A modal logic for reasoning on consistency and completeness of regulations

C. Garion[1] and S. Roussel[1,2] and L. Cholvy[2]

[1] ISAE
10 avenue Edouard Belin,
31055 Toulouse, France
[2] ONERA Centre de Toulouse
2 avenue Edouard Belin
31055 Toulouse, France

Abstract. In this paper, we deal with regulations that may exist in multiagent systems in order to regulate agent behaviour. More precisely, we discuss two properties of regulations, consistency and completeness. After defining what consistency and completeness mean, we propose a way to consistently complete incomplete regulations. This contribution considers that regulations are expressed in a first order deontic logic.

1 Introduction

In a society of agents, a regulation is a set of statements, or norms, which rule the behaviour of agents by expressing what is obligatory, permitted, forbidden and under which conditions. Such a regulation is for instance the one which applies in most countries in EU: *smoking is forbidden in any public area except specific places and in such specific places, smoking is permitted*. Another example of regulation is the one which gives the permissions, prohibitions (and sometimes the obligations) of the different users of a computer system for file reading, file writing and file execution. Regulations are means to regulate agent behaviour so that they can live together. But in order to be useful, regulations must be *consistent* and, in most cases, they must also be *complete*.

Consistency is a property of regulations that has already been given some attention in the literature. For instance, as for confidentiality policies, consistency allows to avoid cases when the user has both the permission and the prohibition to know something [2]. More generally, according to [4] which studies consistency of general kind of regulations, a regulation is consistent if there is no possible situation which leads an agent to *normative contradictions* or *dilemmas* also called in [20] *contradictory conflicts* (a given behaviour is prescribed and not prescribed, or prohibited and not prohibited) and *contrary conflicts* (a given behaviour is prescribed and prohibited). Following this definition, consistency of security policies has then been be studied in [5].

Completeness of regulations has received much less attention. [2] proposes a definition of completeness between two confidentiality policies (for each piece of information, the user must have either the permission to know it or the prohibition to know it), definition which has been adapted in [8] for multilevel security policies.

More recently, we have studied the notion of completeness for particular regulations which are policies ruling information exchanges in a multiagent system [6]. A definition of incompleteness for such policies has been given and a way to reason with incomplete policies has been defined. The approach taken in this work was rather promising and we have extended it for general regulations in [7]. The formal language used in those papers is classical first-order logic (FOL) following the ideas developed in [4]. In particular, deontic notions (obligation, permission, prohibition) are represented using predicate symbols. Because this leads to a rather complicated partition of the language between deontic predicate symbols and predicate symbols representing objects properties, this approach can be criticized. Moreover, deontic notions are classically represented in modal logic since [19, 14]. This is the reason why, in this present paper, we aim at using first order modal logic [12] to express regulations in a more elegant manner. Our objective is thus to reformulate the work described in [7] in a first-order modal framework.

This paper is organised as follows. Section 2 presents the logical formalism used to express regulations, the definitions of consistency and completeness of regulations. Section 3 focuses on the problem of reasoning with an incomplete regulation. Following the approach that has led to the default logic [17] for default reasoning, we present defaults that can be used in order to complete an incomplete regulation. In section 4, we present a particular example of regulation, information exchange policy. Finally, section 5 is devoted to a discussion and extensions of this work will be mentioned.

2 Regulations

The basic formalism used to model regulations is SDL (Standard Deontic Logic), a propositional modal logic [3]. We extend SDL to FOSDL (First-Order Standard Deontic Logic) in order to be able to express complex regulations implicating several agents. This is done in the way developed in [12].

2.1 Language

The alphabet of FOSDL is based on the following sets of non logical symbols: a set $\mathcal{P}$ of predicate symbols, a set $\mathcal{F}$ of function symbols and a modality symbol O representing obligation. The set of functions with arity 0 is called the *constants set* denoted $\mathcal{C}$. We define also the following logical symbols: a set $\mathcal{V}$ of variable symbols, $\neg$, $\vee$, $\forall$, (and). We call a *term* a variable or the application of a function symbol to a term.

We will use roman uppercase letters as predicate symbols, roman lowercase letters as function symbols and $\{x_1, \ldots, x_i, \ldots\}$ as variable symbols.

Definition 1. *The formulae of FOSDL are defined recursively as follows:*

– *if $t_1, \ldots, t_n$ are terms and P a predicate symbol with arity n, then $P(t_1, \ldots, t_n)$ is a formula of FOSDL.*
– *if φ is a formula of FOSDL, then $O\varphi$ is a formula of FOSDL.*
– *if ψ_1 and ψ_2 are formulae of FOSDL and x_1 a variable symbol, then $\neg\psi_1$, $\psi_1 \vee \psi_2$, $\forall x_1\ \psi_1$ are formulae of FOSDL.*

If ψ_1, ψ_2 and ψ_3 are FOSDL formulae and x_1 is a variable symbol, we also define the following abbreviations: $\psi_1 \wedge \psi_2 \equiv \neg(\neg\psi_1 \vee \neg\psi_2)$, $\psi_1 \otimes \psi_2 \otimes \psi_3 \equiv (\psi_1 \wedge \neg\psi_2 \wedge \neg\psi_3) \vee (\neg\psi_1 \wedge \psi_2 \wedge \neg\psi_3) \vee (\neg\psi_1 \wedge \neg\psi_2 \wedge \psi_3)$, $\psi_1 \rightarrow \psi_2 \equiv \neg\psi_1 \vee \psi_2$, $\psi_1 \leftrightarrow \psi_2 \equiv (\neg\psi_1 \vee \psi_2) \wedge (\psi_1 \vee \neg\psi_2)$, $\exists x_1\ \psi_1 \equiv \neg\forall x_1\ \neg\psi_1$.

The modalities for permission, noted P, and prohibition, noted F, are defined from O in the following way:

$$F\varphi \equiv O\neg\varphi$$
$$P\varphi \equiv \neg O\varphi \wedge \neg O\neg\varphi$$

It must be noticed that our definition of permission does not correspond to the usual definition of permission defined in SDL. According to SDL, something is permitted if its negation is not obligatory. However, it has been shown by lawyers [13] that the cases where permission is bilateral (permission to do and permission not to do) are the only valid ones. If not bilateral, permission to do entails obligation to do[1]. Our definition of bilateral permission corresponds to the notion of *optionality*[15] (something is optional iff neither it or its negation is obligatory).

