ECAI 2016
G.A. Kaminka et al. (Eds.)

© 2016 The Authors and IOS Press.
This article is published online with Open Access by IOS Press and distributed under the terms
of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).
doi:10.3233/978-1-61499-672-9-3

When Do Rule Changes Count-As Legal Rule Changes?

Thomas C. King and **Virginia Dignum** and **Catholijn M. Jonker** [1]

Abstract. Institutions regulate societies. Comprising Searle's constitutive counts-as rules, "A counts-as B in context C", an institution ascribes from brute and institutional facts (As), a social reality comprising institutional facts (Bs) conditional on the social reality (contexts Cs). When brute facts change an institution evolves from one social reality to the next. Rule changes are also regulated by *rule-modifying* counts-as rules ascribing rule change in the past/present/future (e.g. a majority rule change vote *counts-as* a rule change). Determining rule change legality is difficult, since changing counts-as rules both alters and is conditional on the social reality, and in some cases hypothetical rule-change effects (e.g. not retroactively criminalising people). However, without a rigorous account of rule change ascriptions, AI agents cannot support humans in understanding the laws imposed on them. Moreover, advances in automated governance design for socio-technical systems, are limited by agents' ability to understand how and when to enact institutional changes. Consequently, we answer "when do rule changes count-as legal rule changes?" in a temporal setting with a novel formal framework.

1 Introduction

Institutions regulate and govern society and have been widely formalised (see Andrighetto *et al.* [2013]). Institutions construct a descriptive and prescriptive social reality from brute facts with Searle's (Searle [1969, 2005]) *counts-as* rules, "A counts-as B in context C". When the brute facts change an institution's social reality evolves according to counts-as rules. Counts-as rules can also be modified over time. In legal systems, secondary counts-as rules ascribe rule change (Biagioli [1997]). We view these rules as rule-modifying counts-as rules - "A counts-as *modifying a rule* in context C".

Yet, it is difficult to determine which rule changes can be made according to rule-modifying counts-as rules. Rules build the social reality, ascribe rule changes conditional on the social reality, and are also subject to being changed. This affects which rule changes are possible in the first place, for example:

- A group of people voting to change a rule counts-as a legal rule change if they constitute the government. A rule change can affect the social reality by redefining it (e.g. who counts-as being in government); rule changes are conditional on the built social reality.
- The UK government voted to *retroactively* require UK residents in a business partnership abroad to pay tax ([Fin, 2008, Sec. 58]), criminalising people in the past. Criminalising retroactive modifications are impossible according to the European Convention of Human Rights ([Council of Europe, 1953, Art. 7]). Rule change affects the social reality (e.g. criminalising people in the past); rule change is conditional on its hypothetical effects (e.g. being impossible if it would criminalise people in the past).
- A monarch or parliament can change laws. The monarch enacts a law obliging all fences are painted white. The parliament retroactively repeals the power for the monarch to enact laws, reversing the fence-painting law enactment. Retroactive rule change affects past rule-modifying counts-as rules; past rule modifications can be unravelled due to retroactive modifications.

An interdependency exists between the counts-as rules that construct a social reality and rule-modifying counts-as rules. Changing counts-as rules affects the past/present/future social reality and can change the modifications which happened in the past up until the present; rule modifications are conditional on the past/present/future social reality and the hypothetical rule change effects. Whether a rule change counts-as a *legal* rule change requires assessing the social reality in which the change takes place and the potential rule change effects, thus affecting whether a rule change is legal in the first place.

A defeasible logic for rule change over time has been proposed (Governatori *et al.* [2005]; Governatori and Rotolo [2010]). But, crucially, not for rule change ascribed by counts-as rules accounting for the interdependency between changing rules and building a social reality. In (Boella and van der Torre [2004]) counts-as rules that regulate rule modifications are formalised, but not in a temporal setting. Yet, there has been little attention paid to formalising rule change regulated by counts-as rules in a temporal setting. This limits endeavours in AI to assist human agents in understanding the laws that govern them. Moreover, whilst AI agents are increasingly used to synthesise normative systems (Morales *et al.* [2014, 2015]), they are held back in enacting institutional rule changes by not understanding how and when laws can be changed.

This raises the question, in a temporal setting - *when do rule changes count-as legal rule changes?* We address this question with a novel formalisation for past/present/future institution rule change ascribed by counts-as rules. Our desiderata being that if a rule change is legal then it occurs, and otherwise it does not and the institution continues to operate 'as usual'. In particular, taking into account the interdependency between ascribing rule change and changing rules. We posit that the most recent rule modifications take precedent and potentially change past modifications. We extend rules commonly found in the literature from being conditional on the present, to the past, different institution versions and hypothetical rule change effects.

We continue with our approach (2). Then we introduce the formalism, comprising a representation (3) and semantics (4). The framework is applied to five case studies (5). Finally, we discuss related work (6) and conclusions (7).

2 Approach

This paper formalises institutional rule change in a temporal setting. Foundational reasoning is required for institutions in a temporal setting, on which our framework is built. We require reasoning about

[1] Delft University of Technology, Netherlands, email addresses: {t.c.king-1, m.v.dignum, c.m.jonker}@tudelft.nl

counts-as rules, “A *counts-as* B in context C”, of two types. Firstly, rules that ascribe institutional events from other institutional events or observable events (brute facts) conditional on the social reality. For example, “the event we call a person paying tax (brute) counts-as paying tax (institutional)”, and “paying tax (institutional) counts-as fulfilling your duties (institutional)”. Secondly, rules which ascribe institutional fluents (institutional facts describing the state of affairs) from institutional events and cause the social reality (institutional state) to change. For example, “signing a business declaration counts-as initiating you are a business partner”. We require reasoning for counts-as rules that cause events to occur and the institutional state to change, which the InstAL (Institution Action Language) framework (Cliffe *et al.* [2007]; Cliffe [2007]) provides. Crucially, InstAL lacks representation and reasoning for rule change ascription.

We extend InstAL’s institutions with counts-as rules that ascriptively regulate rule changes and counts-as rules that themselves are modifiable. Rule modifications activate/deactivate rules in the past/present/future, analogous to enacting regulatory changes. Rule-modifying counts-as rules, “an event A counts-as a *rule-modifying* event B in context C”, ascribe past/present/future rule modifications.

Unlike in InstAL, in this paper institutions evolve dually: 1. when a rule change is ascribed by counts-as rules, the institution evolves to the next version potentially comprising different (active) counts-as rules at different time points, and 2. when observable events occur, each institution version evolves from one state to the next.

For example, an institution starts at version one, only comprising rules which enable rules to be added. A rule is added to the institution on Monday, stating that the tax year’s start causes an obligation to pay tax. Thus, on the Monday the institution evolves to version two, where the tax rule becomes active on the Monday, at which point version two becomes the *current version*. On Tuesday it is the first tax year month, both versions evolve to a new state, in version two the new state contains an obligation to pay tax, but not in version one since the tax rule was activated in version two. Each institution version evolves from state to state, and the institution evolves from one current version to the next when rule change events occur.

Contexts in counts-as rules are extended from being conditional on the present state to also past institution versions and states. This supports representing rule change conditional on its potential retroactive effects. For example, a condition on rule change not criminalising people in a version’s past *compared to the previous version’s past*. To summarise, we extend institutions to evolve along rule version and state timelines according to counts-as rules conditional on past versions and states, and potential rule change effects.

3 Representation

We begin with representing institutions which regulate their own temporal rule modifications.

Definition 1. ***Institution*** *An institution is a tuple $\mathcal{I} = \langle \mathcal{E}, \mathcal{F}, \mathcal{C}, \mathcal{G}, \Delta \rangle$. Institutions are distinguished with a superscript (e.g. $\mathcal{I}^{uk} = \langle \mathcal{E}^{uk}, \mathcal{F}^{uk}, \mathcal{C}^{uk}, \mathcal{G}^{uk}, \Delta^{uk} \rangle$). $\Sigma = 2^{\mathcal{F}}$ denotes all states for $\mathcal{I}$.*

Where:

1. $\mathcal{E} = \mathcal{E}_{obs} \cup \mathcal{E}_{inst} \cup \mathcal{E}_{mod}$ is a finite set of *events* comprising:
   - Observable events $\mathcal{E}_{obs}$ and institutional events $\mathcal{E}_{inst}$.
   - Rule modification events $\mathcal{E}_{mod} = \{mod(op, id, t) \mid op \in \{act, deact\}, id \in \mathbb{ID}, t \in \mathbb{N}\}$ - a rule with the identifier *id* (the identifier set being $\mathbb{ID}$) is activated/deactivated (*op*) at a time *t*.
2. $\mathcal{F} = \mathcal{F}_{dom} \cup \mathcal{F}_{ract}$ is a finite set of *fluents* describing the:
   - Domain $\mathcal{F}_{dom}$.
   - Active rules $\mathcal{F}_{ract} = \{active(id) \mid id \in \mathbb{ID}\}$ identified as *id*.
3. $\mathcal{X}$ is the set of all *contexts* φ expressible in the following grammar for fluents $f \in \mathcal{F}$:

$$\begin{aligned} \varphi ::= &\ \top \mid f \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \rightarrow \varphi \mid P \mid \\ &\ PrS(\varphi) \mid PaS(\varphi) \mid PrV(\phi) \mid PaV(\phi) \\ \phi ::= &\ \varphi \mid NS(\varphi) \end{aligned}$$

