Bringing Action Language C+ to Normative Contexts: Preliminary Report

Dario Garcia-Gasulla and Juan Carlos Nieves

Knowledge Engineering and Machine Learning Group
Universitat Politècnica de Catalunya
C/Jordi Girona 1-3, K2M-201, Barcelona, Spain
Email: {dariog, jcnieves}@lsi.upc.edu

Abstract. C+ is an action language for specifying and reasoning about the effects of actions and the persistence of facts over time. Based on it we present CN+, an operational enhanced form of C+ designed for representing complex normative systems and integrate them easily into the semantics of the causal theory of actions. The proposed system contains a particular formalization of norms using a life-cycle approach to capture the whole normative meaning of a complex normative framework. We discuss this approach and illustrate it with examples.

Key words: Action specification languages, Norm-based systems, Causal logic (C+)

1 Introduction

All real world domains are subject at some level to norms, be them physics, legal or social. Generically, norms define the socially accepted behavior within a society. They are an essential part of any domain, since all behavior within a domain is tied to its norms. In order to represent realistically any domain it is therefore necessary to specify those norms with the same level of detail used for the rest of the domain. Moreover, to achieve functionality norms must be specified using the same terms used to define the domain those norms regulate.

$C+$[6] is an action language based on nonmonotonic causal logic. It is based on the *Principle of Universal Causation* (PUC) and uses causal rules to define the behavior of a domain. Several attempts have been done to extend C+ in order to allow the representation of norms in it [3], [4]. Those attempts opened the path to norm formalization in C+, but did not provide an explicit syntax for expressing the singularities of normative contexts.

In order to produce the tools required for complex normative monitoring and reasoning, we must be able to represent more information than just the current legal state (legal, ilegal) given a normative framework. To produce operational solutions we must also be able to know the exact status of each norm (applicable, respected) in a given situation and how every action will affect the internal status of each norm. A system will perform better knowing which norms is it subject to and which norms is it violating by performing actions which respect the norms

or which stop the system from being subject to them. To do so it is required a formalism which allows the specification of single norms, and which can express the whole variety of states a norm can be in.

In this paper we introduce a *preliminary representation of norms* in C+ which by the use of the same elements that define a domain is able to define the normative framework regulating it. The proposed syntax will allow the fast specification of norms, its parts and its status, in causal logic and the easy integration of those norms within the rest of the domain, contributing this way significantly to the expressiveness of C+.

The rest of the paper is structured as follows: In §2, we introduce some basic concepts of C+ which are relevant for the rest of the paper. In §3, our normative approach is introduced. In §4, we define some observable properties of our normative approach in terms of transition systems. In §5, a short overview of existing normative approaches which are related to our approach is presented. Finally, in the last section we outline our conclusions and future work.

2 Background

C+ [6] is an action language for specifying and reasoning about the effects of actions over time. It is based on nonmonotonic causal logic through which it describes an explicit transition semantics which allow the representation of complex features such as nondeterminism, indirect effects of actions, concurrency of actions, temporariness and inertial behavior of facts. These elements are extremely useful when trying to represent formally a real world domain. Its causal logic is based on the principle *every fact that is caused is satisfied and every fact that is satisfied is caused*. The second part, *every fact that is satisfied is caused*, expresses the 'principle of universal causation' which provides an interesting mathematical simplicity in the semantics of causal theories. Follows an overview of it. Most of what is said next is extracted from [6], and we recommend to read it in order to understand all the details of causal theories.

Based on that causal logic, a *multivalued propositional signature* is a set σ of symbols called *constants* and a nonempty finite set *Dom(c)* of symbols ('the *domain* of c'), for each constant c $\in \sigma$, being *Dom(c)* and σ disjoint. The set σ of symbols can be partitioned into a set σ^f of *fluent constants* and a set σ^a of *action constants*.

An *atom* of signature σ is an expression of the form $c = v$ ('the value of c is v') where $c \in \sigma$ and $v \in Dom(c)$. A *formula* of signature σ is any propositional combination of atoms of σ linked by the propositional connectives from classical logic.

