Simulating the Law in a Multi-Agent System

Matteo Cristani[1,*,†], Francesco Olivieri[2], Guido Governatori[3] and Gabriele Buriola[1,†]

[1] *Dept. of Computer Science, University of Verona, Verona, Italy*
[2] *Independent researcher, Brisbane, Australia*
[3] *Artificial Intelligence and Cyber Futures Institute, Charles Sturt University, Bathurst, Australia*
[1] *Dept. of Computer Science, University of Verona, Verona, Italy*

Abstract

In this paper we define a Multiple Agent System able to simulate an artificial society to be paired with a normative background. The purpose of this architecture shall be the simulation of a law to devise its impact. We analise some existent architecture (GAMA) that has already been used for simulating MAS, with BDI agents. In fact, GAMA technology is insufficient to guarantee certain validity properties and actual computational effectiveness that could instead be provided if we manage the rule system to interpret Defeasible Deontic Logic, a logic framework that satisfies the aforementioned properties. As a first step of an experimental endeavour aiming at law simulation by design, we provide here a theoretical model of the MAS which simulates the society.

Keywords

Defeasible Deontic Logic, Multiple Agent Systems, Law Simulation

1. Introduction

Simulating collective behaviour is a challenging topic of the recent past. There is a rising demand that emerges from a variety of contexts, including the *production of legislation* at broad, for the simulation paradigm can be useful for determining the actual effects of introducing a new norm on a given society. Therefore, the *drafters*, namely those people who are in charge of producing a norm, either when designing a new norm for the general population, or when providing a norm background on the actual domain of a restricted society (a company, an association) or again to govern direct multiparty relationships (as in contracts) consider helpful to apply the norms into a *simulated society* for evaluating the *impact* the new norm has.

We then need a method to *describe* a society, in a way that generates a cycle of multiple agent system evolution steps able to support the impact evaluation we mentioned above.

This concept is part of a series of investigation that are conducted with the purpose of developing a complex system for law evaluation in diverse point of the production process: *by design* (as in the current investigation), after the enforcement, in the application phase. The

WOA 2024: 25th Workshop "From Objects to Agents", July 8-10, 2024, Forte di Bard (AO), Italy

[*] Corresponding author.

[†] These authors contributed equally.

matteo.cristani@univr.it (M. Cristani); francesco.olivieri.phd@gmail.com (F. Olivieri); ggovernatori@csu.edu.au (G. Governatori); gabriele.buriola@univr.it (G. Buriola)

0000-0001-5680-0080 (M. Cristani); 0000-0003-0838-9850 (F. Olivieri); 0000-0002-9878-2762 (G. Governatori); 0000-0002-1612-0985 (G. Buriola)

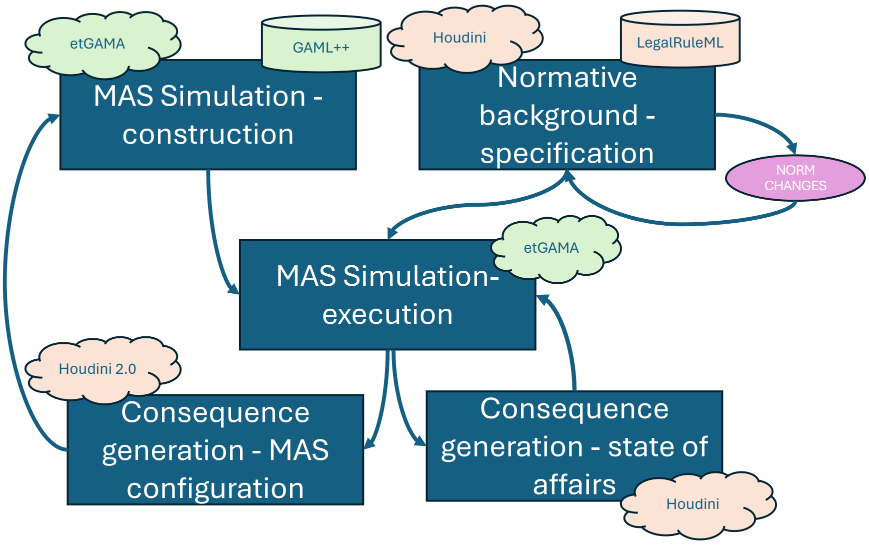
etGAMA
GAML++
MAS Simulation - construction
Houdini
LegalRuleML
Normative background - specification
NORM CHANGES
etGAMA
MAS Simulation- execution
Houdini 2.0
Consequence generation - MAS configuration
Consequence generation - state of affairs
Houdini

Figure 1: A general schema of the architecture we devise for the long-term aim investigation on law simulation.

schema of the application system is illustrated in Figure 1. There are essentially three phases of the application:

1. **MAS Simulation - construction:** a society is generated, possibly based on real-world data such as sociological evidence.

2. **Normative background - specification:** the existing normative background of the society simulated in Phase 1 is defined within an engine for Legal Reasoning, in particular, for the implementation of the aforementioned projects, we use the DDL reasoner *Houdini* [1, 2].

3. **MAS Simulation - execution:** the MAS system defined in Phase 1 is run, and it generates, technically, bunches of *facts* for a Deontic Defeasible Theory, implemented to devise the normative background, as described in Phase 2.

4. **Consequence generation - state of affairs:** the DDL system generates the simulated effects of the applications of the Normative Background towards the MAS system. This *changes* the current state of affairs of the MAS.

5. **Consequence generation - MAS configuration:** the MAS system employs a negative feedback *á la Rosenblueth, Wiener, Bigelow* to resettle the MAS while configuring again the parameters in two levels of feedback layers:

    - As a consequence of the *normative changes*, by modifying the behavioural parameters of the agents (for instance when an action which is permitted becomes forbidden the probability of individuals willing to do that action may decrease);

- As a consequence of the *application of the law* because some agents could have been punished, and therefore they may have been limited in their permits, or superimposed obligations and prohibitions they did not have before.

The above mentioned sequence is repeated in cycles with occasional events of actions performed in the MAS and rules changed in the normative background. Also we leave the evaluation part for further investigation.

A key aspect in modeling human behavior with respect to law compliance is given by the nondeterministic component of actions. The stochastic nature of reality shows up in at least two moments: the actual violation of a norm by a person and the chance that this violation is discovered and prosecuted by the authorities. The former phenomenon is formalized in the model we discuss here through an equation (1) pairing together the expected utility from the violation of a norm, compared with the one relative to the respect of that norm, and the tendency to be compliance with the law, encoded in the model discussed in this paper by a non-negative real number. The latter would call for a probabilistic application of DDL rules and, given its non trivial formalization, is postponed to a further and more specific paper.