   Each expression’s informal meaning is the usual for propositional logic symbols. The operators bear truth in the following cases: (a) P if the context is retroactive (i.e. the state in which P operates on is at a time before the version to which it belongs becomes the current version), and (b) if φ is true in: the previous state ($PrS(\varphi)$), all past states ($PaS(\varphi)$), the same state in the previous version ($PrV(\varphi)$), the same state in all past versions ($PaV(\varphi)$), and the next state ($NS(\varphi)$).The next state operator is restricted to past versions, meaning rules are never conditional on the actual future.
4. $\mathcal{G} : \mathcal{X} \times 2^{\mathcal{E}} \rightarrow 2^{\mathcal{E}_{inst}}$ - is the *event generation function* where $\mathcal{G}(X, E)$ is an event set caused by the events that occur (E) when the context X holds.
5. $\mathcal{C} : \mathcal{X} \times \mathcal{E} \rightarrow 2^{\mathcal{F}_{dom}} \times 2^{\mathcal{F}_{dom}}$ is the *state consequence function* where for a context $X \in \mathcal{X}$ and an event $e \in \mathcal{E}$ the consequence function’s result is notated $\mathcal{C}(X, e) = \langle \mathcal{C}^{\uparrow}(X, e), \mathcal{C}^{\downarrow}(X, e) \rangle$ s.t. the initiated fluent set is $\mathcal{C}^{\uparrow}(X, e)$ and the terminated fluent set is $\mathcal{C}^{\downarrow}(X, e)$
6. $\Delta \subseteq \mathcal{F}$ is the *initial institution state*

For example, the following rule states that if Ada is found guilty (*g(ada)*) then she becomes a criminal (*crim(ada)*). That is, the fluent *crim(ada)* is initiated by the event of being found guilty according to the consequence function ($\mathcal{C}^{\uparrow}$).

$$\mathcal{C}^{\uparrow}(\top, g(ada)) \ni crim(ada)$$

A government rule change (*gmod(act, id, t)*)) that does not retroactively criminalise people counts-as a legal rule change. The condition is in all past retroactive states someone is not a criminal (*crim(ada)*) if in the previous version (prior to rule change) they were not.

$$\begin{aligned} &\mathcal{G}(PaS(P \rightarrow PrV(\neg crim(ada)) \rightarrow \neg crim(ada))), \\ &\quad \{gmod(act,\ id,\ t)\}) \ni act(id,\ t) \end{aligned}$$

In order to reason about modifying specific institutional rules, we tie rule identifiers to the institutional rules they represent. Specifically we map the inputs and single outputs of $\mathcal{G}$ and $\mathcal{C}$ to identifiers (i.e. not the whole set of events or initiated/terminated fluents).

Definition 2. ***Rule Identifier Function*** *A rule identifier function for an event generation function $\mathcal{G} : \mathcal{X} \times 2^{\mathcal{E}} \rightarrow 2^{\mathcal{E}_{inst}}$ is $rid^{\mathcal{G}} : \mathcal{X} \times 2^{\mathcal{E}} \times \mathcal{E}_{inst} \rightarrow \mathbb{ID}$. The rule identifier functions for a consequence function $\mathcal{C} : \mathcal{X} \times \mathcal{E} \rightarrow 2^{\mathcal{F}_{dom}} \times 2^{\mathcal{F}_{dom}}$ are $rid^{\mathcal{C}^{\uparrow}} : \mathcal{X} \times \mathcal{E} \times \mathcal{F}_{dom} \rightarrow \mathbb{ID}$ and $rid^{\mathcal{C}^{\downarrow}} : \mathcal{X} \times \mathcal{E} \times \mathcal{F}_{dom} \rightarrow \mathbb{ID}$.*

So, the previous rule criminalising Ada has the ID *crim0* = $rid^{\uparrow}(\top, g(ada), crim(ada))$. Examples/case studies omit this function.

4 Semantics

This section defines institution semantics, following InstAL’s method using just sets and functions, with the following considerations.

Observable events cause an institution rule version to transition from state to state by generating transitioning events according to the event generation function $\mathcal{G}$ and initiating and terminating fluents according to the consequence function $\mathcal{C}$. An institution transitions from one version of rules to another when rule modifying events are generated by the event generation function $\mathcal{G}$.

An institutional interpretation represents this dual evolution as a tuple $M = \langle R, V \rangle$ where: 1. $V = \langle V_0, ..., V_j \rangle$ is a tuple of versions each comprising a state and event set sequence up to length k with typical element $V_v = \langle S_v, E_v \rangle$. The state sequence for v is $S_v = \langle S_{v:0}, ..., S_{v:k+1} \rangle$ with typical element $S_{v:i} \in \Sigma$ and the event set sequence (the events transitioning between states) is $E_v = \langle E_{v:0}, ..., E_{v:k} \rangle$ with typical element $E_{v:i} \subseteq \mathcal{E}$. States denoted $S_{v:i}$ and event sets $E_{v:t}$ are denoted with the version v to which they belong and their time instant i. 2. $R : [0,k] \rightarrow [0,j]$ is a function stating which institution version is the *current* version for a given time.

R also represents when rule change events occurring in a version can change that version's rules. Rule modification events only change version rules if the institution has not already evolved to a later version. For example, if on Monday a rule is added, then the institution evolves to a new *current* version where that rule is actually added on Monday. When the version evolves, previous versions become *obsolete* from then onwards (e.g. Monday) meaning their rules are not changeable. If $R(i) \leq v$ then an event occurring in version v at time i can modify rules in v since the version is not yet obsolete.

The semantics are defined with respect to the interpretation $M = \langle R, V \rangle$, an institution $\mathcal{I} = \langle \mathcal{E}, \mathcal{F}, \mathcal{C}, \mathcal{G}, \Delta \rangle$, the set of all institutional interpretations $\mathbb{I}$, and an observable event trace $et = \langle O_0, ..., O_k \rangle$.

4.1 Institutional Change

Counts-as rules, causing institution state and version change, are conditional on a context being *modelled* by the state in an *interpretation*.

Definition 3. ***Modelling Context*** *For all $X \in \mathcal{X}$ and $f \in \mathcal{F}$, context models $\langle M, S_{v:t} \rangle \models X$ is defined for $\top$, $\vee$ and $\rightarrow$ w.r.t. $\neg$ and $\wedge$ as usual and for the other symbols as:*

$$\langle M, S_{v:t} \rangle \models f \quad \Leftrightarrow \quad f \in S_{v:t} \tag{3.1}$$

$$\langle M, S_{v:t} \rangle \models \neg\psi \quad \Leftrightarrow \quad \langle M, S_{v:t} \rangle \not\models \psi \tag{3.2}$$

$$\begin{aligned}\langle M, S_{v:t} \rangle \models \psi \wedge \phi \quad \Leftrightarrow \quad & \langle M, S_{v:t} \rangle \models \psi \textbf{ and} \\ & \langle M, S_{v:t} \rangle \models \phi\end{aligned} \tag{3.3}$$

$$\langle M, S_{v:t} \rangle \models P \quad \Leftrightarrow \quad R(t) < v \tag{3.4}$$

$$\langle M, S_{v:t} \rangle \models PrS(\psi) \quad \Leftrightarrow \quad \langle M, S_{v:t-1} \rangle \models \psi \tag{3.6}$$

$$\langle M, S_{v:t} \rangle \models PaS(\psi) \quad \Leftrightarrow \quad \forall t' \in [0, t-1] : \langle M, S_{v:t-1} \rangle \models \psi \tag{3.7}$$

$$\langle M, S_{v:t} \rangle \models PrV(\psi) \quad \Leftrightarrow \quad \langle M, S_{v-1:t} \rangle \models \psi \tag{3.8}$$

$$\langle M, S_{v:t} \rangle \models PaV(\psi) \quad \Leftrightarrow \quad \forall v' \in [0, v-1] : \langle M, S_{v'-1:t} \rangle \models \psi \tag{3.9}$$

$$\langle M, S_{v:t} \rangle \models NS(\psi) \quad \Leftrightarrow \quad \langle M, S_{v:t+1} \rangle \models \psi \tag{3.10}$$

Semantics are as usual for modelling a fluent (3.1), weak negation (3.2) and conjunction (3.3). A state is retroactive if at that time the version is not the current version but it will be in the future (3.4) - for example, if on a Wednesday the institution evolves to a new version, then anything occurring on the Monday is retroactive to the new version (i.e. occurring in the version's past). States model formula as expected for a previous state (3.6), all previous states (3.7), the previous version (3.8), all past versions (3.9) and the next state (3.10).