An *interpretation I* of σ is a function that assigns every constant in σ to an element of its domain. An interpretation *I satisfies* an atom $c = v$ ($I \models c = v$) if *I(c) = v*. The satisfaction relation is extended from atoms to arbitrary formulas according to the usual truth tables from propositional connectives. A *model* of a set X of formulas is an interpretation that satisfies all formulas in X. If X has a model it is said to be *consistent* or *satisfiable*. If every model of a set X of

formulas satisfies a formula F, it is said that X *entails* F ($X \models F$). Two sets of formulas are *equivalent* if they have the same models.

A *causal rule* is an expression of the form: $X \Leftarrow Y$ where X and Y are formulas of σ, called the *head* and the *body* of the rule. That causal rule can be informally interpreted as: *If Y is true there is a cause for X to be true*. Rules with the head $\perp$ are called *constraints*. A *causal theory* is defined as a set of causal rules.

Now we define the concept of *model* for causal theories: Let T be a causal theory, and I be an interpretation of its signature. The *reduct* T^I of T relative to I is the set of heads of all the rules in T whose bodies are satisfied by I. I is a model of T if I is the unique model of T^I. Intuitively, T^I is the set of formulas that are caused, according to the rules of T, under interpretation I.

Every $C+$ action description D of signature (σ^f, σ^a) defines a labelled transition system $\langle S, A, R \rangle$ where:

- S is a nonempty set of *states*. A *state* is an interpretation of the fluent constants σ^f; $S \subseteq I(\sigma^f)$.
- A is a set of *transition labels*, also called *events* or *actions*. An *action* is an interpretation of the action constants σ^a; $A = I(\sigma^a)$.
- R is a set of labelled transitions, $R \subseteq S \times A \times S$.

A state is represented by the set of fluent atoms satisfied in it, and it can be defined as a complete and consistent set of fluent atoms. A formula 'holds' or 'is true' in a state s if s satisfies it. An action a is said to be *executable* in a state s if there is a transition *(s,a,s′)* in R, and *nondeterministic* in s if there are transitions *(s,a,s′)* and *(s,a,s′′)* in R such that *s′ ≠ s′′*.

3   Normative Approach

In order to integrate normative elements into C+ we need a formalization which is coherent with its semantics. Concretely we require the formalization to be compatible with the *fluents and actions* paradigm, from which can be obtained a state-based description of norms. In [9], [8] and [2] a normative analysis is developed which splits a norm content in three parts:

- Activation condition: Is the part of the norm which defines when the norm is active.
- Deactivation (or termination) condition: Is the part of the norm which defines when the norm stops being active.
- Maintenance (or violation) condition: Is the part of the norm which defines when the norm has been violated.

From this approach we obtain two independent elements which fully represent the meaning of any norm, *the norm's condition* and the *norm's content*. The norm's condition defines the situation requirements for a norm to be applicable or not in a given state and it contains both the *activation* and the *deactivation*

condition. The second part, the norm's content, specifies the actions or situations the norm regulates upon and it contains the *maintenance condition*. For example:

- Norm: *If you drive a car under poor visibility conditions, you are obliged to use the car's headlights*
- Condition: *you drive a car under poor visibility conditions*
- Content: *you are obliged to use the car's headlights*

Both the content and the condition define a set of situations in the terms of the symbols of σ. To do so each of them is represented by a formula of signature σ. As an example, the previous norm could be represented by the formulae:

- Condition: $drive\_car = \top \wedge visibility\_conditions = poor$
- Content: $headlights\_on = \top$

Considering those two formulae as satisfiable predicates, it can be proved whether or not an interpretation of signature σ is a model of them (as seen in §2). Taking an interpretation I of signature σ as the definition of the world in a given situation, by checking the satisfiability of I in a norm's formulae we can know the state of the norm in that situation. We can therefore obtain a *norm's life-cycle* based on states, which we already explored in [7], and addapt it to the sintax of $C+$. We can preview four cases based on the fulfillment or not of each of the norm's parts. This way, based on an interpretation of signature σ we can say a norm is:

- Active: The norm's condition is satisfied and therefore the norm is applicable.
- Inactive: The norm's condition is not satisfied and therefore the norm is not in use.
- Violated: The norm's content is not satisfied and therefore the norm is transgressed.
- Respected: The norm's content is satisfied and therefore the norm is fulfilled.