A formula of FOSDL without modality is said to be *objective*. A term of FOSDL without variable symbols is said to be *ground*. The set of all ground terms in FOSDL is said to be the Herbrand universe HU. A formula of FOSDL without variable is said to be *ground*. A formula of FOSDL without the $\vee$, $\wedge$, $\otimes$, $\rightarrow$ nor $\leftrightarrow$ connectives is said to be a *literal*. Finally, we will call a *ground substitution* any function $\chi\ : \mathcal{V} \rightarrow HU$. If $\varphi(x)$ is a FOSDL formula with free variable x, $\varphi(\chi(x))$ is the formula φ in which occurrences of x have been replaced by $\chi(x)$.

2.2 Semantics

Semantics for propositional modal logics are classically defined using Kripke models. Models are defined by a *frame* $\langle\mathcal{W}, \mathcal{R}\rangle$, where $\mathcal{W}$ is a set of worlds and $\mathcal{R}$ an accessibility relation between worlds, and a relation $\Vdash$ between worlds and propositional letters. In the first-order case, we define models using an *augmented* frame and a first-order interpretation instead of $\Vdash$.

The semantics of first-order languages is based on a set of symbols (the *objects of discourse*), called *the domain*. The domain represents the objects on which the predicates will be evaluated by opposition to terms which are purely mathematical notions. In the case of first-order modal logic, we have to choose between

[1] For instance, when smoking is permitted, this implies that not smoking is also permitted. If not, that would mean that smoking would be obligatory.

a *constant domain* augmented frame and a *varying domain* augmented frame. In the first case, the domain is fixed for all the worlds in $\mathcal{W}$, in the second case, each world of $\mathcal{W}$ can have its own domain. We choose here a constant domain. As we study norms concerning only fixed elements, this choice is intuitively justified[2].

Definition 2. *Let $\mathcal{W}$ be a set of worlds, $\mathcal{R}_O$ a relation on $\mathcal{W}^2$ and $\mathcal{D}$ a non empty set of symbols representing the domain, then $\langle \mathcal{W}, \mathcal{R}_O, \mathcal{D} \rangle$ is called a frame.*

To define a model, we have to define an first-order interpretation which is done classically.

Definition 3. *An interpretation $\mathcal{I}$ in a frame $\langle \mathcal{W}, \mathcal{R}_O, \mathcal{D} \rangle$ is an application such that:*

- *for all n-ary function symbol f in $\mathcal{F}$ and all world $w \in \mathcal{W}$, $\mathcal{I}(f, w)$ is a function $\mathcal{D}^n \rightarrow \mathcal{D}$* independent of the world *w;*
- *for all n-ary predicate symbol P in $\mathcal{P}$ and all world $w \in \mathcal{W}$, $\mathcal{I}(P, w)$ is a relation on $\mathcal{D}^n$.*

Notice that we impose a particular condition on the interpretation of functions: the interpretation of a given function f is the same in every world w of $\mathcal{W}$ (this is possible because we use constant domain frames). This restriction allows us to escape from complicated technical details[3], for instance predicate abstraction. See [12] for more details.

Definition 4. *A model $\mathcal{M}$ is a structure $\langle \mathcal{W}, \mathcal{R}_O, \mathcal{D}, \mathcal{I} \rangle$ where $\langle \mathcal{W}, \mathcal{R}_O, \mathcal{D} \rangle$ is a frame and $\mathcal{I}$ an interpretation on $\langle \mathcal{W}, \mathcal{R}_O, \mathcal{D} \rangle$.*

Finally, we only use a class of frames that capture the correct behaviour of the modal operator O by constraining the accessibility relation $\mathcal{R}_O$.

Definition 5. *A FOSDL model is a model $\langle \mathcal{W}, \mathcal{R}_O, \mathcal{D}, \mathcal{I} \rangle$ such that $\mathcal{R}_O$ is serial.*

In order to define a satisfiability relation between models and formulae, we have to define the *valuation* notion which maps variables to elements of $\mathcal{D}$:

Definition 6. *Let $\mathcal{D}$ be a domain. A valuation on $\mathcal{D}$ is a complete function $\mathcal{V} \rightarrow \mathcal{D}$. A valuation σ' is a x-variant of a valuation σ if σ and σ' are identical except on x.*

Let t be a term and $\mathcal{V}(t)$ the set of variables in t, $\chi(t)$ is the term t in which each x_i in $\mathcal{V}(t)$ has been replaced by $\chi(x_i)$.

[2] Notice that varying domain can be useful. For instance in the study of doxastic first-order modal logic, an agent can learn the existence of a particular object, or a new object can appear.

[3] The main problem is to be able to characterize the meaning of a formula such as $OF(c)$ where c is a constant: does it mean that "it is obligatory that the object represented by c *in the current world* has F property" or "it is obligatory that *in each world*, the object represented by c has the F property".

The satisfiability relation $\models$ is defined as follows:

Definition 7. *Let $\mathcal{M} = \langle \mathcal{W}, \mathcal{R}_O, \mathcal{D}, \mathcal{I} \rangle$ a FOSDL model, w a world of $\mathcal{W}$ and σ a valuation on $\mathcal{D}$. Then:*

- *if P is a n-ary predicate symbol and $t_1, \ldots, t_n$ are terms, then $\mathcal{M}, w \models_\sigma P(t1, \ldots, t_n)$ iff $\langle \mathcal{I}(\sigma(t_1), w), \ldots, \mathcal{I}(\sigma(t_n), w) \rangle \in \mathcal{I}(P, w)$.*
- *if ψ is a FOSDL formula, then $\mathcal{M}, w \models_\sigma \neg\psi$ iff $\mathcal{M}, w \not\models_\sigma \psi$.*
- *if ψ_1 and ψ_2 are FOSDL formula, then $\mathcal{M}, w \models_\sigma \psi_1 \vee \psi_2$ iff $\mathcal{M}, w \models_\sigma \psi_1$ or $\mathcal{M}, w \models_\sigma \psi_2$.*
- *if $O\varphi$ is a FOSDL formula, $\mathcal{M}, w \models_\sigma O\varphi$ iff for every $v \in \mathcal{W}$ such that $w\mathcal{R}_O v$ holds, $\mathcal{M}, v \models_\sigma \varphi$.*
- *if ψ is a FOSDL formula, $\mathcal{M}, w \models_\sigma \forall x\ \psi$ iff for all valuations σ' x-variant of σ, $\mathcal{M}, w \models_{\sigma'} \psi$.*

Let ψ be a FOSDL formula. If for all valuations σ $\mathcal{M}, w \models_\sigma \psi$, we will note $\mathcal{M}, w \models \psi$. If $\mathcal{M}, w \models \psi$ for all w in $\mathcal{W}$, we will note $\mathcal{M} \models \psi$. Finally, if $\mathcal{M} \models \psi$ for all FOSDL models $\mathcal{M}$, then we will note $\models \psi$.