An event 'B' occurs when transitioning to a new state in a version according to a rule - "A counts-as B in context C" ($\mathcal{G}$) - if an event 'A' occurs, the context 'C' is modelled by the state and the counts-as rule itself is active in the version's state. Events occurring in response to observable events E are formalised as an event generation operation.

Definition 4. ***Event Generation Operation*** *The event generation operation $GR : \Sigma \times 2^{\mathcal{E}} \times \mathbb{I} \rightarrow 2^{\mathcal{E}}$ is defined such that $GR(S_{v:t}, E, M) = E'$ iff E' only satisfies the following conditions:*

$$E \subseteq E' \tag{4.1}$$

$$\begin{aligned}&\exists X \in \mathcal{X}, e \subseteq E, e' \in \mathcal{G}(X,e) \cap \mathcal{E}_{inst} : id = rid^{\mathcal{G}}(X,e,e'), \\ &\langle M, S_{v:t} \rangle \models X \wedge active(id) \Rightarrow e' \in E'\end{aligned} \tag{4.2}$$

$$\begin{aligned}&\exists X \in \mathcal{X}, e \subseteq E, e' \in \mathcal{G}(X,e) \cap \mathcal{E}_{mod} : id = rid^{\mathcal{G}}(X,e,e'), \\ &\langle M, S_{v:t} \rangle \models X \wedge active(id), R(t) \neq v \Rightarrow e' \in E'\end{aligned} \tag{4.3}$$

$$\begin{aligned}&\exists X \in \mathcal{X}, e \subseteq E, e' \in \mathcal{G}(X,e) \cap \mathcal{E}_{mod} : id = rid^{\mathcal{G}}(X,e,e'), \\ &\langle M, S_{v:t} \rangle \models X \wedge active(id), R(t) = v \Rightarrow (e' \in E' \textbf{ or } e' \notin E')\end{aligned} \tag{4.4}$$

Any fixed point reached after iterative applications of GR is denoted as $GR^{\omega}(S_{v:t}, E, M)$.

Events that have occurred still occur (4.1). If an active rule states an event e causes an event e' in a context modelled by the state, then e *can* cause e' to occur depending on e''s type. Specifically, whether e' is a type that could cause an inconsistency (e.g. removing rules that ascribe rule modifications, for more on the paradox of rule change see Suber [1990]). An event e' always occurs if it is a non-rule-modifying institutional event (4.2) or occurs when the version is obsolete and it cannot modify rules (4.3). Rule modifying events in non-obsolete versions *can* cause rule changes and a potential paradox. So they *optionally* occur in a non-obsolete version where they can cause rule change and/or a paradox (4.4). Hence, GR is *multi-valued.*

Iterating the event generation operation until a *fixed point* is reached obtains all events which occur. At least one fixed point is guaranteed.

Lemma 1. *For any set of events $E \subseteq \mathcal{E}$, interpretation M and state $S_{v:t} \in \Sigma$ there exists a fixed point $GR^{\omega}(S_{v:t}, E, M)$.*

Proof sketch. GR always has a monotonically increasing value (w.r.t. set inclusion) and a finite domain. □

An institution version transitions between states, driven by event occurrences, according to a state transition operation.

Definition 5. ***State Transition Operation*** *The state transition operation $TR : \Sigma \times 2^{\mathcal{E}} \times \mathbb{I} \rightarrow 2^{\mathcal{E}}$ is defined for a state $S_{v:t}$, a set of events $E_{v:t}$ and an interpretation M as:*
$TR(S_{v:t}, E_{v:t}, M) =$

$$\{f \mid f \in S_{v:t} \cap TERM(S_{v:t}, E_{v:t}, M) \textbf{ or} \tag{5.1}$$

$$f \in INIT(S_{v:t}, E_{v:t}, M)\} \tag{5.2}$$

where:
$INIT(S_{v:t}, E_{v:t}, M) =$

$$\begin{aligned}&\{f \mid \exists e \in E_{v:t}, X \in \mathcal{X} : id = rid^{\mathcal{C}^{\uparrow}}(X,e,f), \\ &\quad f \in \mathcal{C}^{\uparrow}(X,e) \cap \mathcal{F}_{dom}, \langle M, S_{v:t} \rangle \models X \wedge active(id) \quad \textbf{or}\end{aligned} \tag{5.3}$$

$$\begin{aligned}&\quad \exists t' \in [0,k], \nexists t'' \in [t',k] : id = rid^{\mathcal{C}^{\uparrow}}(X,e,f), \\ &\quad R(t') \leq v, R(t'') \leq v, mod(act,\ id,\ t) \in E_{v:t'}, \\ &\quad mod(deact,\ id,\ t) \in E_{v:t''}, f = active(id)\}\end{aligned} \tag{5.4}$$

$TERM(S_{v:t}, E_{v:t}, M) =$

$$\begin{aligned}&\{f \mid \exists e \in E_{v:t}, X \in \mathcal{X} : id = rid^{\mathcal{C}^{\downarrow}}(X,e,f), \\ &\quad f \in \mathcal{C}^{\downarrow}(X,e) \cap \mathcal{F}_{dom}, \langle M, S_{v:t} \rangle \models X \wedge active(id) \textbf{ or}\end{aligned} \tag{5.5}$$

$$\begin{aligned}&\quad \exists t' \in [0,k], \nexists t'' \in [t',k] : id = rid^{\mathcal{C}^{\downarrow}}(X,e,f) \\ &\quad R(t') \leq v, R(t'') \leq v, mod(deact,\ id,\ t) \in E_{v:t'}, \\ &\quad mod(act,\ id,\ t) \in E_{v:t''}, f = active(id)\}\end{aligned} \tag{5.6}$$

Transitioning from one state to the next follows common-sense inertia - a fluent holds in a new state if it held in the previous state and was not terminated (5.1) or it was initiated in the previous state (5.2). A domain fluent is initiated/terminated if an event causes it to be according to a rule defined by the state consequence function $\mathcal{C}$ that is active in the current state with a condition (context) that is modelled in the state (5.3 for initiation and 5.5 for termination). A fluent denoting an active rule is initiated/terminated in a state if a rule activating/deactivating event occurs at a time when the version is not obsolete and no contradictory deactivation/activation event occurs at a later time when the version is not obsolete (5.4 for activating rules and 5.6 for deactivating rules). The most recent modifications in a version take precedent if they occur when the version is a non-obsolete version and simultaneous contradictory rule modifications are cancelled.

4.2 Models

Now we define when an interpretation is an institutional model for an observable event set trace. An institutional interpretation is, broadly speaking, an institutional model for an observable event set trace iff: 1. each version evolves according to the event generation and state transition operations, and 2. the institution evolves from one version to another when rules are modified. However, the event generation operation is multi-valued since rule modifications are *optional*. Thus, there are potentially multiple candidate event sets for transitioning between states and therefore multiple interpretations to select as models.

We want to maximise the rule modification events that are not self-contradicting (e.g. not applying modifications that retroactively remove a rule making retroactive rule removal possible). Interpretations are prioritised, denoted as $<$, based on maximising rule modifications. An interpretation has higher priority over another if at the earliest time in the earliest version in which the interpretation differ it contains a superset of rule modifying events compared to the 'same' set for the lower priority interpretation.

Definition 6. ***Prioritised Interpretation*** *Let* $M^0 = \langle R^0, V^0 \rangle \in \mathbb{I}$ *and* $M^1 = \langle R^1, V^1 \rangle \in \mathbb{I}$ *be two interpretations for institution* $\mathcal{I}$ *where:* $V^0 = \langle V_0^0, ..., V_i^0 \rangle$ *with typical element* $V_v^0 = \langle E_v^0, S_v^0 \rangle$ *s.t.* $E_v^0 = \langle E_{v:0}^0, ..., E_{v:k}^0 \rangle$*, and* $V^1 = \langle V_0^1, ..., V_j^1 \rangle$ *with typical element* $V_v^1 = \langle E_v^1, S_v^1 \rangle$ *s.t.* $E_v^1 = \langle E_{v:0}^1, ..., E_{v:k}^1 \rangle$*. The ordering* $<$ *is a relation between interpretations* M^0 *and* M^1 *such that:*

$$\begin{aligned} M^0 < M^1 \quad \Leftrightarrow \qquad & \exists t \in [0,k], \nexists t' \in [0, t\text{-}1] : \\ & v = R^0(t), E_{v:t}^0 \cap \mathcal{E}_{mod} \supset E_{v:t}^1 \cap \mathcal{E}_{mod} \\ & v' = R^0(t'), E_{v':t'}^0 \neq E_{v':t'}^1 \end{aligned}$$

We operationally characterise a model by constructing a 'correct' interpretation. That is, constructing versions comprising correct state transitions and generated events. We could construct each institution version by starting at an initial state and proceeding from one state to the next according to the event generation and state transition operations. However, this would require knowing which rule modification events happen in each version's past, present and future.

