Knowledge Representation and Reasoning

Logics for Artificial Intelligence

Stuart C. Shapiro

Department of Computer Science and Engineering
and Center for Cognitive Science
University at Buffalo, The State University of New York
Buffalo, NY 14260-2000

shapiro@cse.buffalo.edu

Contents

9 Belief Revision/Truth Maintenance

Motivation
Floors Above and Below Ground

```
: andor(1,1){OnFloor(1),OnFloor(2),OnFloor(3),OnFloor(4)}.
: {OnFloor(1), OnFloor(2)} => {Location(belowGround)}.
: {OnFloor(3), OnFloor(4)} => {Location(aboveGround)}.

: perform believe(OnFloor(1))

: list-asserted-wffs
  ~OnFloor(2)
  ~OnFloor(3)
  ~OnFloor(4)
  {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
  {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
  Location(belowGround)
```

```
andor(1,1){OnFloor(4),OnFloor(3),OnFloor(2),OnFloor(1)}
OnFloor(1)
```

Motivation
Disbelieving an Hypothesis

```
: perform disbelieve(OnFloor(1))

: list-asserted-wffs
  {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
  {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
  andor(1,1){OnFloor(4),OnFloor(3),OnFloor(2),OnFloor(1)}
```

Note the absence of `Location(belowGround)`

Moral

If retain derived beliefs (lemmas),

need a way to delete them
when their foundations are removed.

When Needed 1

If the KB contains beliefs about the (some) world,

and the world changes,

and the KB does not have a model of time.

I.e. the beliefs in the KB are of the form,
I believe this is true now.

What's needed

Links from hypotheses to propositions derived from them.

=> and `whendo`

```
: andor(1,1){OnFloor(1),OnFloor(2),OnFloor(3),OnFloor(4)}.
: {OnFloor(1), OnFloor(2)} => {Location(belowGround)}.
: {OnFloor(3), OnFloor(4)} => {Location(aboveGround)}.
: all(x)(whendo(OnFloor(x),believe(HaveBeenOnFloor(x)))).
: perform believe(OnFloor(1))

: list-asserted-wffs
  HaveBeenOnFloor(1)
  ~OnFloor(2)
  ~OnFloor(3)
  ~OnFloor(4)
  {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
  {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
  Location(belowGround)
  andor(1,1){OnFloor(4),OnFloor(3),OnFloor(2),OnFloor(1)}
  OnFloor(1)
```

The Difference Between

=> and `whendo`

```
: perform believe(OnFloor(4))

: list-asserted-wffs
  ~OnFloor(1)
  HaveBeenOnFloor(1)
  ~OnFloor(2)
  ~OnFloor(3)
  {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
  Location(aboveGround)
  {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
  andor(1,1){OnFloor(4),OnFloor(3),OnFloor(2),OnFloor(1)}
  OnFloor(4)
```

Moral

The consequents of `=>`, `v=>`, and `&=>` are derived and retain a connection to their underlying hypotheses.

Whatever is `believe`'d is a hypothesis.

Use `=>`, `v=>`, or `&=>` for logical implications.

Use `whendo(p1,believe(p2))` or `wheneverdo(p1,believe(p2))` for decisions.

Contingent Plans

```
: andor(0,1){Location(BellHall), Location(home)}.

: Location(BellHall) => ActPlan(getMail, go(MailRoom)).

: Location(home) => ActPlan(getMail, go(mailBox)).

: perform believe(Location(BellHall))

: ActPlan(getMail, ?how)?
  ActPlan(getMail,go(MailRoom))

: perform believe(Location(home))

: ActPlan(getMail, ?how)?
  ActPlan(getMail,go(mailBox))
```

Moral

Using this design for contingent plans,

along with retention of lemmas,

depends on belief revision.