2.3 Axiomatics

We will now define an axiom system for FOSDL following the approach presented in[12]. In the following, $\varphi(x)$ denotes a formula in which the variable x may have free occurrences. We will say that a free variable y is *substitutable* for x in $\varphi(x)$ if no free occurrence of x in $\varphi(x)$ is in the scope of $\forall y$ in $\varphi(x)$.

Definition 8 (Axioms). *The formulae of the following forms are axioms:*
(Taut) *all classical FOL tautologies*
(KO) $O(\varphi \to \psi) \to (O\varphi \to O\psi)$
(DO) $O\varphi \to \neg O \neg\varphi$
(Bar1) $O(\forall x\ \varphi) \to \forall x\ O\varphi$
(Bar2) $\forall x\ O\varphi \to O(\forall x\ \varphi)$

Definition 9 (Inference Rules).
(MP) $\dfrac{\varphi \quad \varphi \to \psi}{\psi}$
(Gen) $\dfrac{\varphi}{\forall x\ \varphi}$
(NO) $\dfrac{\varphi}{O\varphi}$

Proposition 1 (Validity and soundness). *The previous system is valid and sound w.r.t. FOSDL semantics.*

The proof is given in [12].

We will define a *proof* of φ from the set of formulae Σ, noted $\Sigma \vdash \varphi$, as a sequence of formulae such that each one of them is an axiom, a formula of Σ, or produced by the application of an inference rule on previous formula.

In the following, $\perp$ will denote every formula that is a contradiction and $\top$ will denote every formula that is a tautology.

2.4 Regulation and integrity constraints modelling

In this section we define the notion of regulation and integrity constraints. First, we define the notion of rule, which is the basic component of a regulation. In this definition, rules have a general form, in particular they can be conditional.

Definition 10. *A rule is a formula of FOSDL of the form* $\forall \overrightarrow{x}\ l_1 \vee \ldots \vee l_n$ *with* $n \geq 1$ *such that:*

1. l_n *is of the form* $O\varphi$ *or* $\neg O\varphi$ *where* φ *is an objective literal*
2. $\forall i \in \{1, \ldots, n-1\}$, l_i *is an objective literal or the negation of an objective literal*
3. *if* x *is a variable in* l_n, *then* $\exists i \in \{1, \ldots, n-1\}$ *such that* l_i *is a negative literal and contains the variable* x
4. $\forall \overrightarrow{x}$ *denotes* $\forall x_1 \ldots \forall x_m$ *where* $\{x_1, \ldots, x_m\}$ *is the set of free variables appearing in* $l_1 \wedge \ldots \wedge l_{n-1}$.

In this definition, constraints (1) and (2) allow rules to be conditionals of the form "if such a condition is true then something is obligatory (resp. permitted or forbidden)". Constraint (3) restricts rules to range-restricted formulae[4]. Finally, rules are *sentences*, i.e. closed formulae, as expressed by constraint (4).

Notice also that we restrict in the definition of rules the formulae that can be defined as obligatory in the regulation: only objective literals can be obligatory or not obligatory.

We will write $\forall \overrightarrow{x}\ l_1 \vee \ldots \vee l_{n-1} \vee P\varphi$ as a shortcut for the two rules $\{\forall \overrightarrow{x}\ l_1 \vee \ldots \vee l_{n-1} \vee \neg O\varphi, \forall \overrightarrow{x}\ l_1 \vee \ldots \vee l_{n-1} \vee \neg O\neg\varphi\}$.

Definition 11. *A regulation is a set of rules.*

Let us consider an example which will help us to illustrate our purpose all along section 2 and 3.

Example 1 *We consider a regulation which rules the behaviour of a driver in front of a traffic light.*

The language needed is defined as follows:

- *green, orange, red, car, truck, bike,* A *and* T *are* 0*-arity functions, i.e. constants.*
- x, y, z, i *and* t *are variables.*
- $D(.)$ *is a predicate symbol that indicates that a term is a driver.*
- $TL(.)$ *is a predicate symbol that indicates that a term is a traffic light.*
- $C(.,.)$ *is predicate symbol that takes for parameters a traffic light and a color and indicates the traffic light color.*

[4] Range-restricted formulae are a decidable subset of domain-independent formulae which have been proved to be the only first order formulae having a meaning in modelling [9]. Notice in particular that by definition of FOSDL language, all variables appearing in l_n are free in l_n.

– $V(.,.)$ *is a predicate symbol that takes for parameters a driver and the type of vehicle he drives.*
– $IFO(.,.)$ *is a predicate symbol that takes for parameters a driver and a traffic light and indicates that the vehicle driven by the driver is in front of the traffic light.*
– $Stop(.,.)$ *is a predicate symbol that takes a driver agent and a traffic light for parameters and that indicates that this agent stops in front of the traffic light.*

Let's now take the three rules (r_0)*: "When a car-driver is in front of a traffic light that is red, he has to stop"* (r_1)*: "When a car-driver is in front of a traffic light that is orange, it is permitted for him to stop"* (r_2)*: "When a car-driver is in front of a traffic light that is green, he must not stop". These rules can be modelled by :*

$(r_0)\forall x \forall t\ D(x) \wedge TL(t) \wedge V(x, car) \wedge C(t, red) \wedge IFO(x,t) \rightarrow OStop(x,t)$
$(r_1)\forall x \forall t\ D(x) \wedge TL(t) \wedge V(x, car) \wedge C(t, orange) \wedge IFO(x,t) \rightarrow PStop(x,t)$
$(r_2)\forall x \forall t\ D(x) \wedge TL(t) \wedge V(x, car) \wedge C(t, green) \wedge IFO(x,t) \rightarrow FStop(x,t)$

2.5 Consistency of regulations

We now define a first notion for regulations, *consistency*. Intuitively, we will say that a regulation is consistent iff we cannot derive from the regulation using the system defined in 2.3 inconsistencies like $OStop(x,t) \wedge FStop(x,t)$. Consistency of a regulation is evaluated under *integrity constraints*, i.e. a set of closed objective formulae which can represent for instance physical constraints or domain constraints. In the following, we will note such an integrity constraints set IC.