To give an example for an observable event set trace $\langle O_0, ..., O_k \rangle$. An institution starts at an initial state only comprising an active rule enabling a government to make retroactive modifications ($\Delta = S_{0:0} = \{active(gov0)\}$). First, a fence is observably built ($O_0 = \{fb\}$, occurring during the first state transition $fb \in E_{0:0} = GR^{\omega}(S_{0:0}, O_0, M)$). But, there is no active rule that causes the next state to be different ($S_{0:1} = TR(S_{0:0}, E_{0:0}) = S_{0:0} = \{active(gov0)\}$). Then, the government votes to retroactively activate a rule in state zero, stating building a fence initiates an obligation to paint it. Consequently, the second state which has already been determined, $S_{0:1}$, seems wrong since it lacks the fence painting rule and its effects. In fact, the institution should transition to a new rule version V_1. This new version should start at the same initial state $S_{1:0} = \Delta$. But, crucially, transition to the next state ($S_{1:0} = TR(S_{1:0}, E_{1:0})$) with the knowledge that in the future of the new version the fence painting rule will be retroactively added at state zero ($S_{1:0}$) and become active in the second state ($S_{1:1}$). State transitions are defined with respect to an interpretation comprising past/present/future rule modification events which might be unknown when each state and transitioning event set is constructed.

We define an interpretation successor operation which addresses the problem of constructing a 'correct' interpretation without the knowledge of each version's past/present/future. The successor operation takes as input a preceding interpretation which supplies versions comprising a past/present/future on which each version in the new succeeding interpretation can be constructed according to *TR* and *GR*. That is, a new interpretation is produced using the version timelines of the previous interpretation, taking into account past/present/future rule modifications from the preceding interpretation's version timelines.

A succeeding interpretation might not be the same as the previous interpretation, since the previous interpretation might have been built without knowledge of its own past/present/future. That is, the new interpretation might differ in its temporal evolution (comparable version timelines in each interpretation being different). Consequently, the succeeding interpretation might have new, previously unknown, rule modification events that also need to be accounted for and thus another succeeding interpretation must be produced.

The idea is to iteratively apply the institution successor operation until a succeeding interpretation is produced that is the same as the previous interpretation. That is, until the operation reaches a fixed point, which is guaranteed according to lemma 3 we give later on. Intuitively, the fixed point characterises an interpretation that is built taking into account its own past/present/future modifications in each version (since it was built with respect to an identical preceding interpretation). Formally, the successor interpretation operation is:

Definition 7. ***Successor Interpretation Operation*** *Let* $et = \langle O_0, ..., O_k \rangle$ *be an observable event trace for* $\mathcal{I}$ *of length k. Let* $M' = \langle R', V' \rangle \in \mathbb{I}$ *be an interpretation such that* $V' = \langle V_0', ..., V_{j'}' \rangle$ *is a tuple of institution versions. The interpretation successor operation* $SUCC : \mathbb{I} \times ET \rightarrow \mathbb{I}$ *is defined for the interpretation* M *w.r.t.* $\mathcal{I}$ *and et such that* $SUCC(M, et) = M'$ *iff* M' *satisfies the following conditions:*

$$\forall v \in [0, j'] : S_{v:0}' = \Delta \tag{7.1}$$

$$\forall v \in [0, j'], t \in [0, k] : E_{v:t}' = GR^{\omega}(S_{v:t}', O_t, M) \tag{7.2}$$

$$\forall v \in [0, j'], t \in [0, k] : S_{v:t+1}' = TR(S_{v:t}', E_{v:t}', M) \tag{7.3}$$

$$R'(t) = \begin{cases} 0, & t = 0, E_{0:t}' \cap \mathcal{E}_{mod} = \emptyset \\ 1, & t = 0, E_{0:t}' \cap \mathcal{E}_{mod} \neq \emptyset \\ R'(t\text{-}1), & t > 0, E_{R(t\text{-}1):t}' \cap \mathcal{E}_{mod}' = \emptyset \\ R'(t\text{-}1)+1, & t > 0, E_{R(t\text{-}1):t}' \cap \mathcal{E}_{mod}' \neq \emptyset \end{cases} \tag{7.4}$$

$$\textit{Given that } V' = \langle V_0', ..., V_{j'}' \rangle, R'(k) = j' \tag{7.5}$$

Every institution version starts at the same initial state (7.1). Each state transition (an event set) in a version is produced by the event generation operation applied to the previous state and the observable events occurring at that time (7.2). The next state in a version is the state produced by the state transition operation applied to the previous state and the transitioning events occurring in that version *with respect to* the preceding institutional interpretation (7.3). That is, transitioning from one state to the next takes into account the rule modification events occurring in the past/present/future of the same version in the preceding interpretation. Rule modifications in the latest version cause the current version to evolve/increment to the next version. If no rule modification takes place the version remains the same or the zeroeth version for the zeroeth time instant (7.4). If a rule modification does take place in the latest version, then the current version at that time incremented by one, or is the first version for the zeroeth time point (7.4). The version sequence only goes up until the current version at the last time instant (7.5).

At least one fixed point for the successor interpretation operation, starting at any initial interpretation, is always guaranteed. A fixed point is denoted as $SUCC^{\omega}(M,et)$. To see why, the general idea is that there always exists a series of successive interpretations that monotonically increase which versions and states they agree on.

The following lemma is used to prove that there always exists a series of such interpretations and therefore that there always exists a fixed point. Informally, the lemma is conditional on there being two successors M' and M'' to any interpretation that agree with each other up until a particular time (h) in a version (j). The consequence is that the second interpretation M'' has the same events at time h and state transition at time $h+1$ in version j as if the event and state transitions were produced with respect to M'''s *own* past/present/future timeline.

Lemma 2. *If $\mathcal{I}$ is an institution, M an interpretation and et an observable event trace of length k for $\mathcal{I}$ and there exists interpretations $M' = SUCC(M,\ et)$ and $M'' = SUCC(M', et)$ where $\exists h \in [0,k], j \in [0,v'], \forall i \in [0,k]$:*

$$\langle V'_0, ..., V'_{j-1}\rangle = \langle V''_0, ..., V''_{j-1}\rangle \tag{A2.1}$$

$$\langle S'_{j:0}, ..., S'_{j:h}\rangle = \langle S''_{j:0}, ..., S''_{j:h}\rangle \tag{A2.2}$$

$$\begin{matrix} mod(op,\ id,\ h) \in E'_{v:i}, \\ R'(i) \leq j \end{matrix} \Leftrightarrow \begin{matrix} mod(op,\ id,\ h) \in E''_{j:i}, \\ R''(i) \leq j \end{matrix} \tag{A2.3}$$

then $E''_{j:h} = GR^{\omega}(S''_{j:h}, O_h, M'')$ and $S''_{j:h+1} = TR(S''_{j:h}, E''_{j:h}, M'')$

Proof sketch. Follows from the assumptions, and definitions 3-5. □

The previous lemma's assumptions can always be met starting from *any* interpretation M. Firstly, since in the worst case, from any interpretation we can obtain a successor starting at the institution's initial state - so both successors agree at least on the initial state. Secondly, by making the non-deterministic choice in the event generation operation to select the same rule modifications for both the successor and the successor to the successor (in the worst case, no rule modifications). We can continue to incrementally produce successive interpretations that monotonically increase the time point they agreed upon. Note that, this may mean backtracking by changing preceding interpretations (e.g. selecting no rule modifications).

Lemma 3. *There exists a fixed point for the interpretation successor operation denoted $SUCC^{\omega}(M,et)$ for any M and et.*

Proof sketch. A proof can be obtained by structural induction, applying Lemma 2, and ensuring each successive interpretation agrees with the preceding interpretation on rule modifications (potentially removing modifications in previous interpretations). □

In fact, there can be multiple fixed points, as exemplified:

Example 4.1. An institution $\mathcal{I}$ contains a legislative rule with the id *leg0* $\in \mathbb{ID}$ stating that an agent, Ada, voting to activate a rule ($vote_a(act,\ id,\ t) \in \mathcal{E}_{obs}$) *counts-as* activating the rule: $\mathcal{G}(\top, \{vote_a(act,\ id,\ t)\}) \ni mod(act,\ id,\ t)$. In the initial state the legislative rule is active $\Delta = \{active(leg0)\}$. In an observable event trace $et = \langle O_0 \rangle$ Ada votes to activate another rule with the id *leg1* $\in \mathbb{ID}$ in the initial state $O_0 = \{vote_a(act,\ leg1,\ 0)\}$.