The research plan is therefore as follows:

- Build the *theoretical model* that is documented in the current paper.
- Implement it within an actual simulation system. In this case we have chosen to employ *GAMA* [3, 4, 5] that has also been showed to adapt to the context of simulating ethically relevant behaviours [3].
- Develop an extension to the markup language GAML++ that expresses the aspects we discuss in this paper, and correspondingly extend GAMA to etGAMA in order to allow simulation of the law.
- Extend Houdini technology to allow *law change* specific management, as well as the utility functions we shall discuss in this paper.
- Build the whole system, where the communication components are managed via json/jquery components in order to align the current java implementation of both Gama and Houdini.

With this research plan in agenda, we now deal with the theoretical model, in the rest of this paper.

For what concerns the structure: Sec. 2 exposes the population model, in particular how it changes during time; Sec. 3 presents the salient characteristics which we focus on, such as age, gender and job; Sec. 4 is dedicated to model law compliance in this *in vitro* society; finally, Sec. 5 summarizes the whole paper.

2. Population model

In this section we provide the population model. First of all, for what concerns time, we adopt a discrete time T starting with $t = 0$ and indexed by natural numbers which may be interpreted as years. Let $\mathbb{P}$ be the countably infinite set of all the people who may, at some point, be part of the population model. At any moment, which in our discrete time means a specific year, the population of the model is given by a finite subset of $\mathbb{P}$; the function $\mathcal{P}\colon T \to \mathcal{P}(\mathbb{P})$[1] associates to every year t a finite subset of $\mathbb{P}$, the *current population* $\mathcal{P}(t)$ in that year denoted by $\mathcal{P}_t$. The starting population $\mathcal{P}_0$, as well as its characteristics, see Sec. 3, is given; whereas the population in $t+1$, i.e. $\mathcal{P}_{t+1}$, depends on four factors: births, deaths, immigration and emigration. Births are given by a birth function $B\colon T \setminus \{0\} \to \mathcal{P}(\mathbb{P})$ which, for every year t except the first one, selects the finite subset of the new born denoted by $\mathcal{P}_t^0$; the function B satisfies reasonable constraints related to births, in particular the following two:

- if $t \neq t'$, then $B(t) \cap B(t') = \emptyset$, i.e. every person can born at most one time;
- $|\mathcal{P}_{t+1}^0| = Br_t \cdot |\mathcal{P}_t^{adult}|$, where $\mathcal{P}_t^{adult}$ is the adult population at time t (see Sec. 3) and Br_t is the *birth rate* in the year t; namely, the number of new born depends on that one of adults via a coefficient.

Moreover, the birth function establishes the gender of the new born; namely there is a function $Bgen: \mathcal{P}_t^0 \to \{male, female\}$ assigning to each new born in $\mathcal{P}_t^0$ its gender, see below for more details.

Deaths are modeled simply by the process that every person lives exactly 80 years. Thus, if $\mathcal{P}_t^{80}$ denotes the subset of $\mathcal{P}_t$ of people in their eighties, then moving from $\mathcal{P}_t$ to $\mathcal{P}_{t+1}$ we simply remove $\mathcal{P}_t^{80}$; see later regarding how age is encoded in the model.

Immigration and emigration are treated similarly, namely we have two functions $Im\colon T \to \mathcal{P}(\mathbb{P})$ and $Em: T \to \mathcal{P}(\mathbb{P})$ selecting for every year t who is immigrating, Im_t, and who is emigrating, Em_t. As before we have some constraints for these functions, in particular:

- $Em_{t+1} \subseteq \mathcal{P}_t$, only people in the current population can emigrate the next year;
- $Im_t \cap Em_t = \emptyset$, a person can not immigrate and emigrate the same year.

Moreover, the immigrate function Im establishes also the age and the gender of the immigrates; namely, there are two functions $ImAge: Im_t \to \{1, \ldots, 80\}$ and $ImGen\colon Im_t \to \{male, female\}$ assigning to each person in Im_t its age and its gender, see below for more details.

All in all, with respect to T population satisfies the following equation:

$$\mathcal{P}_{t+1} = \left(\mathcal{P}_t \setminus \mathcal{P}_t^{80} \setminus Em_{t+1}\right) \cup \mathcal{P}_{t+1}^0 \cup Im_{t+1}.$$

3. Individual Characteristics

Every person $p \in \mathbb{P}$ has some *individual characteristics*, such as age, gender and so on. In this paper the following are considered:

[1] $\mathcal{P}(\mathbb{P})$ denotes the powerset of $\mathbb{P}$.

1. *Age*: this is a natural number between 1 and 80.
2. *Age status*: there are three age status, *young*, *adult*, *elderly* depending on the age.
3. *Gender*: this preliminary model has two genders *male* and *female* which are fixed from birth.
4. *Marital status*: there are three marital status, *bachelor*, *married* and *divorced.*[2]
5. *Job status*: there are four job status, *unemployed*, *public job*, *private job* and *retired.*
6. *Ethical tendency* (also called in places *inclination*): there are three tendencies, *legalist*, *neutral* and *opportunist*; related to tendency there is also a parameter α, see later for more detail.

We see these characteristics as sets, e.g. Age$= \{1, 2, \ldots, 80\}$ and Age status$= \{young, adult, elderly\}$, presenting them one by one after some general considerations.

Excluding the ethical tendency, devoted to modeling law compliance, the other status have been chosen since they represent three of the main social categories producing common and widespread rights and duties, e.g. age for penal responsibility, as well as three of the main characteristics used in demographic and social studies.

Except from gender, all the status may change during time, thus each status depend on both the person $p \in \mathcal{P}_t$ and the current year $t \in T$. Let $\mathcal{L} = Age \times Age\,status \times Gender \times Marital\,status \times Job\,status \times Ethical\,tendency$ be the set of all possible status array, then for every year $t \in T$ there is a function $\ell_t \colon \mathcal{P}_t \to \mathcal{L}$ which associates to each person their status. Moreover, ℓ has, as function, different components one for each status and, for sake of readability, the function determining each status is denoted with an abbreviation of the status itself; thus for a person p and a given year t, $\ell_t(p) = (Age_t(p), AgeStat_t(p), Gender(p), MarStat_t(p), Job_t(p), Legal_t(p))$. We consider now each status to present it showing how it may change during time.

