Legal Knowledge and Information Systems
E. Francesconi et al. (Eds.)

doi:10.3233/FAIA220464

Fundamental Revisions on Constraint Hierarchies for Ethical Norms

Wachara FUNGWACHARAKORN [a] and Kanae TSUSHIMA [a] and Ken SATOH [a]
[a] *National Institute of Informatics, Sokendai University, Tokyo, Japan*

Abstract. This paper studies constraint hierarchies for *ethical norms*, which are unwritten and may be relaxed if they conflict with stronger norms. Since such ethical norms are unwritten, initial representations of ethical norms may contain errors. For correcting those errors, this paper examines fundamental revisions on constraint hierarchies for ethical norms. Although some revisions on representations for ethical norms have been suggested, revisions on constraint hierarchies for ethical norms have not been completely investigated. In this paper, we categorize two fundamental types of revisions on such constraint hierarchies, namely preference revision and content revision. We also compare effects of those revisions in the criteria of syntactic and semantic changes, which are common criteria of revisions on legal theories. From the comparison, we found that preference revision tentatively makes lower syntactic changes. However, its computation is intractable, incomplete, and potentially makes a large number of semantic changes. On the other hand, we show that content revision on constraint hierarchies can make a small number of semantic changes. However, the content revision tentatively produce a large number of syntactic changes. This comparison leads to the possibility of optimization between preference revision and content revision, which we think is an interesting future work.

Keywords. constraint hierarchies, soft constraints, belief revision

1. Introduction

This paper studies representing social norms using constraint hierarchies [1]. Constraint hierarchies consist of two types of constraints: *hard constraints* (or *required constraints*) and *soft constraints* (or *non-required constraints*). This paper takes a simplified distinction as a starting point of analysis of ethical norms along the research of [2,3,4] by distinguishing social norms into two types: *legal norms* represented as *hard constraints* and *ethical norms* represented as *soft constraints*.

Since ethical norms are unwritten, it is infeasible to represent ethical norms correctly in the first place. Therefore, this paper explores common revisions for correcting errors in representations of ethical norms. One common revision used in such representations is preference revision. Preference revision has an advantage as it does not change the content of the representations. However, it suffers from intractability of computations, incompleteness, and possibility of huge ef-

fects from the revision [5]. In contrast to preference revision, we define content revision, which can make a small number of semantic changes. However, content revision suffers from making large syntactic changes. Then, we demonstrate these behaviors in constraint hierarchies and present *DF-contraction* and *DF-expansion* computations for content revision, which are complete and make a small number of semantic changes. After that, we discuss the possibility of optimization between preference revision and content revision.

2. Constraint Hierarchies and Fundamental Revisions

In this paper, we restrict variables to Boolean variables or propositions. A valuation is a mapping θ from the set of all variables to {TRUE,FALSE} and we write Θ as the set of all valuations. Constraints in this paper are hence restricted to propositional logical formulae. For a constraint c, we define an error function $e(c, \theta)$ which returns zero iff θ satisfies c ($c\theta$ holds), and returns one iff θ does not satisfy c. A constraint hierarchy is a set of constraints with strength levels, i.e. $[k]c$ where k is a non-negative real number representing a level of constraint and c is a constraint. Conventionally, constraints with strength level 0 are called *hard constraints* or *required constraints* and other constraints are called *soft constraints* or *required constraints*, where a larger value of the level means the strength is weaker. The level is typically a non-negative integer but, in our problem, we expand the domain of strength into non-negative real numbers so that we can change the level more flexibly. However, a real-number level can still be treated as an integer in practice.

