Representations of Time within Normative MAS

Clara SMITH\textsuperscript{a,1}, Antonino ROTOLO\textsuperscript{b}, Giovanni SARTOR\textsuperscript{c}

\textsuperscript{a}Facultad de Informática, y Fac. Cs. Jurídicas y Sociales, UNLP, Argentina
\textsuperscript{b}CIRSFID, University of Bologna, Italy
\textsuperscript{c}EUI, Firenze; and CIRSFID, University of Bologna, Italy

Abstract. We address some forms of temporal reasoning within normative MAS, focusing on the combination of temporal logics with multi-modal multi-agent logics. We suggest perspectives on how these combinations can be used for modelling aspects of time within lawful provisions, obligations, and legal principles. The main contributions are the new variant of deontic tense logic using hybrid logic, and the combination of time and obligations.

Keywords. Temporal logics, deontic logic, combination of logics, multi-agent systems.

Introduction

The relevance of the relation between time and deontic statements was well-settled in the seventies and eighties. Nowadays, several social and normative multi-agent systems (MAS) capture diverse aspects of time mainly from a modelling point of view. However, working applications of modal temporal logics to those systems are still overlooked. For instance, the account in Artikis \textit{et al.} [1] uses the Event Calculus [2]: time is explicitly coded inside constraints as in e.g. $\text{HoldsAt}(\text{permitted(agent, action)}, t)$ which generically specifies a permission: at time $t$ it is true that the $\text{action}$ is permitted for the $\text{agent}$. Boella \textit{et al.} [3] present a defeasible logic which allows expressing some aspects of FIPA semantics for Agent Communication Languages (ACL) [4]. Instants are modeled as timestamps labelling literals, e.g. $l: \text{t}$ stands for ‘$l$ holds at time $t’$. Within these accounts, thinking back and forth along the time line may be a demanding enterprise; moving along the flow of time means searching the knowledge base for rules and literals tagged with a given instant $t$, perform plausible deductions; then repeat with instant $t+1$ (or resp. $t-1$) and so on. Typical modal temporal formulas such as ‘it has always been the case that $p$', or ‘it is always going to be the case that $p’$ need back and forward searches throughout the knowledge base.

Other MAS handle time in different ways. Castelfranchi and Falcone use for their definition of trust in [5] well-known concepts from dynamic logic (which can be seen as including temporal aspects.) The survey of Meyer \textit{et al.} [6] refers to various works incorporating aspects of time to overcome some inherent difficulties with standard deontic logic; for instance, J. A. van Eck [7] relativizes deontic statements to points in time where they should hold. [6] also refers to Madeiro \textit{et al.} [8], which reduce deontic specifications to first-order temporal ones by interpreting the obligation to do $p$ as the...
property that \( p \) must occur sometime in the future. In Broersen et al. \cite{9} the logic of a dyadic modal operator is studied in isolation: \( O(\rho \leq \delta) \) states that one is being obliged to meet a condition \( \rho \) before a condition \( \delta \) becomes true. The operator is referred as a ‘deontic deadline operator’ meaning that deontic deadlines are interactions between a normative and a temporal dimension. Defeasible Logic is used in Governatori et al. \cite{10}; intervals tag obligations as in e.g. Obl \( \varphi : [t, \max] \), meaning that \( \varphi \) is obligatory between moments \( t \) and \( \max \).

Lately, many MAS are designed as multi-modal systems. Theoretically speaking, the very idea of reasoning about time should extend any MAS of this type consistently. From a computational standpoint, we believe that a minimal functionality for the automatic manipulation of time from a basic modal (and linear) perspective involves few technical adjustments of the multi-modal MAS’ logic.