Age

The age function $Age_t \colon \mathcal{P}_t \to \{1, \ldots, 80\}$ assigns to every person p in the current population $\mathcal{P}_t$ its age $Age_t(p)$. Obviously there is a strict connection between Age_t and Age_{t+1}, more precisely the latter as the following definition:

$$Age_{t+1}(p) := \begin{cases} 1 & \text{if } p \in \mathcal{P}^0_{t+1}, \\ Age_t(p) + 1 & \text{if } p \in \mathcal{P}_t, \\ ImAge_{t+1}(p) & \text{if } p \in Im_{t+1}. \end{cases}$$

The age function $Age_0 \colon \mathcal{P}_0 \to \{1, \ldots, 80\}$ for the starting population $\mathcal{P}_0$ is given. For what concerns notation, given $1 \leqslant n \leqslant 80$, we denote $\mathcal{P}^n_t := \{p \in \mathcal{P}_t \mid Age_t(p) = n\}$.

[2] For sake of simplicity we join together divorced and widowhood and use bachelor for both males and females.

Age status

The age status for a person $p \in \mathcal{P}_t$ depends only on the current age of p, more precisely:

$$AgeStat_t(p) := \begin{cases} young & \text{if } 1 \leqslant Age_t(p) \leqslant 20, \\ adult & \text{if } 21 \leqslant Age_t(p) \leqslant 60, \\ elderly & \text{if } 61 \leqslant Age_t(p) \leqslant 80. \end{cases}$$

We adopt the following notation $\mathcal{P}_t^{adult} := \{p \in \mathcal{P}_t \mid AgeStat_t(p) = adult\}$ and similarly for $\mathcal{P}_t^{young}$ and $\mathcal{P}_t^{elderly}$. Other characteristics or properties may depend on the age status, in particular:

- the transition functions (see below) from one year to the next one for Marital status and Job status;
- the number of new born in the next year, $|\mathcal{P}_{t+1}^0|$, depends on the current number of adult people, $|\mathcal{P}_t^{adult}|$.

Gender

Since gender is fixed, it depends only on the initial conditions for the people in $\mathcal{P}_0$, on the birth gender function $Bgen_t$ for new born and on the immigrate gender function $ImGen_t$ for who immigrates.

Marital status

There are three marital status: bachelor, married, divorced. Young people, i.e. people for which the current age is less than 21, have only one marital status available, namely bachelor; thus

$$MarStat_t(p) = bachelor$$

for all $p \in \mathcal{P}_t^{young}$.

In order to model during years the marital status, as well as the job status, for adults and elderly people (who may have other status beside bachelor) we use Markov chains; more precisely different Markov chains depending on the age status. Excepted for people who have just became adult, i.e. 21 year old people, who are automatically bachelor and people who have just became elderly, i.e. 61 year old people, who preserve the previous status,[3] the transitions between these status are given by a stochastic process with different probabilities for adults and elderly people. Denoting with B, M, D being respectively bachelor, married and divorced and setting to $99\%, 95\%, 5\%, 1\%$ the probabilities involved (which have been arbitrary chosen for this paper but, being editable parameters, could be instantiated with real values coming from statistical investigations), the transition schemes for adults and elderly people are the following:

This means that if at time t a 35-year adult is bachelor there is 95% chance that he is still bachelor at $t + 1$ and a 5% chance that he got married. If the current population is sufficiently

[3] Formally, this means that if $Age_t(p) = 21$ than $MarStat_t(p) = bachelor$ and if $Age_{t+1}(p) = 61$, then $MarStat_{t+1}(p) = MarStat_t(p)$.

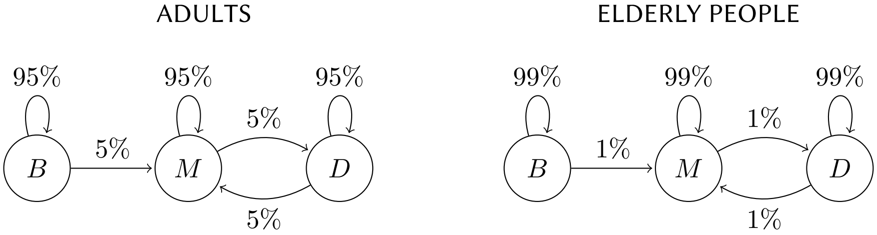

Figure 2: Adults and elderly people marital status transition diagrams.

large these probabilities can be seen as the fractions of the current population changing their status; namely every year the 95% of married adult remain married whereas the 5% divorced.

Moreover, this setting allows for an easy introduction of further constraints. For example, if there is a waiting period of at least n years between the declaration of divorce and a subsequent marriage, this can be encoded in the marital status in the following way: if $MarStat_t(p) = married$ and $MarStat_{t+1} = divorced$, then $MarStat_{t+1+i}(p) = divorced$ for every $i \in \{1, \dots, n\}$.

As before, the marital function $MarStat_0$ for the initial population $\mathcal{P}_0$ as well as the marital status of immigrants are given.

Job status

There are four job status: unemployed, public job, private job and retired. Overall the treatment of the job status is similar to the marital status, for example young people and 21 year old people have only one status available, namely unemployed; the only difference, excepts for probabilities of the transition functions, is that adults and elderly people have two different sets of available status. More precisely adults can have one among: unemployed, public job and private job; whereas elderly people one among: retired, public job and private job. During the transition between adulthood and old age the status remains unchanged except for unemployed people who become retired; formally, if $Age_{t+1}(p) = 61$ then:

$$Job_{t+1}(p) := \begin{cases} retired & \text{if } Job_t(p) = unemployed, \\ public & \text{if } Job_t(p) = public, \\ private & \text{if } Job_t(p) = private. \end{cases}$$

Moreover, as it can be seen from the transition function for elderly people, retirement is irreversible; namely if a person retires than from that year their status will always be retired. Abbreviating with Un, Ret, Pub, Prv respectively being unemployed, retired, public job and private job, the transition functions are as follows:

As before, the job function Job_0 for the initial population $\mathcal{P}_0$ as well as the job status of immigrants are given.

