Combining Non-monotonicity with Modalities and Substructural Operators

Guido Governatori

13 November 2019

TICAMORE Invited speaker rules

Rules for TICAMORE Invited Speakers

- Guido should not tell lies in his presentation
- If Guido tells a lie then he has to explain why
- It ought to be the case that if Guido does not tell a lie then he does not explain why

TICAMORE Invited speaker rules

Rules for TICAMORE Invited Speakers

- Guido should not tell lies in his presentation
- If Guido tells a lie then he has to explain why
- It ought to be the case that if Guido does not tell a lie then he does not explain why
- Guido tells lies in his presentation

TICAMORE Invited speaker rules

Rules for TICAMORE Invited Speakers

- Guido should not tell lies in his presentation
- If Guido tells a lie then he has to explain why
- It ought to be the case that if Guido does not tell a lie then he does not explain why
- Guido tells lies in his presentation

- $\mathrm{OBL}\,\neg lie$
- $lie \rightarrow \mathrm{OBL}\, explain$
- $\mathrm{OBL}(\neg lie \rightarrow \neg explain)$
- lie

TICAMORE Invited speaker rules

Rules for TICAMORE Invited Speakers

- Guido should not tell lies in his presentation
- If Guido tells a lie then he has to explain why
- It ought to be the case that if Guido does not tell a lie then he does not explain why
- Guido tells lies in his presentation

- $\mathrm{OBL}\,\neg \mathit{lie}$
- $\mathit{lie} \rightarrow \mathrm{OBL}\,\mathit{explain}$
- $\mathrm{OBL}(\neg \mathit{lie} \rightarrow \neg \mathit{explain})$
- lie

$\mathrm{OBL}\,\mathit{explain}$ and $\mathrm{OBL}\,\neg \mathit{explain}$

A Legal Example

License for the evaluation of a product

1. The Licensor grants the Licensee a license to evaluate the Product.
2. The Licensee must not publish the results of the evaluation of the Product without the approval of the Licensor; the approval must be obtained before the publication. If the Licensee publishes results of the evaluation of the Product without approval from the Licensor, the Licensee has 24 hours to remove the material.
3. The Licensee must not publish comments on the evaluation of the Product, unless the Licensee is permitted to publish the results of the evaluation.
4. If the Licensee is commissioned to perform an independent evaluation of the Product, then the Licensee has the obligation to publish the evaluation results.
5. This license terminates automatically if the Licensee breaches this Agreement.

Background: Normative Systems and Defeasibility

Key components of Normative Systems

A normative system is a set of clauses (norms).
Norms are modelled as **if** . . . **then** rules

$$A_1, \ldots, A_n \Rightarrow C$$

- Definitional clauses (constitutive rules: defining terms used in a legal context)
- Prescriptive clauses (norms defining "normative effects")
  - obligations
  - permissions
  - prohibitions
  - violations

Key components of Normative Systems

A normative system is a set of clauses (norms).
Norms are modelled as **if** ... **then** rules

$$A_1, \ldots, A_n \Rightarrow C$$

- Definitional clauses (constitutive rules: defining terms used in a legal context)
- Prescriptive clauses (norms defining "normative effects")
  - obligations
  - permissions
  - prohibitions
  - violations

Norms are defeasible (handling exceptions)

Defeasibility: Reasonable results with minimum effort

Defeasibility: Reasonable results with minimum effort

Factual omniscience and (non-)monotonic reasoning

$$PhD \to Uni$$

Defeasibility: Reasonable results with minimum effort

Factual omniscience and (non-)monotonic reasoning

$$\begin{aligned} \mathit{PhD} &\rightarrow \mathit{Uni} \\ \mathit{Weekend} &\rightarrow \neg \mathit{Uni} \\ \mathit{PublicHoliday} &\rightarrow \neg \mathit{Uni} \\ \mathit{Sick} &\rightarrow \neg \mathit{Uni} \end{aligned}$$

Defeasibility: Reasonable results with minimum effort

Factual omniscience and (non-)monotonic reasoning

$$PhD \to Uni$$
$$Weekend \to \neg Uni$$
$$PublicHoliday \to \neg Uni$$
$$Sick \to \neg Uni$$
$$Weekend \wedge VICdeadline \to Uni$$