We explore in this paper three ways of embedding temporal reasoning in multi-modal settings: this is done with the purpose to capture relevant aspects of legal reasoning. In a first alternative, we add temporal modalities to a multi-modal language for agents. This extension does not allow for explicitly referring to time instants in the language; therefore it is useful for modelling abstract temporal properties, or capturing uncertain time provisions such as regulations or obligations conditioned to undetermined events. As a second option, we put together a multi-modal and a hybrid logic. This last logic provides an explicit mechanism for identifying points in time and accounts for the moments at which an event happens. Finally, we combine a tense logic with a deontic logic. The combination is meant to describe the coexistence of a (one-dimensional) temporal point of view and a deontic point of view leading to an underlying ontology of situation-point in time pairs.

We organize the work as follows. Section 1 describes combinations of tense logics with multi-modal logics through temporalization. In Section 1.1, a basic tense logic (understood in the classical way introduced by Prior \cite{11}) is used to temporalize multi-modal MASs. This allows us to express abstract time provisions and lawful principles. Section 1.2 explains the intended semantics and provides the technical aspects of the temporalization (done according to the method in \cite{12}.) Section 2 introduces a new variant of deontic tense logics which amounts to enriching a tense logic with a hybrid logic for naming worlds. Section 3 addresses a possible ontology for testing the validity of formulas with arbitrarily interleaved tense and deontic operators. Some final remarks regarding decidability and expressibility end the paper.

1. Time within MAS

The description of a plain, basic logic of time, restricted to a traditional Kripke-style modal perspective, and which comprises a great variety of systems is usually as follows: the basic language is built using two unary operators, \( F \) and \( P \), a set of propositional letters \( P : p, q, r, \ldots \), and Boolean connectives. Formulas are built using the usual inductive way. The intended interpretation of a formula \( Fp \) is ‘\( p \) will be true at some Future time’, and \( Pp \) is meant to stand for ‘\( p \) was true at some Past time’. Duals for \( F \) and \( P \) are, respectively, \( G \) and \( H \) (‘it is always Going to be the case’ and ‘it Has always been the case’.) The traditional mathematical structures where temporal formulas are interpreted are bidirectional frames \cite[pp. 21]{13}. For now, let us assume a frame is a structure \( T = (T, <) \) where \( T \) is a set of instants of time and \( < \) is a precedence
relation such that if \( s < t \) (\( s, t \in T \)) then we say that \( s \) is earlier than \( t \) [14]. The minimal pointwise tense logic is K4, which is complete w.r.t. the class of transitive frames.

Embeddings of tense logic into the basic modal language were first studied by S. K. Thomason in the mid seventies [15]. The approach has gained popularity recently, as pointed out in [14, Sections 6 and 7]; see also [12]. Moreover, Meyer et al. stressed in [6] that Thomason argued that deontic logic requires a foundation in temporal logic, reducing the obligation of \( p \) to a temporal statement that \( p \) holds in all future worlds.

Different MAS’ modalities define different agents’ features such as intentions, beliefs, agency, and so forth (see e.g. [5, 16, 17, 18]). In the following subsection we add a simple temporal perspective to a multi-modal multi-agent logics.

1.1. Obligations and Undetermined Events

The literature provides a number of different techniques for combining logics, such as products, fibring, fusion, and temporalization (see e.g. [12, 19]). The reasoning pattern we present in this section simply requires the addition of a tense logic on top of a base logic. This allows to express some relevant dynamics of normative systems (e.g., always being obliged to be polite), although no temporal content within obligations (e.g., being obliged to be always polite.) This delicate distinction has been pointed out early in [9].

Following Jones and Sergot’s advice [16], we work with a multi-modal approach for dealing with agents’ attitudes. We have a finite set of agents \( A = \{x, y, z, \ldots\} \) and a countable set of propositions. Complex expressions are formed syntactically from these, plus the following unary modalities: a deontic operator \( O \) represents generic (legal/lawful) obligations, meaning “it is obligatory that” [6,16]; the operator \( \text{Does}_x A \) represents successful agency i.e. agent \( x \) indeed brings about \( A \) [20]. For simplicity, we assume that in expressions like \( \text{Does}_x A \), \( A \) denotes behavioural actions concerning only single conducts of agents such as withdrawal, inform, purchase, payment, etc. (i.e. no modalized formulas occur in the scope of Does.) \( O \) is taken to be a classical KD operator. The logic of Does, instead, is non-normal [21].