Each of the two norm's parts can be displayed by two timelines, one representing the condition state through time and one representing the content state, so that in every moment each norm is defined as *active/inactive* and *respected/violated*. A visual representation of that life-cycle can be seen in Figure 1.

Since the condition and the content of a norm may or may not refer to the same elements, their respective timelines act independently through time, being affected differently by the actions happening in the world. The information regarding the current state of a norm in a given state will be given by the combined state of both elements. The fact that a norm's activation state and violation state are independent of one each other may be controversial (a norm can be violated without being active). That is further discussed and justified at the end of this section.
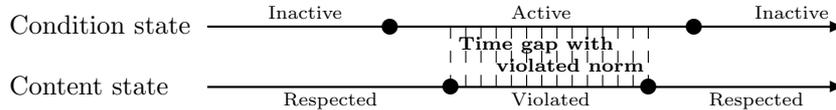

Figure 1: Visual representation of a norm's life-cycle

Definition 1. *Given a multivalued propositional signature σ, a σ-norm is a tuple of the form $\langle act, res \rangle$ such that act and res are two formulae of σ^f. act is called the condition of the norm and res is called the content of the norm.*

The norm used previously could be represented by the next σ-norm:
$$n = \langle drive\_car=\top \wedge visibility\_conditions=poor , headlights\_on=\top \rangle$$

This definition of a norm can be seen from a different perspective. As each of the σ-norm's subparts is a logical formula, we can obtain a norm definition based on the set of states in which the formulae of its σ-norm are satisfied. Concretely a norm is composed by two sets of states, that set containing all the states where the logical formula *act* representing the σ-norm condition is satisfied (which we will call ACT), and that set containing all the states where the logical formula *res* representing the σ-norm content is satisfied (which we will call RES).

Given a multivalued propositional signature σ, we write $\mathcal{I}_\sigma$ to denote the set of all the interpretations defined over σ.

Definition 2. *Let $n = \langle act, res \rangle$ be a σ-norm. A $\mathcal{I}_\sigma$-norm(n) is a tuple of the form $\langle ACT, RES \rangle$ such that $ACT = \{I | I \in \mathcal{I}_\sigma \text{ and } I \models act\}$ and $RES = \{I | I \in \mathcal{I}_\sigma \text{ and } I \models res\}$*

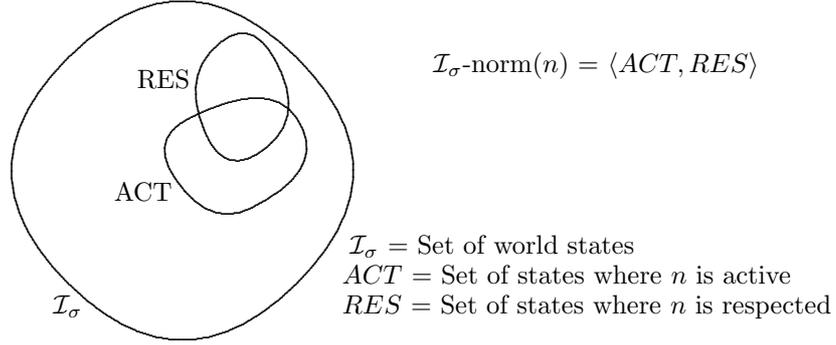A visual representation of Definition 2 can be seen in Figure 2.

$$\mathcal{I}_\sigma\text{-norm}(n) = \langle ACT, RES \rangle$$

$\mathcal{I}_\sigma$ = Set of world states
ACT = Set of states where n is active
RES = Set of states where n is respected

Figure 2: *Visual representation of a $\mathcal{I}_\sigma$-norm's subparts ACT and RES*

From the previous definitions, we can identify different readings of the status of a σ-norm regarding the states of the world and the meaning of the norm.

Definition 3. *Let $n = \langle act, res \rangle$ be a σ-norm and $\mathcal{I}_\sigma$-norm(n) = $\langle ACT, RES \rangle$. If $I \in \mathcal{I}_\sigma$, then the status of the σ-norm n w.r.t. I is:*
– Active if $I \in ACT$ or Inactive if $I \notin ACT$.
And:
– Respected if $I \in RES$ or Violated if $I \notin RES$.