Defeasibility: Reasonable results with minimum effort

Factual omniscience and (non-)monotonic reasoning

$$
\begin{aligned}
\mathit{PhD} &\rightarrow \mathit{Uni}\\
\mathit{Weekend} &\rightarrow \neg \mathit{Uni}\\
\mathit{PublicHoliday} &\rightarrow \neg \mathit{Uni}\\
\mathit{Sick} &\rightarrow \neg \mathit{Uni}\\
\mathit{Weekend} \wedge \mathit{VICdeadline} &\rightarrow \mathit{Uni}
\end{aligned}
$$

VIC= Very Important Conference

Defeasibility: Reasonable results with minimum effort

Factual omniscience and (non-)monotonic reasoning

$$
\begin{aligned}
\mathit{PhD} &\rightarrow \mathit{Uni} \\
\mathit{Weekend} &\rightarrow \neg \mathit{Uni} \\
\mathit{PublicHoliday} &\rightarrow \neg \mathit{Uni} \\
\mathit{Sick} &\rightarrow \neg \mathit{Uni} \\
\mathit{Weekend} \wedge \mathit{VICdeadline} &\rightarrow \mathit{Uni} \\
\mathit{VICdeadline} \wedge \mathit{PartnerBirthday} &\rightarrow \neg \mathit{Uni}
\end{aligned}
$$

Defeasibility: Reasonable results with minimum effort

Factual omniscience and (non-)monotonic reasoning

$$\begin{aligned} PhD &\rightarrow Uni \\ Weekend &\rightarrow \neg Uni \\ PublicHoliday &\rightarrow \neg Uni \\ Sick &\rightarrow \neg Uni \\ Weekend \wedge VICdeadline &\rightarrow Uni \\ VICdeadline \wedge PartnerBirthday &\rightarrow \neg Uni \end{aligned}$$

$$Phd \wedge (\neg Weekend \vee (Weekend \wedge VICdeadline \wedge \neg PartnerBirthday)) \wedge \neg Sick \ldots \rightarrow Uni$$

Defeasible Logic

Why Defeasible Deontic Logic

Rule-based non-monotonic formalism

- Flexible
- Efficient (linear complexity)
- Directly skeptic semantics
- Argumentation semantics
- Constrictive proof theory
- Encompasses other formalisms used in AI and Law
- Applied in several fields/optimised implementations
- Extensible
- Not affected by Deontic Logic Paradoxes

Why Defeasible Deontic Logic

Rule-based non-monotonic formalism

- Flexible
- Efficient (linear complexity)
- Directly skeptic semantics
- Argumentation semantics
- Constructive proof theory
- Encompasses other formalisms used in AI and Law
- Applied in several fields/optimised implementations
- Extensible
- Not affected by Deontic Logic Paradoxes

Defeasible Logic Blueprint

Combination of an efficient non-monotonic logic (defeasible logic) and a deontic logic of violation.

- Derive (plausible) conclusions with the minimum amount of information.
  - Definite conclusions
  - Defeasible conclusions
- Defeasible Theory
  - Facts
  - Strict rules ($A_1, \ldots, A_n \rightarrow B$)
  - Defeasible rules ($A_1, \ldots, A_n \Rightarrow B$)
  - Defeaters ($A_1, \ldots, A_n \leadsto B$)
  - Superiority relation over rules

Reasoning with Defeasible Logic

- Positive defeasible conclusions: meaning that the conclusions can be defeasible proved;
- Negative defeasible conclusions: meaning that one can show that the conclusion is not even defeasibly provable.

Conclusions in Defeasible Logic

A proof is a finite sequence $P = (P(1), \ldots, P(n))$ of tagged literals satisfying four conditions

Conclusions in Defeasible Logic

A proof is a finite sequence $P = (P(1), \ldots, P(n))$ of tagged literals satisfying four conditions

- $+\Delta q$, which is intended to mean that q is definitely provable (i.e., using only facts and strict rules);

Conclusions in Defeasible Logic

A proof is a finite sequence $P = (P(1), \ldots, P(n))$ of tagged literals satisfying four conditions

- $+\Delta q$, which is intended to mean that q is definitely provable (i.e., using only facts and strict rules);
- $-\Delta q$, which is intended to mean that we have proved that q is not definitely provable in D;