From an initial empty interpretation M we have the following successors and interpretations for example 4.1(differences are in **bold**):

$$M^2 = SUCC(M,\ et) = SUCC^{\omega}(M,\ et) \text{ s.t. } V^2 = \langle V^2_0\rangle, R^2(0) = 0, R^2(1) = 0,$$
$$S^2_{0:0} = \{active(leg0)\}, S^2_{0:1} = \{active(leg0)\}, E^2_{0:0} = \{vote_a(act,\ leg1,\ 0)\}$$
$$M^1 = SUCC(M,\ et) = SUCC^{\omega}(M,\ et) \text{ s.t. } V^1 = \langle V^1_0, V^1_1\rangle, R^1(0) = 1, R^1(1) = 1,$$
$$S^1_{0:0} = \{active(leg0)\}, S^1_{0:1} = \{active(leg0)\},$$
$$E^1_{0:0} = \{vote_a(act,\ leg1,\ 0), \boldsymbol{mod(act,\ leg1\ ,\ 0)}\}$$
$$S^1_{1:0} = \{active(leg0)\}, S^1_{1:1} = \{active(leg0)\}, E^1_{1:0} = \{vote_a(act,\ leg1,\ 0)\}$$
$$M^0 = SUCC(M,\ et) = SUCC^{\omega}(M,\ et) \text{ s.t. } V^0 = \langle V^0_0, V^0_1\rangle, R^0(0) = 1, R^0(1) = 1,$$
$$S^0_{0:0} = \{active(leg0)\}, S^0_{0:1} = \{active(leg0)\},$$
$$E^0_{0:0} = \{vote_a(act,\ leg1,\ 0), \boldsymbol{mod(act,\ leg1\ ,\ 0)}\}$$
$$S^0_{1:0} = \{active(leg0)\}, S^0_{1:1} = \{active(leg0), \boldsymbol{active(leg1)}\},$$
$$E^0_{1:0} = \{vote_a(act,\ leg1,\ 0), \boldsymbol{mod(act,\ leg1\ ,\ 0)}\}$$

Each fixed point has different rule modifications. M^2 does not add the rule *leg1*. M^1 contains an attempt to add the rule in the version zero but not in version one. Finally, M^0 adds the rule in the version zero and version one, version one being the current version when the rule is added meaning the rule addition is successful. In fact, the following prioritisation holds $M^0 < M^1 < M^2$ meaning that M^0 maximises successful rule modifications.

Models are interpretations which maximise successful rule modifications. Thus we characterise models by combining the successor interpretation fixed point and interpretation prioritisation. Given an empty interpretation we find a fixed point successor interpretation for a given event set trace (8.1). The fixed point is a model if there is no greater prioritised successor fixed point interpretation (8.2).

Definition 8. ***Models*** *Let $M = \langle R, V\rangle$ be an empty interpretation such that $V = \langle V_0\rangle$, $V_0 = \langle E_0, S_0\rangle$, $E_0 = \langle\rangle$ and $S_0 = \langle\rangle$. The interpretation $M' = \langle R', V'\rangle$ is a model for $\mathcal{I}$ w.r.t. an observable event set trace $et = \langle O_0, ..., O_k\rangle$ iff:*

$$M' = SUCC^{\omega}(M,et) \quad \textbf{\textit{and}} \tag{8.1}$$

$$\textit{There does not exist an } M'' < M' \textit{ meeting 8.1.} \tag{8.2}$$

From lemma 3 and definition 8 we have the following property.

Lemma 4. *There exists at least one model for any institution $\mathcal{I}$ w.r.t. an observable event set trace et.*

These semantics operationalise answering "when does a rule change count-as a legal rule change?". Generally, a rule change counts-as a legal rule change *if and only if* a rule ascribes the change in a context that is consistent with the modification. Models always contain 'legal' rule modifications, defined as fixed point interpretations which maximise rule modifications. So, 'legal' rule-changes occur in at least one model whilst illegal rule changes do not occur at all (the non-deterministic choice for a rule modification to occur in 4.4) and the institution continues to operate 'as usual', meeting our desiderata.

5 Case Studies

Now we apply the framework to concrete case studies. For brevity we use variables to denote: all rule identifiers ($id \in \mathbb{ID}$), all rule change operations ($op \in \{act, deact\}$), and all time instants ($t \in \mathbb{N}$). The first case concerns a simple rule change procedure.

Case 5.1. An institution $\mathcal{I}^{sgov}$ describes a simple government comprising two agents, Ada and Bertrand. Both Ada and Bertrand voting to activate or deactivate a rule in the context that neither are criminals ($crim(ada), crim(bert) \in \mathcal{F}^{sgov}_{dom}$) *counts-as* activating/deactivating the rule. The rule modifying counts-as rules are identified with $leg0 \in \mathbb{ID}$ and formalised as $\mathcal{G}^{sgov}(\neg crim(ada) \wedge \neg crim(bert), \{vote_a(op,\ id,\ t), vote_b(op,\ id,\ t)\}) \ni mod(act,\ id,\ t)$. At time point one Ada and Bertrand vote to add a rule with id *crim0*, $O_1 = \{vote_a(act,\ crim0,\ 1), vote_b(act,\ crim0,\ 1)\}$. The rule identified as *crim0* states that if Ada or Bertrand are found guilty of a crime ($g(ada), g(bert) \in \mathcal{E}^{sgov}_{obs}$) then they become criminals, formally - $\mathcal{C}^{\uparrow}(\top, g(ada)) \ni crim(ada)$ and $\mathcal{C}^{sgov\uparrow}(\top, g(bert)) \ni crim(bert)$. Next, Bertrand is found guilty of a crime $O_2 = \{g(bert)\}$. Finally, Bertrand and Ada vote to deactivate the criminalising rule, $O_3 = \{vote_a(act,\ crim0,\ 3), vote_b(act,\ crim0,\ 3)\}$.

For clarity, models are represented graphically. The model for case 5.1 is shown in Figure 1. Lines represent when domain and active rule fluents hold. We distinguish between whether a fluent holds in a state $S_{v:t}$: 1. retroactively in the version's past and *not* in the previous version, - - - - (i.e. $R(t) < v$ and $\langle M, S_{v-1:t}\rangle \not\models f$), 2. when the version is the *current version*, ——) (i.e. $R(t) = v$), and 3. when the version is *obsolete*, ······ (i.e. $R(t) > v$). Time instants are marked if they have successful or non-successful rule modification events in versions where modifications can have an effect (i.e. non-obsolete versions): 1. ▶ denoting that all the rule modification events occurring in the previous version occur again (i.e. $E_{v:t} \cap \mathcal{E}_{mod} = E_{v-1:t} \cap \mathcal{E}_{mod}$). Meaning, the conditions (contexts) for the rule modifying events to be ascribed are consistent with the version and therefore with applying the rule modifications (the non-deterministic choice to include a rule modification in $E_{v:t}$ according to 4.4 is always made) 2. ◀ denoting that at least one rule modification event which occurred in the previous version does not occur again (i.e. $E_{v:t} \cap \mathcal{E}_{mod} \neq E_{v-1:t} \cap \mathcal{E}_{mod}$). Meaning, the conditions (contexts) for rule modifying events to be ascribed are inconsistent with the version they occur in and therefore with applying the rule modifications (a non-deterministic choice according to 4.4 to *not* include a rule modification is made when building $E_{v:t}$).

Figure 1 shows case 5.1's model. Throughout version zero the legislative rule (*leg0*) is active, stating Ada and Bertrand voting to add a rule counts-as adding a rule. When at time instant one Ada and Bertrand vote to add a new rule (*crim0*), stating people found guilty become criminals, the model succeeds to version one where the new rule is successfully added. At time instant three Bertrand becomes a criminal. When they vote again to modify a rule it is unsuccessful, since rule change is conditional on neither being criminals. Adding a criminalising rule altered the built social reality in version one's future, changing what could be ascribed as a legal rule modification.

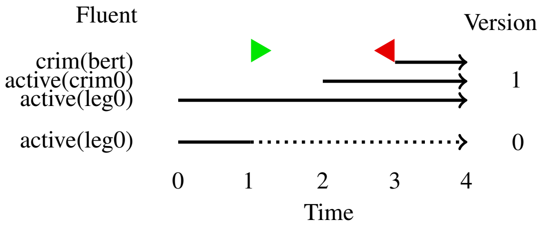

Figure 1. Model for case 5.1 with two institution versions.