It is important to understand the use we make of the concept *violation of a norm*. Since we take the condition and the content of a norm to be independent from each other, a norm can be violated without being active (which may go

against some interpretations of the word *violation*). To represent the fact that a norm is active and violated at the same time, we introduce the concept *infringed*. Only in the situations where the norm is violated and active at the same time we will say the norm's status is *infringed*, which are the states to be considered undesirable by the norm's syntax. The states where a norm is inactive and violated do not infringe the norm, even though it may be advisable to avoid them.

Definition 4. *Let $n = \langle act, res \rangle$ be a σ-norm and $\mathcal{I}_\sigma$-norm$(n) = \langle ACT, RES \rangle$. If $I \in \mathcal{I}_\sigma$, then the σ-norm n is in a* infringing *status w.r.t. I if:*
$-\ I \in ACT \wedge I \notin RES.$

The concept of *infringement* gives us more information about the state of a norm and about the possible effects of actions. Knowing that a norm is violated but not active in a given state allows us to classify the activating actions on that state as *infringing actions*, since the resultant state will result in an *infringed norm*. The same would work for violating actions in states were the norm is active. A visual representation can be seen in Figure 3.
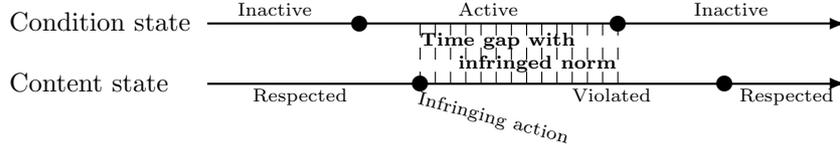

Figure 3: Visual representation of the infringment of a norm

Once we have defined how to specify norms in terms of formulae of the signature of a causal theory, we are in position for defining the concept of a *normative causal theory.*

Definition 5. *Let T be a causal theory of signature σ and N_σ be a finite set of σ-norms. A normative causal theory is a tuple of the form $\langle T, N_\sigma \rangle$.*

The concept of *model* of a normative causal theory is a single generalization of a model of a causal theory: Given a normative causal theory $T_N = \langle T, N_\sigma \rangle$, if I is a model of T then I is a model of T_N. The interesting part of a model I of a normative causal theory $\langle T, N_\sigma \rangle$ is that any $I \in \mathcal{I}_\sigma$ will always induce a particular status to every norm of N_σ since $ACT \cup \overline{ACT} = \mathcal{I}_\sigma$ and $RES \cup \overline{RES} = \mathcal{I}_\sigma$.

Proposition 1. *Let $T_N = \langle T, N_\sigma \rangle$ be a normative causal theory. If I is a model of T_N, then for each $n \in N_\sigma$, the status of n is Active or Inactive and Respected or Violated w.r.t I.*

4   Normative Properties of a Transition System

As was mentioned in Section 2, an action description in C+ can be regarded as a labelled transition system. In this section we are going to define some observable properties a normative causal theory adds to a labelled transition system.

The first definition we introduce is a basic classification of transitions based on the status of a σ-norm. By lack of space, we omit to the definition of an action description of $C+$ (please see [6] for its formal definition).

Definition 6. *Let $\langle S, A, R \rangle$ be a labeled transition system of an action description D, n a σ-norm and $\mathcal{I}_\sigma\text{-}norm(n) = \langle ACT, RES \rangle$. For each $r = (s, a, s\prime) \in R$:*

- r *is an* activating transition *of n if $s \notin ACT$ and $s\prime \in ACT$.*
- r *is a* deactivating transition *of n if $s \in ACT$ and $s\prime \notin ACT$.*
- r *is a* violating transition *of n if $s \in RES$ and $s\prime \notin RES$.*
- r *is a* respecting transition *of n if $s \notin RES$ and $s\prime \in RES$.*
- r *is an* infringing transition *of n if the status of n is not* infringing *in s and the status of n is* infringing *in $s\prime$.*

A visual representation of the states and transitions defined above can be seen in Figure 4.

i:States where n is not active and respected   t1:Infringing and activating transition of n

ii:States where n is active and respected   t2:Infringing and violating transition of n

iii:States where n is infringed   t3:Activating transition of n

iv:States where n is not active and not respected   t4:Respecting transition of n