Conclusions in Defeasible Logic

A proof is a finite sequence $P = (P(1), \ldots, P(n))$ of tagged literals satisfying four conditions

- $+\Delta q$, which is intended to mean that q is definitely provable (i.e., using only facts and strict rules);
- $-\Delta q$, which is intended to mean that we have proved that q is not definitely provable in D;
- $+\partial q$, which is intended to mean that q is defeasibly provable in D;

Conclusions in Defeasible Logic

A proof is a finite sequence $P = (P(1), \ldots, P(n))$ of tagged literals satisfying four conditions

- $+\Delta q$, which is intended to mean that q is definitely provable (i.e., using only facts and strict rules);
- $-\Delta q$, which is intended to mean that we have proved that q is not definitely provable in D;
- $+\partial q$, which is intended to mean that q is defeasibly provable in D;
- $-\partial q$ which is intended to mean that we have proved that q is not defeasibly provable in D.

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove
2. Consider all possible counterarguments to it

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove
2. Consider all possible counterarguments to it
3. Rebut all counterarguments

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove
2. Consider all possible counterarguments to it
3. Rebut all counterarguments
   - Defeat the argument by a stronger one
   - Undercut the argument by showing that some of the premises do not hold

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove
2. Consider all possible counterarguments to it
3. Rebut all counterarguments
   - Defeat the argument by a stronger one
   - Undercut the argument by showing that some of the premises do not hold

1. A is a fact; or

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove
2. Consider all possible counterarguments to it
3. Rebut all counterarguments
   - Defeat the argument by a stronger one
   - Undercut the argument by showing that some of the premises do not hold

1. A is a fact; or
2. there is an applicable rule for A, and either

Proving Conclusions in Defeasible Logic

1. Give an argument for the conclusion you want to prove
2. Consider all possible counterarguments to it
3. Rebut all counterarguments
   - Defeat the argument by a stronger one
   - Undercut the argument by showing that some of the premises do not hold

1. A is a fact; or
2. there is an applicable rule for A, and either
   1. all the rules for $\neg A$ are discarded (i.e., not applicable) or
   2. every applicable rule for $\neg A$ is weaker than an applicable rule for A.

. . . formally

$+\partial$: If $P(n+1) = +\partial q$ then

1) $+\Delta q \in P(1..n)$, or

2) $-\Delta{\sim}q \in P(1..n)$ and

2.1) $\exists r \in R_{sd}[q]$: $\forall a \in A(r) + \partial a \in P(1..n)$ and

2.2) $\forall s \in R[{\sim}q]$ either $\exists a \in A(s) : -\partial a \in P(1..n)$ or

$\exists t \in R[q]$: $\forall a \in A(t) + \partial a \in P(1..n)$ and $t \succ s$.

Example

Facts: $A_1,\ A_2,\ B_1,\ B_2$

Rules: $r_1 : A_1 \Rightarrow C$

$r_2 : A_2 \Rightarrow C$

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Superiority relation:

$r_1 > r_3$

$r_2 > r_4$

$r_5 > r_1$

Example

Facts: A_1, A_2, B_1, B_2

Rules: $r_1 : A_1 \Rightarrow C$

$r_2 : A_2 \Rightarrow C$

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Superiority relation:

$r_1 > r_3$

$r_2 > r_4$

$r_5 > r_1$

Phase 1: Argument for C

Example

Facts: A_1, A_2, B_1, B_2

Rules: $r_1 : A_1 \Rightarrow C$

$r_2 : A_2 \Rightarrow C$

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Superiority relation:

$r_1 > r_3$

$r_2 > r_4$

$r_5 > r_1$

Phase 1: Argument for C

A_1 (Fact), $r_1 : A_1 \Rightarrow C$

Example

Facts: A_1, A_2, B_1, B_2

Rules:
$$r_1 : A_1 \Rightarrow C$$
$$r_2 : A_2 \Rightarrow C$$
$$r_3 : B_1 \Rightarrow \neg C$$
$$r_4 : B_2 \Rightarrow \neg C$$
$$r_5 : B_3 \Rightarrow \neg C$$