Motivation
Sea Creatures

```
: all(x)(andor(0,1){Ako(x, mammal), Ako(x, fish)}).

: all(x)(LiveIn(x, water) => Ako(x, fish)).

: all(x)(BearYoung(x, live) => Ako(x, mammal)).

: LiveIn(whales, water).

: BearYoung(whales, live).
```

Motivation
Are Whales Fish or Mammals?

```
: Ako(whales, ?x)?

A contradiction was detected within context default-defaultct.
The contradiction involves the newly derived proposition:
    Ako(whales,mammal)

and the previously existing proposition:
    ~Ako(whales,mammal)
```

SNeBR Options

```
You have the following options:
 1. [C]ontinue anyway, knowing that a contradiction is derivable;
 2. [R]e-start the exact same run in a different context which is
    not inconsistent;
 3. [D]rop the run altogether.

  (please type c, r or d)
=><= r

In order to make the context consistent you must delete at least
one hypothesis from each of the following sets of hypotheses:
  (wff5 wff4 wff3 wff2 wff1)
```

Culprit Sets

```
 In order to make the context consistent you must delete
  at least one hypothesis from the set listed below.
An inconsistent set of hypotheses:
 1 :   BearYoung(whales,live)
          (2 supported propositions: (wff6 wff5) )
 2 :   LiveIn(whales,water)
          (3 supported propositions: (wff8 wff7 wff4) )
 3 :   all(x)(BearYoung(x,live) => Ako(x,mammal))
          (2 supported propositions: (wff6 wff3) )
 4 :   all(x)(LiveIn(x,water) => Ako(x,fish))
          (3 supported propositions: (wff8 wff7 wff2) )
 5 :   all(x)(andor(0,1){Ako(x,fish),Ako(x,mammal)})
          (2 supported propositions: (wff8 wff1) )
```

Choosing the Culprit

```
  Enter the list number of a hypothesis to examine or
  [d] to discard some hypothesis from this list,
  [a] to see ALL the hypotheses in the full context,
  [r] to see what you have already removed,
  [q] to quit revising this set, or
  [i] for instructions

  (please type a number OR d, a, r, q or i)
=><= d

  Enter the list number of a hypothesis to discard,
  [c] to cancel this discard, or [q] to quit revising this set.
=><= 4

...
```

Resultant KB

```
: list-asserted-wffs
  ~Ako(whales,fish)
  ~(all(x)(LiveIn(x,water) => Ako(x,fish)))
  Ako(whales,mammal)
  BearYoung(whales,live)
  LiveIn(whales,water)
  all(x)(BearYoung(x,live) => Ako(x,mammal))
  all(x)(andor(0,1){Ako(x,fish),Ako(x,mammal)})
```

Moral
When Needed 2

If accepting information from multiple sources,

or just one possibly inconsistent source,

need a way to recognize contradictions,

and to find the culprit,

and to delete it,

and its implications.

What's Needed

Links from derived propositions

to hypotheses they were derived from.

Relevance Logic (R)
Motivation
Paradoxes of Implication 1
Anything Implies a Truth

1.	A	*Hyp*
2.	B	*Hyp*
3.	A	*Reit*, *1*
4.	$B \Rightarrow A$	$\Rightarrow I$, *2–3*
5.	$A \Rightarrow (B \Rightarrow A)$	$\Rightarrow I$, *1–4*

But it seems that B had nothing to do with deriving A.

Motivation of R

Paradoxes of Implication 2

A Contradiction Implies Anything

$$
\begin{array}{lll}
1. & A \wedge \neg A & \mathit{Hyp} \\
2. & \quad \neg B & \mathit{Hyp} \\
3. & \quad A \wedge \neg A & \mathit{Reit}, 1 \\
4. & \quad A & \wedge E, 3 \\
5. & \quad \neg A & \wedge E, 3 \\
6. & \neg\neg B & \neg I, 2\text{--}5 \\
7. & B & \neg E, 6 \\
8. & (A \wedge \neg A) \Rightarrow B & \Rightarrow I, 1\text{--}7
\end{array}
$$

But it seems that $\neg B$ had nothing to do with deriving the contradiction.