The next case presents an institution $\mathcal{I}^{uk}$ representing the UK's legislation rules. The cases are based on past changes to a court decision on UK tax laws (Pad), and past changes to tax laws (Fin [2008]). The UK government can unconditionally enact any rules effective at any time. Observable events where the government activates/deactivates a rule (*gmod(op, id, t)*) count-as modifying the rule (*mod(op, id, t)*. Legislative rules identified as $leg0 \in \mathbb{ID}$ cause rule activations $\mathcal{G}^{uk}(\top, \{gmod(act,\ id,\ t)\}) \ni mod(act,\ id,\ t)$ and legislative rules identified as $leg1 \in \mathbb{ID}$ cause rule deactivations $\mathcal{G}^{uk}(\top, \{gmod(deact,\ id,\ t)\}) \ni mod(deact,\ id,\ t)$. A model $M^{uk} = \langle R^{uk}, V^{uk}\rangle$ is produced for an observable event trace $et = \langle O_0, O_1, O_2, O_3, O_4\rangle$ for $\mathcal{I}^{uk}$. The model comprises four versions $V^{uk} = \langle V^{uk}_0, V^{uk}_1, V^{uk}_2, V^{uk}_3\rangle$. We begin the case:

Case 5.2. A rule states that any UK resident (e.g. person *a* resides in the UK -*r(a, uk)*) in a business partnership in the UK (*p(a, uk)*) or elsewhere such as Jersey (*p(a, jers)*) in the first tax year month is obliged to pay tax (*oblt*). We have for all locations $L \in \{uk, jers\}$ a tax rule $\mathcal{C}^{uk\uparrow}(r(a,\ uk) \wedge p(a,\ L), mon1) \ni oblt$ identified as $tax0 \in \mathbb{ID}$. Initially the legislative rules *leg0* and *leg1*, and the tax rule *tax0* are active ($\Delta^{uk} = \{active(leg0), active(leg1), active(tax0)\}$). At time point one it is the first tax year month ($O_1 = \{mon1\}$). Following a court challenge (Pad) the government retroactively replaces the tax rule with id *tax0* with a new rule with id *tax1* ($O_2 = \{gmod(deact,\ tax0,\ 0), gmod(act,\ tax1,\ 0)\}$). The new rule, *tax1*, states that only people in a UK business partnership are obliged to pay tax - $\mathcal{C}^{uk\uparrow}(r(a,\ uk) \wedge p(a,\ uk), mon1) \ni oblt$.

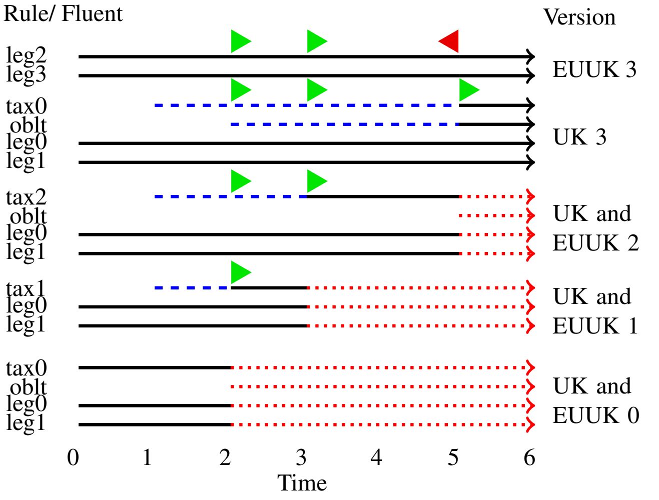

Figure 2. Model for case 5.2 with four institution versions for the institution $\mathcal{I}^{uk}$ (denoted UK) and a model for case 5.3 with four versions for the institution $\mathcal{I}^{euuk}$ (denote EUUK). Institutions $\mathcal{I}^{uk}$ and $\mathcal{I}^{euuk}$ have identical versions 0 to 2. Not shown, in all states person 'a' is in a Jersey-based business partnership (*p(a,jersey)*) and is a UK resident (*r(a,uk)*).

In Figure 2 version zero (V^{uk}_0) obliges person 'a' to pay tax in state two. In state two ($S^{uk}_{0:2}$) the current institution version changes when the rule obliging UK residents to pay tax (*tax0*) is replaced with the rule obliging UK business partners to pay tax (*tax0*) ($act(t1,0), deact(t0,0) \in E^{uk}_{0:2}$) to version one ($V^{uk}_1$ s.t. $R^{uk}(2) = 1$). Due to this change, the version one (V^{uk}_1), does not oblige tax to be paid in its third state ($S^{uk}_{1:2}$) since person *a* resides in the UK but is in a Jersey-based business partnership.

Case 5.2 (Continued)**.** The government partially reverses the tax change at time point three. This is by retroactively replacing the rule obliging people in a UK business partnership to pay tax (*tax1*) with new rule identified as *tax2* ($O_3 = \{gmod(deact,\ tax1,\ 0), gmod(act,\ tax2,\ 0)\}$). The new rule obliges

UK *residents* in a business partnership to pay tax if it does not criminalise them retroactively (i.e. in a retroactive state an obligation to pay tax is initiated conditional on the obligation holding in the next state of the previous version). For all locations $L \in \{uk, jersey\}$ the rule is $\mathcal{C}^{uk\uparrow}((r(a,\ uk) \wedge p(a,\ L)) \rightarrow (P) \rightarrow PrV(NS(oblt))), mon1) \ni oblt$. Next, it is the first tax year month again ($O_4 = \{mon1\}$).

In Figure 2 version two (V_2^{uk}), like version zero, does not oblige 'a' to pay tax in the past. But, it does oblige them to pay tax after the second time the first tax year month occurs ($mon1 \in E_{2:4}^{uk}$).

Case 5.2 (Continued)**.** The UK government decides to reverse the previous judgements going back to the original rule set ($O_5 = \{gmod(deact,\ tax2,\ 0), gmod(act,\ tax0,\ 0)\}$).

In Figure 2, version three (V_3^{uk}) reverts to the original legislation. Thus we have the same situation as if the legislation in version zero had not been modified. That is, an obligation to pay tax after the first occurrence of the first tax year month ($mon1 \in E_{3:1}^{uk}$).

The next case is a variation on the previous describing an institution $\mathcal{I}^{euuk}$, incorporating EU human rights law.

Case 5.3. The European Convention on Human Rights [Council of Europe, 1953, Art. 7] (ECHR) blocks retroactive legislative modifications that *criminalise* formerly innocent people. The institution $\mathcal{I}^{euuk}$ contains the same rules as $\mathcal{I}^{uk}$ with the same identifiers minus the legislative rules *leg0* and *leg1*. Instead, legislative rules state that observable rule modifications *count-as* rule modifications *conditional* on the changes not retroactively criminalising people. In all states where rules are being applied retroactively, if there is not an obligation to pay tax in the previous version then there must not be an obligation to pay tax in the current version. We have rules with the identifier *l2*: $\mathcal{G}^{euuk}(PaS(P \rightarrow PrV(\neg oblt) \rightarrow \neg oblt)), \{gmod(act,\ id,\ t)\}) \ni act(id,\ t)$, and rules with the identifier *l3*: $\mathcal{G}^{euuk}(PaS(P \rightarrow PrV(\neg oblt) \rightarrow \neg oblt)), \{gmod(deact,\ id,\ t)\}) \ni deact(id,\ t)$. Initially, person 'a' is in a Jersey based business partnership (*p(a,jersey)*) and is a UK resident (*r(a,uk)*), and the first tax rule and the legislative rules conditional on being non retroactively criminalising are active such that $\Delta^{euuk} = \{p(a,uk), r(a,uk), tax0, l2, l3\}$. The same events occur as in case 5.2, $et = \langle \emptyset, \{mon1\},\ \{gmod(deact,\ t0,\ 0), gmod(act,\ t1,\ 0)\},\ \{gmod(deact,\ t1,\ 0),\ gmod(act,\ t2,\ 0)\},\ \{mon1\}, \{gmod(deact,\ t2,\ 0),\ gmod(act,\ t0,\ 0)\}\rangle$.

Figure 2 shows a model M^{euuk} for $\mathcal{I}^{euuk}$. The first three versions are identical to our previous case 5.2 (where the UK's legislature was not constrained by EU rules blocking retroactively criminalising modifications), since the first two rule modifications do not criminalise people retroactively. Unlike in our previous case 5.2, the version two contains no tax rules. The reason being that tax rule two - "obliging uk residents in a business partnership to pay tax but on the condition that if it is retroactive then those people were obliged to pay tax in the previous version", is deactivated since its deactivation does not criminalise retroactively. On the other hand, tax rule zero - "any UK resident in a business partnership in the first tax year month is obliged to pay tax" ($\mathcal{C}^{uk\uparrow}(r(a,\ uk) \wedge p(a,\ L), mon1) \ni oblt$) is not reactivated, even though it was reactivated in our previous case 5.2. Its reactivation would retroactively criminalise people if activated in version three, meaning its activation does not occur since legislative rule - *l2*: $\mathcal{G}^{euuk}(PaS(P \rightarrow PrV(\neg oblt) \rightarrow \neg oblt)), \{gmod(act,\ id,\ t)\}) \ni act(id,\ t)$ - has a condition that is not met.

The next cases look at modifying legislative rules themselves.