Superiority relation:
$$r_1 > r_3$$
$$r_2 > r_4$$
$$r_5 > r_1$$

Phase 1: Argument for C

A_1 (Fact), $r_1 : A_1 \Rightarrow C$

Phase 2: Possible counterarguments

Example

Facts: A_1, A_2, B_1, B_2

Rules: $r_1 : A_1 \Rightarrow C$

$r_2 : A_2 \Rightarrow C$

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Superiority relation:

$r_1 > r_3$

$r_2 > r_4$

$r_5 > r_1$

Phase 1: Argument for C

A_1 (Fact), $r_1 : A_1 \Rightarrow C$

Phase 2: Possible counterarguments

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Example

Facts: A_1, A_2, B_1, B_2

Rules: $r_1 : A_1 \Rightarrow C$

$r_2 : A_2 \Rightarrow C$

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Superiority relation:

$r_1 > r_3$

$r_2 > r_4$

$r_5 > r_1$

Phase 1: Argument for C

A_1 (Fact), $r_1 : A_1 \Rightarrow C$

Phase 2: Possible counterarguments

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Phase 3: Rebut the counterarguments

Example

Facts: A_1, A_2, B_1, B_2

Rules: $r_1 : A_1 \Rightarrow C$

$r_2 : A_2 \Rightarrow C$

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Superiority relation:

$r_1 > r_3$

$r_2 > r_4$

$r_5 > r_1$

Phase 1: Argument for C

A_1 (Fact), $r_1 : A_1 \Rightarrow C$

Phase 2: Possible counterarguments

$r_3 : B_1 \Rightarrow \neg C$

$r_4 : B_2 \Rightarrow \neg C$

$r_5 : B_3 \Rightarrow \neg C$

Phase 3: Rebut the counterarguments

r_3 weaker than r_1

r_4 weaker than r_2

r_5 is not applicable

Defeasible Deontic Logic

Extending the language of Defeasible Logic

- extend the language with the deontic operators OBL, PERM, PERM_W and PERM_S.
- extend the language with the reparation operator $\otimes$. Permitted only in the head/conclusion of rules.
- two classes of rules:
  - constitutive: $A_1, \ldots, A_n \hookrightarrow B$
  - prescriptive: $A_1, \ldots, A_n \hookrightarrow_{\mathrm{OBL}} B$

Extending the proof mechanism of Defeasible Logic

- To prove $\mathsf{OBL}\, p_n$ from a rule

$$A \Rightarrow_{\mathsf{OBL}} p_1 \otimes \cdots \otimes p_{n-1} \otimes p_n$$

  we have to show that $\mathsf{OBL}\, p_1, \ldots, \mathsf{OBL}\, p_n$ and $\neg p_1, \ldots, \neg p_{n-1}$ are provable.
- To disprove $\mathsf{OBL}\, p_n$ from a rule

$$A \Rightarrow_{\mathsf{OBL}} p_1 \otimes \cdots \otimes p_{n-1} \otimes p_n$$

  that at least one among $\mathsf{OBL}\, p_1, \ldots, \mathsf{OBL}\, p_n$ and $p_1, \ldots, p_{n_1}$ is rejected.

Conclusion and Deontic Operators

- $\mathsf{OBL}\, p$ is proved iff $+\partial_{\mathsf{OBL}}\, p$ is proved
- $\mathsf{OBL}\, p$ is rejected iff $-\partial_{\mathsf{OBL}}\, p$ is proved
- $\mathsf{PERM}_w\, p$ is proved iff $-\partial_{\mathsf{OBL}} {\sim} p$ is proved
- $\mathsf{PERM}_w\, p$ is rejected iff $+\partial_{\mathsf{OBL}} {\sim} p$ is proved

Modelling Permissions

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

Modelling Permissions

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

General prohibition to Uturn:

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

Permission to Uturn if Uturn permitted sign:

$$arr_{40e}: \mathit{UturnPermittedSign} \hookrightarrow \mathit{Uturn}.$$

Modelling Permissions

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

General prohibition to Uturn:

$$arr_{40a} : AtTrafficLigths \Rightarrow_{\mathrm{OBL}} \neg Uturn.$$

Permission to Uturn if Uturn permitted sign:

$$arr_{40e} : UturnPermittedSign \Rightarrow_{\mathrm{OBL}} Uturn.$$

Modelling Permissions

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

General prohibition to Uturn:

$$arr_{40a} : AtTrafficLigths \Rightarrow_{\mathrm{OBL}} \neg Uturn.$$

Permission to Uturn if Uturn permitted sign:

$$arr_{40e} : UturnPermittedSign \rightsquigarrow_{\mathrm{OBL}} Uturn.$$

From definitions to obligations

Can we conclude OBL B

$$A_1, \ldots, A_n \Rightarrow B$$

and

$$\mathrm{OBL}\, A_1, \ldots, \mathrm{OBL}\, A_n$$

What is a breach of the license?