Ethical tendency

Ethical tendency aims to formalize the behavior of people with respect to the violation of law. The main idea is that the violation of a norm by a person depends on the expected utility of that

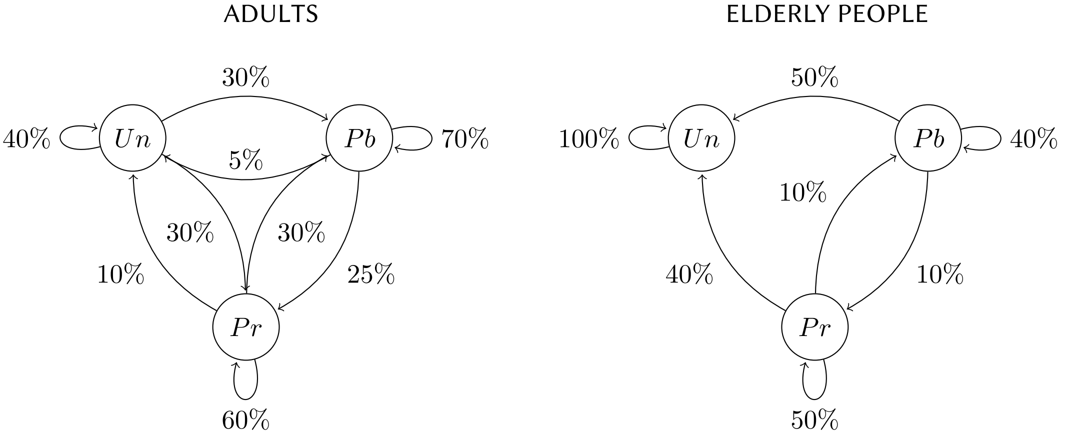

Figure 3: Adults and elderly people job transition diagram.

person violating the norm compared with the expected utility respecting the norm together with the Ethical tendency of that person, i.e. the general propensity to respect the law. Let η be a norm and p a person; if we denote with $\mu_p(\eta^+)$ the expected utility of p in complying with η and with $\mu_p(\eta^-)$ the expected utility of p in violating η, than the probability of p of violating η is given by:

$$P(p,\eta) := \begin{cases} 0 & \text{if } \mu_p(\eta^+) \geqslant \mu_p(\eta^-), \\ 1 - e^{-\alpha_p k} & \text{if } k = \mu_p(\eta^-) - \mu_p(\eta^+) > 0. \end{cases} \tag{1}$$

α_p is a parameter representing the tendency of p to violate the law, there are three cases corresponding to the three Ethical tendency status:

- $\alpha_p = 0$, in this case p does not violate the law, no matter how high is the expected utility in the violation; in this case p is a *legalist.*
- $0 < \alpha_p < +\infty$, in this case p may violate η if $\mu_p(\eta^-) > \mu_p(\eta^+)$ and the probability increases as $k = \mu_p(\eta^-) - \mu_p(\eta^+)$ increases; in this case p is legally *neutral.*
- $\alpha_p = +\infty$, in this case p violates η as soon as $\mu_p(\eta^-) > \mu_p(\eta^+)$; in this case p is an *opportunist.*

For a quantitative estimation of the role of α_p, if $\alpha_p = 1$ and $k = \mu_p(\eta^-) - \mu_p(\eta^+) = 1$ then there is a probability of $1 - e^{-1} \simeq 63\%$ that p violate η.

The Ethical tendency of a person p may change during time according a transition function, in this case we assign the same transition process to the three age status. Denoting with L, N, O being respectively legalist, neutral and opportunist we adopt the following status transitions (again the probabilities are editable parameters):

When the ethical tendency of a person p becomes *neutral*, the transition function also assigns to α_p a strictly positive real value. As before the legal function $Legal_0$ of the starting population $\mathcal{P}_0$, as well as the ones for new born and immigrants, are given.[4]

[4] Another possibility would be to assign to new born legal tendencies that reflect the proportion of legal tendencies in the adult population or in the whole population.

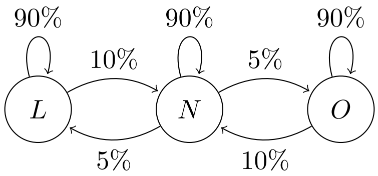

Figure 4: Young, adults and elderly people ethical tendencies transition diagram.

4. Law Compliance

The conceptualisation of law compliance is taken from the current literature on DDL and derived models. In particular we presuppose that an agent *enforces* freely literals that are taken as *feasible actions*. On the other hand, when an agent performs a particular action, for which, as we specified before, she has a specific tendency (a probability of doing that action), the legal system takes care of that. In particular, we assume that the *normative background* is actually constructed as a **DDT**, and feasible actions are literals of that theory.

Defeasible Logic [6, 7] is a simple, flexible, and efficient rule-based non-monotonic formalism. Its strength lies in its constructive proof theory, allowing it to draw meaningful conclusions from (potentially) conflicting and incomplete knowledge base. In non-monotonic systems, more accurate conclusions can be obtained when more pieces of information become available.

Many variants of Defeasible Logic have been proposed for the logical modelling of different application areas, specifically agents [8, 9, 10], legal reasoning [11, 12] and workflows from a business process compliance perspective [13, 14].

In this research we focus on the Defeasible Deontic Logic (henceforth DDL) framework [15] that allows us to determine what prescriptive behaviours are in force in a given situation. For detailed descriptions of how to adopt DDL for legal reasoning we refer the reader to [16].

We start by defining the language of a Defeasible Deontic Theory (henceforth a DDT)

Let PROP be a set of propositional atoms, and Lab be a set of arbitrary labels (the names of the rules). We use lower-case Roman letters to denote literals and lower-case Greek letters to denote rules.

Accordingly, $\mathrm{PLit} = \mathrm{PROP} \cup \{\neg l \,|\, l \in \mathrm{PROP}\}$ is the set of *plain literals*; the set of *deontic literals* is $\mathrm{ModLit} = \{\Box l, \neg\Box l \mid l \in \mathrm{PLit} \wedge \Box \in \{\mathsf{O}, \mathsf{P}\}\}$ and, finally, the set of *literals* is $\mathrm{Lit} = \mathrm{PLit} \cup \mathrm{ModLit}$. The *complement* of a literal l is denoted by $\sim l$: if l is a positive literal p then $\sim l$ is $\neg p$, and if l is a negative literal $\neg p$ then $\sim l$ is p. We will not have specific rules nor modality for prohibitions, as we will treat them according to the standard duality that something is forbidden iff the opposite is obligatory (i.e., $\mathsf{O}\neg p$).

Definition 1 (Defeasible Deontic Theory). *A* DDT D *is a triple* $(F, R, >)$*, where* F *is the set of facts,* R *is the set of rules, and* $>$ *is a binary relation over* R *(called superiority relation).*

Specifically, the set of facts $F \subseteq \mathrm{PLit}$ denotes simple pieces of information that are always considered to be true, like “Sylvester is a cat”, formally $cat(Sylvester)$. In this paper, we subscribe to the distinction between the notions of obligations and permissions, and that of norms, where the norms in the system determine the obligations and permissions in force in a normative system. A DDT is meant to represent a normative system, where the rules encode the norms of the system, and the set of facts corresponds to a case. As we will see below, the rules are used to conclude the institutional facts, obligations and permissions that hold in a case. Accordingly, we do not admit obligations and permissions as facts of the theory.