Case 5.4. An institution $\mathcal{I}^{p}$ describes a parliament that can retroactively modify rules through a majority vote $pvote(act,\ id,\ t) \in \mathcal{E}_{obs}$. The legislative rules are identified the id $parl0 \in \mathbb{ID}$ for activating rules $\mathcal{G}^{p}(\top, \{pvote(act,\ id,\ t)\}) \ni mod(act,\ id,\ t)$ and with the id $parl1 \in \mathbb{ID}$ for deactivating rules $\mathcal{G}^{p}(\top, \{pvote(deact,\ id,\ t)\}) \ni mod(deact,\ id,\ t)$. In the initial state all rules are active such that $active(id) \in \Delta$. In an observable event set trace $tr = \langle O_0, O_1 \rangle$ at time point one the parliament votes to retroactively remove the rule which ascribes retroactive modifications ($O_1 = \{pvote(deact,\ parl1,\ 0)\}$.

Depicted in Figure 3 a single model $M^p = \langle R^p, V^p \rangle$ comprises two institution versions $V^p = \langle V_0^p, V_1^p \rangle$. An event occurs in version zero at time instant one, where the parliament votes to retroactively modify a rule and the corresponding rule modification event occurs ($E_{0:1}^{p} = \{pvote(deact,\ parl1,\ 0), mod(deact,\ parl1,\ 0)\}$). Consequently the institution transitions to version one ($R^p(1) = 1$). Importantly, in version one, the same rule modifying event *does not occur*. The reason being, if the modification event did occur then the rule *parl1* ascribing the modification event - $\mathcal{G}^{p}(\top, \{pvote(deact,\ id,\ t)\}) \ni mod(deact,\ id,\ t)$ - would be inactive in version one state one $S_{1:1}^{p}$, and the deactivation could not occur in the first place (contradiction). This exemplifies how the formalism always guarantees a model, paradoxical rule modifications do not occur if they make the rule modifying event impossible in the first place.

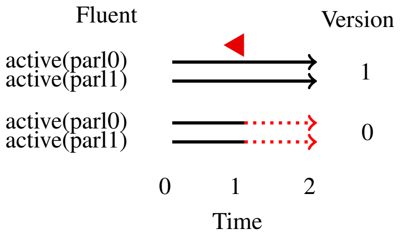

Figure 3. Model for case 5.4

The next case extends the previous case 5.4:

Case 5.5. This case describes an institution $\mathcal{I}^{mp}$ where a monarch and a parliament can retroactively modify rules, including all the rules from the previous case's institution $\mathcal{I}^{p}$. Additionally, a rule identified as $fence0 \in \mathbb{ID}$ states that if a fence is built $fb \in \mathcal{E}_{obs}^{mp}$ it is obliged the fence is painted white $oblpf \in \mathcal{F}_{inst}^{mp}$ - $\mathcal{C}^{mp\uparrow}(\top, fb) \ni oblpf$. A rule identified as *mon0* states the monarch issuing a rule change decree $mdecree(act,\ id,\ t) \in \mathcal{E}_{obs}^{mp}$ to activate a rule counts-as activating the rule - $\mathcal{G}^{mp}(\top, \{mdecree(act,\ id,\ t)\}) \ni mod(act,\ id,\ t)$. A rule identified as *mon1* state the monarch issuing a decree to deactivate a rule counts-as deactivating the rule $\mathcal{G}^{mp}(\top, \{mdecree(deact,\ id,\ t)\}) \ni mod(deact,\ id,\ t)$. All legislative rules are initially active, but the fence painting rule is not (s.t. $active(fence0) \notin \Delta^{mp}$). At time point one the parliament votes for the fence-painting obligation rule to be activated, ($O_1 = \{pvote(act, fence0,\ 1)\}$), a fence is built ($O_2 = \{fb\}$), the monarch issues by decree the fence-building rule to be retroactively deactivated at the time it was activated, cancelling its activation ($O_3 = \{mdecree(deact, fence0,\ 1)\}$). Finally, the parliament votes to retroactively disenable the monarch from deactivating rules ($O_4 = \{pvote(deact,\ mon1,\ 0)\}$).

Depicted in Figure 4 the model $M^{mp} = \langle R^{mp}, V^{mp} \rangle$ comprises four versions $V^{mp} = \langle V_0^{mp}, V_1^{mp}, V_2^{mp}, V_3^{mp} \rangle$. At version zero time instant zero the parliament votes to add the rule obliging built fences to be painted white, causing a rule modification event ($E_{0:1}^{mp} = \{pvote(act, fence0,\ 1), mod(act, fence0,\ 1)\}$) and the institution to transition to the version one ($R^{mp}(1) = 1$) where the same modification occurs ($E_{1:1}^{mp} = \{pvote(act, fence0,\ 1), mod(act, fence0,\ 1)\}$).

In the version one time instant two building a fence ($fb \in E^{mp}_{1:2}$) causes an obligation to paint the fence $oblpf \in S^{mp}_{1:3}$. At time instant three the monarch retroactively deactivates the fence painting rule ($mdecree(deact, fence0,\ 1) \in E^{mp}_{1:3}$) causing the institution to transition to the version two ($R^{mp}(3) = 2$) where the modification takes effect ($mdecree(deact, fence0,\ 1) \in E^{mp}$). Consequently, the fence painting obligation rule is deactivated and its effects (an obligation) no longer hold. When the parliament retroactively removes the ability for the monarch to deactivate rules the institution transitions to the final version three ($R^{mp}(4) = 3$) where the parliament's retroactive rule removal takes effect ($pvote(deact,\ mon1,\ 0), mod(deact,\ mon1,\ 0) \in E^{mp}_{3:4}$) causing the monarch's modifications to be unravelled (note that at the final version's third time instant the monarch's rule modification is unsuccessful even though it was successful in the previous version). Consequently, the fence painting obligation rule and its effects (an obligation) is reinstated by retroactively removing the ability to deactivate the fence painting rule.

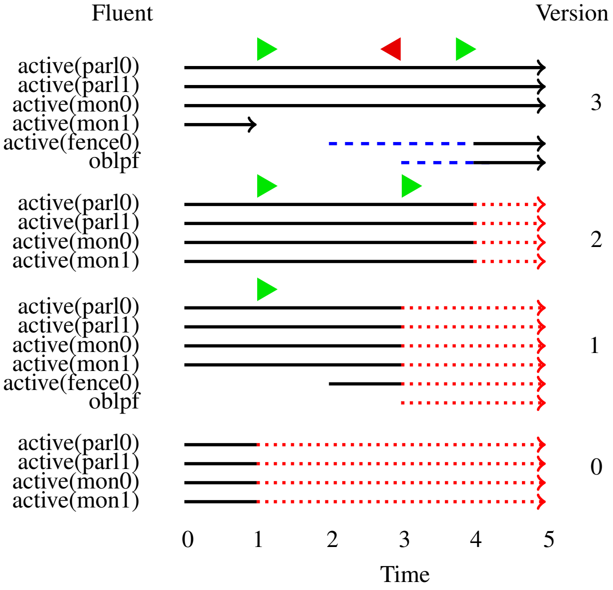

Figure 4. Model for case 5.5 with four institution versions.

6 Related Work

In (Governatori *et al.* [2005]; Governatori and Rotolo [2010]) a defeasible logic is proposed for temporal rule modification operations. Operations include, in (Governatori and Rotolo [2010]), complete rule removal (annulment) and removing immediate rule effects (abrogation). Meta-rules are used to introduce rule changes, which bear similarity to our rule-modifying counts-as rules. However, the meta-rules are only conditional on a single state. For comparison, we formalise richer conditions on past versions, states and hypothetical rule changes required to capture a number of important examples we address (such as rule change being non-retroactively criminalising). The focus of these papers is on rule change operations found in the legal domain, rather than the relation between ascribing a social reality and rule modifications with counts-as rules. In (López and Luck [2003]; López *et al.* [2006]) electronic institutions are specified in the Z specification language where legislation norms restrict legislative actions. The conditions for legislation norms are less expressive than our proposal and the authors do not consider the interdependency between changing rules in the past/present/future and the built social reality. On the other hand, in Boella and van der Torre [2004] rule modifications ascribed by counts-as rules are formalised where there is such a potential interdependency, but the setting is non-temporal.

Our proposal is thematically related to work in the institutional/normative reasoning sphere, in particular work on: 1. constitutive rule classes (Grossi *et al.* [2005, 2006, 2008]; Grossi [2008, 2011]), 2. norm change postulates (Boella and van der Torre [2004]), 3. detecting and/or resolving norm inconsistencies (Jiang *et al.* [2015]; Jiang [2015]; Kollingbaum *et al.* [2007]; Corapi *et al.* [2011]; Li [2014]; Vasconcelos *et al.* [2008]), rectifying *non-compliant* institutions (King *et al.* [2015a,b]) and 4. temporal norm updates (Alechina *et al.* [2013]; Knobbout *et al.* [2014]). However, these papers do not look at rule change legality ascribed by constitutive rules over time.