License for the evaluation of a product

1. The Licensor grants the Licensee a license to evaluate the Product.
2. The Licensee must not publish the results of the evaluation of the Product without the approval of the Licensor; the approval must be obtained before the publication. If the Licensee publishes results of the evaluation of the Product without approval from the Licensor, the Licensee has 24 hours to remove the material.
3. The Licensee must not publish comments on the evaluation of the Product, unless the Licensee is permitted to publish the results of the evaluation.
4. If the Licensee is commissioned to perform an independent evaluation of the Product, then the Licensee has the obligation to publish the evaluation results.
5. This license terminates automatically if the Licensee breaches this Agreement.

Uncompensable Violation

$$\Rightarrow_{\mathsf{OBL}} A \otimes B \otimes C$$

and $\neg A$, $\neg B$, $\neg C$

DDL formally

A rule $r \in R[q,j]$ is *body-applicable* iff for all $a_i \in A(r)$:

1. if $a_i = \Box l$ then $+\partial_\Box l \in P(1..n)$ with $\Box \in \{\mathrm{OBL}, \mathrm{PERM}, \mathrm{PERM}_w, \mathrm{PERM}_s\}$;
2. if $a_i = \neg\Box l$ then $-\partial_\Box l \in P(1..n)$ with $\Box \in \{\mathrm{OBL}, \mathrm{PERM}, \mathrm{PERM}_w, \mathrm{PERM}_s\}$;
3. if $a_i = l \in Lit$ then $+\partial l \in P(1..n)$.

A rule $r \in R[q,j]$ is *body-discarded* iff $\exists a_i \in A(r)$ such that

1. if $a_i = \Box l$ then $-\partial_\Box l \in P(1..n)$ with $\Box \in \{\mathrm{OBL}, \mathrm{PERM}, \mathrm{PERM}_w, \mathrm{PERM}_s\}$;
2. if $a_i = \neg\Box l$ then $+\partial_\Box l \in P(1..n)$ with $\Box \in \{\mathrm{OBL}, \mathrm{PERM}, \mathrm{PERM}_w, \mathrm{PERM}_s\}$;
3. if $a_i = l \in Lit$ then $-\partial l \in P(1..n)$.

DDL formally

A rule $r \in R$ is *body-p-applicable* iff

1. if $r \in R^{\mathrm{OBL}}$ and it is body-applicable; or
2. if $r \in R^{C}$ and, $A(r) \neq \emptyset$, $A(r) \subseteq PLit$ and $\forall a_i \in A(r)$, $+\partial_{\mathrm{OBL}} a_i \in P(1..n)$.

A rule $r \in R$ is *body-p-discarded* iff

1. if $r \in R^{\mathrm{OBL}}$ and it is not body-applicable; or
2. if $r \in R^{C}$ and either $A(r) = \emptyset$ or $A(r) \cap Lit \neq \emptyset$ and $\exists a_i \in A(r)$, $-\partial_{\mathrm{OBL}} a_i \in P(1..n)$.

DDL formally

A rule $r \in R[q, j]$ such that $C(r) = c_1 \otimes \cdots \otimes c_n$ is *applicable* for literal q at index j, with $1 \leq j < n$, in the condition for $\pm\partial_{\mathrm{OBL}}$ iff

1. r is body-p-applicable; and
2. for all $c_k \in C(r)$, $1 \leq k < j$, $+\partial_{\mathrm{OBL}} c_k \in P(1..n)$ and $(-\partial c_i \in P(1..n)$ or $+\partial{\sim}c_i \in P(1..n))$.

