Chapter x
Logics for Legal Dynamics

Guido Governatori and Antonino Rotolo

Abstract Legal dynamics is an important aspect of legal reasoning that inspired the area of belief revision. While formal models of belief revision have been thoroughly examined, the formalisation of legal dynamic has been mostly neglected. In this contribution we propose Temporal Defeasible Logic to model legal dynamics. We build such a logic in steps starting from basic defeasible logic, and we show how to use it to model different forms of modifications such as derogations, textual modifications, abrogation and annulment.

x.1 Introduction

One peculiar feature of the law is that it necessarily takes the form of a dynamic normative system [Kelsen, 1991, Hart, 1994]. Despite the importance of norm-change mechanisms, the logical investigation of legal dynamics was for long time underdeveloped. However, research is rapidly evolving and recent contributions exist.

In the Eighties a promising research effort was devoted by Alchourrón, Gärdenfors and Makinson [1985] to develop a logical model (AGM) for also modeling norm change. As is well-known, the AGM framework distinguishes three types of change operation over theories. Contraction is an operation that removes a specified sentence ϕ from a given theory Γ (a logically closed set of sentences) in such a way that Γ is set aside in favor of another theory Γ^-_ϕ which is a subset of Γ not containing ϕ. Expansion operation adds a given sentence ϕ to Γ so that the resulting theory Γ^+_ϕ is the smallest logically closed set that contains both Γ and ϕ. Revision operation adds ϕ to Γ but it is ensured that the resulting theory Γ^*_ϕ be consistent. Alchourrón,

Guido Governatori
NICTA and Queensland University of Technology, Australia

Antonino Rotolo
CIRSFID and Department of Legal Studies, University of Bologna, Italy

Gärdenfors and Makinson argued that, when Γ is a code of legal norms, contraction corresponds to norm derogation (norm removal) and revision to norm amendment.

AGM framework has the advantage of being very abstract but works with theories consisting of simple logical assertions. For this reason, it is perhaps suitable to capture the dynamics of obligations and permissions, not of legal norms. In fact, it is essential to distinguish norms from obligations and permissions [Boella et al., 2009, Governatori and Rotolo, 2010]: the latter ones are just possible effects of the application of norms and their dynamics do not necessarily require to remove or revise norms, but correspond in most cases to instances of the notion of *norm defeasibility* [Governatori and Rotolo, 2010]. Very recently, some research has been carried out to reframe AGM ideas within rule-based logical systems, which take this distinction into account [Stolpe, 2010, Rotolo, 2010]. However, also these attempts suffer from some drawbacks, as they fail to handle the following aspects of legal norm change:

1. the law usually regulate its own changes by setting specific norms whose peculiar objective is to change the system by stating what and how other existing norms should be modified;
2. since legal modifications are derived from these peculiar norms, they can be in conflict and so are defeasible;
3. legal norms are qualified by temporal properties, such as the time when the norm comes into existence and belongs to the legal system, the time when the norm is in force, the time when the norm produces legal effects, and the time when the normative effects hold.

Hence, legal dynamics can be hardly modelled without considering defeasibility and temporal reasoning. Some works (see, in particular, [Governatori and Rotolo, 2010]) have attempted to address these research issues. All norms are qualified by the above mentioned different temporal parameters and the modifying norms are represented as defeasible meta-rules, i.e., rules where the conclusions are temporalised rules.

This work reports on this research line and shows how suitable temporal extensions of Defeasible Logic can do a good job in faithfully modelling interesting aspects of legal dynamics.

The layout of the chapter is as follows. In the next section we are going to provide an introduction to Defeasible Logic and to show how to use it for modelling legal knowledge and legal reasoning. In particular in Section x.2.1 we present Basic Defeasible Logic; then in Section x.2.2 we introduce Deontic operators: this allows us to distinguish between constitutive and prescriptive rules and to speak about deontic effects. The next step is to extend the logic obtained so far with time (Section x.2.3). The final logic we describe is the extension with meta-rules rules about rules meant to captures norms about norms, which are essential to model norm dynamics. Section x.3 first introduces the types of changes we examine in this contribution (Section x.3), and then in Sections x.3.2, x.3.3 we discuss how textual modifications and derogations are handled by the logic, and to conclude with repeal type modifications (abrogation and annulment, Sections x.3.5 and x.3.7); in Section x.3.6 we also shortly analyse the challenges posed to logical systems by retroactive modifications. Section x.4 offers a short overview of alternative approaches.

x.2 Logic for Norms

In this section we are going to illustrate and provide the foundations of our logical framework.

Our choice of logic to represent norms falls on Defeasible Logic [Nute, 1994]. Defeasible Logic is a simple, flexible, extensible non-monotonic formalism.

x.2.1 Basic Defeasible Logic

Knowledge in Defeasible logic is structured in three components

- A set of facts (corresponding to indisputable statements represented as literals, where a literal is either an atomic proposition or its negation).
- A set of rules. A rule establishes a connection between a set of premises and a conclusion. In particular, for reasoning with norms, it is reasonable to assume that a rule provides the formal representation of a norm. Accordingly, the premises encode the conditions under which the norm is applicable, and the conclusion is the normative effect of the norm.
- A preference relation over the rules. The preference relation just gives the relative strength of rules. It is used in contexts where two rules with opposite conclusions fire simultaneously, and determines that one rule overrides the other in that particular context.

Formally, the knowledge in the logic is organised in Defeasible Theories, where a Defeasible Theory D is a structure

$$(F,R,\prec) \tag{1}$$

where F is the set of facts, R is the set of rules, and $\prec$ is a binary relation over the set of rules, i.e., $\prec \subseteq R \times R$.[1]

As we have alluded to above, a rule is formally a binary relation between, a set premises and a conclusion. Thus if Lit is the set of literals, the set Rule of all rules is:

$$\text{Rule} \subseteq 2^{\text{Lit}} \times \text{Lit}. \tag{2}$$

[1] Defeasible Logic does not impose any property for $\prec$. However, in many application is useful to assume that the transitive closure to be acyclic to prevent situations where, at the same time a rule overrules another rule and it is overridden by it.

Accordingly, a rule is an expression with the following form:[2]

$$r\colon a_1,\ldots,a_n \hookrightarrow c \tag{3}$$

where r is a unique label identifying the rule. Given that a rule is a relation, we can ask what is the strength of the link between the premises and the conclusion. We can distinguish three different strengths: (i) given the premises the conclusion always holds, (ii) given the premises the conclusion holds sometimes, and (iii) given the premises the opposite of the conclusions does not hold. Therefore, to capture theses types Defeasible Logic is equipped with three types of rules: *strict rules*, *defeasible rules* and *defeaters*. We will use $\rightarrow$, $\Rightarrow$ and $\rightsquigarrow$ instead of $\hookrightarrow$ to represent, respectively, strict rules, defeasible rules and defeaters. We will continue to use $\hookrightarrow$ for a rule when the strength is either not known or irrelevant.

Given a rule like rule r in (3) we use the following notation to refer to the various elements of the rule. $A(r)$ denotes the *antecedent* or *premises* of the rule, in this case, $\{a_1,\ldots,a_n\}$, and $C(r)$ denotes the *conclusion* or *consequent*, that is, c. From time to time we use *head* and *body* of a rule to refer, respectively, to the consequent and to the antecedent of the rule.

Strict rules are rules in the classic sense: whenever the premises are indisputable so is the conclusion. Strict rules can be used to model legal definitions that do not admit exceptions, for example the definition of minor: "'minor' means any person under the age of eighteen years". This definition can be represented as

$$age(x) < 18yrs \rightarrow minor(x). \tag{4}$$

Defeasible Rules are rules such that the conclusions normally or typically follows from the premises, unless there are evidence or reasons to the contrary.

Defeaters are rules that do not support directly the derivation of a conclusion, but that can be used to prevent a conclusion.

We illustrate defeasible rules and defeaters with the help of the definition of complaint from the Australian Telecommunication Consumer Protections Code 2012 TCP-C268_2012 May 2012 (TCPC).

> **Complaint** means an expression of dissatisfaction made to a Supplier in relation to its Telecommunications Products or the complaints handling process itself, where a response or Resolution is explicitly or implicitly expected by the Consumer.
>
> An initial call to a provider to request a service or information or to request support is not necessarily a Complaint. An initial call to report a fault or service difficulty is not a Complaint. However, if a Customer advises that they want this initial call treated as a Complaint, the Supplier will also treat this initial call as a Complaint.
>
> If a Supplier is uncertain, a Supplier must ask a Customer if they wish to make a Complaint and must rely on the Customer's response.