7 Conclusions

This paper answers "when do rule changes count-as legal rule changes?" with a novel formalism. Our framework formalises reasoning about institutional rule change over time ascribed by counts-as rules. A novel semantics defines how an institution evolves from one social reality to the next and from one version of rules to another. Under the proposed semantics counts-as rules define the past/present/future social reality. In turn, rule modifications change counts-as rules in the past/present/future and therefore the constructed social reality.

A rule change counts-as a legal rule change if and only if - 1. the rule change is ascribed by counts-as rules, conditional on a context which can include the potential changes to the social reality the rule modification *would* make. 2. the rule change results are consistent with the context the rule change is conditional on. In particular, taking into account the rule modification's past/present/future effect on counts-as rules, any changes to previous rule modifications, and the rule modification being 'undone' by future modifications. Legal rule changes always occur. Meeting our desiderata, illegal rule changes do not occur and the institution continues to operate 'as usual'.

There are many avenues for future work. First, extending the framework to deal with further cases. In particular, rules which explicitly block retroactive modifications altogether (e.g. [USC, Art. 1 Sec. 9 Cl. 3] "No Bill of Attainder or ex post facto Law shall be passed"). Preventing retroactive modifications can be expressed as a lack of a rule ascribing past rule changes, but not rules which *block* past rule changes. Second, the fixed-point institutional model characterisation can be implemented in any adequately expressive language. One possibility is Answer-Set Programming (Gebser *et al.* [2011b]; Gelfond and Lifschitz [1988]) as used in the InstAL framework (Cliffe *et al.* [2007]). Third, agent planning for rule changes, such as by building on existing agent-planning in Answer-Set Programming (Gebser *et al.* [2011a]; Lifschitz [1999]). Fourth, looking at agents bringing about legal rules, known as social commitments (e.g. promises, contracts), through locutions (Austin [1975]). In particular, looking at how social commitments can be created with the special role of ascribing changes to social commitments, such as by building on Event Calculus based frameworks (Chesani *et al.* [2012]; Günay and Yolum [2012]).

ACKNOWLEDGEMENTS

This research was supported by the SHINE [2] project of TU Delft. The authors would like to thank the anonymous referees for their input.

[2] http://shine.tudelft.nl

REFERENCES

Natasha Alechina, Mehdi Dastani, and Brian Logan. Reasoning about normative update. *IJCAI International Joint Conference on Artificial Intelligence*, pages 20–26, 2013.

Giulia Andrighetto, Guido Governatori, Pablo Noriega, and Leendert van der Torre. *Normative Multi-Agent Systems*, volume 4. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

John Langshaw Austin. *How To Do Things With Words*. Oxford university press, 1975.

Carlo Biagioli. Towards a legal rules functional micro-ontology. In *1st LegOnt Workshop on Legal Ontologies.*, 1997.

Guido Boella and Leendert van der Torre. Regulative and Constitutive Norms in Normative Multiagent Systems. *KR'04*, pages 255–265, 2004.

Federico Chesani, Paola Mello, Marco Montali, and Paolo Torroni. Representing and monitoring social commitments using the event calculus. *Autonomous Agents and Multi-Agent Systems*, 27(1):85–130, jun 2012.

Owen Cliffe, Marina De Vos, and Julian Padget. Answer set programming for representing and reasoning about virtual institutions. *Computational Logic in Multi-Agent Systems*, pages 60–79, 2007.

Owen Cliffe. *Specifying and Analysing Institutions in Multi-Agent Systems Using Answer Set Programming*. PhD thesis, University of Bath, 2007.

Domenico Corapi, Alessandra Russo, Marina De Vos, Julian Padget, and Ken Satoh. Normative design using inductive learning. *TPLP*, 4-5:783–799, 2011.

Council of Europe. European Convention on Human Rights, 1953.

Finance Act 2008, Chapter 9 (United Kingdom), 2008.

Martin Gebser, Roland Kaminski, Murat Knecht, and Torsten Schaub. Plasp: A prototype for PDDL-based planning in ASP. *Logic Programming and Nonmonotonic Reasoning*, pages 358–363, 2011.

Martin Gebser, Benjamin Kaufmann, and Roland Kaminski. Potassco: The Potsdam answer set solving collection. *AI Communications*, 24(2):107 – 124, 2011.

Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In *ICLP/SLP*, pages 1070 – 1080, 1988.

Guido Governatori and Antonino Rotolo. Changing Legal Systems: Legal Abrogations and Annulments in Defeasible Logic. *Logic Journal of IGPL*, 18(1):157–194, 2010.

Guido Governatori, Monica Palmirani, Regis Riveret, Antonino Rotolo, and Giovanni Sartor. Norm modifications in defeasible logic. In *In Proceedings of JURIX'05*, pages 13–22, 2005.

Davide Grossi, John-Jules Meyer, and Frank Dignum. Modal logic investigations in the semantics of counts-as. *ICAIL '05: Proceedings of the 10th international conference on Artificial intelligence and law*, pages 1–19, 2005.

Davide Grossi, John Jules Ch Meyer, and Frank Dignum. Classificatory aspects of counts-as: An analysis in modal logic. *Journal of Logic and Computation*, 16(5):613–643, 2006.

Davide Grossi, J. J Ch Meyer, and Frank Dignum. The many faces of counts-as: A formal analysis of constitutive rules. *Journal of Applied Logic*, 6(2):192–217, 2008.

Davide Grossi. Pushing Anderson's Envelope: The Modal Logic of Ascription. In *9th International Conference on Deontic Logic in Computer Science (DEON 2008)*, pages 263–277, 2008.

Davide Grossi. Norms as ascriptions of violations: An analysis in modal logic. *Journal of Applied Logic*, 9(2):95–112, 2011.

A Günay and P Yolum. Detecting conflicts in commitments. *Declarative Agent Languages and Technologies IX*, pages 51–66, 2012.

Jie Jiang, Jeremy Pitt, and Ada Diaconescu. Rule Conflicts in Holonic Institutions. *2015 IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops*, pages 49–54, 2015.

Jie Jiang. *Organizational Compliance: An Agent-based Model for Designing and Evaluating Organizational Interactions*. PhD thesis, TU Delft, Delft University of Technology, 2015.

Thomas C King, Tingting Li, Marina De Vos, Virginia Dignum, Catholijn M Jonker, Julian Padget, and M Birna Van Riemsdijk. A Framework for Institutions Governing Institutions. In *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*, pages 473–481, 2015.

Thomas C King, Tingting Li, Marina De Vos, Catholijn M Jonker, Julian Padget, and M Birna Van Riemsdijk. Revising Institutions Governed by Institutions for Compliant Regulations. In *Coordination, Organizations, Institutions and Norms, LNCS 9628*. 2015.

Max Knobbout, Mehdi Dastani, and John-jules Ch Meyer. Reasoning about Dynamic Normative Systems. *Logics in Artificial Intelligence, Lectured Notes in Computer Science*, 8761:628–636, 2014.

Martin J. Kollingbaum, Wamberto W. Vasconcelos, Andres García-Camino, and Timothy J. Norman. Managing conflict resolution in norm-regulated environments. *ESAW 2007*, 2007.

Tingting Li. *Normative Conflict Detection and Resolution in Cooperating Institutions*. PhD thesis, University of Bath, 2014.

Vladimir Lifschitz. Answer set planning. *16th International Conference on Logic Programming*, (2):23–37, 1999.

Fabiola López Y López and Michael Luck. Modelling Norms for Autonomous Agents. In *Proceedings of The Fourth Mexican Conference on Computer Science*, pages 238–245. IEEE Computer Society, 2003.

Fabiola López y López, Michael Luck, and Mark D'Inverno. A normative framework for agent-based systems. *Computational and Mathematical Organization Theory*, 12(2-3):227–250, oct 2006.

Javier Morales, M Lopez-Sanchez, Juan A. Rodriguez-Aguilar, Michael Wooldridge, and Wamberto Vasconcelos. Minimality and Simplicity in the On-line Automated Synthesis of Normative Systems. In *Proceedings of the 13th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2014)*, pages 109–116, 2014.

Javier Morales, Maite López-Sánchez, Juan Antonio Rodríguez-Aguilar, Michael Wooldridge, and Wamberto Vasconcelos. Synthesising Liberal Normative Systems. In *Proceedings of the 15th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2015)*, pages 433–441, 2015.

Padmore v IRC (1987) STC 36 affirmed by the Court of Appeal (1989) STC 493.

John R. Searle. *Speech acts: An essay in the philosophy of language*. Cambridge university press, 1969.

John R. Searle. What is an institution? *Journal of Institutional Economics*, 1:1–22, 2005.

Peter Suber. *The Paradox of Self-Amendment: A Study of Law, Logic, Omnipotence, and Change*. Peter Lang International Academic Publishers, 1990.

The United States Constitution.

Wamberto W. Vasconcelos, Martin J. Kollingbaum, and Timothy J. Norman. Normative conflict resolution in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 19(2):124–152, nov 2008.