{C}$. Also, note that in Section 2 we implicitly assumed that the action identifiers in the quantificational operator are known to all the agents, they are *public knowledge*. This assumption is dropped here. Indeed, one may have a commitment to do a specific action c_i at some point and prevent other agents from knowing it or knowing when that action will take place. The semantic clauses we introduced can be modified to mirror these cases.

5 Modeling with quantificational operators

Our first example is in the area of planning. There are two agents, say Anthony ($\mathcal{A}_1$) and Bill ($\mathcal{A}_2$), and a project that must be finished by a certain time. Let us say that there are 3 time-steps before the deadline (step-1, step-2, and step-3) and that Anthony cannot work at the project at time-step 1 since at that time he has to meet his doctor. We use action identifier a for the action Anthony does at this step. Later, he is working full time on the project. Regarding Bill, he will work on the project except at the time-step 2 when he has to meet with the office manager. Bill does not know what the meeting is about. We represent this case in our language with the following formula (φ stands for "the project is finished"):

$$\begin{bmatrix} a & \exists x & \exists z \\ \exists y & \forall u & \exists v \end{bmatrix} \varphi \tag{1.7}$$

The first row describes Anthony's attitude toward the project during the three time-steps, while the second row describes Bill's attitude. Note that the universal quantifier marks the time-step when Bill acts without regards for the project since his action at that time depends on what his office manager asks him to do. If Anthony and Bill are risk-averse and co-operative agents, all the actions that instantiate variables x, z, y, v should be chosen together as described by clause 5_1, where we now allow $[X]$ to be a multi-column quantificational operator (the clause applies to multi-column operators without change). If the two agents work independently from each other (non co-operative agents), then we should adopt clause 5_3 (which also extends to multi-column operators).

We may want to model the case where the agents have a predefined plan for the first two time-steps only. For instance, suppose they agreed on a plan the day before when they knew they where going to be in different places during time-steps 1 and 2 without the possibility of sharing information. Also, let us assume that after time-step 2 they meet so that the decision about the third time-step can be postponed to that time. This situation is described by formula

$$\begin{bmatrix} a & \exists x \\ \exists y & \forall u \end{bmatrix} \begin{bmatrix} \exists z \\ \exists v \end{bmatrix} \varphi.$$

Note the change of the order in which quantified variables are instantiated: the value of u is known when choosing a value for z and v (Section 4).

The second example we consider comes from robotics. Here there are two agents whose goal is to pick up an object but none of them can do it alone. If φ stands for "the object is lifted", the situation is described by formula

$$\begin{bmatrix} \exists x \\ \exists y \end{bmatrix} \varphi \wedge \neg \begin{bmatrix} \forall x \\ \exists y \end{bmatrix} \varphi \wedge \neg \begin{bmatrix} \exists x \\ \forall y \end{bmatrix} \varphi. \tag{1.8}$$

Consider interpretation 5_1. If (1.8) is true, the agents can execute actions that make φ true (first conjunct) but the first or the second agent cannot bring about φ without the collaboration of the other agent (second and third conjuncts). It is possible to make a stronger claim adding the following as conjuncts: $\begin{bmatrix} \forall x \\ \exists y \end{bmatrix} \neg\varphi$, $\begin{bmatrix} \exists x \\ \forall y \end{bmatrix} \neg\varphi$. These tell us that each agent can force φ to be false, i.e., each can prevent the system from reaching any state where φ holds.

6 Related work and conclusions

We looked at the variety of semantics for quantificational operators extending the work in [Bo$_1$05a]. [Bo$_1$03] provides an interpretation for the quantificational modal operators that relies entirely on game-theoretic semantics. [Bo$_1$05b] studies the formal properties of an interpretation along the lines of 5_1. in the framework of standard Kripke semantics.

The formalism we adopted has been influenced by the notion of Henkin (branching) quantifiers [He$_1$61, Wa$_2$70]. Note that there is an ontological discrepancy between the notion of agent in multi-agent systems (where agents are internal components) and the formal notion of player as used in game-theory (players are external components that act to interpret the formalism); a distinction that has not received enough attention in the literature.