Here is a (simplified) formal representation:

[2] More correctly, we should use $r\colon \{a_1,\ldots,a_n\} \hookrightarrow c$. However, to improve readability, we drop the set notation for the antecedent of rule.

$$tcpc_1: ExpressionDissatisfaction \Rightarrow Complaint$$
$$tcpc_2: InformationCall \Rightarrow \neg Complaint$$
$$tcpc_3: ProblemCall, FirstCall \leadsto Complaint$$
$$tcpc_4: AdviseComplaint \Rightarrow Complaint$$

where $tcpc_1 \prec tcpc_2$ and $tcpc_2 \prec tcpc_4$.

The first rule $tcpc_1$ sets the basic conditions for something to be a complaint. On the other hand, rule $tcpc_2$ provides an exception to the first rule, and rule $tcpc_4$ is an exception to the exception provided by rule $tcpc_2$. Finally, $tcpc_3$ does not alone warrant the call to be a complaint (though, it does not preclude the possibility that the call turns out to be a complaint; hence the use of a defeater to capture this case).

Defeasible Logic is a constructive logic. This means that at the heart of it we have its prof theory, and for every conclusion we draw from a defeasible theory we can provide a proof for it, giving the steps used to reach the conclusion, and at the same time, providing a (formal) explanation or justification of the conclusion. Furthermore, the logic distinguishes *positive* and *negative* conclusion, and the strength of a conclusion. This is achieved by labelling each step in a derivation with a proof tag. As usual a derivation is a (finite) sequence of formulas, each obtained from the previous ones using inference conditions.

Let D be a Defeasible Theory. The following are the proof tags we consider for basic Defeasible Logic:

$+\Delta$ if a literal p is tagged by $+\Delta$, then this means that p is provable using only the facts and strict rules in a defeasible theory. We also say that p is *definitely provable* from D.

$-\Delta$ if a literal p is tagged by $-\Delta$, then this means that p is refuted using only the facts and strict rules in a defeasible theory. In other terms, it indicates that the literal p cannot be proved from D using only facts and strict rules. We also say that p is *definitely refuted* from D.

$+\partial$ if a literal p is tagged by $+\partial$, then this means that p is *defeasibly provable* from D.

$-\partial$ if a literal p is tagged by $-\partial$, then this means that p is *defeasibly refutable* from D.

Some more notation is needed before explaining how tagged conclusions can be asserted. Given a set of rules R, we use R_x to indicate particular subsets of rules: R_s for strict rules, R_d for defeasible rules, R_{sd} for strict or defeasible rules, R_{dft} for defeaters; finally $R[q]$ denotes the rules in R whose conclusion is q.

There are two ways to prove $+\Delta p$ at the n-th step of a derivation: the first is that p is one of the facts of the theory. The second case is when we have a strict rule r for p and all elements in the antecedent of r have been definitely proved at previous steps of the derivation.

For $-\Delta p$ we have to argue that there is no possible way to derive p using facts and strict rules. Accordingly, p must not be one of the facts of the theory, and second for every rule in $R_s[p]$ (all strict rules which are able to conclude p) the rule cannot be applied, meaning that at least one of the elements in the antecedent of the rule has already refuted (definitely refuted). The base case is where the literal to be refuted is not a fact and there are no strict rules having the literal as their head.

Defeasible derivations have a three phases argumentation like structure[3]. To show that $+\partial p$ is provable at step n of a derivation we have to:[4]

1. give an argument for p;
2. consider all counterarguments for p; and
3. rebut each counterargument by either:
   a. showing that the counterargument is not valid;
   b. providing a valid argument for p defeating the counterargument.

In this context, in the first phase, an argument is simply a strict or defeasible rule for the conclusion we want to prove, where all the elements are at least defeasibly provable. In the second phase we consider all rules for the opposite or complement of the conclusion to be proved. Here, an argument (counterargument) is not valid if the argument is not supported.[5] Here "supported" means that all the elements of the body are at least defeasibly provable.

Finally to defeasibly refute a literal, we have to show that either, the opposite is at least defeasible provable, or show that an exhaustive search for a constructive proof for the literal fails (i.e., there are rules for such a conclusion or all rules are either 'invalid' argument or they are not stronger than valid arguments for the opposite).

Consider again the set of rules encoding the TCPC 2012 definition of complaint. Assume to have a situation where there is an initial call from a customer who is dissatisfied with some aspects of the service received so far where she asks for some information about the service. In this case rules $tcpc_1$ and $tcpc_2$ are both applicable (we assume that the facts of the case include the union of the premises of the two rules, but *AdviseComplaint* is not a fact). Here, $tcpc_2$ defeats $tcpc_1$, and $tcpc_4$ cannot be used. Hence, we can conclude $-\partial AdviseComplaint$ and consequently $+\partial \neg Complaint$ and $-\partial Complaint$. However, if the customer stated that she wanted to complain for the service, then the fact *AdviseComplaint* would appears in the facts. Therefore we can conclude $+\partial AdviseComplaint$, making then rule $tcpc_4$ applicable, and we can reverse the conclusions, namely: $+\partial Complaint$ and $-\partial \neg Complaint$.

---

[3] The relationships between Defeasible Logic and argumentation are, in fact, deeper than the similarity of the argumentation like proof theory. Governatori et al. [2004] prove characterisation theorems for defeasible logic variants and Dung style argumentation semantics [Dung, 1995]. In addition Governatori [2011] proved that the Carneades argumentation framework [Gordon et al., 2007], widely discussed in the AI and Law literature, turns out to be just a syntactic variant of Defeasible Logic.

[4] Here we concentrate on proper defeasible derivations. In addition we notice that a defeasible derivations inherit from definite derivations, thus we can assert $+\partial p$ if we have already established $+\Delta p$.

[5] It is possible to give different definition of support to obtain variants of the logic tailored for various intuitions of non-monotonic reasoning. Billington et al. [2010] show how to modify the notion of support to obtain variants capturing such intuitions, for example by weakening the requirements for a rule to be supported: instead of being defeasibly provable a rule is supported if it is possible to build a reasoning chain from the facts ignoring rules for the complements.

While the Defeasible Logic we outlined in this section and its variants are able to model different features of legal reasoning (e.g., burden of proof [Governatori and Sartor, 2010] and proof standards [Governatori, 2011] covering and extending the proof standards discussed in [Gordon et al., 2007]), we believe that a few important characteristics of legal reasoning are missing. First we do not address the temporal dimension of norms (and, obviously, this is of paramount importance to model norm dynamics), and second, we do not handle the normative character of norms: norms specify what are the obligations, prohibitions and permissions in force and what are the conditions under which they are in force. In the next sections we are going to extend Defeasible Logic with (1) deontic operators, to capture the normative nature of norms and (2) time, to model the temporal dimensions used in reasoning with norms.

x.2.2 Defeasible Deontic Logic

Norms in a normative system can have (among others, but typically) the following functions:

1. to define the terms and concepts used in the normative system, and
2. to prescribe the behaviours the subjects of the normative system are meant to comply with.

The distinction just introduced is that of *constitutive* rules and *prescriptive rules*. The "mode" of the behaviours prescribed by the prescriptive rules is determined by deontic modalities (e.g., obligation, prohibition, permission). The Defeasible Logic presented in the previous section accounts for constitutive rules. To model prescriptive rules we have (i) to extend the language with deontic operators (ii) to use again the idea that rules are just binary relations and add a dimension, that is the *mode*, in the classification of rules. Hence rules can be classified according to their strength as well as their mode.

In this contribution we concentrate on the following deontic operators: O, P and F, respectively for obligation, permission and prohibition. In the language of Defeasible Deontic Logic the set of literals Lit is partitioned in *plain literals* and *deontic literals*. A plain literal is a literal in the sense of basic defeasible logic, while a deontic literal is obtained by placing a plain literal in the scope of a deontic operator or a negated deontic operator. Accordingly, expressions like $\mathsf{O}l$, $\neg\mathsf{P}l$ and $\mathsf{F}\neg l$ are deontic literals, where l is plain literal.

In Defeasible Deontic Logic rules are defined with the following signature

$$\mathrm{Rule}\colon 2^{\mathrm{Lit}} \times \mathrm{PlainLit} \tag{5}$$

where PlainLit is the set of all plain literals. This means that the antecedent of a rule can contain both plain and deontic literal, but in any case the conclusion is plain literal. Thus the question is if the conclusions of rules are plain literals, where do we get deontic literals? The answer is that we have two different modes of for the rules.

The first mode is that of constitutive rule, where the conclusion is an assert with the same mode as it appears in the rule (i.e., as an institutional fact); the second mode is that of prescriptive rule, where the conclusion is asserted with a deontic mode (where the deontic mode corresponds to one of the deontic operators). Accordingly, a Defeasible Deontic Theory is a structure

$$(F, R_C, R_{\mathsf{O}}, \prec) \tag{6}$$

where R_C is a set of constitutive rules, and R_{O} is a set of prescriptive rules. Constitutive rules behaves as the rules in Basic Defeasible Logic, and we continue to use $\hookrightarrow$ to denote the arrow of a constitutive rule. $\hookrightarrow_{\mathsf{O}}$ for the arrow of a prescriptive rule.

