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

1. Introduction 3

2. Propositional Logic 15

3. Predicate Logic 82

4. Summary 177

2 Propositional Logic

Logics that do not analyze information below the level of the sentence (proposition).

1. CarPool World: A motivational "micro-world" 16

2. The "Standard" Propositional Logic 17

2.1 CarPool World

- Tom and Betty carpool to work.
- On any day, either Tom drives Betty or Betty drives Tom.
- In the former case, Tom is the driver and Betty is the passenger.
- In the latter case, Betty is the driver and Tom is the passenger.

Sentences	
Betty drives Tom.	Tom drives Betty.
Betty is the driver.	Tom is the driver.
Betty is the passenger.	Tom is the passenger.

2.2 The "Standard" Propositional Logic

1. Syntax 18

2. Semantics 22

3. Proof Theory 55

Syntax of the "Standard" Propositional Logic

Atomic Propositions

- Any letter of the alphabet, e.g.: P
- Any letter of the alphabet with a numeric subscript, e.g.: Q_3
- Any alphanumeric string, e.g.: *Tom is the driver*

is an atomic proposition.

Well-Formed Propositions (WFPs)

1. Every atomic proposition is a wfp.

2. If P is a wfp, then so is $(\neg P)$.

3. If P and Q are wfps, then so are

   (a) $(P \wedge Q)$ (b) $(P \vee Q)$

   (c) $(P \Rightarrow Q)$ (d) $(P \Leftrightarrow Q)$

4. Nothing else is a wfp.

Parentheses may be omitted
or square brackets may be used instead.

Alternative Symbols

$\neg$: $\sim$

$\wedge$: &

$\vee$: $|$

$\Rightarrow$: $\rightarrow$ $\supset$

$\Leftrightarrow$: $\leftrightarrow$ $\equiv$

Examples of WFPs

$$\neg(A \wedge B) \Leftrightarrow (\neg A \vee \neg B)$$

$$\textit{Tom is the driver} \Rightarrow \textit{Betty is the passenger}$$

$$\textit{Betty drives Tom} \Leftrightarrow \neg \textit{Tom is the driver}$$

Semantics of Atomic Propositions

Extensional Semantics:

- Relative to an interpretation/model/situation.
- Either True or False.

Intensional Semantics:

- Dependent on a Domain.
- Independent of any specific interpretation/model/situation.
- Statement in a previously understood language (e.g. English) that allows truth value to be determined in any specific situation.
- Often omitted, but shouldn't be.

Intensional CarPool World Semantics

$[Betty\ drives\ Tom]$ = Betty drives Tom to work.

$[Tom\ drives\ Betty]$ = Tom drives Betty to work.

$[Betty\ is\ the\ driver]$ = Betty is the driver of the car.

$[Tom\ is\ the\ driver]$ = Tom is the driver of the car.

$[Betty\ is\ the\ passenger]$ = Betty is a passenger in the car.

$[Tom\ is\ the\ passenger]$ = Tom is a passenger in the car.

Alternative Intensional Seesaw World Semantics

$[Betty\ drives\ Tom]$ = Tom drives Betty to work.

$[Tom\ drives\ Betty]$ = Betty drives Tom to work.

$[Betty\ is\ the\ driver]$ = Tom is a passenger in the car.

$[Tom\ is\ the\ driver]$ = Betty is a passenger in the car.

$[Betty\ is\ the\ passenger]$ = Tom is the driver of the car.

$[Tom\ is\ the\ passenger]$ = Betty is the driver of the car.

Alternative Seesaw World

Syntax/Intensional Semantics

$[A]$ = Betty drives Tom to work.

$[B]$ = Tom drives Betty to work.

$[C]$ = Betty is the driver of the car.

$[D]$ = Tom is the driver of the car.

$[E]$ = Betty is a passenger in the car.

$[F]$ = Tom is a passenger in the car.

Intensional Semantics Moral

- Don't omit.
- Don't presume.
- No "pretend it's English semantics".

Seesaw World
Extensional Semantics

	Denotation in Situation				
Proposition	1	2	3	4	5
Betty drives Tom	True	True	True	False	False
Tom drives Betty	True	True	False	True	False
Betty is the driver	True	True	True	False	False
Tom is the driver	True	False	False	True	False
Betty is the passenger	True	False	False	True	False
Tom is the passenger	True	False	True	False	False

Note: n propositions $\Rightarrow 2^n$ possible situations.

6 propositions in CarPool World
$\Rightarrow 2^6 = 64$ different situations.

Intensional Semantics of WFPs

$[\neg P]$ = It is not the case that $[P]$.

$[P \wedge Q]$ = $[P]$ and $[Q]$.

$[P \vee Q]$ = Either $[P]$ or $[Q]$ or both.

$[P \Rightarrow Q]$ = If $[P]$ then $[Q]$.

$[P \Leftrightarrow Q]$ = $[P]$ if and only if $[Q]$.

Example CarPool World Intensional WFP Semantics

$[Betty\ drives\ Tom \Leftrightarrow \neg Tom\ is\ the\ driver]$
= Betty drives Tom to work
if and only if Tom is not the driver.