Let H be a constraint hierarchy. We write H_k be the set of all constraints in H with strength level k. We follow the definitions in [1] by defining:

$$\begin{aligned} S_0 &= \{\theta \in \Theta |\ \forall c \in H_0, c\theta\ holds\} \\ S &= \{\sigma \in S_0 |\ \forall \theta \in S_0, \neg better(\theta, \sigma, H)\} \end{aligned} \tag{1}$$

Intuitively, S_0 is a set of valuations that satisfy all constraints in the set of hard constraints H_0, and S is a valuation in S_0 that satisfies soft constraints as much as possible. A member of S is thus called a solution to H and S is called a solution set. We say H is trivial if $S = S_0$ otherwise we say it is non-trivial. *better* in (1) is an arbitrary comparator between two valuations θ and σ with respect to constraints H. In our problem, we choose the most basic one called *locally-better*. θ is *locally-better* than σ if there exists a strength $k > 0$ such that (i) for every strength stronger than k, the error after applying θ is equal to that after applying σ, i.e. $\forall i \in (0, k)\ \forall p \in H_i\ e(p, \theta) = e(p, \sigma)$, and (ii) at strength k, the error is strictly less than at least one constraint and less than or equal for all the rest, i.e. $\exists q \in H_k\ e(q, \theta) < e(q, \sigma) \wedge \forall r \in H_k\ e(r, \theta) \leq e(r, \sigma)$.

When the solution is unexpected, it means that the solution set is different from the intended set in the user's mind. In concept learning [6], a user works as a membership query that can answer correctly whether or not a queried valuation is intended. The revision task is to find a new constraint hierarchy H' such that a set of solutions to H' satisfies the results of the membership query, i.e. an intended

valuation must be a solution to H' and an unintended valuation must not be a solution to H'. Furthermore, the revision should change from H to H' as little as possible. This criterion is often called *minimal revision*. There are basically two metrics for counting the number of changes: one is based on syntax and another is based on semantics. Our syntax-based metric is adapted from Theory Distance Metric [7]. The metric is defined as follows.

Definition 1 (Syntax-based Metric)**.** *Let H, H_r, and H' be constraint hierarchies. A revision transformation r is such that $r(H) = H_r$, and H_r is obtained from H by edit operations as follows:*

1. *creating a new constraint with a strength level and one literal*
2. *adding one literal using $\vee$ (a logical or) or $\wedge$ (a logical and) to a constraint in H (adding parentheses may be needed to reduce ambiguity but it does not count as an operation)*
3. *removing one literal from a constraint in H (a constraint is deleted if the constraint has no literal left)*
4. *changing a strength of constraint (Operation 4 is specific to constraint hierarchies.)*

The syntactic changes between H and H' are determined by the smallest number of applying the revision transformation r to revise H into H', i.e. $H' = r^n(H)$ and there is no $m < n$ such that $H' = r^m(H)$.

On the other hand, our semantics-based metric is based on the change of the solutions. The metric is defined as follows.

Definition 2 (Semantics-based Metric)**.** *Let H and H' be constraint hierarchies. Let S and S' be the set of solutions to H and H' respectively. The semantic changes between H and H' are determined by the size of symmetric difference S and S', i.e. the number of valuations that belong to only one set.*

Next, we explore two fundamental types of revisions, namely preference revision and content revision. Preference revision refers to a revision that changes only the strengths of constraints but not their contents. In other words, it uses only Operation 4 in Definition 1. Changing a strength of constraint (Operation 4 in Definition 1) can be considered as an operation with the lowest cost for syntactic changes because the content of the constraints is still kept. Hence, preference revision is often considered to be the lowest cost for syntactic changes also. However, preference revision also has some limitations [5]:

1. (Intractable) Although an evaluation table is given, finding how to change a constraint to satisfy the user's intention is *NP-complete* since we need to use *better* to compare the target solution to other valuations.
2. (Incomplete) Only changing a strength of constraint cannot change the solutions in some constraint hierarchies. An obvious example is a constraint hierarchy with only one constraint.
3. (Huge Effect) Although preference revision often makes a small number of syntactic changes, it sometimes makes a large number of semantic changes.

Let us demonstrate Limitation 3 using a constraint hierarchy representing a classic ethical example of the Righteous Lies Problem.