The main idea is that given the constitutive defeasible rule

$$a_1, \ldots, a_n \Rightarrow_C b \tag{7}$$

we can assert b, given $a_1, \ldots, a_n$, thus the behaviour of constitutive rule is just the normal behaviour of rules we examined in the previous section. For prescriptive rules the behaviour is a different. From the rule

$$a_1, \ldots, a_n \Rightarrow_{\mathsf{O}} b \tag{8}$$

we conclude $\mathsf{O}b$ when we have $a_1, \ldots, a_n$. Thus we conclude the obligation of the conclusion of the rule, not just the conclusion of the rule.

The reasoning mechanism is essentially the same as that of basic defeasible presented in Section x.2.1. The differences are an argument can only be attacked by an argument of the same type. Thus if we have an argument consisting of a constitutive rule for p, a counterargument should be a constitutive rule for $\sim p$. The same applies for prescriptive rule. An exception to this is when we have a constitutive rule for p such that all its premises are provable as obligations. In this case the constitutive rule behaves like a prescriptive rule, and can be use as a counterargument for a prescriptive rule for $\sim p$, or the other way around. The second difference is that now the proof tags are labelled with either C, e.g., $+\partial_C p$, (for constitutive conclusions) or with O, e.g., $-\partial_{\mathsf{O}} q$ (for prescriptive conclusions). Accordingly, when we are able to derive $+\partial_{\mathsf{O}} p$ we can say that $\mathsf{O}p$ is provable.

This feature poses the question of how we model the other deontic operators (i.e., permission and prohibition). As customary in Deontic Logic, we assume the following principles governing the interactions of the deontic operators.[6]

$$\mathsf{O}{\sim}l \equiv \mathsf{F}l \tag{9}$$

$$\mathsf{O}l \wedge \mathsf{O}{\sim}l \rightarrow \bot \tag{10}$$

$$\mathsf{O}l \wedge \mathsf{P}{\sim}l \rightarrow \bot \tag{11}$$

Principle (9) provides the equivalence of a prohibition with a negative obligation (i.e., obligation not). The second and the third are rationality postulates stipulating that it

[6] In the three formulas below $\rightarrow$ is the material implication of classical logic.

is not possible to have that something and its opposite are at the same time obligatory (10) and that a normative system makes something obligatory and its opposite is permitted (11). (9) gives us the immediate answer on how prohibition is modeled. A rule giving a prohibition can be modelled just as a prescriptive rule for a negated literal. This means that to conclude $\mathsf{F}p$ we have to derive $+\partial_{\mathsf{O}}\neg p$.

Consider Section 40 of the Australian Road Rules (ARR)[7]

Making a U–turn at an intersection with traffic lights
A driver must not make a U–turn at an intersection with traffic lights unless there is a U–turn permitted sign at the intersection.

The prohibition of making U-turns at traffic lights can be encoded by the following rule:

$$arr_{40a}: AtTrafficLigths \Rightarrow_{\mathsf{O}} \neg Uturn.$$

In a situation where *AtTrafficLights* is given we derive $+\partial_{\mathsf{O}}\neg Uturn$ which corresponds to $\mathsf{F}Uturn$.

The pending issue is how to model permissions. Two types of permissions have been discussed in literature following von Wright [1963] and Alchourrón and Bulygin [1984]: (i) weak permission, meaning that there is no obligation to the contrary; and (ii) strong permission, a permission explicitly derogates an obligation to the contrary. In this case we have an exception. For both types of permission we have that the obligation to the contrary does not hold. Defeasible Deontic Logic is capable to handle the two types of permission is a single shot if we establish that $\mathsf{P}p$ is captured by $-\partial_{\mathsf{O}}{\sim}p$. The meaning of $-\partial_{\mathsf{O}}p$ is that p is refuted as obligation, or that it is not possible to prove p as an obligation; hence it means that we cannot establish that p is obligatory, thus there is no obligation contrary to ${\sim}p$.

The final aspect we address is how to model strong permissions. Remember that strong permissions are meant to be exceptions. Exceptions in Defeasible Logic can be easily captured by rules for the opposite plus a superiority relation. Accordingly, this could be modelled by

$$arr_{40e}: UturnPermittedSign \Rightarrow_{\mathsf{O}} Uturn.$$

and $arr_{40a} \prec arr_{40e}$. We use a prescriptive defeasible rule for obligation to block the prohibition to U-turn. But, since arr_{49e} prevails over arr_{49a}, we derive that U-turn is obligatory, i.e., $+\partial_{\mathsf{O}}Uturn$.

Thus, when permissions derogate to prohibitions (or obligations), there are good reasons to argue that defeaters for O are suitable to express an idea of strong permission[8]. Explicit rules such as $r: a \rightsquigarrow_{\mathsf{O}} q$ state that a is a specific reason for blocking the derivation of $\mathsf{O}\neg q$ (but not for proving $\mathsf{O}q$), i.e., this rule does not support any

[7] This norm makes use of "must not", to see that "must not" is understood as prohibition in legal documents see, the Australian National Consumer Credit Protection Act 2009, Section 29, whose heading is "Prohibition on engaging in credit activities without a licence", recites "(1) A person must not engage in a credit activity if the person does not hold a licence authorising the person to engage in the credit activity".

[8] The idea of using defeaters to introduce permissions was introduced by Governatori et al. [2005b].

conclusion, but states that $\neg q$ is deontically undesirable. Accordingly, we can rewrite the derogating rule as

$$arr_{40e}: UturnPermittedSign \rightsquigarrow_{\mathsf{O}} Uturn.$$

In this case, given *UturnPermittedSign* we derive $-\partial_{\mathsf{O}}\neg Uturn$.

For an in-depth presentation of Defeasible Deontic Logic, its properties and a detailed analysis of how to use it to model obligations and permissions (and several ways to do it) we refer the reader to [Governatori et al., 2013a].

x.2.3 Defeasible Deontic Logic with Time

The extension of Defeasible Logic with deontic operators makes the the logic more expressive and more capable of representing aspects of legal reasoning insofar as it allows us to consider the important distinction between constitutive rules and prescriptive rules, and to differentiate among normative effects. However, a key element is still missing: time. Very often norms have temporal parameters and Deontic Defeasible Logic is not able to reasons about them. In this section we are going to extend the logic with temporal parameters. In particular we are going to *temporalise* the logic. This means that we attach a temporal parameter to the atomic elements of the logic, i.e., to the atomic propositions. For the logic we assume a discrete totally ordered set of instants of time $\mathcal{T} = \{t_0, t_1, t_2, \dots\}$. Based on this we can introduce the notion of *temporalised literals*. Thus if l is a plain literal, i.e., $l \in$ PlainLit, and $t \in \mathcal{T}$ then l^t is a temporalised literals. The intuitive interpretation of l^t is that l is true (or holds) at time t. We use TempLit to denote the set of temporalised literals. Deontic literals are now obtained from temporalised literals using the same conditions as in Section x.2.2; thus a deontic literal is an expression like $\mathsf{O}l^t$, where its natural reading is that l is obligatory at time t, or that the obligation of l is in force at time t. Finally, given a time instant t and $y \in \{pers, tran\}$ we call the combination of (t, y) *duration specification*, and literals labelled with a duration specification duration literals. A duration literal has the form $l^{(t,y)}$. We denote the set of duration literals DurLit. The set of literals is now composed by the set of temporalised literals and the set of deontic literals, namely Lit = DeonLit ∪ TempLit. The signature of rules is now

$$\text{Rule}\colon 2^{\text{Lit}} \times \text{DurLit} \tag{12}$$

this means that a rule has the following form

$$r\colon a_1^{t_1}, \dots, a_n^{t_n} \hookrightarrow_X c^{(t,y)} \tag{13}$$

where $X \in \{C, \mathsf{O}\}$, specifying whether the rules is a constitutive or a prescriptive one, and $y \in \{tran, pers\}$ indicating whether the conclusion of the rule is either *transient* or *persistent*.

The idea behind the distinction between a transient and persistent conclusion is whether the conclusion is guaranteed to hold for a single instant or it continues to hold until it is terminated. This is particular relevant for prescriptive rules, since their conclusions are obligations (or, in general deontic effects), and obligations, once triggered, remain in force until they are complied with, violated, or explicitly terminated. Accordingly we can use the duration specification $(t, tran)$ to indicate that on obligation is in force at a specific time t, and must be fulfilled at that time, while the duration specification $(t, pers)$ establishes that an obligation enters in force at time t.