First, we will define consistency of a regulation in a particular *state of the world*. Intuitively, states of the world are syntactic representations of classical first-order interpretations. They can also be assimilated to classical Herbrand models.

Definition 12 (state of the world). *A state of the world s is a complete and consistent set of objective ground literals.*

A state of the world is a syntactical representation of a Herbrand interpretation. Thus, for any n-ary predicate symbol P, any ground terms $t_1, \ldots, t_n$ and any state of the world s, either $P(t_1, \ldots, t_n) \in s$ or $\neg P(t_1, \ldots, t_n) \in s$. In the following, when describing a state of the world, we will omit the negative literals for readability.

Definition 13. *Let IC be a set of integrity constraints and s a state of the world. s is* consistent *with IC iff $s, IC \not\vdash \bot$.*

Definition 14. *Let ρ be a regulation, IC a set of integrity constraints and s a state of the world consistent with IC. ρ is consistent according to IC in s iff $\rho, IC, s \not\vdash \bot$.*

Example 2 *Let us resume example 1. Let us consider that IC contains two constraints: (1) a traffic light has a unique color and this color can be green, orange or red, and (2) a driver drives one and only one type of vehicle. Thus* $IC = \{\forall t\ TL(t) \rightarrow C(t, green) \otimes C(t, orange) \otimes C(t, red), \forall x \forall y \forall z\ D(x) \wedge V(x,y) \wedge V(x,z) \rightarrow y = z\}$[5].

Let s be the state of the world $\{D(A),\ TL(T), IFO(A,T),\ V(A, car),\ C(T, red)\}$.

First, s is such that $s, IC \not\vdash \perp$. Let us consider a regulation ρ that contains the three rules (r_0), (r_1) and (r_2). In this case, $\rho, IC, s \not\vdash \perp$ (because the only deontic literal that can be deduced from ρ, IC and s is $OStop(A,T)$). Thus, ρ is consistent according to IC in s.

Definition 15 (consistency of a regulation). *Let ρ be a regulation and IC a set of integrity constraints. ρ is consistent according to IC iff for all states of the world s such that $s, IC \not\vdash \perp$ then $\rho, IC, s \not\vdash \perp$.*

2.6 Completeness of regulations

Informally, a regulation is totally complete as soon as it prescribes the behaviour any agent should have in any situation. We can wonder if this definition really makes sense: can or must a regulation take into account all possible situations? Thus, we suggest to define a partial completeness restricted to two ground formulae φ and ψ: φ represents a particular situation in which we want to evaluate the regulation and ψ a predicate ruled by the regulation. Thus, we want a regulation be complete for φ and ψ iff in any situation where φ is true, it is obligatory (resp. permitted, forbidden) that ψ.

This leads to the following definition:

Definition 16. *Let IC be a set of integrity constraints, ρ be a regulation consistent according to IC and s a state of the world consistent with IC. Let $\varphi(\overrightarrow{x})$ and $\psi(\overrightarrow{x})$ two objective formulae, $\overrightarrow{x}$ representing free variables in φ and $\psi(\overrightarrow{x})$ meaning that the free variables in ψ are a subset of $\overrightarrow{x}$. ρ is $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$-*complete *according to IC in s for $\vdash$ iff for all ground substitutions χ such that $s \vdash \varphi(\chi(\overrightarrow{x}))$:*

$$\rho, s \vdash O\psi(\chi(\overrightarrow{x}))\ or$$
$$\rho, s \vdash F\psi(\chi(\overrightarrow{x}))\ or$$
$$\rho, s \vdash P\psi(\chi(\overrightarrow{x}))$$

Example 3 *Let us consider the state of the world $s_0 = \{D(A),\ TL(T), IFO(A,T),\ V(A, Car), C(T, red)\}$. Consider ρ and IC defined in example 2. s_0 is consistent with IC and $\rho, s \vdash O(Stop(A,T))$. Let's take $\varphi_0(x,t) \equiv TL(t) \wedge D(x) \wedge IFO(x,t)$ and $\psi_0(x,t) \equiv Stop(x,t)$. $s_0, IC \vdash IFO(A,T)$ and $\rho, IC, s_0 \vdash O(Stop(A,T))$. Thus, ρ is $(\varphi_0(x,t), \psi_0(x,t))$-complete according to IC in s_0 for $\vdash$.*

Let us now consider the state of the world $s_1 = \{D(A),\ TL(T), IFO(A,T),\ V(A, Truck),\ C(T, red)\}$. s_1 is consistent with IC. $s_1, IC \vdash IFO(A,T)$ but $\rho, IC, s_1 \not\vdash$

[5] The introduction of equality is done in the same way as in [12].

$O\psi_0(A,T)$, $\rho, IC, s_1 \not\vdash P\psi_0(A,T)$ and $\rho, IC, s_1 \not\vdash F\psi_0(A,T)$. Thus, ρ is $(\varphi_0(x,t), \psi_0(x,t))$-incomplete according to IC in s_1 for $\vdash$. In fact, no rule of the regulation can be applied as the vehicle is not a car but a truck.

The previous definition can be generalized as follows:

Definition 17 (completeness of a regulation). *Let IC a set of integrity constraints and ρ be a regulation. Let $\varphi(\overrightarrow{x})$ and $\psi(\overrightarrow{x})$ be two objective formula with the same meaning as in definition 16. ρ is $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$-*complete *according to IC for $\vdash$ iff for every state of the world s consistent with IC, ρ is $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$-complete according to IC in s for $\vdash$.*

Completeness is an important issue for a regulation. For a given situation, without any behaviour stipulated, any behaviour could be observed and thus consequences could be quite important. With an incomplete regulation, we could (1) detect the "holes" of the regulation and send them back to the regulation designers so that they can correct them or (2) detect the "holes" of the regulation and apply on those holes some completion rules to correct them. The first solution could be quite irksome to be applied (the number of holes could be quite important and thus correct them one by one quite long). Therefore, we put in place the second solution.