Having settled the MAS’s operators to work with, we bring in the temporal modalities. \( F \) allows us to represent future obligations, and also the future extinction of an obligation. Examples follow.

Example 1. Basic Temporal MAS; obligations in the future. According to art. 566 of the Argentinian Civil Code, and to art. 1183 of the Italian CC, a deadline may be established for complying with an obligation. Similarly, the emergence of obligations can be postponed to a future moment in time; or a future moment may be established for extinction of obligations. Intuitively, time works in such scenarios as a modality of the obligation; the obligation is what is being modalized. \( F \) permits to naturally express that an obligation will hold sometimes in the future: \( F(O \text{Does}_x A) \) means that “it will be true in some future time that there is the obligation for agent \( x \) to do \( A \)”.

We can also express that there will be a future time in which an obligation will not hold: \( F(\neg(O(\text{Does}_x A))) \) means that “it will be true in some future time that there is no obligation for agent \( x \) to do \( A \)”.

Furthermore, \( F \) may also apply to conditional obligations: \( F(p \rightarrow O(\text{Does}_x A)) \) means that there will be an instant in the future when if \( p \) holds then there is the obligation for agent \( x \) to do \( A \). Similarly, \( F(p \rightarrow \neg(O(\text{Does}_x A))) \) means that there will be an instant in the future when if \( p \) holds then there is no obligation for agent \( x \) to do \( A \). As other instances of this kind of statements, consider:
1) \( F(O(\text{Does y Pay})) \), there is a future moment in which agent \( y \) will have to pay;
2) \( F(\neg O(\text{Does y Pay})) \), there is a future moment in which there is no obligation for \( y \) to pay;
3) \( F(\text{Does z Ask} \rightarrow (O \text{ Does y Pay})) \), in the future there will be a moment in which, if \( z \) asks for the payment then \( y \) has the obligation to pay;
4) \( P(\text{Does y Pays} \land (\neg O(\text{Does y Pay}))) \), there was a past moment in which it was the case that \( y \) paid although he had no obligation to do so.

These statements express interesting normative positions; nonetheless, they fail to capture temporally limited obligations since they do not provide sufficient information to agents \( y \) and \( z \). To be able to comply with an obligation starting in the future (1), \( y \) needs to know what is the precise time when the obligation will start to hold (it may not be enough to know that it will hold in the future.) Similarly, for being able to reject future compliance with an obligation terminating in the future (as in 2), and in, e.g. ‘I will not work for you tomorrow since my work contract expires today’) \( y \) needs to know when the obligation will terminate. For (3) to be satisfied, it is sufficient that there is one future instant in which \( z \) ask for payment, or the payment is provided (using \( F \) as we do, to avoid a violation the payment must happen at the time of asking.)

We bring in \( G \), the dual of \( F \), for modelling the persistence of an obligation (or its absence.) \( G A \) stands for “\( A \) will always hold in the future”; e.g. \( \text{Dies} \rightarrow G(\neg O(\text{Does x A})) \) means that if \( x \) dies, s/he will never have the obligation to do \( A \). For example, the death of a person who has been instituted as head of a life-rent extinguishes forever the obligation to pay (art. 2070 ACC, art. 1873 ICC.) Note that simple formulas like the former consequent \( G(\neg O(\text{Does x A})) \) are indeed powerful. This expression can be considered as a possible formulation for the legal institution called prescription, which in one of it forms establishes that “by the time designed by law, the debtor is free from its obligation” meaning that there will be a future moment in time in which the creditor cannot pursue her/his legal right in court (art. 4017 ACC, art. 2934 ICC.) However, to model a prescription more is required. In Section 2 we supplement our account with an explicit way of naming points in time, and time intervals.