The set of rules R contains three *types* of rules: *strict rules*, *defeasible rules*, and *defeaters*. Rules are also of two *kinds*:

- *Constitutive rules* (non-deontic rules) R^{C} model constitutive statements (count-as rules);
- *Deontic rules* to model prescriptive behaviours, which are either *obligation rules* R^{O} that determine when and which obligations are in force, or *permission rules* which represent *strong* (or *explicit*) permissions R^{P}.

Lastly, $> \subseteq R \times R$ is the *superiority* (or *preference*) relation, which is used to solve conflicts in case of potentially conflicting information.

A theory is *finite* if the set of facts and rules are so. We only focus on finite theories.

A strict (constitutive) rule is a rule in the classical sense: whenever the premises are indisputable, so is the conclusion.

On the other hand, defeasible rules are to conclude statements that can be defeated by contrary evidence. In contrast, defeaters are special rules whose only purpose is to prevent the derivation of the opposite conclusion.

A prescriptive behaviour like "Passing on zebra crossing is not permitted when the traffic light for pedestrian is red" can be formalised via the general permissive rule

$$AtZebraCross \Rightarrow_{\mathsf{P}} Pass$$

and the exception through the obligation rule

$$Pedestrian\_traffic\_light\_red \Rightarrow_{\mathsf{O}} \neg Pass.$$

Following the ideas of [17], obligation rules gain more expressiveness with the *compensation operator* $\otimes$ for obligation rules, which is to model reparative chains of obligations. Intuitively, $a \otimes b$ means that a is the primary obligation, but if for some reason we fail to obtain, to comply with, a (by either not being able to prove a, or by proving $\sim a$) then b becomes the new obligation in force. This operator is used to build chains of preferences, called $\otimes$-expressions.

The formation rules for $\otimes$-expressions are as follows (i) every plain literal is an $\otimes$-expression and (ii) if A is an $\otimes$-expression and b is a plain literal then $A \otimes b$ is an $\otimes$-expression [15].

In general an $\otimes$-expression has the form '$c_1 \otimes c_2 \otimes \cdots \otimes c_m$', and it appears as consequent of a rule '$A(\alpha) \hookrightarrow_{\mathsf{O}} C(\alpha)$' where $C(\alpha) = c_1 \otimes c_2 \otimes \cdots \otimes c_m$; the meaning of the $\otimes$-expression is: if the rule is allowed to draw its conclusion, then c_1 is the obligation in force, and only when c_1 is violated then c_2 becomes the new in force obligation, and so on for the rest of the elements in the chain. In this setting, c_m stands for the last chance to comply with the prescriptive behaviour enforced by α, and in case c_m is violated as well, then we will result in a non-compliant situation.

For instance, the previous prohibition to pass on pedestrian cross in case of red can foresee a compensatory fine, like

$$Pedestrian\_traffic\_light\_red \Rightarrow_{\mathsf{O}} \neg Pass \otimes PayFine$$

that has to be paid in case someone passes the pedestrian cross when the light is red.

It is worth noticing that we admit $\otimes$-expressions with only one element. The intuition, in this case, is that the obligatory condition does not admit compensatory measures or, in other words, that it is impossible to recover from its violation.

In this paper, we focus exclusively on the defeasible part of the logic ignoring the monotonic component given by the strict rules; consequently, we limit the language to the cases where the rules are either defeasible or defeaters. From a practical point of view, the restriction does not effectively limit the expressive power of the logic: a defeasible rule where there are no rules for the opposite conclusion, or where all rules for the opposite conclusion are weaker than the given defeasible rules, effectively behaves like a strict rule. Formally a rule is defined as below.

Definition 2 (Rule). *A* rule *is an expression of the form $\alpha\colon A(\alpha) \hookrightarrow_{\Box} C(\alpha)$, where*

1. $\alpha \in \mathrm{Lab}$ *is the unique name of the rule;*
2. $A(\alpha) \subseteq \mathrm{Lit}$ *is the set of antecedents;*
3. *An arrow $\hookrightarrow \in \{\Rightarrow, \leadsto\}$ denoting, respectively, defeasible rules, and defeaters;*
4. $\Box \in \{\mathsf{C}, \mathsf{O}, \mathsf{P}\}$*;*
5. *its consequent $C(\alpha)$, which is either*
   a) *a single plain literal $l \in \mathrm{PLit}$, if either (i) $\hookrightarrow \equiv \leadsto$ or (ii) $\Box \in \{\mathsf{C}, \mathsf{P}\}$, or*
   b) *an $\otimes$-expression, if $\Box \equiv \mathsf{O}$.*

If $\Box = \mathsf{C}$ then the rule is used to derive non-deontic literals (constitutive statements), whilst if $\Box$ is O or P then the rule is used to derive deontic conclusions (prescriptive statements). The conclusion $C(\alpha)$ is, as before, a single literal in case $\Box = \{\mathsf{C}, \mathsf{P}\}$; in case $\Box = \mathsf{O}$, then the conclusion is an $\otimes$-expression. $\otimes$-expressions can only occur in prescriptive rules though we do not admit them on defeaters (Condition 5.(a).i), see [15] for a detailed explanation.

We use some abbreviations on sets of rules. The set of defeasible rules in R is $R_{\Rightarrow}$, the set of defeaters is R_{dft}. $R^{\Box}[l]$ is the rule set appearing in R with head l and modality $\Box$, while $R^{\mathsf{O}}[l, i]$ denotes the set of obligation rules where l is the i-th element in the $\otimes$-expression. Given that the consequent of a rule is either a single literal or an $\otimes$-expression (that can be understood as a sequence of elements, and then as an ordered set), in what follows we are going to abuse the notation and use $l \in C(\alpha)$. $R^{\Box}$ is the set of rules $\alpha\colon A(\alpha) \hookrightarrow_{\Box} C(\alpha)$ such that α *appears in R*. For a theory as determined by Definitions 1 and 2, α appears in R means that $\alpha \in R$; thus R^{P} is the set of permissive rules appearing in R. We use $R^{\Diamond}$ and $R^{\Diamond}[l]$ as shorthands for $R^{\mathsf{O}} \cup R^{\mathsf{P}}$ and $R^{\mathsf{O}}[l] \cup R^{\mathsf{P}}[l]$, respectively. The abbreviations can be combined. Finally, a literal l appears in a theory D, if there is a rule $\alpha \in R$ such that $l \in A(\alpha) \cup C(\alpha)$.