3 Reasoning with incomplete regulations

3.1 Defaults for completing regulation

Reasoning with incomplete information is a classical problem in logic and artificial intelligence: can we infer something about an information that is not present in a belief base? Several approaches have been defined, but we are here interested in one: default reasoning. The principle of default reasoning is quite simple: if an information is not contradictory with the informations that can be classically deduced from the belief base, then we can deduced another information from the belief base. A classical example is the following: let us suppose that we believe that "*every bird flies*", that "*penguins do not fly*" and "*penguins are birds*". Of course, the representation of this set of formulae in FOL is inconsistent (a bird which is also a penguin flies and do not fly at the same time). In fact, the first rule "*every bird flies*" is a default: "*if a is bird and it is not inconsistent with the belief base that a flies, then a flies*"[6]. If a is a penguin, then "a flies" cannot be deduced, and it cannot be deduced that a is a penguin, then we can deduce that a flies.

Default logic is a non-monotonous extension of first-order logic introduced by Reiter [17] in order to formalize default reasoning. We will here follow the presentation of Besnard given in [1].

[6] Notice that in this case, the information that is not contradictory with the belief base and the information newly deduced are the same.

A default d is a configuration $\frac{P : J_1, \ldots, J_n}{C}$ where $P, J_1, \ldots, J_n, C$ are first-order closed sentences. P is called the *prerequisite* of d, $J_1, \ldots, J_n$ the *justification* of d and C the *consequence* of d. A default theory $\Delta = (D, F)$ is composed of a set of objective closed formulae F (facts) and a set of defaults.

A default theory (D, F) can be given in a *surface form* (D', F) on condition that
$D = \{\frac{P(\overrightarrow{a}) : J_1(\overrightarrow{a}), \ldots, J_n(\overrightarrow{a})}{C(\overrightarrow{a})} \ : \ \frac{P(\overrightarrow{x}) : J_1(\overrightarrow{x}), \ldots, J_n(\overrightarrow{x})}{C(\overrightarrow{x})} \in D'$ and $\overrightarrow{a}$ is a ground term$\}$

and every element of D' is of the form $\frac{P(\overrightarrow{x}) : J_1(\overrightarrow{x}), \ldots, J_n(\overrightarrow{x})}{C(\overrightarrow{x})}$ where $P(\overrightarrow{x}), J_1(\overrightarrow{x}), \ldots, J_n(\overrightarrow{x}), C(\overrightarrow{x})$ are first-order sentences with free variables occurring in $\overrightarrow{x}$.

Using defaults we obtain *extensions*, i.e. sets of formulae that are deduced monotonically and non-monotonically from F. Let $\Delta = (D, F)$ be a default theory where defaults contains only closed formulae, then a extension of Δ is a set of formulae E verifying the following conditions:

1. $F \subseteq E$
2. $Th(E) = E$ where $Th(E) = \{\varphi \ : \ E \vdash \varphi\}$
3. if $\frac{P : J_1, \ldots, J_n}{C}$ is a default of D, then if $P \in E$ and J_1 is consistent with E, ..., J_n is consistent with E, then $C \in E$

Default theories can have many extensions or no extensions at all. Reiter showed in [17] that if F is consistent and if (D, F) has an extension, then this extension is consistent. He showed also that any normal and closed default theory has at least one extension.

Here, we are not interested in the fact that a given objective formula ψ is believed but in the fact that a given regulation deduces that it is obligatory, forbidden or tolerated (those cases are the only ones due to the D axiom of O). Thus, if the regulation is incomplete for an objective formula ψ (i.e. it does not deduce neither $O\psi$ nor $F\psi$ nor $P\psi$), then it can only be completed by assuming that $O\psi$ can be deduced, or $P\psi$, or $F\psi$. This leads to the three sets of defaults which are described in the following.

In the following, let IC be a set of integrity constraints, ρ be a consistent regulation according to IC and s be a state of the world consistent with IC. Let $\varphi(\overrightarrow{x})$ and $\psi(\overrightarrow{x})$ be two objective formulae verifying definition 16.

Definition 18. *Let $E_F(\overrightarrow{x})$, $E_P(\overrightarrow{x})$ and $E_O(\overrightarrow{x})$ be three objective formulae such that their respective set of free variables is in $\overrightarrow{x}$. We define a set of configuration as follows:*

$(DF_{\varphi,\psi})$ $\frac{\varphi(\overrightarrow{x}) \wedge E_F(\overrightarrow{x}) \ : \ F\psi(\overrightarrow{x})}{F\psi(\overrightarrow{x})}$

$(DP_{\varphi,\psi})$ $\frac{\varphi(\overrightarrow{x}) \wedge E_P(\overrightarrow{x}) \ : \ P\psi(\overrightarrow{x})}{P\psi(\overrightarrow{x})}$

$(DO_{\varphi,\psi})$ $\frac{\varphi(\overrightarrow{x}) \wedge E_O(\overrightarrow{x}) \ : \ O\psi(\overrightarrow{x})}{O\psi(\overrightarrow{x})}$

A $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$*-completeness default theory for* ρ *and* s *is a default theory* $\Delta_{\rho,s}(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$ *whose surface form is given by* $(\{DF_{\varphi,\psi}, DP_{\varphi,\psi}, DO_{\varphi,\psi}\}, \rho \cup s)$

We can complete an incomplete regulation so that $\psi(\overrightarrow{x})$ is forbidden ($DF_{\varphi,\psi}$), permitted ($DP_{\varphi,\psi}$) or obligatory ($DO_{\varphi,\psi}$) depending on $E_F(\overrightarrow{x})$, $E_P(\overrightarrow{x})$ and $E_O(\overrightarrow{x})$. Following Reiter, we define a new inference relation $\vdash_*$ defined as follows:

Definition 19. *Let* γ *be a formula of FOSDL.* $\rho, s \vdash_* \gamma$ *iff there is an extension* E_γ *of* $\Delta_{\rho,s}(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$ *such that* $\gamma \in E_\gamma$.