1.2. Semantics and Temporalization of Simple Normative MAS

For combining a temporal and a multi-modal multi-agent logics we follow the temporalization technique [12,19]. The temporal logic and the MAS’s logic will interact in a restricted one-way manner. Intuitively, the temporalization amounts to place the temporal machinery on top of the MAS. We restrict ourselves to well-formed atomic formulas whose outermost symbol is a temporal operator. Formally, the behaviour of such a system is captured by a model \((T, <, g, t_0)\). The outer frame \((T, <)\) corresponds to the temporal evolution of the system; \( t_0 \in T \) is the initial point in time.

The system evolves through time in the sense that new generic/individual obligations/permissions are settled while some others become obsolete or prescribe (consider both general and relativised obligtions: \( O^a \) is a deontic operator meaning “it is obligatory in the interest of agent \( a \) that” [22]. The function \( g \) is such that for every \( t \in T \), \( g(t) \) returns a model for the MAS itself. According to [18,21] build a multi-relational frame of the form

\[
F = < A, W, O_1, \{O^i\}_{i \in G}, \{D\}_{i \in G} >
\]
where $A$ is a finite set of agents; $W$ is a set of situations, or possible worlds; $O$ is the accessibility relation for the deontic operator, which is serial (standard KD semantics); \( \{O_i\}_{i \in G} \) is the set of accessibility relations w.r.t. relativised obligations, which are serial (usual KD\(_n\) semantics); \( \{D_i\}_{i \in G} \) is a family of sets of accessibility relations $D_i$ w.r.t. Does; which are reflexive, serial, and pointwise closed under intersection.

A multi-relational model is in its turn a structure of the form $M = <F,V>$ where $F$ is a multi-relational frame as above, and $V$ is a valuation function defined as follows:

- standard Boolean conditions;
- \( V(w, O A) = 1 \) iff $\forall v (w O v \rightarrow V(v, A) = 1)\);
- \( V(w, O_i A) = 1 \) iff $\forall v (w O_i v \rightarrow V(v, A) = 1)\);
- \( V(w, Does_i A) = 1 \) iff $\exists D_i \in D_i$ such that $\forall v (w D_i v \rightarrow V(v, A) = 1)\).

This way, $(T, <, g, t_0)$ amounts to a model for the logic of a temporalized MAS. According to these definitions, soundness and completeness for the resulting logic follow directly from [18,20,21]. Note that although in the resulting logic we can write temporal formulas, the formulas of the nested MAS (with no temporal operators) are unambiguously evaluated w.r.t. a non-temporal model $g(t)$.

Now the technical aspects of the temporalization. Call $T$ the basic propositional tense logic in Section 1, and call $L$ the multi-modal logic for the MAS. We can safely assume that $L$ is an extension of propositional logic. Following [12], we partition the set of formulas in $L$ into two subsets: Boolean formulas, $B_L$, and monolithic formulas, $M_L$. A formula $\hat{A}$ belongs to $B_L$ if its outermost operator is a Boolean connective; otherwise it belongs to $M_L$. It is also assumed that there is no intersection among the set of connectives of $T$ and $L$. Call $T(L)$ the temporalization of $L$ by means of $T$.

**Temporalization: Syntax.** Let $L_L$ denote the language of $L$, and $L_T$ denote the language of $T$. The language $L_{TIL}$ of $T(L)$ – over the set of proposition letters $P$ – is obtained by replacing the formation rule of sentences in $L_T$ that says “every proposition letter in $P$ is a formula” by the formation rule:

"every monolithic formula in $L_L$ is a formula."

As pointed out in [19], this replacement can be matched with a process called “fuzzling” or layering, characterized by the fact that the formulas in the base system can be substituted for atoms of the top system. A model for $T(L)$ has the structure $M = <T, <, g, t_0>$ where $(T, <)$ is the frame for the tense logic $T$, and $g$ is the aforementioned total function mapping worlds in the set $T$ to models for the multi-modal logics $L$. Let us call $K_L$ the set of models for $L$; then $g: W \rightarrow K_L$.