What's Needed

A way to determine when a hypothesis is really used to derive another wff.

When a hypothesis is **relevant** to a conclusion.

R
Relevance Logic
The Logic of Relevant Implication

Syntax: The same as Standard FOL.

Intensional Semantics: The same as Standard FOL.

Extensional Semantics: The same as Standard FOL for terms.
For wffs: a four-valued logic, using True, False, Neither, and Both.

KB Interpretations of R's 4 Truth Values

True	true
False	false
Neither	unknown
Both	contradictory, "I've been told both."

R Proof Theory

Structural Rules of Inference

$$\begin{array}{r|l} i. & A, \{n\} \quad Hyp \\ \hline & \end{array} \qquad \begin{array}{r|l} i. & A, \alpha \\ & \vdots \\ j. & A, \alpha \quad Rep, i \end{array} \qquad \begin{array}{r|l} i. & \quad A, \alpha \\ & \quad \vdots \\ & \begin{array}{|l} \vdots \\ A, \alpha \quad Reit, i \end{array} \\ j. & \end{array}$$

where n is a new integer.

R Rules for $\Rightarrow$

$i.$	$A, \{n\}$	Hyp
	$\vdots$	
$j.$	$B, \alpha,\ s.t.\ n \in \alpha$	
k	$(A \Rightarrow B), \alpha - \{n\}$	$\Rightarrow I, i\text{–}j$

$i.$	A, α	
	$\vdots$	
$j.$	$(A \Rightarrow B), \beta$	
$k.$	$B, \alpha \cup \beta$	$\Rightarrow E, i, j$

How the Paradoxes of Implication are Blocked 1

$$
\begin{array}{l|l}
1. & A, \{1\} \quad Hyp \\
\hline
2. & \quad \begin{array}{|l} B, \{2\} \quad Hyp \\ \hline A, \{1\} \quad Reit, 1 \end{array} \\
3. &
\end{array}
$$

Can't then apply $\Rightarrow I$

R Rules for $\wedge$

$$\begin{array}{r|ll}
i. & A, \alpha & \\
 & \vdots & \\
j. & B, \alpha & \\
k. & (A \wedge B), \alpha & \wedge I, i, j
\end{array}$$

$$\begin{array}{r|ll}
i. & (A \wedge B), \alpha & \\
j. & A, \alpha & \wedge E, i
\end{array}
\qquad\qquad
\begin{array}{r|ll}
i. & (A \wedge B), \alpha & \\
j. & B, \alpha & \wedge E, i
\end{array}$$

Why $\wedge I$ Requires the Same OS

If Not

1.	$A, \{1\}$	*Hyp*
2.	$B, \{2\}$	*Hyp*
3.	$A, \{1\}$	*Reit*, *1*
4.	$(A \wedge B), \{1, 2\}$	$\wedge I$?
5.	$A, \{1, 2\}$	$\wedge E$
6.	$(B \Rightarrow A), \{1\}$	$\Rightarrow I, 2-5$
7.	$(A \Rightarrow (B \Rightarrow A)), \{\}$	$\Rightarrow I, 1-6$

Reconstruct paradox of implication.

Note: Empty os means a theorem.

Extended Rule for $\wedge I$

$$\begin{array}{r|ll} i. & A, \alpha & \\ & \vdots & \\ j. & B, \beta & \\ k. & (A \wedge B), (\alpha \cup \beta)^* & \wedge I, i, j \end{array}$$

Can't apply $\wedge E$ to an extended wff.