A modal version of Hintikka Independent-friendly logic [Hi$_1$Sa$_4$96], which comprises Henkin quantifiers, has been proposed in [Br$_0$Fr$_4$02b]. The aim of the authors is to isolate a notion of bisimulation (model equivalence) that corresponds to their modal system. Related to our work is also the logic **ATL** [Al$_2$He$_2$Ku$_1$02] and its extension **ATEL** [vHWo$_3$02]. The relationship is better analyzed through *Coalition Logic* (**CL**) introduced in [Pa$_7$02]. The connections between **CL** and **ATL** are presented in [Go$_2$01, Go$_2$Ja$_2$04]. Interestingly, the encoding of **CL** into our formalism enables the use of Kripke structures for **CL**. (More precisely, **CL** is semantically equivalent to a fragment of our logic with the semantics given by clause 5_1, cf. [Bo$_1$07]). Other frameworks, like **KARO** [vLvHMe$_3$98] and the variety of systems following the BDI approach [Ra$_3$Ge$_1$91] or the *Intention Logic* [Co$_1$Le$_1$90], adopt combinations of different modalities or exploit full first-order logic. These are very expressive systems and differ in their motivations from our approach. We refer the reader to [vHVe$_2$02, vHWo$_3$03, vDvHKo$_4$07] for overviews on this area of research.

We have shown how to produce different interpretations for modal operators built out of action identifiers, variables, and quantified variables. Our stand is that when there is a number of practical constraints to capture, semantic pluralism could help. We showed how the same language can distinguish and characterize different systems in a flexible way making it possible to describe uniformly what might seem a plethora of heterogeneous cases. Then, formal and reliable comparisons of apparently disparate phenomena become possible at the semantical level. Our approach has some drawbacks as well. The quantificational operators inherit some restrictions of Dynamic Logic, in particular the rigid structure in finite steps. (Extensions with constructs on action identifiers or temporal modalities have not been studied yet.) On the technical side, although adding quantificational modal operators does not make the resulting logic necessarily undecidable, this happens in many cases when equality (over action) is present. For instance, one can see that the theory in [$\mathrm{Bo}_1$03] is undecidable by embedding first-order logic augmented with a binary predicate via $A(x,y) \longmapsto \begin{bmatrix} x \\ y \end{bmatrix} p_0$, for some atomic p_0. For an example in the opposite sense, [$\mathrm{Bo}_1$05b] gives a complete and decidable logic for the class of multi-relational Kripke frames.

References

[$\mathrm{Al_0Le_0}$04] J.J. Alferes & J.A. Leite, eds. *Logics in Artificial Intelligence: 9th European Conference, JELIA 2004*, *Lecture Notes in Computer Science* 3229. Springer, 2004.

[$\mathrm{Al_1Fi_0Sa_3}$91] J.F. Allen, R. Fikes & E. Sandewall, eds. *Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91).* Morgan Kaufmann, 1991.

[$\mathrm{Al_2He_2Ku_1}$02] R. Alur, T. Henzinger & O. Kupferman. Alternating-time temporal logic. *Journal of the ACM* 49(5):672–713, 2002.

[$\mathrm{Ba_5Ma_3}$05] S. Bandini & S. Manzoni, eds. *AI*IA 2005: Advances in Artificial Intelligence, 9th Congress of the Italian Association for Artificial Intelligence, Milan, Italy, September 21–23, 2005, Proceedings*, *Lecture Notes in Computer Science* 3673. Springer, 2005.

[$\mathrm{Be_0}$+02] B. Bennett, C. Dixon, M. Fisher, U. Hustadt, E. Franconi, I. Horrocks & M. de Rijke. Combinations of modal logics. *Artificial Intelligence Review* 17(1):1–20, 2002.

[$\mathrm{Bo_1}$03] S. Borgo. A multi-agent modal language for concurrency with non-communicating agents. In [$\mathrm{Ma_6}$Mü$\mathrm{Pe_1}$03, pp. 40–50].

[$\mathrm{Bo_1}$05a] S. Borgo. Modal operators with adaptable semantics for multi-agent systems. In [$\mathrm{Ba_5Ma_3}$05, pp. 186–197].

[$\mathrm{Bo_1}$05b] S. Borgo. Quantificational modal logic with sequential Kripke semantics. *Journal of Applied Non-Classical Logics* 15(2):137–188, 2005.

[$\mathrm{Bo_1}$07] S. Borgo. Coalitions in action logic. In [$\mathrm{Ve_0}$07, pp. 1822–1827].

[$\mathrm{Br_0Fr_4}$02b] J.C. Bradfield & S.B. Fröschle. On logical and concurrent equivalences. *Electronic Notes in Theoretical Computer Science* 52(1):32–45, 2002.

[$\mathrm{Br_2He_3Tr}$05] J. Broersen, A. Herzig & N. Troquard. From coalition logic to STIT. *Electronic Notes in Theoretical Computer Science* 157(4):23–35, 2005.

[$\mathrm{Ca_4Jo_1}$02] C. Castelfranchi & W. Johnson, eds. *First International Joint Conference on Autonomous Agents and Multiagent Systems.* ACM Press, 2002.