**Temporalization: Semantics.** Given a model $M$ for $T(L)$, and a valuation function in $M$, the semantics for $T(L)$ is obtained by replacing the clause for $T$ that says

\[ M, t \models p \text{ iff } p \in V(t), \text{ whenever } p \in P, \]

by the clause:

\[ M, t \models A \text{ iff } g(t) \models_L A, \text{ whenever } A \in M_L. \]

Once a formula has entered a component of the model it can never come back to the top level [12]. Subsequently, in the present layout we can not test validity of statements such as ‘there is a future time in which it will be obligatory that $A$ will always be the case’: $F(O(GA))$. In Section 3 we address a possible ontology for testing the validity of these kind of formulas.
2. Hybrid Temporal Normative MAS

We work next with a hybrid temporal logic [13, Chapter 7] and [23,24], an extension of the basic temporal logic in Section 1. This logic will subsequently be the one used to temporalize. A hybrid temporal language helps us to treat points in time as “first class objects”, by naming them individually and directly. For example, if we are working with months, we can assign the names J, F, M, A, ... to particular worlds in the domain, the same way we give constant names to specific elements in any domain (e.g. ‘0’ and ‘1’ within the integers.) Being able to locate points in time leads us to ask whether a certain event will happen in a given future moment, like in: “will April be rainy?”, or if it has happened in a past time, as in: “were the company flights rescheduled in August?”. Finally, we may want to give a name to the current moment because, e.g., it is the moment in which we are in a position to fulfill our and other parties’ expectations; assuming a domain of days we can propose: “let’s sign the contract today, December 27th”, or: “you may start working from now, December 27th, on”. This possibility is functional for indicating a date in lawful acts.

We proceed as follows. Let us take the basic modal language and add a second sort of atomic formulas. These atoms are called nominals, written i, j, k ... A nominal i names a world by being true in that world and nowhere else; i.e. if i is a nominal the formula i holds if and only if the current location is named i. Old and new atoms are combined to form complex formulas in the usual way (as in i ∧ p.) For direct access to worlds, basic hybrid logics provides a ‘@’ operator. This operator, called a satisfaction operator, allows us to write formulas such as @i A which is true at any point in a model if and only if A is satisfied at the unique point named by i. For example, the fact that an obligation A is claimable by time i can be written as @i  A. We may eventually have a family of @i operators in our language; hybrid logics are essentially multi-modal logics and satisfaction operators are normal modal operators. Note that @i A expressions play the same role of HoldsAt(A, i) propositions and i:t facts. Hybrid logics may also include the downarrow binder ↓i. which creates a name i and assigns it to the currently evaluated world.

Regarding the temporalization, it remains almost unaltered when considering T a hybrid logic. L_T is extended –for T to become hybrid– with the new sort of nominals plus the symbols @ and ↓. Two examples follow.

**Example 2. Hybrid-temporal MAS:** termed and claimable obligations. Consider the domain of days. The formulas ↓i.e → @i,G(O(Does,x,A)) and ↓j.F(↓j.e) ∧ @j.G(¬(O(Does,y,A))) stand for “when the event e occurs, from that day on it will be the case that agent x is obliged to do A”, and “from now on, there will be a day (j) in which the event e will happen, and from that day on it will be the case that there is no obligation for x to do A”, respectively.

**Example 3. Hybrid-temporalised MAS with relativised obligations:** payment made before due time. The expression (↓i.(Does,x,Pay) ∧ @j.F(↓j.O′(Does,y,Pay))) → @i.¬G(O′ Does, Reimburse) establishes that if agent x pays on day i and there is a future day j in which it is obligatory in the interest of agent y that x pays, then there is no obligation for y to reimburse such payment (art. 571 ACC.)
2.1. Deadlines, Intervals

We have been working with abstract temporal properties, and with uncertain time provisions. Then we added the possibility of explicitly referring to points in time. But for several applications all these appear to be not enough. For example, consider that agent \(x\) works for agent \(y\), starting on day \(i\): \(\Diamond_i \Diamond (\text{Does } \text{Workfor}(y))\). Similarly, write the fact that \(x\) will not be obliged to work for \(y\) from day \(j\) on as: \(\Box_j \neg \Diamond (\text{Does } \text{Workfor}(y))\). Though both prescriptions make sense separately, they may lead to a contradictory situation: in the future there are instants in which \(x\) is obliged to work and obliged not to work for \(y\) (for getting the contradiction, ensure that time is linear or that \(i\) and \(j\) belong to the same branch in a tree structure.)