Example 1 (Righteous Lies Problem)**.** *Suppose there is a situation where we need to tell lies to protect others and the only way to protect others is to tell lies (tell_lies* $\leftrightarrow$ *protect). From an ethical point of view, we should protect others, we should not tell lies, and it is common to prioritize protecting others over not telling lies. However, we later realize that protecting criminals could be a case of protecting others (prot_criminal* $\rightarrow$ *protect). Hence, we can represent the current setting as the following constraint hierarchy as shown on the left. For ease of exposition, we write hard, strong, weak instead of* $0, 1, 2$ *respectively.*

$[hard]\ tell\_lies \leftrightarrow protect$
$[hard]\ prot\_criminal \rightarrow protect$
$[strong]\ protect$
$[weak]\ \ \neg tell\_lies$

$[hard]\ tell\_lies \leftrightarrow protect$
$[hard]\ prot\_criminal \rightarrow protect$
$[strong]\ \neg tell\_lies$
$[weak]\ \ protect$

Table 1. Effect of preference revision on solutions to the Righteous Lies Problem

tell_lies	*protect*	*prot_criminal*	Old Solution ?	Intended ?	New Solution ?
TRUE	TRUE	TRUE	yes	no	no
TRUE	TRUE	FALSE	yes	-	no
FALSE	FALSE	FALSE	no	-	yes

Table 1 shows the effect of preference revision on solutions to the Righteous Lies Problem. The old solutions indicate that we should tell lies to protect others, regardless whether they are criminals. Then, we may not intend to protect criminals otherwise it causes more losses to society. However, the new solution from the preference revision as shown on the right gives unexpected results as it re-prioritizes not telling lies over protecting others. From a logical point of view, this revision is unexpected because it makes too many semantic changes.

In contrast to preference revision, let us define content revision as a revision that changes only the content of the constraints but not their strengths. In other words, it uses only Operation 1-3 in Definition 1. Following the definition, we can define two computations for content revision on non-trivial constraint hierarchies. The first computation is *DF-contraction*, for contracting the set of solutions to exclude an unintended solution. The second computation is *DF-expansion*, for expanding the set of solutions to include an intended valuation that satisfies all hard constraints but not yet a solution. These computations can always revise constraint hierarchies with only one semantic change as the following theorems.

Theorem 1. *Given a non-trivial constraint hierarchy* H*. Let* S *be the set of solutions to* H*,* $\theta \in S$*, clause*(θ) *be a conjunctive clause corresponding to an valuation* θ *(e.g. clause*$(a = TRUE, b = FALSE) = a \wedge \neg b$*).* DF-contraction *revises* H *into* H' *as follows.*

for every soft constraint $c \in H$ *such that* $c\theta$ *holds*

remove clause(θ) *from* c *in disjunctive form*

If H' is non-trivial, then DF-contraction *is correct (i.e. θ is not a solution to H'), complete (i.e. can always find H'), makes only one semantic change (i.e. θ is the only member of the symmetric difference between S and S').*

Theorem 2. *Given a non-trivial constraint hierarchy H. Let S be the set of solutions to H, $\theta \in S_0 \setminus S$ where S_0 is the set of all valuations that satisfy all hard constraints, as defined in (1) , clause(θ) be a conjunctive clause corresponding to an valuation θ (e.g. clause$(a = TRUE, b = FALSE) = a \wedge \neg b$).* DF-expansion *revises H into H' as follows.*

for every soft constraint $c \in H$ such that $c\theta$ does not hold

add clause(θ) with $\vee$ (a logical or) to c

DF-expansion *is correct (i.e. θ is a solution to H'), complete (i.e. can always find H'), and makes only one semantic change (i.e. θ is the only member of the symmetric difference between S and the set S' of solutions to H').*

Let us illustrate *DF-contraction* in the Righteous Lies Problem example (Example 1), we can revise the constraint by removing ($tell\_lies \wedge protect \wedge prot\_criminal$) from *protect*, which can be considered in disjunctive form as follows.