$\mathcal{I}_\sigma$ set of world states   Norm n $\langle$ACT,RES$\rangle$
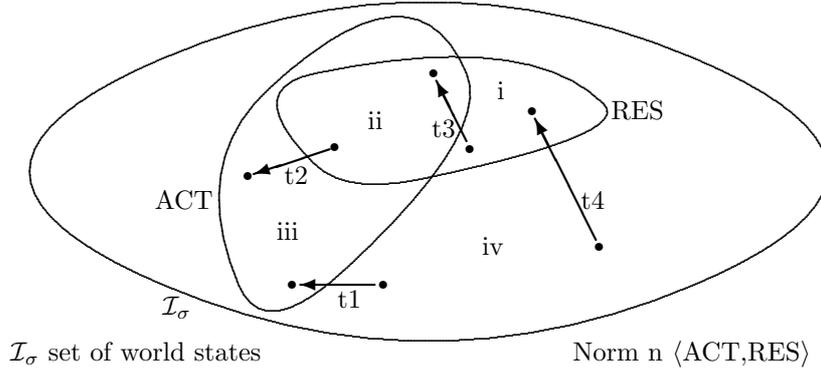
Figure 4: Visual representation of the transitions between the states of a norm

Given $r_1, r_2 \in R$ such that $\langle S, A, R \rangle$ is a labeled transition system, we say that r_1 and r_2 are *normative-different* with respect to a σ-norm if they are different transitions in terms of Definition 6.

Proposition 2. *Let $\langle S, A, R \rangle$ be a labeled transition system of an action description D and n be σ-norm. If $r_1 = (s, a, s\prime), r_2 = (s, a, s\prime\prime) \in R$ such that r_1 and r_2 are normative-different, then $s\prime$ and $s\prime\prime$ define different status for n.*

Essentially this proposition suggest that two normative-different transitions in a transition systems necessarily get a different status of a given norm.
As an example of the integration of the elements and properties seen until now we will next see an example of how to formalize a norm and its involved elements in CN+. Lets consider the following norm:

If there is a standing elder or pregnant woman, you must leave them your seat.

The norm formalized as in Definition 1, σ-norm $n=\langle act,res\rangle$ where act (activated) and res (respected) are two logical formulas, would be represented as:

$$n = \langle\ sitting = \top\ , standing\_elder = \bot\ \wedge standing\_pregnant = \bot\ \rangle$$

From the previous formalization and following Definition 2, the set of states ACT and RES could be analyzed. Examples of the actions which would affect the state the σ-norm n could be:

- Activating action: *sit_down*
- Deactivating action: *stand_up*
- Respecting action: *elder_sit_down*
- Violating action: *appear_standing_pregnant*
- Infringing action: *appear_standing_elder* if $sitting = \top$

In this example the set of states where the norm is active are all those where you are sitting. At the same time, the set of states where the norm is respected are those where there is no pregnant or elder person standing. Both activating actions and violating actions can be infringing actions if performed in certain situations. Sitting down when an elder is standing is an activating and infringing action, and the appearance of an elder if you are sitting is a violating and infringing action.

By representing norms this way we not only represent the content and condition of individual norms, but also capture information regarding the reasons that cause the infringement of a norm and the resultant states. With that knowledge we can analyze the states related to a norm breaking event, the previous and posterior states, and by studying the related information we can learn about the behavior within the domain with regard to norms. Also, it is important to note that, since CN+ defines norms using the same symbols used to define the domain, CN+ norms can be as detailed and complex as the domain itself and can be implemented in C+ implementations, such as CCalc[6].

5   Other Approaches

Regarding the other attempts to integrate normative elements into *C+*, the most relevant ones are [3] and [4]. In [3] a more organizational approach is taken, using as main element the roles of the agents instead of the actions as done in this article. The authors define the necessary rules and constraints to represent the *Contract Net Protocol* with *C+'s labelled transition system*. To do so they split the social norms into four types depending on their meaning within roles. This approach is specially interesting regarding Multi Agent Systems, since norms are specified thinking in the interaction between different roles with different goals. For each norm's type a set of rules is proposed, but the resultant normative formalization is quite specific and complex, and therefore difficult to generalize

to other contexts.