The inference mechanism extends that of Defeasible Deontic Logic taking into account the temporal and durations specification. To assert that p holds at time t we have two ways:

1. Give an argument for p at time t';[9]
2. Evaluate all counterarguments against it. Here, we have a few cases:
   a. If the duration specification of p is $(t, tran)$ $(t' = t)$, then, the counterargument must be for the same time t given that p is ensured to hold only for t.
   b. If the duration specification of p is $(t', pers)$, then t' can precede t and we can 'carry' over the conclusion from previous times. In this case, the counterarguments we have to consider are all rules whose conclusion has a duration specification (t'', z) such that $t' \leq t'' \leq t$.
3. Rebut the counterarguments. This is the same as the corresponding step of basic defeasible logic, the only thing to pay attention to is that when we rebut with a stronger argument, the stronger argument should have t'' in the duration specification of the conclusion.

The general idea of the conditions outline above is that, as we have already alluded to, it is possible to assert that something holds at time t, because it did hold at time t', $t' < t$, by persistence, but there must be no reasons to terminate it. Thus new information defeats previous one.

To illustrate the intuition we just described consider Section 8.2.1.a of the Australian Telecommunications Consumers Protection Code 2012 (TCPC 2012).

> A Supplier must take the following actions to enable this outcome:
>
> (a) **Demonstrate fairness, courtesy, objectivity and efficiency:** Suppliers must demonstrate, fairness and courtesy, objectivity, and efficiency by:
>
> (i) Acknowledging a Complaint:
>
> A. immediately where the Complaint is made in person or by telephone;
>
> B. within 2 Working Days of receipt where the Complaint is made by email; ....

The normative fragment above can be represented by the following set of rules:

[9] We equate arguments with rules, thus this is the same as saying that there is (defeasible) rule such that all the elements in its antecedent are provable and the conclusion is $p^{(t', y)}$.

$$tcpc_1 : Complaint^t, inPerson^t \Rightarrow_{\mathsf{O}} Acknowledge^{(t,tran)}$$
$$tcpc_2 : Complaint^t \Rightarrow_{\mathsf{O}} Acknowledge^{(t,pers)}$$
$$tcpc_3 : Complaint^t \leadsto_{\mathsf{O}} \neg Acknowledge^{(t+2d,tran)}$$

Rule $tcpc_1$ covers the case of a complaint made in person of by phone. Given that the complaint must be acknowledged immediately, we can use the duration specification $(t, tran)$, where t is the time when the complaint is received. The *tran* specification implies that the obligation to acknowledge the complain is in force only at t and not acknowledging at t results in a violation. For the case regulated by paragraph B, we use two rules. The first $tcpc_2$ is to initiate the obligation (at the same time t when the complaint is received, while $tcpc_3$ gives the deadline by when the content of the obligation must be fulfilled. Notice that we use a defeater to terminate the obligation.

Suppose we have a complaint by email on day 10. From this we can derive $+\partial_{\mathsf{O}} Acknowledge^{10}$ from rule $tcpc_2$. By persistence we have that $+\partial_{\mathsf{O}} Acknowledge^{11}$. On day 12 the effect of rule $tcpc_3$ kicks in, and we have $-\partial_{\mathsf{O}} Acknowledge^{12}$.

For thorough presentations of temporal defeasible logic, its properties and application to modelling obligation with time and deadlines we refer the reader to [Governatori et al., 2005b, 2007a, Governatori and Rotolo, 2013].

x.2.4 From Rules to Meta-Rules

The temporal Defeasible Logic just presented allows us to reasons about the times specified inside norms, but it is not able to capture the lifecycle of norms. To obviate this problem Governatori and Rotolo [2010] propose to consider a legal system as a time-series of its versions, where each version is obtained from previous versions by some norm changes, e.g., norms entering in the legal system, modification of existing norms, repeals of existing norms, ... This means that we can represent a legal system LS as a sequence

$$LS(t_1), LS(t_2), \ldots, LS(t_j) \tag{14}$$

where each $LS(t_i)$ is the snapshot of the rules (norms) in the legal system at time t_i. Graphically it can be represented by the picture in Figure x.1

A *rule* is a relation between a set of premises (conditions of applicability of the rule) and a conclusion. In this paper the admissible conclusions are either literals or rules themselves; in addition the conclusions and the premises will be qualified with the time when they hold. We consider two classes of rules: *meta-rules* and *proper rules*. Meta-rules describe the inference mechanism of the institution on which norms are formalised and can be used to establish conditions for the creation and modification of other rules or norms, while proper rules correspond to norms in a normative system. In what follows we will use Rule to denote the set of rules, and MetaRules for the set of meta-rules, i.e., rules whose consequent is a rule.

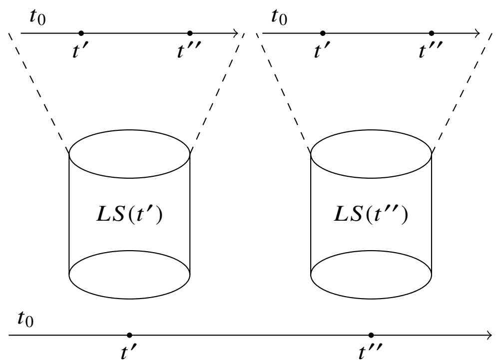


Fig. x.1 Legal System at t' and t''

A *temporalised rule* is either an expression $(r\colon \bot)^{(t,x)}$ (the void rule) or $(r\colon \emptyset)^{(t,x)}$ (the empty rule) or $(r\colon A \hookrightarrow_X B)^{(t,x)}$, where r is a rule label, A is a (possibly empty) set of temporalised literals, $X \in \{C, \mathsf{O}\}$, B is a duration literal, $t \in \mathcal{T}$ and $x \in \{tran, pers\}$.

We have to consider two temporal dimensions for norms in a normative system. The first dimension is when the norm is in force in a normative system, and the second is when the norm exists in the normative system from a certain viewpoint. So far temporalised rules capture only one dimension, the time of force. To cover the other dimension we introduce the notion of temporalised rule with viewpoint. A *temporalised rule with viewpoint* is an expression

$$(r\colon A \hookrightarrow_X B)^{(t,x)}@(t', y), \tag{15}$$

where $(r\colon A \hookrightarrow_X B)^{(t,x)}$ is a temporalised rule, $t' \in \mathcal{T}$ and $y \in \{tran, pers\}$.

Finally, we introduce meta-rules, that is, rules where the conclusion is not a simple duration literal but a temporalised rule. Thus a *meta-rule* is an expression

$$(s\colon A \hookrightarrow (r\colon B \hookrightarrow_X C)^{(t',x)})@(t, y), \tag{16}$$

where $(r\colon B \hookrightarrow_X C)^{(t',x)}$ is a temporalised rule, $r \neq s$, $t \in \mathcal{T}$ and $y \in \{tran, pers\}$. Notice that meta-rules carry only the viewpoint time (the validity time) but not the "in force" time. The intuition behind this is that meta-rules yield the conditions to modify a legal system. Thus they specify what rules (norms) are in a normative system, at what time the rules are valid, and the content of the rules. Accordingly, these rules must have an indication when they have been inserted in a normative system, but then they are universal (i.e., apply to all instants) within a particular instance of a normative system.

Every temporalised rule is identified by its rule label and its time. Formally we can express this relationship by establishing that every rule label r is a function

$$r\colon \mathcal{T} \mapsto \text{Rule}. \tag{17}$$

Thus a temporalised rule r^t returns the value/content of the rule 'r' at time t. This construction allows us to uniquely identify rules by their labels[10], and to replace rules by their labels when rules occur inside other rules. In addition there is no risk that a rule includes its label in itself. In the same way a temporalised rule is a function from $\mathcal{T}$ to Rule, we will understand a temporalised rule with viewpoint as a function with the following signature:

$$\mathcal{T} \mapsto (\mathcal{T} \mapsto \text{Rule}). \tag{18}$$

As we have seen above a legal system LS is a sequence of versions $LS(t_0), LS(t_1), \dots$. The temporal dimension of viewpoint corresponds to a version while the temporal dimension temporalising a rule corresponds to the time-line inside a version. Thus the meaning of an expression $r^{t_v}@t_r$ is that we take the value of the temporalised rule r^{t_v} in $LS(t_r)$. Accordingly, a version of LS is just a repository (set) of norms (implemented as temporal functions).

