Nested Rules in Defeasible Logic

Insu Song and Guido Governatori

School of Information Technology and Electrical Engineering
The University of Queensland, Australia
{insu,guido}@itee.uq.edu.au

12 November 2005

Look Ahead

- What?
- Why?
- Where?
- How?

What's a nested rule?

```
<rule>
  <head>
    <rule>
      <head>D</head>
      <body>C</body>
    </rule>
  </head>
  <body>
    <rule>
      <head>B</head>
      <body>A</body>
    </rule>
  </body>
</rule>
```

Classical Logic

$$(A \to B) \to (C \to D)$$

Evaluates

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Classical Logic

$$(A \to B) \to (C \to D)$$

Evaluates

$$(A \vee \neg C \vee D) \wedge (\neg B \vee \neg C \vee D)$$

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Conditional Logic

$$(A \mathrel{\Box\!\!\rightarrow} B) \mathrel{\Box\!\!\rightarrow} (C \mathrel{\Box\!\!\rightarrow} D)$$

Evaluates

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Conditional Logic

$$(A \mathrel{\Box\!\!\rightarrow} B) \mathrel{\Box\!\!\rightarrow} (C \mathrel{\Box\!\!\rightarrow} D)$$

Evaluates

$$(A \mathrel{\Box\!\!\rightarrow} B) \mathrel{\Box\!\!\rightarrow} (C \mathrel{\Box\!\!\rightarrow} D)$$

What's a nested rule?

```
<rule>
  <head>
    <rule>
      <head>D</head>
      <body>C</body>
    </rule>
  </head>
  <body>
    <rule>
      <head>B</head>
      <body>A</body>
    </rule>
  </body>
</rule>
```

Logic Programming

```
(D :- C) :- (B :- A)
```

Evaluates

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Logic Programming

`(D :- C) :- (B :- A)`

Evaluates

Syntax Error

What's a nested rule?

```
<rule>
  <head>
    <rule>
      <head>D</head>
      <body>C</body>
    </rule>
  </head>
  <body>
    <rule>
      <head>B</head>
      <body>A</body>
    </rule>
  </body>
</rule>
```

Nonmonotonic Consequence Relation

$$(A \mathrel{|\!\sim} B) \mathrel{|\!\sim} (C \mathrel{|\!\sim} D)$$

Evaluates

What's a nested rule?

```
<rule>
  <head>
    <rule>
      <head>D</head>
      <body>C</body>
    </rule>
  </head>
  <body>
    <rule>
      <head>B</head>
      <body>A</body>
    </rule>
  </body>
</rule>
```

Nonmonotonic Consequence Relation

$$(A \mid\!\sim B) \mid\!\sim (C \mid\!\sim D)$$

Evaluates

Semantic nonsense

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Defeasible Logic

$$(A \mathrel{|\!\sim} B) \mathrel{|\!\sim} (C \mathrel{|\!\sim} D)$$

Evaluates

Semantic nonsense

Gabbay (1995), Artosi, Governatori and Rotolo (2002)

$$(A \mathrel{\Box\!\!\rightarrow} B) \mathrel{|\!\sim} (C \mathrel{\Box\!\!\rightarrow} D)$$

What's a nested rule?

```
<rule>
  <head>
    <rule>
      <head>D</head>
      <body>C</body>
    </rule>
  </head>
  <body>
    <rule>
      <head>B</head>
      <body>A</body>
    </rule>
  </body>
</rule>
```

Defeasible Logic

$$(A \Rightarrow B) \Rightarrow (C \Rightarrow D)$$

Evaluates

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Defeasible Logic

$$(A \Rightarrow B) \Rightarrow (C \Rightarrow D)$$

Evaluates (before today)

Syntax error and semantic nonsense

What's a nested rule?

```
<rule>
  <head>
     <rule>
       <head>D</head>
       <body>C</body>
     </rule>
  </head>
  <body>
     <rule>
       <head>B</head>
       <body>A</body>
     </rule>
  </body>
</rule>
```

Defeasible Logic

$$(A \Rightarrow B) \Rightarrow (C \Rightarrow D)$$

Evaluates (before today)

Syntax error and semantic nonsense

From today

$$(A \Rightarrow B) \vdash_{\Rightarrow} (C \Rightarrow D)$$

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Given $A \Rightarrow (B \Rightarrow C)$ we need A to prove $B \Rightarrow C$