Definition 3 (Tagged modal formula). *A* tagged modal formula *is an expression of the form* $\pm\partial_{\Box}l$*, with the following meanings*

- $+\partial_{\Box}l$*:* l *is* defeasibly provable *(or simply provable) with mode* $\Box$*;*
- $-\partial_{\Box}l$*:* l *is* defeasibly refuted *(or simply refuted) with mode* $\Box$*.*

Accordingly, the meaning of $+\partial_{\mathsf{O}}p$ is that p is provable as an obligation, and $-\partial_{\mathsf{P}}\neg p$ is that we have a refutation for the permission of $\neg p$. Similarly, for the other combinations.

As we will shortly see (Definitions 5 and 6), one of the key ideas of DDL is that we use tagged modal formulas to determine which formulas are (defeasibly) provable or rejected given a theory and a set of facts (used as input for the theory). Therefore, when we have asserted the tagged modal formula $+\partial_{\mathsf{O}}l$ in a derivation (see Definition 4 below), we can conclude that the obligation of l ($\mathsf{O}l$) follows from the rules and the facts and that we used a prescriptive rule to derive l; similarly for permission (using a permissive rule). However, the C modality is silent, meaning that we do not put the literal in the scope of the C modal operator, thus for $+\partial_{\mathsf{C}}l$, the derivation simply asserts that l holds (and not that $\mathsf{C}l$ holds, even if the two have the same meaning). For the negative cases (i.e., $-\partial_{\Box}l$), the interpretation is that it is not possible to derive l with a given mode. Accordingly, we read $-\partial_{\mathsf{O}}l$ as it is impossible to derive l as an obligation. For $\Box \in \{\mathsf{O}, \mathsf{P}\}$ we are allowed to infer $\neg\Box l$, giving a constructive interpretation of the deontic modal operators. Notice that this is not the case for C, where we cannot assert that $\sim l$ holds (this would require $+\partial_{\mathsf{C}}{\sim}l$); in the logic, failing to prove l does not equate to proving $\neg l$. We will use the term *conclusions* and tagged modal formulas interchangeably.

Definition 4 (Proof). *Given a DDT* D*, a proof* P *of length* m *in* D *is a finite sequence* $P(1), P(2), \ldots, P(m)$ *of tagged modal formulas, where the proof conditions hold.*

$P(1..n)$ denotes the first n steps of P, and we also use the notational convention $D \vdash \pm\partial_{\Box}l$, meaning that there is a proof P for $\pm\partial_{\Box}l$ in D.

Core notions in DL are that of *applicability/discardability.* As knowledge in a defeasible theory is circumstantial, given a defeasible rule like '$\alpha\colon a, b \Rightarrow_{\Box} c$', there are four possible scenarios: the theory defeasibly proves both a and b, the theory proves neither, the theory proves one but not the other. Naturally, only in the first case, where both a and b are proved, we can use α to *support/try to conclude* $\Box c$. Briefly, we say that a rule is *applicable* when every antecedent's literal has been proved at a previous derivation step. Symmetrically, a rule is *discarded* when one of such literals has been previously refuted. Formally:

Definition 5 (Applicability). *Assume a deontic defeasible theory* $D = (F, R, >)$*. We say that rule* $\alpha \in R^{\mathsf{C}} \cup R^{\mathsf{P}}$ *is* applicable *at* $P(n+1)$*, iff for all* $a \in A(\alpha)$

1. *if* $a \in \mathrm{PLit}$*, then* $+\partial_{\mathsf{C}}a \in P(1..n)$*,*
2. *if* $a = \Box q$*, then* $+\partial_{\Box}q \in P(1..n)$*, with* $\Box \in \{\mathsf{O}, \mathsf{P}\}$*,*
3. *if* $a = \neg\Box q$*, then* $-\partial_{\Box}q \in P(1..n)$*, with* $\Box \in \{\mathsf{O}, \mathsf{P}\}$*.*

We say that rule $\alpha \in R^{\mathsf{O}}$ *is* applicable at index i and $P(n+1)$ *iff Conditions 1–3 above hold and*

4. *$\forall c_j \in C(\alpha)$, $j < i$, then $+\partial_{\mathsf{O}} c_j \in P(1..n)$ and $+\partial_{\mathsf{C}} {\sim} c_j \in P(1..n)$.*[5]

Definition 6 (Discardability). *Assume a deontic defeasible theory D, with $D = (F, R, >)$. We say that rule $\alpha \in R^{\mathsf{C}} \cup R^{\mathsf{P}}$ is* discarded *at $P(n+1)$, iff there exists $a \in A(\alpha)$ such that*

1. *if $a \in \mathrm{PLit}$, then $-\partial_{\mathsf{C}} l \in P(1..n)$, or*
2. *if $a = \Box q$, then $-\partial_{\Box} q \in P(1..n)$, with $\Box \in \{\mathsf{O}, \mathsf{P}\}$, or*
3. *if $a = \neg\Box q$, then $+\partial_{\Box} q \in P(1..n)$, with $\Box \in \{\mathsf{O}, \mathsf{P}\}$.*

We say that rule $\alpha \in R^{\mathsf{O}}$ is discarded at index i and $P(n+1)$ *iff either at least one of the Conditions 1–3 above hold, or*

4. *$\exists c_j \in C(\alpha)$, $j < i$ such that $-\partial_{\mathsf{O}} c_j \in P(1..n)$, or $-\partial_{\mathsf{C}} {\sim} c_j \in P(1..n)$.*

Discardability is obtained by applying the principle of *strong negation* to the definition of applicability. The strong negation principle applies the function that simplifies a formula by moving all negations to an innermost position in the resulting formula, replacing the positive tags with the respective negative tags, and the other way around; see [19]. Positive proof tags ensure that there are effective decidable procedures to build proofs; the strong negation principle guarantees that the negative conditions provide a constructive and exhaustive method to verify that a derivation of the given conclusion is not possible. Accordingly, Condition 3 of Definition 5 allows us to state that $\neg\Box p$ holds when we have a (constructive) failure to prove p with mode $\Box$ (for obligation or permission), thus it corresponds to a constructive version of negation as failure.

We are ready to formalise the proof conditions, as in [15]. We start with positive proof conditions for constitutive statements. In the following, we shall omit the explanations for negative proof conditions, when trivial, reminding the reader that they are obtained through the application of the strong negation principle to the positive counterparts.

Definition 7 (Constitutive Proof Conditions).