A Further extension. We expand our account with obligations spanning until an event terminates them. This allows us to write sentences like “if agent \(a\) damages agent \(b\), until the obligation is extinguished it is obligatory that \(a\) pays \(b\)”, which responds to the more general pattern “\(p\) will be the case, and until that happens, \(q\) will hold”: \(U(p, q)\). Those \(p\) properties are called guarantee properties in the computational literature [13, 14]. While \(U\) (until) searches forwards, \(S(p, q)\) – the since operator – looks backwards.

Let us fit these operators in our hybrid-temporal multi-modal account. The usual satisfaction definition for \(U\) is: \(t \models U(\phi, \psi)\) iff there is a \(v > t\) such that \(v \models \phi\) and, for all \(s\) with \(t < s < v\): \(s \models \psi\). This is to be interpreted on frames with structure \((T, <)\).

This semantics for \(U\) suits the intended behaviour of models \((T, <, g, t_0)\) in Section 1.2. (Correspondingly, define \(S\) as a search in the opposite direction in the time line.) Note that the final point in which \(\psi\) is true is the immediate \(s\) before \(v\).

It is well known that \(U\) and \(S\) are not definable in the basic modal language (see e.g. [13,14]); on the other hand, \(F\) and \(P\) are definable in a language with \(U\) and \(S\). Thus, such a language is stronger. Extend again \(L_T\), now with \(U\) and \(S\), and \(S\)-formulas are built straightforwardly from Boolean connectives and propositions in the usual way. The temporalization remains untouched, except for the inclusion of \(U\) and \(S\) at the upper level.

Example 4. Hybrid-temporal MAS with Until; deal example. Agents \(a\) and \(b\) make the following deal: “\(a\) has the obligation to pay \(b\) until \(a\) indeed pays or June arrives”. This deal can be written as: \(U(((\text{Does } a \text{ Pay}) \lor \text{June}), \text{O}_b \text{Does } \text{Pay})\); June is here a nominal (and nominals play the role of propositional constants.) A proper payment before June will release \(a\) from his obligation. It follows from the example (and from \(U\)’s satisfaction definition) that the obligation to pay holds in all months previous to June and up to the month \(a\) pays (if on time); when June comes, the obligation will no longer hold, according to what settled.

2.2. Regarding the Use of \(U\) and \(S\) within \(T(L)\), and the use of \(O(\rho \leq \delta)\).

The logic underlying \(O(\rho \leq \delta)\) as presented in [9] is not hybridized; i.e. it does not allow to explicitly handle names for points in time. (Few adjustments are needed to get \(O(\rho \leq \delta)\)’s logic hybridized.)

\(O(\rho \leq \delta)\) always moves forward. With a U-S tool we are equipped with a back-and forth search through time. In our account it is easy to express e.g. the rule “since the rise of the European Union one must use unified currency within the member states,
old local currency is no longer accepted": \( S(\text{raiseUE}, O(\text{Does}, \text{PayEuros}) \land F(\text{Does}, \text{PayLocalCurrency})) \). (\( F \) here stands for “forbidden”, recall \( Op \equiv \neg P \neg p \), and \( \neg Pp \equiv Fp \).)