DDL formally $+\partial_{\mathrm{OBL}}$

$+\partial_{\mathrm{OBL}}$: If $P(n+1) = +\partial_{\mathrm{OBL}} q$ then
(1) $\exists r \in R_d^{\mathrm{OBL}}[q,i] \cup R_d^{C}[q]$ such that r is applicable for q, and
(2) $\forall s \in R[\sim q, j]$, either
 (2.1) s is discarded, or
 (2.2) $\exists t \in R[q,k]$ such that t is applicable for q and $s \prec t$

DDL formally $+\partial_{\mathsf{PERM}_s}$

$+\partial_{\mathsf{PERM}_s}$: If $P(n+1) = +\partial_{\mathsf{PERM}_s} q$ then
(1)$+\partial_{\mathsf{OBL}} q \in P(1..n)$ or
(2.1) $\exists r \in R^{\mathsf{OBL}}_{def}[q] \cup R^{C}_{def}[q]$ such that r is body-p-applicable, and
(2.2) $\forall s \in R[\sim q, j]$, either
(2.2.1) s is discarded, or
(2.2.2) $\exists t \in R_{[}q, k]$ such that t is applicable for q and $s \prec t$

DDL formally (other permissions)

$+\partial_{\mathsf{PERM}_w}$: If $P(n+1) = +\partial_{\mathsf{PERM}_w} q$ then
$(1) -\partial_{\mathsf{OBL}} {\sim} q \in P(1..n)$.

$+\partial_{\mathsf{PERM}}$: If $P(n+1) = +\partial_{\mathsf{PERM}} q$ then
$(1) +\partial_{\mathsf{PERM}_s} q \in P(1..n)$ or
$(2) +\partial_{\mathsf{PERM}_w} q \in P(1..n)$.

DDL formally (proving violation

$+\partial_\perp$: If $P(n+1) = +\partial_\perp$ then
(1)$\exists R_d^{\mathrm{OBL}} \cup R_d^C$ such that
(1.1) r is body-p-applicable and
(1.2) $\forall c_i \in C(r)\ +\partial_{\mathrm{OBL}} c_i \in P(1..n)$ and
either $-\partial_C c_i \in P(1..n)$ or $+\partial{\sim}c_i \in P(1..n)$.

Closing the circle

License for the evaluation of a product

1. The Licensor grants the Licensee a license to evaluate the Product.
2. The Licensee must not publish the results of the evaluation of the Product without the approval of the Licensor; the approval must be obtained before the publication. If the Licensee publishes results of the evaluation of the Product without approval from the Licensor, the Licensee has 24 hours to remove the material.
3. The Licensee must not publish comments on the evaluation of the Product, unless the Licensee is permitted to publish the results of the evaluation.
4. If the Licensee is commissioned to perform an independent evaluation of the Product, then the Licensee has the obligation to publish the evaluation results.
5. This license terminates automatically if the Licensee breaches this Agreement.

Making sense of the license

- C0 the use of the product is forbidden;
- C1 if a license is granted, then the use of the product is permitted;
- C2 the publication of the result of the evaluation is forbidden;
- C2c the removal of the illegally published results within the allotted times compensates the illegal publication;
- C2e the publication of the results of the evaluation is permitted if approval is obtained before publication;
- C3 commenting about the evaluation is forbidden;
- C3e commenting about the evaluation is permitted, it publication of the results is permitted;
- C4 publication of the results of evaluation is obligatory, if commissioned for an independent evaluation;
- C4x the use of product is obligatory, if commissioned for an independent evaluation;
- C5 the use of the product is forbidden, if there is a violation of the above conditions.

Formalising the Agreement

$$
\begin{aligned}
r_0 &: \; \Rightarrow_{\mathrm{OBL}} \neg \mathit{use} \\
r_1 &: \mathit{license} \rightsquigarrow_{\mathrm{OBL}} \mathit{use} \\
r_2 &: \; \Rightarrow_{\mathrm{OBL}} \neg \mathit{publish} \otimes \mathit{remove} \\
r_{2e} &: \mathit{approval} \rightsquigarrow_{\mathrm{OBL}} \mathit{publish} \\
r_3 &: \; \Rightarrow_{\mathrm{OBL}} \neg \mathit{comment} \\
r_{3e} &: \mathrm{PERM}\, \mathit{publish} \rightsquigarrow_{\mathrm{OBL}} \mathit{comment} \\
r_4 &: \mathit{commission} \Rightarrow_{\mathrm{OBL}} \mathit{publish} \\
r_{4x} &: \mathit{commission} \Rightarrow_{\mathrm{OBL}} \mathit{use} \\
r_5 &: \bot \Rightarrow_{\mathrm{OBL}} \neg \mathit{use}
\end{aligned}
$$