[$\mathrm{Co_1Le_1}$90] P.R. Cohen & H.J. Levesque. Intention is choice with commitment. *Artificial Intelligence* 42(2–3):213–261, 1990.

[$\mathrm{Go_2}$01] V. Goranko. Coalition games and alternating temporal logics. In [vB01d, pp. 259–272].

[$\mathrm{Go_2Ja_2}$04] V. Goranko & W. Jamroga. Comparing semantics of logics for multi-agent systems. *Synthese* 139(2):241–280, 2004.

[$\mathrm{Ha_1Ko_6Ti_1}$00] D. Harel, D. Kozen & J. Tiuryn. *Dynamic Logic.* The MIT Press, 2000.

[$\mathrm{He_1}$61] L. Henkin. Some remarks on infinitely long formulas. In *Infinitistic Methods: Proceedings of the Symposium on Foundations of Mathematics, Warsaw, 2–9 September 1959*, pp. 167–183. Pergamon Press and Państwowe Wydawnictwo Naukowe, 1961.

[$\mathrm{Hi_1Sa_4}$96] J. Hintikka & G. Sandu. Game-theoretical semantics. In [vBtM96, pp. 361–410].

[$\mathrm{Ho_6}$01] J. Horty. *Agency and Deontic Logic.* Oxford University Press, 2001.

[$\mathrm{Kr_2Mo_4}$95] M. Krynicki & M. Mostowski. Henkin quantifiers. In [$\mathrm{Kr_2Mo_4Sz}$95, pp. 193–262].

[Kr_2Mo_4Sz95] M. Krynicki, M. Mostowski & L.W. Szczerba, eds. *Quantifiers: Logics, Models and Computation.* Kluwer Academic Publishers, 1995.

[$Ma_6MüPe_103$] V. Mařík, J.P. Müller & M. Pechoucek, eds. *Multi-Agent Systems and Applications III, 3rd International Central and Eastern European Conference on Multi-Agent Systems, CEEMAS 2003, Prague, Czech Republic, June 16–18, 2003, Proceedings*, *Lecture Notes in Computer Science* 2691. Springer, 2003.

[Pa_702] M. Pauly. A modal logic for coalitional power in games. *Journal of Logic and Computation* 12(1):149–166, 2002.

[Pe_4Ma_902] L. Petrosjan & V. Mazalov, eds. *Game Theory and Applications.* Nova Science Publishers, 2002.

[Ra_3Ge_191] A. Rao & M. Georgeff. Modeling rational agents within a BDI-architecture. In [$Al_1Fi_0Sa_391$, pp. 473–484].

[vB01d] J. van Benthem, ed. *Proceedings of the 8th Conference on Theoretical Aspects of Rationality and Knowledge (TARK VIII).* Morgan Kaufmann, 2001.

[vBtM96] J. van Benthem & A. ter Meulen, eds. *Handbook of Logic and Language.* Elsevier, 1996.

[$vHVe_202$] W. van der Hoek & R. Verbrugge. Epistemic logic: a survey. In [Pe_4Ma_902, pp. 53–94].

[$vHWo_302$] W. van der Hoek & M.J. Wooldridge. Tractable multiagent planning for epistemic goals. In [Ca_4Jo_102, pp. 1167–1174].

[$vHWo_303$] W. van der Hoek & M.J. Wooldridge. Towards a logic of rational agency. *Logic Journal of the IGPL* 11(2):135–159, 2003.

[$vHWo_305$] W. van der Hoek & M.J. Wooldridge. On the logic of cooperation and propositional control. *Artificial Intelligence* 164(1–2):81–119, 2005.

[$vDvHKo_407$] H. van Ditmarsch, W. van der Hoek & B. Kooi. *Dynamic Epistemic Logic*, *Synthese Library* 337. Springer, 2007.

[$vLvHMe_398$] W. van Linder, W. van der Hoek & J.-J.C. Meyer. Formalizing abilities and opportunities of agents. *Fundamenta Informaticae* 34(1–2):53–101, 1998.

[$Ve_0$07] M.M. Veloso, ed. *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6–12, 2007.* 2007.

[$Wa_2$70] W. Walkoe. Finite partially-ordered quantification. *Journal of Symbolic Logic* 35(4):535–555, 1970.

[$We_1$99] G. Weiss, ed. *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence.* The MIT Press, 1999.

[Wö04] S. Wölfl. Qualitative action theory: A comparison of the semantics of Alternating-Time Temporal Logic and the Kutschera-Belnap approach to agency. In [$Al_0Le_0$04, pp. 70–81].