Extensional Semantics of WFPs

$[\![\neg P]\!]$ is True if $[\![P]\!]$ is False. Otherwise, it is False.

$[\![P \wedge Q]\!]$ is True if $[\![P]\!]$ is True and $[\![Q]\!]$ is True. Otherwise, it is False.

$[\![P \vee Q]\!]$ is False if $[\![P]\!]$ is False and $[\![Q]\!]$ is False. Otherwise, it is True.

$[\![P \Rightarrow Q]\!]$ is False if $[\![P]\!]$ is True and $[\![Q]\!]$ is False. Otherwise, it is True.

$[\![P \Leftrightarrow Q]\!]$ is True if $[\![P]\!]$ and $[\![Q]\!]$ are both True, or both False. Otherwise, it is False.

Extensional Semantics Truth Tables

P	True	False
$\neg P$	False	True

P	True	True	False	False
Q	True	False	True	False
$P \wedge Q$	True	False	False	False
$P \vee Q$	True	True	True	False
$P \Rightarrow Q$	True	False	True	True
$P \Leftrightarrow Q$	True	False	False	True

Notice that each column of these tables represents a different situation.

Material Implication

$P \Rightarrow Q$ is True when P is False.

So,
If the world is flat, then the moon is made of green cheese is considered True if *if . . . then* is interpreted as material implication.

Example CarPool World Truth Table

$Betty\ drives\ Tom$	True	True	False	False
$Tom\ is\ the\ driver$	True	False	True	False
$\neg Tom\ is\ the\ driver$	False	True	False	True
$Betty\ drives\ Tom \Leftrightarrow \neg Tom\ is\ the\ driver$	False	True	True	False

Semantic Properties of WFPs

- A wfp is **satisfiable** if it is True in at least one situation.

- A wfp is **contingent** if it is True in at least one situation and False in at least one situation.

- A wfp is **valid** ($\models P$) if it is True in every situation.
A valid wfp is also called a **tautology**.

- A wfp is **unsatisfiable** or **contradictory**
if it is False in every situation.

Examples

P	True	True	False	False
Q	True	False	True	False
$\neg P$	False	False	True	True
$Q \Rightarrow P$	True	True	False	True
$P \Rightarrow (Q \Rightarrow P)$	True	True	True	True
$P \wedge \neg P$	False	False	False	False

$\neg P$, $Q \Rightarrow P$, and $P \Rightarrow (Q \Rightarrow P)$ are satisfiable,

$\neg P$ and $Q \Rightarrow P$ are contingent,

$P \Rightarrow (Q \Rightarrow P)$ is valid,

$P \wedge \neg P$ is contradictory.

Logical Implication

$\{A_1, \ldots, A_n\}$ **logically implies** B in logic $\mathcal{L}$

$$A_1, \ldots, A_n \models_{\mathcal{L}} B$$

if B is True in every situation in which every A_i is True.

and if $\mathcal{L}$ is assumed,

$$A_1, \ldots, A_n \models B$$

Some Relations

$$A_1, \ldots, A_n \models B$$
iff
$$A_1 \wedge \cdots \wedge A_n \models B$$

If $n = 0$, we have validity, $\models B$,
i.e., B is True in every situation.

Domain Knowledge (Rules)

Used to reduce the set of situations to those that “make sense”.

To see if
Domain Rules $\models$ *Query*
restrict situations to those that satisfy Domain Rules.

Domain Rules for CarPool World

$$Betty\ \ is\ \ the\ \ driver \Leftrightarrow \neg Betty\ \ is\ \ the\ \ passenger$$

$$Tom\ \ is\ \ the\ \ driver \Leftrightarrow \neg Tom\ \ is\ \ the\ \ passenger$$

$$Betty\ \ drives\ \ Tom \Rightarrow Betty\ \ is\ \ the\ \ driver \wedge Tom\ \ is\ \ the\ \ passenger$$

$$Tom\ \ drives\ \ Betty \Rightarrow Tom\ \ is\ \ the\ \ driver \wedge Betty\ \ is\ \ the\ \ passenger$$

$$Tom\ \ drives\ \ Betty \vee Betty\ \ drives\ \ Tom$$

Sensible CarPool World Situations

The only 2 of the 64 in which all domain rules are True:

Proposition	Denotation in Situation 3	Denotation in Situation 4
Betty drives Tom	True	False
Tom drives Betty	False	True
Betty is the driver	True	False
Tom is the driver	False	True
Betty is the passenger	False	True
Tom is the passenger	True	False
$\mathit{Betty\ drives\ Tom} \Leftrightarrow \neg \mathit{Tom\ is\ the\ driver}$	True	True

Significant Metatheorem

$A_1, \ldots, A_n \models P$ if and only if $\models A_1 \wedge \cdots \wedge A_n \Rightarrow P$.

So deciding validity and logical implication are equivalent.

Wang's Algorithm*
A Semantic Inference Method

To Show $A \models P$:

- Try to construct a truth table line in which A is True and P is false.