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Given $A \Rightarrow (B \Rightarrow C)$ we need A to prove $B \Rightarrow C$
Given $B \Rightarrow C$ we need B to prove C

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Given $A \Rightarrow (B \Rightarrow C)$ we need A to prove $B \Rightarrow C$
Given $B \Rightarrow C$ we need B to prove C
To prove C from $A \Rightarrow (B \Rightarrow C)$ we need A and B

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Given $A \Rightarrow (B \Rightarrow C)$ we need A to prove $B \Rightarrow C$
Given $B \Rightarrow C$ we need B to prove C
To prove C from $A \Rightarrow (B \Rightarrow C)$ we need A and B
Therefore to prove C, $A \Rightarrow (B \Rightarrow C)$ behaves like $A \wedge B \Rightarrow C$.

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Given $A \Rightarrow (B \Rightarrow C)$ we need A to prove $B \Rightarrow C$
Given $B \Rightarrow C$ we need B to prove C
To prove C from $A \Rightarrow (B \Rightarrow C)$ we need A and B
Therefore to prove C, $A \Rightarrow (B \Rightarrow C)$ behaves like $A \wedge B \Rightarrow C$.
So do we need nested rules?

Why nested rules?

$$A \Rightarrow (B \Rightarrow C)$$

Given $A \Rightarrow (B \Rightarrow C)$ we need A to prove $B \Rightarrow C$
Given $B \Rightarrow C$ we need B to prove C
To prove C from $A \Rightarrow (B \Rightarrow C)$ we need A and B
Therefore to prove C, $A \Rightarrow (B \Rightarrow C)$ behaves like $A \wedge B \Rightarrow C$.
So do we need nested rules?

$$(B \Rightarrow C) \Rightarrow D$$

Confidential

Confidential

A company has the policy that all confidential documents must be encrypted when they are sent by email, and no confidential document can be sent to people outside the company. A document is classified as confidential when its disclosure would harm the interests of the company. Let us suppose we have a document d describing the details of an application for a patent. Here we have that if the document is disclosed before the grant of the patent then the knowledge in it will be classified as public domain, and if something is public domain, other concurrent companies can use the technology described in the document. But if the technology is used by other companies then its usage will generate less revenue than if it were secret and this will harm the interest of the company.

Confidential formalised

- r1: $(Disclose(x) \Rightarrow HarmInterests) \Rightarrow Confidential(x)$
- r2: $Confidential(x) \Rightarrow Encrypt(x)$
- r3: $Disclose(x) \Rightarrow PublicDomain(x)$
- r4: $PublicDomain(x) \Rightarrow FreeUseOf(x)$
- r5: $FreeUseOf(x) \Rightarrow LessRevenue(x)$
- r6: $LessRevenue(x) \Rightarrow HarmInterests$

Nested rules are not classical

$$(A \Rightarrow B) \Rightarrow C) \not\equiv (A, B) \Rightarrow C$$

Nested rules are not classical

$$(A \Rightarrow B) \Rightarrow C) \not\equiv (A, B) \Rightarrow C$$

$$(Disclose(x) \Rightarrow HarmInterests) \Rightarrow Confidential(x)$$

has a different meaning from

$$(Disclose(x) \wedge HarmInterests) \Rightarrow Confidential(x)$$

Nested rules are not classical

$$(A \Rightarrow B) \Rightarrow C \not\equiv (\not A \vee B) \Rightarrow C$$

Nested rules are not classical

$$(A \Rightarrow B) \Rightarrow C \not\equiv (\not{A} \vee B) \Rightarrow C$$

$$(Disclose(x) \Rightarrow HarmInterests) \Rightarrow Confidential(x)$$

has a different meaning from

$$\neg Disclose(x) \Rightarrow Confidential(x)$$
$$HarmInterests) \Rightarrow Confidential(x)$$

Applications of nested rules

- Security
- Classification
- Configuration
- Autonomic computing
- Affective computing
- Normative reasoning
- Legal drafting
- Knowledge base revision

Nested Rules in Defeasible Logic

DL is a rule-based non-monotonic formalism

- Flexible
- Efficient (linear complexity)
- Directly skeptic semantics
- Argumentation semantics
- Constructive proof theory
- Applied in several fields/optimised implementations
- Extensible

Defeasible Logic Theory

$$(F, R, >)$$

- Facts
- Strict rules ($A \rightarrow B$)
- Defeasible rules ($A \Rightarrow B$)
- Defeaters ($A \leadsto B$)
- Superiority relation over rules

Conclusions in Defeasible Logic

- $+\Delta q$, which is intended to mean that the literal q is definitely provable, using only strict rules.
- $-\Delta q$, which is intended to mean that q is provably not definitely provable (finite failure).
- $+\partial q$, 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.