Accordingly, given a rule r, the expression $r^t@t'$ gives the value of the rule (set of premises and conclusion of the rule) at time t in the repository t'. The content of a void rule, e.g., $(r: \bot)^t@t'$ is $\bot$, while for the empty rule the value is the empty set. This means that the void rule has a value for the combination of the temporal parameters, while for the empty rule, the content of the rule does not exist for the given temporal parameters. Another way to look at the difference between the empty rule and the void rule is to consider that a rule is a relationship between a set of premises and a conclusion. For the void rule this relationship is between the empty set of premises and the empty conclusion; thus the rule exists but it does not produce any conclusion. For the empty rule, the relationship is empty, thus there is no rule. Alternatively, we can think of the function corresponding to temporalised rules as a partial function, and the empty rule identifies instants when the rule is not defined.

For a transient fully temporalised literal $l^{(t,x)}@(t', tran)$ the reading is that the validity of l at t is specific to the legal system corresponding to repository associated to t', while $l^{(t,x)}@(t', pers)$ indicates that the validity of l at t is preserved when we move to legal systems after the legal system identified by t'. An expression $r^{(t,tran)}$ sets the value of r at time t and just at that time, while $r^{(t,pers)}$ sets the values of r to a particular instance for all times after t (t included).

We will often identify rules with their labels, and, when unnecessary, we will drop the labels of rules inside meta-rules. Similarly, to simplify the presentation and when possible, we will only include the specification whether an element is persistent or transient only for the elements for which it is relevant for the discussion at hand.

Meta-rules describe the inference mechanism of the institution on which norms are formalised and can be used to establish conditions for the creation and modification of other rules or norms, while proper rules correspond to norms in a normative system. Thus a temporalised rule r^t gives the 'content' of the rule 'r' at time t; in legal terms it tells us that norm r is in force at time t. The expression

[10] We do not need to impose that the function is an injective: while each label should have only one content at any given time, we may have that different labels (rules) have the same content.

$$(p^{t_p}, q^{t_q} \Rightarrow (p^{t_p} \Rightarrow_{\mathsf{O}} s^{(t_s,pers)})^{(t_r,pers)})@(t, tran) \tag{19}$$

means that, for the repository at t, if p is true at time t_p and q at time t_q, then $p^{t_p} \Rightarrow_{\mathsf{O}} s^{(t_s,pers)}$ is in force from time t_r onwards.

A legal system is represented by a temporalised defeasible theory, called *normative theory*, i.e., a structure

$$(F, R, R^{\text{meta}}, \prec) \tag{20}$$

where F is a finite set of facts (i.e., fully temporalised literals), R is a finite set of prescriptive and constitutive rules, R^{meta} is a finite set of meta rules, and $\prec$, the superiority relation over rules is formally defined as $\mathcal{T} \mapsto (\mathcal{T} \mapsto \text{Rule} \times \text{Rule})$. accounting that we can have different instances of the superiority relation depending on the legal systems (external time) and the time when the rules involved in the superiority are evaluated[11].

In the current logic a conclusion has a form like: $+\partial t@t'\ p^{t_p}$, meaning that the conclusion that p holds at time t_p is derivable at time t using the information included in the version of the legal system at time t'.

The inference mechanism with meta-rules is essentially an extension of that of temporal defeasible logic, but it involves more steps. Rules are no longer just given, but they can be derived from meta-rules. Thus to prove $+\partial t@t' p^{t_p}$ the first thing to do is to see if it is possible to derive a rule r having p^{t_p} as its head. But we have to derive such rule at the appropriate time. Here, we want to remember that a rule is a function from time (validity time or version of a legal system) to time (when a rule is in force in a version of a legal system) to the content of the rule (relationship between a set of premises and a conclusion). The basic intuition is that a rule corresponds to a norm, and there could be several modifications of a norm, thus deriving a rule means to derive one of such modifications. As we shall see in the next section a meta-rule (or more generally a set of meta-rules) can be used to encode a modification of a norm. In general it is possible to have multiple (conflicting) modifications of a norm. Accordingly, to derive a rule, we have to check that there are no conflicting modifications[12] or the conflicting modifications are weaker than the current modification. The final consideration is that in this case we have two temporal dimensions, and the persistence applies to both. Thus we can have persistence inside a legal system, thus we can conclude $+\partial t''@t'\ p^{t_p}$ from $+\partial t@t'\ p^{t_p}$, where $t < t''$ as well as persistence over versions, thus $+\partial t@t''\ p^{t_p}$ from $+\partial t@t'\ p^{t_p}$, where $t' < t''$.

[11] For instance, if we have $s \prec^{2007}_{Monday} r$ and $r \prec^{2007}_{Tuesday} s$, it means that, according to the regulation in force in 2007, on Monday rule s is stronger than rule r, but on Tuesday r is stronger than s.

[12] Two meta-rules are conflicting, when the two meta-rules have the same rule as their head, but with a different content.

x.3 Modelling Legal Changes

x.3.1 Types of Legal Change

Norm changes in the law can be explicit or implicit [Governatori et al., 2005a, 2007b, Governatori and Rotolo, 2010]:

Explicit: The law introduces norms whose peculiar objective is to change the system by specifying what and how other existing norms should be modified;

Implicit: the legal system is revised by introducing new norms which are not specifically meant to modify previous norms, but which change in fact the system because they are incompatible with such existing norms and prevail over them. (The new norms prevail because, for example, have a higher ranking status in the hierarchy of the legal sources or because have been subsequently enacted.)

The most interesting case is when we deal with explicit modifications, which permit to classify a large number of modification types. In general, we have different types of modifying norms, as their effects (the resulting modifications) may concern, for example, the text of legal provisions, their scope, or their time of force, efficacy, or applicability, or their own existence or validity [Guastini, 1998, Governatori et al., 2005a, 2007b].

Derogation is an example of scope change: a norm n supporting a conclusion p and holding at the national level may be derogated by a norm n' supporting a different conclusion p' within a regional context. Hence, derogation corresponds to introducing one or more exceptions to n. Temporal changes impact on the target norm in regard to its date of force (the time when the norm is "usable"), date of effectiveness (when the norm in fact produces its legal effects) or date of application (when conditions of norm applicability hold). An example of change impacting on time of force is when a norm n is originally in force in 2007 but a modification postpones n to 2008. Substitution is an example of textual modification, as it generically replaces some textual components of a provision with other components. For instance, some of its applicability conditions are replaced by other conditions. Finally, we have modifications on norm validity and existence, such as abrogation and annulment. For instance, an annulment is usually seen as a kind of repeal, as it makes a norm invalid and removes it from the legal system. As we will see, its peculiar effect applies *ex tunc*: annulled norms are prevented to produce all their legal effects, independently of when they are obtained.

x.3.2 Modifications of Scope: Derogation

Derogations are modifications of norm scope. A fictional example from the Italian constitution (enacted in 1948) is the following:

Example 1 (Derogation).

[**Target of the modification**] Article 3 (1) All citizens have equal social status and are equal before the law, without regard to their sex, race, language, religion, political opinions, and personal or social conditions.

[**Modification enacted in 2014 and effective in 2015**] In derogation to the provisions set out in Article 3, paragraph 1, of the Constitution, the citizens who are resident in Bologna may have different social status, but this modification will be effective only in 2015, when Italy will be no longer in EU.

From the logical point of view, derogation can be simply modeled by adding exceptions, in particular defeaters. Using meta-rules, Example 1 can be captured as follows[13].

Example 2 (Derogation (cont'd)). Let $D = (F, R, R^{meta}, \prec)$ be a normative theory such that

$$Art.\ 3 : (Citizen^x \Rightarrow_{\bigcirc} Equal\_status^x)^{(1948,pers)} @ (1948, tran) \in R$$

Example 1 is modeled by stating that R^{meta} includes the following meta-rule

$$derog_{Art.3} : (\sim EU^x \Rightarrow (r' : Citizen^x, Resident\_Bologna^x \rightsquigarrow_{\bigcirc} \sim Equal\_status^x)^{(2015,pers)}) @ (2014, pers)$$

and that $\prec$ is as follows (where $t \geq 2015$)[14]:

$$\{s \prec^{2014}_{2015} r' : s \in R[Equal\_status^x] \text{ and } A(s) \cap \partial^-(D) \neq \emptyset\} \in \prec$$
$$\{mr \prec^{2014}_{2015} derog_{Art.3} : mr \in R^{meta}[\sim r^t] \text{ and } A(mr) \cap \partial^-(D) \neq \emptyset\} \in \prec$$

Notice that the above conditions on $\prec$ ensures that this operation minimises the impact of the added meta-rule and the related defeater. In fact, the operation works on art. 3 (and any other similar provision) only when any conflicting meta-rule and art. 3 are applicable.

x.3.3 Textual Modifications: Substitution

Consider a textual modification such as substitution, which typically replaces some textual components of a provision with other textual components. Another fictional (but this time reasonable!) example from the Italian constitution is the following:

Example 3 (Substitution).