\( O(\rho \subseteq \delta) \) is a specific operator that deals with the deontic aspect of deadlines, while \( U \) and \( S \) are well understood temporal modalities that do not refer to obligations at all. We are (relatively) free to combine both \( U \) and \( S \) not only with \( O \) but also with other modalities. For example, suppose that agent \( y \) does not believe that agent \( x \) is travelling, and says “I won’t believe he is travelling until he shows the ticket to me”: \( U(\text{Does}, \text{ShowsTicket}, \neg \text{Bel}_y \text{Travels}). \) This expression amounts to what we can call a deadline belief (in the same line [9] refers to deadline intentions.)

Regarding the way of counting intervals within the law, consider the formula \( U(\phi, O\delta) \). From \( U \)’s definition we get that the guarantee \( \phi \) “starts to hold” at the first point following the current world (i.e. the one where the \( U \)-formula is tested.) This complies with the legal notion of notification: within the domain of days, if one is notified in accordance to the legal procedures, the (eventually triggered) deadlines usually begin on the day after the reception of the notification. Next, notice that the last point in which \( \phi \) holds is the point just before the deadline: for example, within the domain of days, just the day before. One can comply with what settled at most at the instant before \( \phi \) holds. This should not be seen as a drawback but as a matter of modeling; suppose the statement: “new t-shirts for the team must be received for the final match”. Within the domain of days, even within the domain of hours or minutes, t-shirts can not be received by the final match, but at least in the instant ( whatsoever) before. To consider deadlines which allow the fulfilment of an obligation at the instant of expiration, as in ‘you have until Friday to pay your debt!’, strengthen the satisfaction condition for \( U \):

\[
t \models U(\phi, \psi) \iff \text{there is a } v > t \text{ such that } v \models \phi, \text{ and for all } s \text{ with } t < s < v: s \models \psi, \text{ and } v \models \psi.
\]

This definition better suits the lawful notion of deadline: fulfil what required yet at the very last moment of expiration (as settled in, e.g., art. 27 ACC).

3. An Ontology for a Combination of Time and Obligations

\( F(O\delta) \) is a ‘temporalized’ formula, while \( O(F\delta) \) is not. We now combine logics for writing and testing the validity of formulas with arbitrarily interleaved deontic and tense modalities. Assume two structures \((W,R)\) and \((T,\prec)\) which are respectively the underlying ontologies where a deontic point of view and a temporal point of view are interpreted. Think of both structures as being traditional Kripke models: \((W,R)\) represents a multigraph over situations, \((T,\prec)\) represents a valid time line. Next, build an ontology \( W \times T \) of pairs \((\text{situation, point in time})\) representing the intuition “this situation, at this time”. For example, the pair \((\text{switch, 5 pm})\) captures the status of a switch at 5 pm, while \((\text{switch, 6 pm})\) recalls the switch at 6 pm.

**Combination. Syntax.** Let \( L_L \) denote the language of \( L \) (the logic for the MAS), and \( L_T \) denote the language of \( T \) (the basic temporal logic). The language \( L_{LT} \) is obtained by taking the union of the formation rules for the combination of \( L_L \) and \( L_T \). Unlike the case of \( L_{TLL} \), \( F(O\delta) \) and \( O(F\delta) \) are both formulas of \( L_{LT} \).
Combination. Semantics. Assume two structures \((W, R)\) and \((T, <)\) where to respectively test the validity of deontic formulas and temporal formulas. Interpret \(\mathcal{L}_{L,T}\) formulas over a combined model \(M = (W \times T, R, <, V)\), where: \(W \times T\) is a a set of pairs situation-point in time; \(R\) is the accessibility relation for the deontic operator, which is serial (standard KD semantics); \(<\) is the accessibility relation for the temporal operator (\(K_4\) semantics); and \(V : W \times T \rightarrow P \omega (P)\) is a function assigning to each pair in \(W \times T\) the set of proposition letters in \(P\) which are true. The definition of a formula in \(\mathcal{L}_{L,T}\) being satisfied in a model \(M = (W \times T, R, <, V)\) at state \((w,t)\) amounts to:

1. \(M, (w,t) \models p\) iff \(p \in V(w,t)\) for all \(p \in P\);
2. \(M, (w,t) \models \neg A\) iff not \(M, (w,t) \models A\);
3. \(M, (w,t) \models A \lor B\) iff \(M, (w,t) \models A\) or \(M, (w,t) \models B\);
4. \(M, (w,t) \models O A\) iff \((\forall v \in W)\) (if \(wRv\) then \(M, (v,t) \models A\));
5. \(M, (w,t) \models F A\) iff \((\exists s \in T)\) \((t < s \text{ and } M, (w,s) \models A)\).