Conclusions in Defeasible Logic

- $+\Delta q$, which is intended to mean that the literal q is definitely provable, using only strict rules.
- $-\Delta q$, which is intended to mean that q is provably not definitely provable (finite failure).
- $+\partial q$, 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.

A derivation is a sequence of tagged expressions where each tagged expression is obtained from the previous ones by some proof conditions (inference rules)

Proofs and Nested rules

- In standard defeasible logic every time we have to use a rule we check whether the rule is given in the set of rules.
- In defeasible logic with nested rules instead we derive rules (and we have to check the strength of the derivation).
- We have 12 proof tags (proof conditions) for rules (4 proof tags $\pm\Delta$ and $\pm\partial$ and 3 types of rules ($\rightarrow$, $\Rightarrow$, $\rightsquigarrow$).

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).
- The rule is directly derivable as it were a normal literal

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).
- The rule is directly derivable as it were a normal literal

$$C \in F \qquad C \Rightarrow (A \hookrightarrow B) \in F$$

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).
- The rule is directly derivable as it were a normal literal

$$C \in F \qquad C \Rightarrow (A \hookrightarrow B) \in F$$

- The rule is a "derived rule".

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).
- The rule is directly derivable as it were a normal literal

$$C \in F \qquad C \Rightarrow (A \hookrightarrow B) \in F$$

- The rule is a "derived rule".
  - $(F, R, >) \not\vdash \#B$ and

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).
- The rule is directly derivable as it were a normal literal

$$C \in F \qquad C \Rightarrow (A \hookrightarrow B) \in F$$

- The rule is a "derived rule".
  - $(F, R, >) \not\vdash \#B$ and
  - $(\emptyset, R \cup \hookrightarrow_{\#} A, >) \vdash \#B$

Proving rules

$$(F, R, >) \vdash +\#A \hookrightarrow B$$

- $A \hookrightarrow B \in R$ (standard condition in Defeasible logic).
- The rule is directly derivable as it were a normal literal

$$C \in F \qquad C \Rightarrow (A \hookrightarrow B) \in F$$

- The rule is a "derived rule".
  - $(F, R, >) \not\vdash \#B$ and
  - $(\emptyset, R \cup \hookrightarrow_{\#} A, >) \vdash \#B$

$$A \Rightarrow C; \quad C \Rightarrow B$$

Future Work

Done

- Increase in the expressive power of Defeasible Logic
- Reasonable / stable proof conditions for 1 level of nested rules
- Negative results (loops)
- Looked at some possible applications

To do

- complete the proof conditions (different variants exists for many proof tags, different intuition of the meaning of the proof tag)
- study under which conditions the negative results can be avoided
- study of the complexity
- deeper study of the logical property of the logic
- implementation (???)

References

Antoniou, G., Billington, D., Governatori, G., and Maher, M. J. 2001.
Representation results for defeasible logic.
ACM Transactions on Computational Logic 2, 2, 255–287.

Artosi, A. and Governatori, G. 1998.
A tableaux methodology for deontic conditional logics.
In *ΔEON'98, 4th International Workshop on Deontic Logic in Computer Science*. CIRFID, Bologna, 65–81.

Artosi, A., Governatori, G., and Rotolo, A. 2002.
Labelled tableaux for non-monotonic reasoning: Cumulative consequence relations.
Journal of Logic and Computation 12, 6, 1027–1060.

Dov M. Gabbay. 1995.
Conditional implications and non-monotonic consequence.
In G. Crocco, L. Fariñas del Cerro, and A. Herzig, editors, *Conditionals: from philosophy to computer science*, Oxford University Press, Oxford, 337–360.