[**Target of the modification**] Article 3 (1) All citizens have equal social status and are equal before the law, without regard to their sex, race, language, religion, political opinions, and personal or social conditions.

---

[13] In the remainder of the paper, when temporal parameters are not essential we will not specify them and will just add a superscript x.

[14] Recall that, for any rule s, $A(s)$ denotes the set of antecedents of s, while $\partial^-(D)$ stands for the set of negative conclusions of the theory D. i.e., the literals occurring in conclusions of the form $-\partial$.

[**Modification enacted and effective in 2014**] In the Article 3, paragraph 1 of the Italian constitution the expression “citizens” is replaced with “human beings".

This can be represented by the normative theory $D = (F, R, R^{meta}, \prec)$ such that

$$Art.\ 3 : (Citizen^x \Rightarrow_{\mathsf{O}} Equal\_status^x)^{(1948,pers)} @ (1948, tran) \in R,$$

the substitution is modelled by the following meta-rule in R^{meta}

$$sub_{Art.\ 3} : (\Rightarrow (Art.\ 3 : Human\_being^x \Rightarrow_{\mathsf{O}} Equal\_status^x)^{(2014,pers)}) @ (2014, pers)$$

and $\prec$ is as follows (where $t \geq 2014$):

$\{s \prec^{2014}_{2014} Art.\ 3^{2014} : s \in R[Equal\_status^x] \text{ and } A(s) \cap \partial^-(D) \neq \emptyset\} \in \prec$
$\{sub_{Art.\ 3} \prec^{2014}_{2014} Art.\ 3^{2013}\} \in \prec$
$\{mr \prec^{2014}_{t} sub^{2014}_{Art.\ 3}, mr \in R^{meta}[\sim Art.\ 3^{2014}] \text{ and } A(mr) \cap \partial^-(D) \neq \emptyset\} \in \prec .$

x.3.4 Temporal Modifications

Temporal modifications are performed by meta-rules that change norms in regard to their time of force, efficacy, or applicability. Consider this example:

Example 4 (Temporal modification).

[**Target of the modification**] Legislative Act n. 124, 23 July 2008.
[. . .]
Art. 8. This legislative act is in force since the date of publication of the *Gazzetta Ufficiale* [23 August 2008]

[**Modification enacted and effective at 1 August 2008**] Legislative Act n. 124, 23 July 2008 is in force since 1 January 2009.

Example 4 is reconstructed as follows.

Example 5 (Temporal modification (cont'd)). For the sake of simplicity, assume that the content of Legislative Act n. 124 is $a^x_1, \ldots a^x_n \Rightarrow_{\mathsf{O}} b^x$. Hence, we have that R^{meta} contains the following meta-rule modeling the enactment of Legislative Act n. 124;

$$mr : (\Rightarrow (L.\ 124 : a^x_1, \ldots a^x_n \Rightarrow_{\mathsf{O}} b^x)^{(\text{23 August 2008}, pers)}) @ (\text{23 July 2008}, pers).$$

The modification at hand is expressed by having in R^{meta} other two meta-rules mr' and mr'' such that

$$temp'_{L.\ 124} : (\leadsto \sim(L.\ 124 : a^x_1, \ldots a^x_n \Rightarrow_{\mathsf{O}} b^x)^{(\text{23 August 2008}, pers)}) @ (\text{1 August 2008}, pers)$$

$$temp''_{L.\ 124} : (\Rightarrow (L.\ 124 : a^x_1, \ldots a^x_n \Rightarrow_{\mathsf{O}} b^x)^{(\text{1 January 2009}, pers)}) @ (\text{1 August 2008}, pers)$$

such that $(temp'_{L.\ 124} \prec^{\text{23 August 2008}}_{\text{1 August 2008}} mr) \in \prec$.

x.3.5 Modifications on Norm Validity and Existence: Annulment vs. Abrogation

The expression *repeal* is sometimes used to generically denote the operation of norm withdrawal. However, at least two forms of withdrawal are possible: annulment and abrogation.

An *annulment* makes the target norm invalid and removes it from the legal system. Its peculiar effect applies *ex tunc*: annulled norms are prevented to produce all their legal effects, independently of when they are obtained. Annulments typically operate when the grounds (another norm) for annulling are hierarchically higher in the legal system than the target norm which is annulled: consider when a legislative provision is annulled (typically by the Constitutional Court) because it violates the constitution.

An *abrogation* works differently; the main point is usually that abrogations operate *ex nunc* and so do not not cancel the effects that were obtained from the target norm before the modification. If so, it seems that abrogations cannot operate retroactively. In fact, if a norm n_1 is abrogated in 2012, its effects are no longer obtained after then. But, if a case should be decided at time 2013 but the facts of the case are dated 2011, n_1, if applicable, will anyway produce its effects because the facts held in 2011, when n_1 was still in force (and abrogations are not retroactive). Accordingly, n_1 is still in the legal system, even though is no longer in force after 2012. Abrogations typically operate when the grounds (another norm) for abrogating is placed at the same level in the hierarchy of legal sources of the target norm which is abrogated: consider when a legislative provision is abrogated by a subsequent legislative act.

Consider this case:

Example 6 (Abrogation vs Annulment).

> [**Target of the modification**] Legislative Act n. 124, 23 July 2008
> Art. 1. With the exception of the cases mentioned under the Articles 90 and 96 of the Constitution, criminal proceedings against the President of the Republic, the President of the Senate, the President of the House of Representatives, and the Prime Minister, are suspended for the entire duration of tenure. [...]

In case of abrogation, we could have that the legislator enacts the following provision:

> [**Abrogation enacted and effective at 1 January 2011**] Legislative Act n. 124, 23 July 2008 is abrogated.

In case of (judicial) annulment, we would rather have

> [**Annulment enacted and effective at 1 January 2011**] On account of Art. 3 of the Constitution [...] the Constitutional Court hereby declares the constitutional illegitimacy of Art. 1 of the Act n. 124, 23 July 2008.

As we have recalled, the difference between the two cases is that the annulment has retroactive effects. In particular, let us focus on the following provisions from the Italian penal code:

Art. 157 Italian of Penal Code – Terms of statute-barred penal provisions.
When the the terms for statute-barred penal effects expire, the corresponding crime is canceled [. . .]
Art. 158 Italian Penal Code – Effectiveness of the terms of statute-barred penal provisions
The effectiveness of terms of statute-barred penal provisions begins starting from the time when the crime was committed.
Art. 159 Italian of Penal Code – Suspension of time limits for statute-barred penal effects.
The terms for statute-barred penal effects [. . .] are suspended whenever the criminal proceedings are suspended under any legislative provisions [. . .]

Consider a hypothetical case where the Italian Prime Minister is accused in 2007 of accepting bribes at the beginning of 2006. Clearly, if Legislative Act n. 124 is abrogated in 2011, since abrogation has no retroactive effects, art. 159 of Italian Penal Code applied from 2008 to 2011, and so the counting of terms has been suspended between these two years. Hence, from the perspective of 2011 (immediately after the abrogation) the relevant time passed is two years and six months (2006, 2007, and until July 2008). Instead, if the act is annulled in 2011, more time has passed from the perspective of 2011, because it is as if the Legislative Act n. 124 were never enacted: from 2006 until 2011.

As we can see, modeling retroactive legal modifications is far from obvious. The logical model proposed in [Governatori and Rotolo, 2010] and recalled in Section x.2.3 offers a solution. In the next section we will illustrate the intuition and apply to the above example of annulment and abrogation.

x.3.6 Intermezzo – Temporal Dynamics and Retroactivity

As we have previously argued, if $t_0, t_1, \ldots, t_j$ are points in time, the dynamics of a legal system LS can be captured by a time-series $LS(t_0), LS(t_1), \ldots, LS(t_j)$ of its versions. Each version of LS is like a norm repository: the passage from one repository to another is effected by legal modifications or simply by temporal persistence. This model is suitable for modeling complex modifications such as retroactive changes, i.e., changes that affect the legal system with respect to legal effects which were also obtained before the legal change was done.

The dynamics of norm change and retroactivity need to fully make use of the time-line within each version of LS (the time-line placed on top of each repository in Figure x.2). Clearly, retroactivity does not imply that we can really change the past: this is “physically” impossible. Rather, we need to set a mechanism through which we are able to reason on the legal system from the viewpoint of its current version but *as if* it were revised in the past: when we change some $LS(i)$ retroactively, this does not mean that we modify some $LS(k)$, $k < i$, but that we move back from the perspective of $LS(i)$. Hence, we can “travel” to the past along this inner time-line, i.e., from the viewpoint of the current version of LS where we modify norms.