Moreover, we will note $Th_*(E) = \{\varphi \ : \ E \vdash_* \varphi \text{ and } \varphi \text{ is closed}\}$.

Notice that we define here what Reiter calls an existential inference. There are of course other sorts of inference, for instance universal, but as we will show in section 3.2 we will obtain only one extension in the cases we are interested in, so the different kinds of inference are identical.

The next step is to define the conditions under which the regulation is complete and consistent with this new inference. This will be addressed in the next section.

3.2 Consistency and completeness of the completed regulation

First, we extend the definitions 15, 16 and 17 by using $\vdash_*$ instead of $\vdash$ in those definitions. To distinguish the new notions of consistency and completeness from the old ones, we will use $*$ as a prefix (for instance we will write "$*$-consistency") or write explicitly "for $\vdash_*$" (for instance, we will write "consistent for $\vdash_*$").

The main result about completeness and consistency of the regulation obtained by using the default theory defined previously is expressed by the following proposition.

Proposition 2. *Let us consider a set of integrity constraints* IC*, a regulation* ρ *consistent according to* IC *and a state of the world* s *consistent with* IC *and such that* $\rho \cup s$ *is consistent. Let* $\varphi(\overrightarrow{x})$ *and* $\psi(\overrightarrow{x})$ *be two objective formulae verifying definition 16 and* $\Delta_{\rho,s}(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$ *the corresponding default theory.*

The following propositions are equivalent:

1. *for every vector* $\overrightarrow{a}$ *of ground terms, if* $s \vdash \varphi(\overrightarrow{a})$, $\rho, s \not\vdash O\psi(\overrightarrow{a})$, $\rho, s \not\vdash P\psi(\overrightarrow{a})$ *and* $\rho, s \not\vdash F\psi(\overrightarrow{a})$ *(i.e.* ρ *is not* $(\varphi(\overrightarrow{a}), \psi(\overrightarrow{a}))$*-complete in* s*), then* $s \vdash E_F(\overrightarrow{a}) \otimes E_P(\overrightarrow{a}) \otimes E_F(\overrightarrow{a})$.
2. ρ *is consistent and* $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$*-complete for* $\vdash_*$ *in* s.

This proposition characterizes necessary and sufficient conditions for the defaults to consistently complete an incomplete regulation. More precisely, this proposition says that if every time the regulation does not prescribe a behaviour one and only one E_i is true, then the defaults consistently complete the regulation (because one and only one default is applied for a particular $\psi(\overrightarrow{a})$).

Example 4 *Consider the state of the world* $s_1 = \{D(A), TL(T), IFO(A,T), V(A, truck), C(T, red)\}$ *from the last example.* ρ *is incomplete in* s_1 *for* $\varphi_0(x,t) \equiv D(A) \wedge TL(T) \wedge IFO(A,T)$ *and* $\psi_0(x,t) \equiv Stop(A,T)$ *in* s_1.

Let's take $E_F(x,t) = V(x, truck) \wedge C(t, green)$, $E_P(x,t) = V(x, truck) \wedge C(t, orange)$ *and* $E_O(x,t) = V(x, truck) \wedge C(t, red)$*, then* $s_1 \vdash E_O(A,T)$*. Thus,* ρ *is consistent and* $(\varphi_0(x,t), \psi_0(x,t))$*-complete for* $\vdash_*$ *in* s_1.

Even if this necessary and sufficient condition is interesting in theory, it is not really useful for practical purposes. In fact, to verify that this condition is satisfied, we would have to detect every "hole" in the regulation. This detection is an operation we want to avoid. Thus, we try to find more general conditions that are still sufficient but not necessary for the completion rules to consistently complete the regulation. We present two immediate corollaries of the previous definition.

Corollary 1. *If* $s \vdash \forall \overrightarrow{x}\ \varphi(\overrightarrow{x}) \rightarrow E_O(\overrightarrow{x}) \otimes E_F(\overrightarrow{x}) \otimes E_P(\overrightarrow{x})$ *then* ρ *is consistent and* $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$*-complete according to IC for* $\vdash_*$ *in* s.

Example 5 *Consider the state of the world* $s_2 = \{D(A), TL(T), IFO(A,T), V(A, bike), C(T, red)\}$. s_2 *is consistent with* IC*. Consider the regulation defined in example 1.*

This time, let us consider $E_F(x,t) = C(t, green)$, $E_P(x,t) = C(t, orange)$ *and* $E_O(x,t) = C(t, red)$. $s_2 \vdash E_O(A,T)$*. Thus,* ρ *is* $*$*-consistent and* $*$*-complete for* $\varphi_0(x,t)$ *and* $\psi_0(x,t)$ *in* s_2*. But we also have* $s_1 \vdash E_O(A,T)$*, so* ρ *is* $*$*-consistent and* $(\varphi_0(x,t), \psi(x,t))$*-complete for* $\vdash_*$ *in* s_1*. Those more general* E_i *allow us to have a regulation complete for any type of vehicle.*

Corollary 2. *If* $IC \vdash \forall \overrightarrow{x}\ E_O(\overrightarrow{x}) \otimes E_F(\overrightarrow{x}) \otimes E_P(\overrightarrow{x})$ *then* ρ *is consistent and* $(\varphi(\overrightarrow{x}), \psi(\overrightarrow{x}))$*-complete according to IC for* $\vdash_*$.

Example 6 $IC \vdash \forall t\ C(t, red) \otimes C(t, green) \otimes C(t, orange)$*. Thus* ρ *is* $*$*-consistent and* $(\varphi_0(x,t), \psi_0(x,t))$*-complete for* $\vdash_*$.

IC specifies that a traffic light has one and only one color among three colors Red, Orange and Green. If there is one E_i *for each color, we are sure that whatever the situation is, we can apply one and only one default if there is a "hole" in the regulation.*

Another alternative would be to take fixed E_i. For example, we could take one E_i equal to $\top$ and the two others to $\bot$. We have three cases:

- suppose that $E_F \equiv \top$, $E_P \equiv \bot$ and $E_O \equiv \bot$. In this case, according to completion rules, everything that is not specified as obligatory or permitted by the regulation is forbidden. This strict behaviour could be observed for regulations that rule a highly secured system where each action has to be explicitly authorized before being performed.