where $r_0 \prec r_1$, $r_o \prec r_{4x}$, $r_1 \prec r_5$, $r_{4x} \prec r_5$ $r_2 \prec r_{2e}$, $r_3 \prec r_{3e}$.

Pragmatic Oddity

Pragmatic Oddity

- There should be no fence
- If there is a fence, the fence should be white
- There is a fence

The obligations in force are

- There should be no fence
- The fence should be white

Pragmatic Oddity

- There should be no fence
- If there is a fence, the fence should be white
- There is a fence

The obligations in force are

- There should be no fence
- The fence should be white

How to prevent

It should be the case that there is no fence and the fence is white

Solutions

- Simple solution: logic without $(\mathrm{OBL}\, a \wedge \mathrm{OBL}\, b) \rightarrow \mathrm{OBL}(a \wedge b)$

Solutions

- Simple solution: logic without $(\mathrm{OBL}\,a \wedge \mathrm{OBL}\,b) \to \mathrm{OBL}(a \wedge b)$
- What about normative systems with

$$\cdots \to \mathrm{OBL}\,a, \qquad \cdots \to \mathrm{OBL}\,b, \qquad \mathrm{OBL}(a \wedge b) \to \ldots$$

Solutions

- Simple solution: logic without $(\mathrm{OBL}\, a \wedge \mathrm{OBL}\, b) \rightarrow \mathrm{OBL}(a \wedge b)$
- What about normative systems with

$$\cdots \rightarrow \mathrm{OBL}\, a, \qquad \cdots \rightarrow \mathrm{OBL}\, b, \qquad \mathrm{OBL}(a \wedge b) \rightarrow \ldots$$

- Restricted $(\mathrm{OBL}\, a \wedge \mathrm{OBL}\, b) \rightarrow \mathrm{OBL}(a \wedge b)$ to cases where a and b are independent of the violation of each other.

DDL Solution

- extend the language to allow $\mathrm{OBL}(a \wedge b)$ in the antecedent of rules.

$+\partial_{\mathrm{OBL}\wedge}$: If $P(n+1) = +\partial_{\mathrm{OBL}} p \wedge q$ then
(1) $+\partial_{\mathrm{OBL}} p \in P(1..n)$ and
(2) $+\partial_{\mathrm{OBL}} q \in P(1..n)$ and
(3) if $P(k) = +\partial_{\mathrm{OBL}} p$ $(k \leq n)$,then $+\partial{\sim}q \notin P(1..k)$ and
(4) if $P(k) = +\partial_{\mathrm{OBL}} q$ $(k \leq n)$, then $+\partial{\sim}p \notin P(1..k)$.

Example

Rules/Facts

$$r_1 : \cdots \Rightarrow_{\mathrm{OBL}} a \otimes b \qquad r_2 : \cdots \Rightarrow_{\mathrm{OBL}} b \qquad \neg a$$

Derivations	(1)	$+\partial \neg a$	fact
	(2)	$+\partial_{\mathrm{OBL}} a$	from r_1
	(2)	$+\partial_{\mathrm{OBL}} b$	from r_1 and (1) and (2)

(1)	$+\partial_{\mathrm{OBL}} a$	from r_1
(2)	$+\partial_{\mathrm{OBL}} b$	from r_2
(3)	$+\partial \neg a$	fact
(4)	$+\partial_{\mathrm{OBL}} a \wedge b$	from (1) and (2)

Future Work

- complexity and implementation of logic with pragmatic oddity
- Logic for free choice permission
- Logic where $a, b \Rightarrow c$ is different from $b, a \Rightarrow c$
- Logic where $a, a \Rightarrow c$ is different from $a \Rightarrow c$
- complexity, termination and implementation of resource based defeasible logics.

Questions?

Guido Governatori

guido.governatori@data61.com.au