Figure x.2 shows a case where the legal system LS and its norm r persist from time t' to time t'' and can have effects immediately from t'. Now, the figure represents the situation where r is retroactively repealed at t'' by stating that the modification

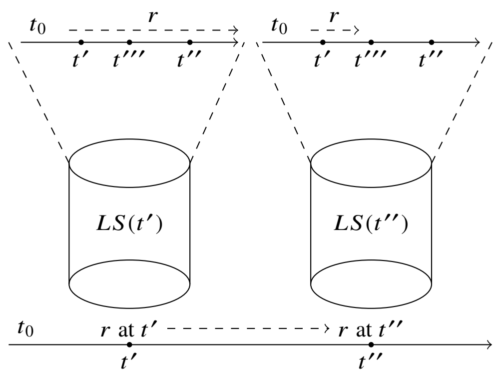
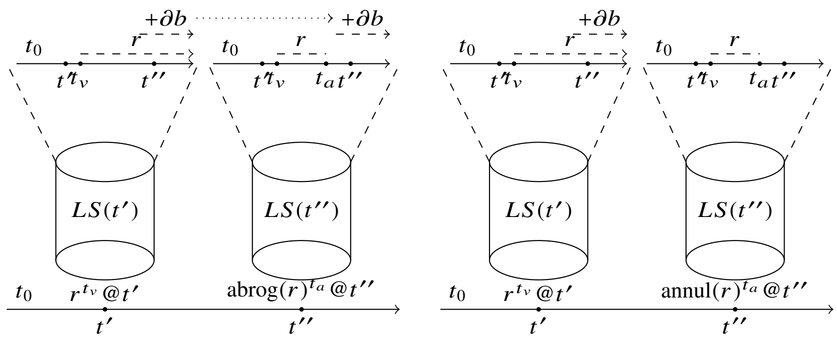


Fig. x.2 Legal System at t' and t''



(a) Abrogation. In $LS(t')$ rule r produces a persistent effect b. Literal b carries over by persistence to $LS(t'')$ even if r is no longer in force.

(b) Annulment. In $LS(t')$ rule r is applied and produces a persistent effect b. Since r is annulled in $LS(t'')$, b must be undone as well.

Fig. x.3 Abrogation and Annulment

applies from t''' (which is between t' and t'') onwards. The difference between abrogation and annulment is illustrated in Figures 3(a) and 3(b).

x.3.7 Modifications on Norm Validity and Existence: Annulment vs. Abrogation (Cont'd)

On account of our previous considerations, the cases in Example 6 can be reconstructed as follows.

Example 7 (Abrogation vs Annulment (cont'd)). First of all, for the sake of simplicity let us

- only consider the case of Prime Minister (Legislative Act n. 124 mentions other institutional roles),
- assume that the dates of enactment and effectiveness coincide and are generically 2008,
- the duration of tenure covers a time span from 2008 to 2012,

and formalize the corresponding fragment of art. 1 of Legislative Act n. 124 (23 July 2008) as follows:

$$\mathit{L.\ 124}\colon (\mathit{Crime}^{x}, \mathit{Tenure}^{x+y} \Rightarrow_{\mathsf{O}} \mathit{Suspended}^{(x+y,tran)})^{(2008,pers)})@(2008,pers)$$

The duration of tenure spanning from 2008 to 2012 is represented as follows:

$$\mathit{r1}\colon (\mathit{Elected}^{2008} \Rightarrow_{\mathsf{O}} \mathit{Tenure}^{(2008,pers)})^{(2008,pers)})@(2008,pers)$$
$$\mathit{r2}\colon (\mathit{Elected}^{2008} \rightsquigarrow_{\mathsf{O}} \neg\mathit{Tenure}^{2012})^{(2008,pers)})@(2008,pers)$$

Arts. 157-159 of the Italian Penal Code state the following:

$$\mathit{Art.\ 157}\colon (\mathit{Crime}^{x}, \mathit{Terms}^{x+y} \Rightarrow_{\mathsf{O}} \mathit{CrimeCancelled}^{(x+y,pers)})^{(z,pers)})@(z,pers)$$
$$\mathit{Art.\ 158}\colon (\mathit{Crime}^{x} \Rightarrow_{\mathsf{O}} \mathit{Terms}^{(x,pers)})^{(z,pers)})@(z,pers)$$
$$\mathit{Art.\ 159}\colon (\mathit{Crime}^{x}, \mathit{Suspended}^{x+y} \Rightarrow_{\mathsf{O}} \neg\mathit{Terms}^{(x+y,tran)})^{(z,pers)})@(z,pers)$$

As proposed by Governatori and Rotolo [2010], the distinction between abrogation and annulment requires the distinguish between *void* rules and *empty* rules. The content of a void rule, e.g., $(r\colon \bot)^{t}@t'$ is $\bot$, while for the empty rule the value is the empty set. This means that the void rule has value for the combination of the temporal parameters, while for the empty rule, the content of the rule does not exist for the given temporal parameters.

Given a rule $(r\colon A \Rightarrow b^{t_b})^{t_r}@t$, the abrogation of r at t_a in repository t' is basically obtained by having in the theory the following meta-rule

$$abr_r\colon\ \Rightarrow (r\colon \bot)^{(t_a,pers)})@(t',pers) \tag{21}$$

where $t' > t$. The abrogation simply terminates the applicability of the rule. More precisely this operation sets the rule to the void rule. The rule is not removed from the system, but it has now a form where no longer can produce effects. In the case of the Legislative Act n. 124 (23 July 2008) we would have

$$abr_{L.\ 124}\colon\ \Rightarrow (L.\ 124\colon \bot)^{(2011,pers)})@(2011,pers)$$

Hence, we can derive, for example

- $+\partial_{\mathsf{O}} x@x\ \mathit{Suspended}^{x}$, $2008 \le x \le 2010$;
- $-\partial_{\mathsf{O}} x@x\ \mathit{Terms}^{x}$, $2008 \le x \le 2010$;
- $-\partial_{\mathsf{O}} 2011@2011\ \mathit{Suspended}^{2011}$;
- $+\partial_{\mathsf{O}} 2011@2011\ \mathit{Terms}^{2011}$.

This is in contrast to what we do for annulment where the rule to be annulled is set to the empty rule. This essentially amounts to removing the rule from the repository.

From the time of the annulment the rule has no longer any value. All past effects are thus blocked as well.

The definition of a modification function for annulment depends on the underlying variants of the logic, in particular whether conclusions persist across repositories. Minimally, the operation requires the introduction of a meta-rule setting the rule r to be annulled to $\emptyset$, with the time when the rule is annulled and the time when the meta-rule is inserted in the legal system:

$$(annul_r\colon \Rightarrow (r\colon \emptyset)^{(t_a,pers)})@(t',pers) \tag{22}$$

Hence,

$$(annul_{L.\ 124}\colon \Rightarrow (L.\ 124\colon \emptyset)^{(2008,pers)})@(2011,pers)$$

If we assume that conclusions persist over repositories we need some additional technical machinery to block pasts effects from previous repositories. In this case, since $L.$ 124 is modeled as a transient rule, we have basically to add a defeater like the following[15]:

$$((annul_{ef}\colon \rightsquigarrow_{\mathsf{O}} \neg Suspended^{2008})^{(2008,pers)})@(2011,pers)$$

Hence, we now have, for example

- $-\partial_{\mathsf{O}} x@2011\ Suspended^{x}$, $2008 \leq x$;
- $+\partial_{\mathsf{O}} x@2011\ Terms^{x}$, $2008 \leq x$.

x.4 State of the Art

Alchourrón and Makinson were the first to logically study the changes of a legal code [Alchourrón and Makinson, 1981, 1982, Alchourrón and Bulygin, 1981]. The addition of a new norm n causes an enlargement of the code, consisting of the new norm plus all the regulations that can be derived from n. Alchourrón and Makinson distinguish two other types of change. When the new norm is incoherent with the existing ones, we have an *amendment* of the code: in order to coherently add the new regulation, we need to reject those norms that conflict with n. Finally, *derogation* is the elimination of a norm n together with whatever part of the legal code that implies n.

Alchourrón, Gärdenfors and Makinson [1985] inspired by the works above proposed the so called AGM framework for belief revision. This area proved to a very fertile one and the phenomenon of revision of logical theories has been thoroughly investigated. It is then natural to ask if belief revision offers a satisfactory framework for the problem of norm revision. Some of the AGM axioms seem to be rational requirements in a legal context, whereas they have been criticized when imposed on

[15] The general procedure to block conclusions when conclusions persist over repositories can be very complex: for all details, see [Governatori and Rotolo, 2010].

belief change operators. An example is the *success* postulate, requiring that a new input must always be accepted in the belief set. It is reasonable to impose such a requirement when we wish to enforce a new norm or obligation. However, it gives rise to irrational behaviors when imposed to a belief set, as observed in [Gabbay et al., 2003].