– suppose that $E_F \equiv \bot$, $E_P \equiv \top$ and $E_O \equiv \bot$. We are here in the opposite situation, meaning that everything that is not obligatory or forbidden is permitted. This "tolerant" behaviour could be observed for regulations for dimmed secured systems where everything that is not forbidden or obligatory is implicitly permitted.
– suppose that $E_F \equiv \bot$, $E_P \equiv \bot$ and $E_O \equiv \top$. In this case, every action that is not forbidden or permitted has to be performed.

4 Examples of regulations: information exchange policies

An information exchange policy is a regulation which prescribes the behaviour of agents in a multiagent system regarding information communication. To describe such policies, we need five predicate symbols: *Agent*, *Info*, *Receive*, *Topic* and *Tell*. $Agent(x)$ means that x is an agent, $Info(i)$ means that i is an information, $Receive(x, i)$ means that agent x receives information i. $Topic(i, t)$ means that information i deals with topic t. $Tell(x, i, y)$ means that agent x tells agent y an information i. We also define constants a, b, i_1, *EqtCheck*, *ExpRisk*, *Meeting* and *EqtOutOfOrder*.

The consistency of such policies is defined by definition 14. The completeness of such policies is defined by instantiated definition 16 with the following specific formula:

$$\begin{aligned}\varphi(x,i,y) \equiv & Agent(x) \land Info(i) \land Receive(x,i) \land \\ & Agent(y) \land \neg(x = y) \\ \psi(x,i,y) \equiv & Tell(x,i,y)\end{aligned}$$

This leads to the following definition:

Definition 20. *Let IC a set of integrity constraints, s a state of the world consistent with IC and ρ a regulation consistent in s according to IC. ρ is complete according to IC in s for $\vdash$ iff for all ground substitution χ such that $s \vdash Agent(\chi(x)) \land Info(\chi(y)) \land Receive(\chi(x), \chi(i)) \land Agent(\chi(y)) \land \neg(\chi(x) = \chi(y))$:*

$$\begin{aligned}&\rho, s \vdash O\,Tell(\chi(x), \chi(i), \chi(y)) \text{ or} \\ &\rho, s \vdash F\,Tell(\chi(x), \chi(i), \chi(y)) \text{ or} \\ &\rho, s \vdash P\,Tell(\chi(x), \chi(i), \chi(y))\end{aligned}$$

Thus, default are the following:

$(DF_{\varphi,\psi})$ $\dfrac{\varphi(x,i,y) \land E_F(x,i,y) : F\,Tell(x,i,y)}{F\,Tell(x,i,y)}$

$(DP_{\varphi,\psi})$ $\dfrac{\varphi(x,i,y) \land E_P(x,i,y) : P\,Tell(x,i,y)}{P\,Tell(x,i,y)}$

$(DO_{\varphi,\psi})$ $\dfrac{\varphi(x,i,y) \land E_O(,i,y) : O\,Tell(x,i,y)}{O\,Tell(x,i,y)}$

Results proved in section 3 remain valid. In particular, we still have the three cases:

- $E_F \equiv \top$, $E_P \equiv \bot$ and $E_O \equiv \bot$.
  This applies to highly secured multiagent systems in which any communication action should be explicitly obligatory or permitted before being performed.
- $E_F \equiv \bot$, $E_P \equiv \top$ and $E_O \equiv \bot$.
  This case applies to lowly secured systems in which any communication action which is not explicitly forbidden is permitted.
- $E_F \equiv \bot$, $E_P \equiv \bot$ and $E_O \equiv \top$.
  In this case, unless explicit mentioned, sending information is obligatory.

In order to illustrate this, consider the example of a firm in which there is a manager and two employees. Consider a policy π_0 with only one rule which states that "*Managers are required not to inform their employees about any equipment checking information*". The rule is modelled by[7]

$\forall x \forall i \forall y\ Manager(x) \wedge Employee(y) \wedge Receive(x,i) \wedge Topic(i, EqtChk) \rightarrow O\neg Tell(x,i,y)$

Let us consider $IC = \emptyset$ (there is no integrity constraints) and the state of the world $s_0 = \{Agent(a), Agent(b), Manager(a), Employee(b), Info(i_1), Topic(i_1, ExpRisk), Receive(a, i_1)\}$. In this situation, a is a manager and b an employee. a has received information i_1 whose topic is "Explosion Risk".

As π_0 contains only one rule and s_0 is consistent with IC, π_0 is consistent in s_0.

However we have $s_0 \vdash Agent(a) \wedge Info(i_1) \wedge Receive(a, i_1) \wedge Agent(b) \wedge \neg(a = b)$ but $\pi_0, s_0 \not\vdash O(Tell(a, i_1, b))$ and $\pi_0, s_0 \not\vdash P(Tell(a, i_1, b))$ and $\pi_0, s_0 \not\vdash F(Tell(a, i_1, b))$. Thus, π_0 is incomplete for $\vdash$.

Incompleteness comes from the fact that the policy prescribes the behaviour of the manager if he/she receives an information about "Equipment Verification" but it does not prescribe anything as for information about "Explosion Risk". The policy does not state what the manager should do when he/she receives information about "Risk Explosion".

In order to complete the previous policy, we could take:

$E_F(x,y,i) = Topic(i, EqtChk)$, $E_P(x,y,i) = \bot$ and $E_O(x,y,i) = Topic(i, ExpRisk)$. Such a choice forces the manager to tell its employees about "Risk Explosion" information. We can verify that π_0 is complete and consistent for $\vdash_*$ in s_0 for $\varphi(x,i,y)$ and $\psi(x,i,y)$.

Let consider now that IC contains the constraint "*An information has one and only one topic and this topic can be EqtChk, ExpRisk, Meeting or EqtOutOfOrder*". Take:

$$
\begin{aligned}
E_F(x,y,i) \equiv\ & Topic(i, EqtChk) \vee \\
& Topic(i, Meeting) \\
E_P(x,y,i) \equiv\ & Topic(i, EqtOutOfOrder) \\
E_O(x,y,i) \equiv\ & Topic(i, ExpRisk)
\end{aligned}
$$

We can apply the corollary 2 to conclude that π_0 is $*$-complete and $*$-consistent for $\varphi(x,i,y)$ and $\psi(x,i,y)$.

[7] The predicate names are obvious thus we do not formally define the language.