R Rules for $\neg$

$$
\begin{array}{r|l}
i. & \begin{array}{|ll} A, \{n\} & Hyp \\ \hline \vdots & \\ B, \alpha \ s.t. \ n \in \alpha & \\ \neg B, \alpha & \end{array} \\
j+2. & \neg A, \alpha - \{n\} \qquad \neg I, i\text{–}(j+1)
\end{array}
$$

$$
\begin{array}{r|ll}
i. & \neg\neg A, \alpha & \\
j. & A, \alpha & \neg E, i
\end{array}
$$

Extended R Rule for $\neg I$

$i.$	$A, \{n\}$	Hyp
	$\vdots$	
$j.$	B, α	
$j+1.$	$\neg B, \beta$	
$j+2.$	$\neg A, ((\alpha \cup \beta) - \{n\})^*$ $s.t.$ $n \in (\alpha \cup \beta)$	$\neg I, i$–$(j+1)$

How the Paradoxes of Implication are Blocked 2

$$
\begin{array}{r|l}
1. & \begin{array}{ll} (A \wedge \neg A), \{1\} & \mathit{Hyp} \end{array} \\ \hline
2. & \quad \begin{array}{|ll} \neg B, \{2\} & \mathit{Hyp} \\ \hline (A \wedge \neg A), \{1\} & \mathit{Reit}, \mathit{1} \\ A, \{1\} & \wedge E, \mathit{3} \\ \neg A, \{1\} & \wedge E, \mathit{3} \end{array}
\end{array}
$$

Can't then apply $\neg I$

R is a **paraconsistent** logic:
a contradiction does not imply anything whatsoever.

R Rules for $\vee I$

$$\begin{array}{r|l} i. & A, \{n\} \\ j. & (A \vee B), \{n\} \quad \vee I, i \end{array} \qquad\qquad \begin{array}{r|l} i. & B, \{n\} \\ j. & (A \vee B), \{n\} \quad \vee I, i \end{array}$$

Irrelevance of Disjunctive Syllogism

1.	$((A \vee B) \wedge \neg A), \{1\}$	Hyp	
2.	$\neg A, \{1\}$	$\wedge E, 1$	
3.	$(A \vee B), \{1\}$	$\wedge E, 1$	
4.	$A, \{2\}$	Hyp	
5.	$\neg B, \{3\}$	Hyp	
6.	$A, \{2\}$	$Reit, 4$	
7.	$\neg A, \{1\}$	$Reit, 2$	
8.	$\neg\neg B$	$\neg I, 5–7$	**Not valid in R**
9.	B	$\neg E$	
10.	B	Hyp	
11.	B	$Rep, 10$	
12.	B	$\vee E, 3–11$	

So $\vee$ is just truth-functional, and don't need $\vee E$

R Rules for Intensional OR ($\oplus$)

$$\begin{array}{r|ll}
i. & (\neg A \Rightarrow B), \alpha & \\
 & \vdots & \\
j. & (\neg B \Rightarrow A), \alpha & \\
j+1. & (A \oplus B), \alpha & \oplus I, i, j
\end{array}$$

$$\begin{array}{r|ll}
i. & (A \oplus B), \alpha & \\
 & \vdots & \\
j. & \neg A, \beta & \\
j+1. & B, \alpha \cup \beta & \oplus E
\end{array}
\qquad
\begin{array}{r|ll}
i. & (A \oplus B), \alpha & \\
 & \vdots & \\
j. & \neg B, \beta & \\
j+1. & A, \alpha \cup \beta & \oplus E
\end{array}$$

R Rules for $\Leftrightarrow$

$$
\begin{array}{r|ll}
i. & (A \Rightarrow B), \alpha & \\
 & \vdots & \\
j. & (B \Rightarrow A), \alpha & \\
j+1. & (A \Leftrightarrow B), \alpha & \Leftrightarrow I, i, j
\end{array}
$$

$$
\begin{array}{r|cl}
i. & A, \alpha & \\
 & \vdots & \\
j. & (A \Leftrightarrow B), \beta & \\
j+1. & B, \alpha \cup \beta & \Leftrightarrow E, i, j
\end{array}
\qquad
\begin{array}{r|cl}
i. & B, \alpha & \\
 & \vdots & \\
j. & (A \Leftrightarrow B), \beta & \\
j+1. & A, \alpha \cup \beta & \Leftrightarrow E, i, j
\end{array}
$$