$+\partial_{\mathsf{C}} l$: If $P(n+1) = +\partial_{\mathsf{C}} l$ then
(1) $l \in F$, or
(2) (1) ${\sim} l \notin F$, and
(2) $\exists \beta \in R^{\mathsf{C}}_{\Rightarrow}[l]$ s.t. β is appl., and
(3) $\forall \gamma \in R^{\mathsf{C}}[{\sim} l]$ either
(1) γ is disc., or
(2) $\exists \zeta \in R^{\mathsf{C}}[l]$ s.t.
(1) ζ is appl. and
(2) $\zeta > \gamma$.

$-\partial_{\mathsf{C}} l$: If $P(n+1) = -\partial_{\mathsf{C}} l$ then
(1) $l \notin F$ and either
(2) (1) ${\sim} l \in F$, or
(2) $\forall \beta \in R^{\mathsf{C}}_{\Rightarrow}[l]$, either β is disc., or
(3) $\exists \gamma \in R^{\mathsf{C}}[{\sim} l]$ such that
(1) γ is appl., and
(2) $\forall\ \zeta \in R^{\mathsf{C}}[l]$, either
(1) ζ is disc., or
(2) $\zeta \not> \gamma$.

[5] As discussed above, we are allowed to move to the next element of an $\otimes$-expression when the current element is violated. To have a violation, we need (i) the obligation to be in force, and (ii) that its content does not hold. $+\partial_{\mathsf{O}} c_i$ indicates that the obligation is in force. For the second part we have two options. The former, $+\partial_{\mathsf{C}} {\sim} c_i$ means that we have "evidence" that the opposite of the content of the obligation holds. The latter would be to have $-\partial_{\mathsf{C}} c_i \in P(1..n)$ corresponding to the intuition that we failed to provide evidence that the obligation has been satisfied. The former option implies the latter one. For a deeper discussion on the issue, see [18].

A literal is defeasibly proved if: it is a fact, or there exists an applicable, defeasible rule supporting it (such a rule cannot be a defeater) and all opposite rules are either discarded or defeated. To prove a conclusion, not all the work has to be done by a stand-alone (applicable) rule (the rule witnessing Condition (2.2)): all the applicable rules for the same conclusion (may) contribute to defeating applicable rules for the opposite conclusion. Both γ as well as ζ may be defeaters. Below we present the proof conditions for obligations.

Definition 8 (Obligation Proof Conditions).

$+\partial_{\mathsf{O}} l$: *If* $P(n+1) = +\partial_{\mathsf{O}} l$ *then*
  $\exists \beta \in R^{\mathsf{O}}_{\Rightarrow}[l, i]$ *s.t.*
  (1) β *is applicable at index* i *and*
  (2) $\forall \gamma \in R^{\mathsf{O}}[\sim l, j] \cup R^{\mathsf{P}}[\sim l]$ *either*
    (1) γ *is discarded (at index* j*), or*
    (2) $\exists \zeta \in R^{\mathsf{O}}[l, k]$ *s.t.*
      (1) ζ *is applicable at index* k *and*
      (2) $\zeta > \gamma$.

$-\partial_{\mathsf{O}} l$: *If* $P(n+1) = -\partial_{\mathsf{O}} l$ *then*
  $\forall \beta \in R^{\mathsf{O}}_{\Rightarrow}[l, i]$ *either*
  (1) β *is discarded at index* i*, or*
  (2) $\exists \gamma \in R^{\mathsf{O}}[\sim l, j] \cup R^{\mathsf{P}}[\sim l]$ *s.t.*
    (1) γ *is applicable (at index* j*), and*
    (2) $\forall \zeta \in R^{\mathsf{O}}[l, k]$ *either*
      (1) ζ *is discarded at index* k*, or*
      (2) $\zeta \not> \gamma$.

(i) in Condition (2) γ can be a permission rule as explicit, opposite permissions represent exceptions to obligations, whereas ζ (Condition 2.2) must be an obligation rule as a permission rule cannot reinstate an obligation, and that (ii) l may appear at different positions (indices i, j, and k) within the three $\otimes$-chains. Below, we introduce the proof conditions for permissions.

Definition 9 (Permission Proof Conditions).

$+\partial_{\mathsf{P}} l$: *If* $P(n+1) = +\partial_{\mathsf{P}} l$ *then*
  (1) $+\partial_{\mathsf{O}} l \in P(1..n)$*, or*
  (2) $\exists \beta \in R^{\mathsf{P}}_{\Rightarrow}[l]$ *s.t.*
    (1) β *is appl. and*
    (2) $\forall \gamma \in R^{\mathsf{O}}[\sim l, j]$ *either*
      (1) γ *is disc. at index* j*, or*
      (2) $\exists \zeta \in R^{\mathsf{P}}[l] \cup R^{\mathsf{O}}[l, k]$ *s.t.*
        (1) ζ *is appl. (at index* k*) and*
        (2) $\zeta > \beta$.

$-\partial_{\mathsf{P}} l$: *If* $P(n+1) = -\partial_{\mathsf{P}} l$ *then*
  (1) $-\partial_{\mathsf{O}} l \in P(1..n)$*, and*
  (2) $\forall \beta \in R^{\mathsf{P}}_{\Rightarrow}[l]$ *either*
    (1) β *is disc. or*
    (2) $\exists \gamma \in R^{\mathsf{O}}[\sim l, j]$ *s.t.*
      (1) γ *is appl. at index* j *and*
      (2) $\forall \zeta \in R^{\mathsf{P}}[l] \cup R^{\mathsf{O}}[l, k]$ *either*
        (1) ζ *is disc. (at index* k*), or*
        (2) $\zeta \not> \beta$.

Condition (1) allows us to derive a permission from the corresponding obligation. Thus it corresponds to the $\mathsf{O}a \rightarrow \mathsf{P}a$ axiom of Deontic Logic. Condition (2.2) considers as possible counter-arguments *only* obligation rules in situations where both $\mathsf{P}l$ and $\mathsf{P}\neg l$ hold are allowed. We refer the readers interested in a deeper discussion on how to model permissions and obligations in DDL to [15].

The set of positive and negative conclusions of a theory is called *extension*. The extension of a theory is computed based on the literals that appear in it; more precisely, the literals in the Herbrand Base of the theory $HB(D) = \{l, \sim l \in \text{PLit} | \, l \text{ appears in } D\}$.

Definition 10 (Extension). *Given a DDT D, we define the* extension *of D as* $E(D) = (+\partial_{\mathsf{C}}, -\partial_{\mathsf{C}}, +\partial_{\mathsf{O}}, -\partial_{\mathsf{O}}, +\partial_{\mathsf{P}}, -\partial_{\mathsf{P}})$*, where* $\pm\partial_{\Box} = \{l \in HB(D) | \, D \vdash \pm\partial_{\Box} l\}$*, with* $\Box \in \{\mathsf{C}, \mathsf{O}, \mathsf{P}\}$.