In [4] an extended form of $C+$ is proposed called $nC+$ which uses deontic concepts in order to represent normative aspects of domains. $nC+$ was used as inspiration for the one presented in this article. It uses a coloring system which labels states and transitions as green (legal) or red (illegal) to represent states where norms are respected or not. Restrictions are discussed and examples provided. $nC+$ instead of formalizing norms in the terms of the signature as we do, represents the normative meaning in the states and transitions of the system. This requires the addition of new components to the transition system, one for stating the permitted states and one for stating the permitted transitions. Our approach is able to label states and transitions as valid or not as $nC+$ does, while giving information regarding the reasons for it, which $nC+$ fails to do since it does not specify independently norms or its parts. While $nC+$ can represent the normative system (always as a whole) and give information about the global state of the world, it can not monitor the specific status of single norms, and therefore cannot use that information for advanced normative monitoring and reasoning.

6 Conclusions and Future work

Based on the norm's lifecycle introduced in §3, which captures the whole normative meaning and behavior of a given norm, the proposed sintaxis allows the representation of a complete normative framework in the terms of causal logic. By using the same tools used to define the world, CN+ can the state of a set of norms within the domain. This approach provides the basics for normative monitoring and normative reasoning, facilitating in an intuitive way the analysis of the normative situation of a state. By studying the status of all norms in a given state and how those change though time affected by actions, CN+ can help discover the state of norms in a future or past state. By the use of C+ expressiveness power, CN+ can formalize complex laws (as complex as the domain) making it a potentially useful tool to support decision making tasks in strongly legislated domains.

In that scope, C+ has already been used to formalize complex scenarios [1] [5] using CCalc, a query oriented implementation of C+. Based on that, and with the goal of providing a working environment with integrated norms, we are currently working on an implementation of CN+ in CCalc's syntax. Proving that CN+ can be easily implemented in CCalc by formalizing human laws actually in use would reinforce the idea that C+ is a good and operational solution to model domains and that CN+ can capture all the whole meaning of a real normative context and fully integrate it into C+.

Acknowledgement

215890). The views expressed in this paper are not necessarily those of the ALIVE consortium.

References

1. V. Akman, S. T. Erdoğan, J. Lee, V. Lifschitz, and H. Turner. Representing the zoo world and the traffic world in the language of the causal calculator. *Artif. Intell.*, 153(1-2):105–140, 2004.
2. H. Aldewereld. Autonomy vs. conformity: An institutional perspective on norms and protocols. *Knowledge Eng. Review*, 24(4):410–411, 2009.
3. A. Artikis, M. J. Sergot, and J. V. Pitt. Specifying norm-governed computational societies. *ACM Trans. Comput. Log*, 10(1), 2009.
4. R. Craven and M. J. Sergot. Agent strands in the action language nC+. *J. Applied Logic*, 6(2):172–191, 2008.
5. D. Garcia-Gasulla, J. C. Nieves, and U. Cortés. Reasoning about actions for the management of urban wastewater systems using a causal logic. In *In 2010 International Congress on Environmental Modelling and Software Modelling for Environment's Sake, 5th Biennal Meeting, Ottawa, Canada.*, 2010.
6. E. Giunchiglia, J. Lee, V. Lifschitz, N. McCain, and H. Turner. Nonmonotonic causal theories. *Artif. Intell.*, 153(1-2):49–104, 2004.
7. J. C. Nieves, D. Garcia-Gasulla, M. Aulinas, and U. Cortés. Using situation calculus for normative agents in urban wastewater systems. In *In PAAMS'2010*, volume 70 of *Advances in Intelligent And Soft Computing*, pages 247–257, Salamanca, Spain, April 2010. Springer.
8. N. Oren, S. Panagiotidi, J. Vázquez-Salceda, S. Modgil, M. Luck, and S. Miles. Towards a formalisation of electronic contracting environments. In J. F. Hübner, E. T. Matson, O. Boissier, and V. Dignum, editors, *COIN@AAMAS&AAAI*, volume 5428 of *Lecture Notes in Computer Science*, pages 156–171. Springer, 2008.
9. J. Vázquez-Salceda. *Role of Norms and Electronic Institutions in Multi-Agent Systems: The Harmonia Framework.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2004.