R Rules for $\forall$

$$
\begin{array}{r|l}
i. & \begin{array}{|ll} A(a), \{n\} & \qquad\qquad\qquad Hyp \\ \hline \vdots & \\ B(a), \alpha \ s.t.\ n \in \alpha & \end{array} \\
j+1. & \forall x(A(x) \Rightarrow B(x)), \alpha\{n\} \quad \forall I, i\text{–}j
\end{array}
$$

$$
\begin{array}{r|ll}
i. & A(t), \alpha & \\
 & \vdots & \\
j. & \forall x(A(x) \Rightarrow B(x)), \beta & \\
j+1. & B(t), \alpha \cup \beta & \forall E, i, j
\end{array}
$$

Where a is an arbitrary individual not otherwise used in the proof.

Note $\forall$ only governs $\Rightarrow$.

R Rules for $\exists$

$$
\begin{array}{r|ll}
i. & A(t), \alpha & \\
 & \vdots & \\
j. & \exists x A(x), \alpha & \exists I, i
\end{array}
$$

$$
\begin{array}{r|ll}
i. & \exists x A(x), \alpha & \\
 & \vdots & \\
j. & A(s), \alpha & \exists E, i
\end{array}
$$

Where s is a Skolem constant not otherwise used in the proof,

Why the Subproof Contours?

1. To keep track of assumptions for each derived wff.
   But this is accomplished by os.

2. To differentiate hypotheses from derived wffs.
   Introduce support: $\langle\{hyp \mid der \mid ext\}, os\rangle$
   with origin tag and origin set.

Restriction Sets

Extend support to $\langle \{hyp \mid der \mid ext\}, os, rs \rangle$

where rs is a restriction set:
a set of sets, α, s.t. $os \cup \alpha$ is (known to be) inconsistent.

SNePS KB

The SNePS KB consists of a collection of supported wffs.

A wff may have more than one support if it was derived in multiple ways.

Every implemented rule of inference specifies how the derived wff is derived from its parent(s) and how its support is derived from the support(s) of its parent(s).

Contexts and Belief Spaces

A context is a set of hypotheses.

A belief space defined by a context c is the set conaining every wff whose os is a subset of c.

SNePSLOG Example

```
: expert
...
: andor(1,1){OnFloor(1),OnFloor(2),OnFloor(3),OnFloor(4)}.
  wff5: andor(1,1){OnFloor(4),OnFloor(3),
                   OnFloor(2),OnFloor(1)}
        {<hyp,{wff5},{}>}

: {OnFloor(1), OnFloor(2)} => {Location(belowGround)}.
  wff7: {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
        {<hyp,{wff7},{}>}

: {OnFloor(3), OnFloor(4)} => {Location(aboveGround)}.
  wff9: {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
        {<hyp,{wff9},{}>}
```

```
: perform believe(OnFloor(1))
...
: describe-context
((assertions (wff9 wff7 wff5 wff1)) (restriction nil)
 (named (default-defaultct)))
```

```
: list-asserted-wffs
  wff13: ~OnFloor(2)  {<der,{wff1,wff5},{}>}
  wff12: ~OnFloor(3)  {<der,{wff1,wff5},{}>}
  wff11: ~OnFloor(4)  {<der,{wff1,wff5},{}>}
  wff9: {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
        {<hyp,{wff9},{}>}
  wff7: {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
        {<hyp,{wff7},{}>}
  wff6: Location(belowGround)  {<der,{wff1,wff7},{}>}
  wff5: andor(1,1){OnFloor(4),OnFloor(3),
                   OnFloor(2),OnFloor(1)}
        {<hyp,{wff5},{}>}
  wff1: OnFloor(1)  {<hyp,{wff1},{}>}
```

```
: perform disbelieve(OnFloor(1))
...
: describe-context
((assertions (wff9 wff7 wff5)) (restriction nil)
 (named (default-defaultct)))

: list-asserted-wffs
  wff9: {OnFloor(4),OnFloor(3)} v=> {Location(aboveGround)}
        {<hyp,{wff9},{}>}
  wff7: {OnFloor(2),OnFloor(1)} v=> {Location(belowGround)}
        {<hyp,{wff7},{}>}
  wff5: andor(1,1){OnFloor(4),OnFloor(3),
                  OnFloor(2),OnFloor(1)}
        {<hyp,{wff5},{}>}
```