$(tell\_lies \wedge protect \wedge prot\_criminal) \vee (\neg tell\_lies \wedge protect \wedge prot\_criminal) \vee$
$(tell\_lies \wedge protect \wedge \neg prot\_criminal) \vee (\neg tell\_lies \wedge protect \wedge \neg prot\_criminal)$

Hence, *protect* is revised into

$(\neg tell\_lies \wedge protect \wedge prot\_criminal) \vee$
$(tell\_lies \wedge protect \wedge \neg prot\_criminal) \vee (\neg tell\_lies \wedge protect \wedge \neg prot\_criminal)$

or $protect \wedge (\neg tell\_lies \vee \neg prot\_criminal)$ in minimal form. As a result, *DF-contraction* gives the following constraint hierarchy.

$[hard]$ $tell\_lies \leftrightarrow protect$
$[hard]$ $prot\_criminal \rightarrow protect$
$[strong]$ $protect \wedge (\neg tell\_lies \vee \neg prot\_criminal)$
$[weak]$ $\neg tell\_lies$

Table 2. New solutions to the content revision in the Righteous Lies Problem example

tell_lies	*protect*	*prot_criminal*	Old Solution ?	Intended ?	New Solution ?
TRUE	TRUE	TRUE	yes	no	no
TRUE	TRUE	FALSE	yes	-	yes
FALSE	FALSE	FALSE	no	-	no

Table 2 shows the new solution to this revision, which makes only one semantic change. However, the revision makes three syntactic changes. One reason for the drawback of content revision is that it does not consider interactions between constraints, as opposed to preference revision. In future work, it is interesting to propose an optimization between preference revision and content revision to adjust between syntactic changes and semantic changes from both types of revision.

3. Conclusion

This paper presents constraint hierarchies for representing ethical norms, which refer to norms that can be conflicted hence they could not be all satisfied sometimes. Such norms fit well with soft constraints in constraint hierarchies as they divide constraints into hard constraints, which must be all satisfied, and soft constraints, which should be satisfied as much as possible. A solution to a constraint hierarchy is intuitively a valuation that satisfies all the hard constraints and no other valuations that satisfy a larger set of soft constraints. This paper also investigates two fundamental types of revision on constraint hierarchies to cover the changes of ethical norms. The first type is preference revision, which changes only the strengths of the constraints but not their contents. Hence, it benefits from making fewer changes in the syntactic sense. However, it cannot always change the solutions as intended and can make a large number of changes in the semantic sense. The second type is content revision, which changes only the contents of constraints but not their strengths. We introduce *DF-contraction* and *DF-expansion* computations for content revisions on constraint hierarchies. They can always change the solutions as intended with only one semantic change, but they mostly make a large number of syntactic changes. Hence, an optimization between preference revision and content revision is an interesting future work.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers, JP17H06103 and JP19H05470 and JST, AIP Trilateral AI Research, Grant Number JPMJCR20G4, Japan.

References

[1] A. Borning, B. Freeman-Benson and M. Wilson, Constraint hierarchies, *LISP and symbolic computation* **5**(3) (1992), 223–270.
[2] J. Greene, F. Rossi, J. Tasioulas, K.B. Venable and B. Williams, Embedding ethical principles in collective decision support systems, in: *Thirtieth AAAI conference on artificial intelligence*, 2016.
[3] L. Dennis, M. Fisher, M. Slavkovik and M. Webster, Formal verification of ethical choices in autonomous systems, *Robotics and Autonomous Systems* **77** (2016), 1–14.
[4] K. Satoh, J.-G. Ganascia, G. Bourgne and A. Paschke, Overview of RECOMP project, in: *International Workshop on Computational Machine Ethics, International Conference on Principles of Knowledge Representation and Reasoning*, 2021.
[5] G. Governatori, F. Olivieri, S. Scannapieco and M. Cristani, Superiority based revision of defeasible theories, in: *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, Springer, 2010, pp. 104–118.
[6] D. Angluin, Queries and concept learning, *Machine learning* **2**(4) (1988), 319–342.
[7] J. Wogulis and M.J. Pazzani, A methodology for evaluating theory revision systems: Results with Audrey II, in: *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, Morgan Kaufmann, San Francisco,CA,USA, 1993, pp. 1128–1134.