5 Conclusion

In this paper, we addressed the problem of analysing consistency and completeness of regulations which may exist in a society of agents in order to rule their behaviour.

More specifically, we have defined a modal logical framework and showed how to express a regulation within this framework. We then have reminded of a definition of consistency and we have defined what meant completeness for a regulation. The definition of completeness we gave is rather general. We also dealt with incomplete regulations and proposed a way for completing them by using defaults. We have established several results which show when these defaults consistently complete a regulation.

Although these notions (except defaults) were present in [6, 7], we have extended these previous papers in two points:

– first, we use a first-order modal logic to represent regulations. This allows us to clearly distinguish between the properties with which the deontic notions deal from the deontic notions and we keep the expressiveness of FOL for objects properties.
– second, the approach taken in the previous papers to complete a regulation was to extend the CWA (Closed World Assumption) defined by Reiter in order to complete first-order databases [16]. We choose here to use default reasoning, which is a more elegant solution to complete regulations.

The notion of completeness developed here is in fact a kind of local completeness, in the sense that we require to have $O(\psi(\overrightarrow{x}))$, $P(\psi(\overrightarrow{x}))$ or $F(\psi(\overrightarrow{x}))$ only for a specific context represented by formula $\varphi(\overrightarrow{x})$. That looks close to the notion of completeness introduced in the databases domain by [18, 10], who noticed that some of the integrity constraints that are expressed on a database are rules about what the database should know (i.e. these are rules about what should be deduced in the database). For instance, the integrity constraint expressing that "any employee has got a phone number, a fax number or a mail address" expresses in fact that, for any employee known by the database, the database knows its phone number, its fax number or its mail address[8]. As first mentioned by Reiter [18], this integrity constraint expresses a kind of local completeness of the database. Reiter's defaults can be used in order to complete such a database in case of incompleteness. For instance, one of the rules can be that if the database does contain any required information (no phone number, no fax number, no mail address) for a given employee but if the department that employee works in is known, then it can be assumed that its phone number is the phone number of its department.

Studying the formal link between the notion of completeness introduced in this paper and that notion of local completeness constitutes one interesting extension of this work.

[8] Notice that this does not prevent the fact that in the real world, an employee of the company has no telephone number, no fax number and no mail address

Furthermore, in order to deal with more general regulations, this present work must be extended. In particular, we have to extend it by considering more notions, among them time and action. Indeed, as it is shown in [11], the issue of time is very important when speaking about obligations and we will have to consider different types of time among which, at least, the time of validity of the norms and the deadlines beared on the obligations. Notice also for instance that in most of the examples of this paper, the predicates concerned by deontic operators represent actions (tell, stop etc.). The adding of a dynamic modal operator and/or temporal operator may be interesting. We will thus obtain a multimodal logic with strong expressiveness.

Finally, we developed a really simple model of the deontic notions by using SDL and lots of classical problem in deontic logic are not handled here: norms with exceptions, contrary-to-duties, collective obligations etc. Another extension of this work will be to define a logic that can deal with these problems.

References

1. P. Besnard. *An introduction to default logic*. Springer-Verlag, 1989.
2. P. Bieber and F. Cuppens. Expression of confidentiality policies with deontic logic. In *Deontic logic in computer science: normative system specification*, pages 103–121. John Wiley and Sons, 1993.
3. B. F. Chellas. *Modal logic, an introduction*. Cambridge University Press, 1980.
4. L. Cholvy. Checking regulation consistency by using SOL-resolution. In *International Conference on Artificial Intelligence and Law*, pages 73–79, 1999.
5. L. Cholvy and F. Cuppens. Analyzing consistency of security policies. In *1997 IEEE Symposium on Security and Privacy*, pages 103–112. IEEE, 1997.
6. L. Cholvy and S. Roussel. Reasoning with incomplete information exchange policies. In K. Mellouli, editor, *Symbolic and Quantitative Approaches to Reasoning with Uncertainty, 9th European Conference, ECSQARU'07*, number 4724 in Lecture Notes in Articial Intelligence, pages 683–694. Springer-Verlag, 2007.
7. L. Cholvy and S. Roussel. Consistency and completeness of regulations. In *Proceedings of he third International Workshop on Normative Multiagent Systems (NORMAS'08)*, pages 51–65, 2008.
8. F. Cuppens and R. Demolombe. A modal logical framework for security policies. In *Lectures Notes in Artificial Intelligence*, volume 1325, page 1997. Springer, 1997.
9. R. Demolombe. Syntactical characterization of a subset of domain independent formulas. *Journal of the Association for Computer Machinery*, 39(1):71–94, 1982.
10. R. Demolombe. Database validity and completeness: another approach and its formalisation in modal logic. In Enrico Franconi and Michael Kifer, editors, *Proc. of the 6th International Workshop on Knowledge Representation meets Databases (KRDB'99)*, pages 11–13. CEUR-WS.org, 1999.
11. R. Demolombe, P. Bretier, and V. Louis. Norms with deadlines in dynamic deontic logic. In Gerhard Brewka, Silvia Coradeschi, Anna Perini, and Paolo Traverso, editors, *Proceedings of ECAI 2006, 17th European Conference on Artificial Intelligence*, pages 751–752. IOS Press, 2006.
12. M. Fitting and R. L. Mendelsohn. *First-order modal logic*. Kluwer Academic, 1999.
13. C. Groulier. *Normes permissives et droit public*. PhD thesis, Université de Limoges, 2006. Available on `http://www.unilim.fr/scd/theses/accesdoc.html`. In French.

14. R. Hilpinen, editor. *Deontic logic*. Reidel Publishing Company, 1971.
15. P. Mc Namara. Deontic logic. Stanford Encyclopedia of Philosophy, http://plato.stanford.edu/entries/logic-deontic/.
16. R. Reiter. On closed world databases. In J. Minker J.-M. Nicolas H. Gallaire, editor, *Logic and Databases*. Plenum Publications, 1978.
17. R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1,2), 1980.
18. R. Reiter. What should a database know? *Journal of Logic Programming*, 14(1,2):127–153, 1992.
19. G. H. von Wright. Deontic logic. *Mind*, 60:1–15, 1951.
20. E. Vranes. The definition of "norm conflict" in international law and legal theory. *The European Journal of International Law*, 17(2):395–418, 2006.