SNePSLOG Example of $\neg I$

```
wff1: all(x)(andor(0,1){Ako(x,fish),Ako(x,mammal)})
      {<hyp,{wff1},{}>}
wff2: all(x)(LiveIn(x,water) => Ako(x,fish))
      {<hyp,{wff2},{}>}
wff3: all(x)(BearYoung(x,live) => Ako(x,mammal))
      {<hyp,{wff3},{}>}
wff4: LiveIn(whales,water)  {<hyp,{wff4},{}>}
wff5: BearYoung(whales,live)  {<hyp,{wff5},{}>}
```

```
: Ako(whales, ?x)?
...
A contradiction was detected within context default-defaultct.
The contradiction involves the newly derived proposition:
    wff6: Ako(whales,mammal)
          {<der,{wff3,wff5},{{wff1,wff2,wff4}}>}

and the previously existing proposition:
    wff8: ~Ako(whales,mammal)
          {<der,{wff1,wff2,wff4},{{wff3,wff5}}>}
...
In order to make the context consistent you must delete
at least one hypothesis from each of the following sets
of hypotheses:
  (wff5 wff4 wff3 wff2 wff1)
```

The Culprit Set

```
1 :  wff5: BearYoung(whales,live)
           {<hyp,{wff5},{{wff1,wff2,wff3,wff4}}>}
         (2 supported propositions: (wff6 wff5) )
2 :  wff4: LiveIn(whales,water)
           {<hyp,{wff4},{{wff1,wff2,wff3,wff5}}>}
         (3 supported propositions: (wff8 wff7 wff4) )
3 :  wff3: all(x)(BearYoung(x,live) => Ako(x,mammal))
           {<hyp,{wff3},{{wff1,wff2,wff4,wff5}}>}
         (2 supported propositions: (wff6 wff3) )
4 :  wff2: all(x)(LiveIn(x,water) => Ako(x,fish))
           {<hyp,{wff2},{{wff1,wff3,wff4,wff5}}>}
         (3 supported propositions: (wff8 wff7 wff2) )
5 :  wff1: all(x)(andor(0,1){Ako(x,fish),Ako(x,mammal)})
           {<hyp,{wff1},{{wff2,wff3,wff4,wff5}}>}
         (2 supported propositions: (wff8 wff1) )
```

KB after deleting `wff2`

```
wff10: ~Ako(whales,fish)
      {<der,{wff1,wff3,wff5},{{wff2,wff4}}>}
wff9: ~(all(x)(LiveIn(x,water) => Ako(x,fish)))
      {<ext,{wff1,wff3,wff4,wff5},{{wff2}}>}
wff6: Ako(whales,mammal)
      {<der,{wff3,wff5},{{wff1,wff2,wff4}}>}
wff5: BearYoung(whales,live)
      {<hyp,{wff5},{{wff1,wff2,wff3,wff4}}>}
wff4: LiveIn(whales,water)
      {<hyp,{wff4},{{wff1,wff2,wff3,wff5}}>}
wff3: all(x)(BearYoung(x,live) => Ako(x,mammal))
      {<hyp,{wff3},{{wff1,wff2,wff4,wff5}}>}
wff1: all(x)(andor(0,1){Ako(x,fish),Ako(x,mammal)})
      {<hyp,{wff1},{{wff2,wff3,wff4,wff5}}>}
```