A scan through the structure is done according to which operator is being tested: \(O\) makes one move along the first component of the current world (alternative situations); \(F\) moves along the second component of the current world (points in time).

Example 5. Combination of Time and Obligations; the security norms for a switch system. Distinguish a propositional variable, let us call it \(\text{safe}\). Given a model \(M\), a pair \((w,t)\) is called \textit{safe} if \(\text{safe} \in V(w,t)\). Within this context, define a pair \((w,t)\) as being \textit{secure} iff \(M, (w,t) \models O(G(\text{safe}))\). For \(wRw'\) and \(t<t'\), it is clear that \((w,t)\) may be a safe “state” without \((w',t')\) or \((w',t')\) being so. Assume a switch-time ontology: we have e.g. the pairs \((w_1, 5pm)\), \((w_2, 6pm)\), \((w_1, 5pm)\), \((w_2, 6pm)\), among other pairs in \(W \times T\). Now suppose a configuration where \((T, <)\) amounts to the obvious time line, \(W = \{w_1, w_2\}\) are the two switches of the system, \(R = \{w_1Rw_2, w_1Rw_2, w_2Rw_2\}\), and assume \(w_1\) is safe at 5pm i.e. \(M, (w_1, 5pm) \models \text{safe}\), being \((w_1, 5pm)\) “this” world. How do we test for security in this world (i.e. the validity of the combined formula \(O(G(\text{safe}))\)?) For testing this, the movements along the multigraph are determined by: \(M, (w_1, 5pm) \models G(\text{safe})\) iff \((\forall v \in W)\) (if \(w_1Rv\) then \(M, (v, 5pm) \models G(\text{safe})\)); which amounts to test \((\forall v \in W)\) (if \(w_1Rv\) then \(M, (v, 5pm) \models ((\forall s \in T)\) \((t < s \text{ and } M, (v,s) \models \text{safe})\)).

The ontology of time and obligations outlined here leads us to a wider interplay of modalities than those accounts proposed in the previous sections; such interplay was not expressible in \(\mathcal{L}_{T,L}\). Our account here amounts to to put together the expressivity of two separate logic systems \((W, R)\) and \((T, <)\). In \((W \times T, R, <)\) we retain the independency between the two source logics and we gain the fact that the resultant account is based on their fully combined language.

4. Final Remarks

From the logical point of view, the temporalizations in this paper are simple; their simplicity is support for their usefulness and robustness, and also keeps the systems manageable, decidable, and suitable for further studies and extensions. Regarding decidability, it is well known that the logic that results from the temporalization of a decidable logic and a basic tense logic is also decidable. PSPACE algorithms have
been devised for a number of well-known logics including the temporal counterparts of K, T, K4 and S4. Theorem 7.1 in [13, pp. 436] settles that the satisfiability problem for basic hybrid logics is PSPACE-complete. Finite model checking algorithms for temporalizations are available in [12] (a temporalization using a hybrid logic is suggested in that paper.) Bolander and Blackburn provide terminating tableau systems for a number on non-transitive hybrid logics extending K such as the logic of irreflexive, antisymmetric frames [25]; they also provide –for hybrid tense logic enriched with a universal modality– a terminating tableau calculus for the logic of transitive frames. Finally, U and S-formulas are complete w.r.t. frames (T,\(<\)) such that (T,\(<\)) is a well-ordered flow of time called Dedekind complete order, such as the total order of the natural numbers [13]. This background provides a strong platform where to build proof procedures for the temporalized normative MAS we outlined.

References