5. Conclusions and further extensions

While expressing some concerns and the plan to correct macroscopic limits of current simulation platforms, in particular, GAMA, we value the idea that it is a relevant step to enterprise the definition of a credible system to actually simulate the effects on a society of the introduction of a new norm. The discussion we provided upon the notion of *class* to which a particular individual of the MAS belongs (while classes do not constitute a partition) is devised as a support to the usage of the rules in the Deontic counterpart of the simulator. However, when the system will be completely built, we shall have three specific strengths that differentiate our approach to any other ones already discussed in the simulation literature for agents' modelling that are inspired by the approach followed for the GAMA framework:

- We consider the class hierarchy described in terms of transitions, the results of the *computation of the extension* of the constituent rules as devised above. There is no such a thing as *a priori class*, but only classes built as result of a set of rules;
- Probabilistic and utility-driven models attain at agents' definition, not the normative background that is solely prescriptive;
- There exists a system of negative feedback, able to re-devise the properties of the class hierarchy, the probabilities and the consequential behaviors of the agents.

We followed here the concepts expressed in the researches by Riveret et al. [20, 21] and also discussed further on by Governatori et al. [9].

Finally, the ductility of defeasible deontic logic would allow, in a further extension, to include in the model not only law compliance, but also ethical and personal behaviors.

References

[1] M. Cristani, G. Governatori, F. Olivieri, L. Pasetto, F. Tubini, C. Veronese, A. Villa, E. Zorzi, The architecture of a reasoning system for defeasible deontic logic, in: Procedia Computer Science, 2023, pp. 4214–4224. doi:10.1016/j.procs.2023.10.418.

[2] M. Cristani, G. Governatori, F. Olivieri, L. Pasetto, F. Tubini, C. Veronese, A. Villa, E. Zorzi, Houdini (unchained): an effective reasoner for defeasible logic, in: CEUR Workshop Proceedings, 2022, pp. 1–16.

[3] P. Taillandier, D.-A. Vo, E. Amouroux, A. Drogoul, Gama: A simulation platform that integrates geographical information data, agent-based modeling and multi-scale control, in: LNCS, 2012, p. 242 – 258. doi:10.1007/978-3-642-25920-3_17.

[4] A. Drogoul, E. Amouroux, P. Caillou, B. Gaudou, A. Grignard, N. Marilleau, P. Taillandier, M. Vavasseur, D.-A. Vo, J.-D. Zupker, Gama: Multi-level and complex environment for agent-based models and simulations, in: AAMAS 2013, 2013, p. 1361 – 1362.

[5] E. Amouroux, T.-Q. Chu, A. Boucher, A. Drogoul, Gama: An environment for implementing and running spatially explicit multi-agent simulations, in: LNCS, 2009, p. 359 – 371. doi:10.1007/978-3-642-01639-4_32.

[6] D. Nute, Defeasible logic, in: Handbook of Logic in Artificial Intelligence and Logic Programming, Oxford University Press, 1987.

[7] G. Antoniou, D. Billington, G. Governatori, M. J. Maher, Representation results for defeasible logic, ACM Trans. Comput. Log. (2001) 255–287. doi:10.1145/371316.371517.

[8] K. Kravari, N. Bassiliades, A survey of agent platforms, Journal of Artificial Societies and Social Simulation (2015) 11. doi:10.18564/jasss.2661.

[9] G. Governatori, F. Olivieri, S. Scannapieco, A. Rotolo, M. Cristani, The rationale behind the concept of goal, Theory Pract. Log. Program. (2016) 296–324. URL: https://doi.org/10.1017/S1471068416000053. doi:10.1017/S1471068416000053.

[10] M. Dastani, G. Governatori, A. Rotolo, L. van der Torre, Programming cognitive agents in defeasible logic, in: LPAR 2005 Conference, Montego Bay, Jamaica, LNAI, Springer, 2005, pp. 621–636.

[11] G. Governatori, A. Rotolo, Changing legal systems: Legal abrogations and annulments in defeasible logic, Logic Journal of the IGPL (2009) 157–194. doi:10.1093/jigpal/jzp075.

[12] M. Cristani, F. Olivieri, A. Rotolo, Changes to temporary norms, in: ICAIL 2017, 2017, pp. 39–48. doi:10.1145/3086512.3086517.

[13] G. Governatori, F. Olivieri, S. Scannapieco, M. Cristani, Designing for compliance: Norms and goals, in: RuleML 2011, LNCS, Springer, 2011, pp. 282–297. doi:10.1007/978-3-642-24908-2_29.

[14] F. Olivieri, M. Cristani, G. Governatori, Compliant business processes with exclusive choices from agent specification, LNCS (2015) 603–612.

[15] G. Governatori, F. Olivieri, A. Rotolo, S. Scannapieco, Computing strong and weak permissions in defeasible logic, J. Philos. Log. (2013) 799–829. URL: https://doi.org/10.1007/s10992-013-9295-1. doi:10.1007/s10992-013-9295-1.

[16] G. Governatori, A. Rotolo, G. Sartor, Logic and the law: Philosophical foundations, deontics, and defeasible reasoning, in: D. Gabbay, J. Horty, X. Parent, R. van der Meyden, L. van der Torre (Eds.), Handbook of Deontic Logic and Normative Systems, College Publications, London, 2021, pp. 657–764.

[17] G. Governatori, A. Rotolo, Logic of violations: A gentzen system for reasoning with contrary-to-duty obligations, Australasian Journal of Logic (2006) 193–215. URL: http://ojs.victoria.ac.nz/ajl/article/view/1780.

[18] G. Governatori, Burden of compliance and burden of violations, in: A. Rotolo (Ed.), 28th Annual Conference on Legal Knowledge and Information Systems, Frontiers in AI and Applications, IOS Press, Amsterdam, 2015, pp. 31–40.

[19] G. Governatori, V. Padmanabhan, A. Rotolo, A. Sattar, A defeasible logic for modelling policy-based intentions and motivational attitudes, Log. J. IGPL (2009) 227–265. doi:10.1093/jigpal/jzp006.

[20] R. Riveret, A. Rotolo, G. Sartor, Probabilistic rule-based argumentation for norm-governed learning agents, Artificial Intelligence and Law (2012) 383–420. doi:10.1007/s10506-012-9134-7.

[21] R. Riveret, G. Contissa, A. Rotolo, J. Pitt, Law enforcement in norm-governed learning agents, in: AAMAS 2013, 2013, pp. 1151–1152.