The AGM operation of contraction is perhaps the most controversial one, due to some postulates such as recovery [Governatori and Rotolo, 2010, Wheeler and Alberti, 2011], and to elusive nature of legal changes such as derogations and repeals, which are all meant to contract legal effects but in remarkably different ways [Governatori and Rotolo, 2010]. Standard AGM framework is of little help here: it has the advantage of being very abstract—it works with theories consisting of simple logical assertions—but precisely for this reason it is more suitable to capture the dynamics of obligations and permissions than the one of legal norms. In fact, it is hard in AGM to represent how the same set of legal effects can be contracted in many different ways, depending on how norms are changed. For this reason, previous works [Governatori et al., 2005a, 2007b, Governatori and Rotolo, 2010] proposed to combine a rule-based system like Defeasible Logic with some forms of temporal reasoning.

Difficulties behind AGM have been considered and some research has been carried out to reframe AGM ideas within reasonably richer rule-based logical systems able to capture the distinction between norms and legal effects [Stolpe, 2010, Rotolo, 2010]. However, these attempts suffer from some drawbacks: they fail to handle reasoning on deontic effects and are based on a very simple representation of legal systems.

Another limit of standard AGM framework is that it is very abstract and so it is hard to model the distinction between norm change and the change of normative effects (such as obligation change). This difficulty has been addressed in logical frameworks combining AGM ideas with richer rule-based logical systems, such as standard or Defeasible Logic [Rotolo, 2010, Governatori et al., 2013b] or Input/Output Logic [Boella et al., 2009, Stolpe, 2010]. Wheeler and Alberti [2011] suggested a different route, i.e., employing in the law existing techniques—such as iterated belief change, two-dimensional belief change, belief bases, and weakened contraction—that can obviate problems identified in [Governatori and Rotolo, 2010] for standard AGM.

x.5 Summary

This work reports on research on extensions of Defeasible Logic to faithfully model aspects of legal dynamics. In particular, different temporal variants of the logic capture different temporal and deontic aspects of the norm-modification process. These variants increase the expressive power of the logic and it allows us to also represent meta-rules describing norm-modifications by referring to a variety of possible time-lines through which conclusions, rules and derivations can persist over time. We identified several temporal constraints that permit to allow for, or block, persistence with respect to specific time-lines. We described some issues related to norm modifications and versioning and we illustrated the techniques with some relevant modifications such as annulment, abrogation, substitution and derogation. In particular, we illustrated the problem of how legal effects of ex-tunc modifications, such as annulment, can be blocked after the modification applied. The idea we suggested is to block persistence of derivations across repositories. In other words, the conclusions of the annulled rule will only be derived in the repository in which the modification does not occur. The proposed methodology illustrates the possibilities of the formalism and we intend to apply it to the logical analysis of a larger corpus of norm-modifications.

Acknowledgements NICTA is funded by the Australian Government through the Department of Communications and the Australian Research Council through the ICT Centre of Excellence Program.

References

Carlos E. Alchourrón and Eugenio Bulygin. The expressive conception of norms. In Risto Hilpinen, editor, *New Studies in Deontic Logic*, pages 95–125. D. Reidel, Dordrecht, 1981.

Carlos E. Alchourrón and Eugenio Bulygin. Permission and permissive norms. In W. Krawietz et al., editor, *Theorie der Normen*. Duncker & Humblot, 1984.

Carlos E. Alchourrón and David C. Makinson. Hierarchies of regulations and their logic. In Risto Hilpinen, editor, *New Studies in Deontic Logic*, pages 125–148. D. Reidel, Dordrecht, 1981.

Carlos E. Alchourrón and David C. Makinson. The logic of theory change: Contraction functions and their associated revision functions. *Theoria*, 48:14–37, 1982.

Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50:510–530, 1985.

David Billington, Grigoris Antoniou, Guido Governatori, and Michael J. Maher. An inclusion theorem for defeasible logic. *ACM Transactions in Computational Logic*, 12(1):article 6, 2010.

Guido Boella, Gabriella Pigozzi, and Leendert van der Torre. A normative framework for norm change. In *8th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, pages 169–176. IFAAMAS, 2009.

Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321–358, 1995.

Dov M. Gabbay, Gabriella Pigozzi, and John Woods. Controlled revision - an algorithmic approach for belief revision. *Journal of Logic and Computation*, 13(1):3–22, 2003.

Thomas F. Gordon, Henry Prakken, and Douglas Walton. The Carneades model of argument and burden of proof. *Artificial Intelligence*, 171(10-11):875–896, 2007.

Guido Governatori. On the relationship between Carneades and Defeasible Logic. In Kevin Ashley, editor, *The 13th International Conference on Artificial Intelligence and Law (ICAIL 2011)*, pages 31–40. ACM, 2011.

Guido Governatori and Antonino Rotolo. Changing legal systems: Legal abrogations and annulments in defeasible logic. *The Logic Journal of IGPL*, 18(1):157–194, 2010.

Guido Governatori and Antonino Rotolo. Computing temporal defeasible logic. In Leora Morgenstern, Petros S. Stefaneas, François Lévy, Adam Wyner, and Adrian Paschke, editors, *7th International Symposium on Theory, Practice, and Applications of Rules on the Web (RuleML 2013)*, volume 8035 of *LNCS*, pages 114–128, 2013.

Guido Governatori and Giovanni Sartor. Burdens of proof in monological argumentation. In Radboud Winkels, editor, *The Twenty-Third Annual Conference on Legal Knowledge and Information Systems (Jurix 2010)*, pages 57–66, Amsterdam, 2010. IOS Press.

Guido Governatori, Michael J. Maher, David Billington, and Grigoris Antoniou. Argumentation semantics for defeasible logics. *Journal of Logic and Computation*, 14(5):675–702, 2004.

Guido Governatori, Monica Palmirani, Régis Riveret, Antonino Rotolo, and Giovanni Sartor. Norm modifications in defeasible logic. In Marie-Francine Moens and Peter Spyns, editors, *The Eighteenth Annual Conference on Legal Knowledge and Information Systems (Jurix 2005)*, pages 13–22, Amsterdam, 2005a. IOS Press.

Guido Governatori, Antonino Rotolo, and Giovanni Sartor. Temporalised normative positions in defeasible logic. In Anne Gardner, editor, *10th International Conference on Artificial Intelligence and Law (ICAIL 2005)*, pages 25–34. ACM Press, 2005b.

Guido Governatori, Joris Hulstijn, Régis Riveret, and Antonino Rotolo. Characterising deadlines in temporal modal defeasible logic. In Mehmet A. Orgun and John Thornton, editors, *20th Australian Joint Conference on Artificial Intelligence*, volume 4830 of *Lecture Notes in Artificial Intelligence*, pages 486–496, Heidelberg, 2007a. Springer.

Guido Governatori, Antonino Rotolo, Régis Riveret, Monica Palmirani, and Giovanni Sartor. Variants of temporal defeasible logic for modelling norm modifications. In Radboud Winkels, editor, *Proceedings of 11th International Conference on Artificial Intelligence and Law (ICAIL 2007)*, pages 155–159, New York, 2007b. ACM Press.

Guido Governatori, Francesco Olivieri, Antonino Rotolo, and Simone Scannapieco. Computing strong and weak permissions in defeasible logic. *Journal of Philosophical Logic*, 42(6):799–829, 2013a.

Guido Governatori, Antonino Rotolo, Francesco Olivieri, and Simone Scannapieco. Legal contractions: a logical analysis. In *ICAIL*, pages 63–72, 2013b.

Riccardo Guastini. *Teoria e dogmatica delle fonti*. Giuffré, Milan, 1998.

H.L.A. Hart. *The Concept of Law*. Clarendon, Oxford, 1994.

Hans Kelsen. *General Theory of Norms*. Clarendon, Oxford, 1991.

Donald Nute. Defeasible logic. In Dov M Gabbay, Christopher John Hogger, and John Alan Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3, pages 353–395. Oxford University Press, Oxford, 1994.

Antonino Rotolo. Retroactive legal changes and revision theory in defeasible logic. In Guido Governatori and Giovanni Sartor, editors, *Proceedings of the 10th International Conference on Deontic Logic in Computer Science (DEON 2010)*, volume 6181 of *LNAI*, pages 116–131. Springer, 2010.

Audun Stolpe. Norm-system revision: theory and application. *Artificial Intelligence and Law*, 18 (3):247–283, 2010.

Georg Henrik von Wright. *Norm and action: A logical inquiry*. Routledge and Kegan Paul, 1963.

Gregory R. Wheeler and Marco Alberti. No revision and no contraction. *Minds and Machines*, 21 (3):411–430, 2011.