- If you succeed, $A \not\models P$.

- If you fail, $A \models P$.

*Wang, H. 1960. Toward Mechanical Mathematics. IBM Journal of Research and Development 4, 2-22. Reprinted in K. M. Sayre and F. J. Crosson (Eds.) The Modeling of Mind: Computers and Intelligence. Simon and Schuster, New York, 1963.

Wang's Algorithm

```
wang(Twfps, Fwfps) {
  /*
   * Twfps and Fwfps are sets of wfps.
   * Returns True if there is no model
   * that satisfies Twfps and falsifies Fwfps;
   * Otherwise, returns False.
  */
```

Wang Algorithm

if $Twfps$ **and** $Fwfps$ **intersect, return True;**

if every $A \in Twfps \cup Fwfps$ **is atomic, return False;**

if $(P = \neg A) \in Twfps$**, return wang(**$Twfps - P$**,** $Fwfps \cup \{A\}$**);**

if $(P = \neg A) \in Fwfps$**, return wang(**$Twfps \cup \{A\}$**,** $Fwfps - P$**);**

Wang Algorithm

```
if (P = A ∧ B) ∈ Twfps,
   return wang(Twfps − P ∪ {A, B}, Fwfps);
if (P = A ∧ B) ∈ Fwfps,
   return wang(Twfps, Fwfps − P ∪ {A})
          and wang(Twfps, Fwfps − P ∪ {B});

if (P = A ∨ B) ∈ Twfps,
   return wang(Twfps − P ∪ {A}, Fwfps);
          and wang(Twfps − P ∪ {B}, Fwfps);
if (P = A ∨ B) ∈ Fwfps,
   return wang(Twfps, Fwfps − P ∪ {A, B})
```

Wang Algorithm

```
if (P = (A ⇒ B)) ∈ Twfps,
   return wang(Twfps − P, Fwfps ∪ {A})
          and wang(Twfps − P ∪ {B}, Fwfps);
if (P = (A ⇒ B)) ∈ Fwfps,
   return wang(Twfps ∪ {A}, Fwfps − P ∪ {B});

if (P = (A ⇔ B)) ∈ Twfps,
   return wang(Twfps − P ∪ {A, B}, Fwfps)
          and wang(Twfps − P, Fwfps ∪ {A, B});
if (P = (A ⇔ B)) ∈ Fwfps,
   return wang(Twfps ∪ {A}, Fwfps − P ∪ {B})
          and wang(Twfps ∪ {B}, Fwfps − P ∪ {A});
}
```

Implemented Wang Function

`(wang '(`$A_1, \ldots, A_n$`) '(`P`))`
Returns `t` if $A_1, \ldots, A_n \models P$;
`nil` otherwise.

Using Wang's Algorithm on a Tautology

```
(wang              '()    '((A => (B => A))))
 0: (wang          nil    ((A => (B => A))))
   1: (wang        (A)    ((B => A)))
     2: (wang      (B A)  (A))
     2: returned t
   1: returned t
 0: returned t
t
```

Using Wang's Algorithm on a Non-Tautology

```
(wang                   '() '((A => (A & B))))
 0: (wang               nil ((A => (A & B))))
   1: (wang             (A) ((A & B)))
     2: (wang           (A) (A))
     2: returned t
     2: (wang           (A) (B))
     2: returned nil
   1: returned nil
 0: returned nil
nil
```

Using Wang's Algorithm to see if

$$TD, TD \Rightarrow BP, BP \Rightarrow \neg BD \models \neg BD$$

```
(wang                  '(TD (TD => BP) (BP => (~ BD))) '((~ BD)))
 0: (wang              (TD (TD => BP) (BP => (~ BD)))  ((~ BD)))
   1: (wang            (BP TD (BP => (~ BD)))          ((~ BD)))
     2: (wang          ((~ BD) BP TD)                  ((~ BD)))
     2: returned t
     2: (wang          (BP TD)                         (BP (~ BD)))
     2: returned t
   1: returned t
   1: (wang            (TD (BP => (~ BD)))             (TD (~ BD)))
   1: returned t
 0: returned t
t
```

Properties of Wang's Algorithm

Wang's Algorithm is **sound**.

If `(wang A '(B)) = t` then $A \models B$

Wang's Algorithm is **complete**.

If $A \models B$ then `(wang A '(B)) = t`

Wang's Algorithm is a **decision procedure**.

For any valid inputs `A`, `B`,

`(wang A '(B))` terminates

and returns `t` iff $A \models B$

Alternative View of Wang Function

(**`wang`** *KB Query*)
Returns `t` if the *Query* follows from the *KB*
`nil` otherwise.

Semantic Tableaux

Wang's Algorithm
is a version of the Semantic Tableau method
due to E. Beth, *The Foundations of Mathematics*, (Amsterdam: North Holland), 1959.

There has been much work done on this approach since.

2.2.3 Proof Theory of the Standard Propositional Logic

- Specifies when a given wfp can be derived correctly from a set of (other) wfps.

$$A_1, \ldots, A_n \vdash P$$

- Determines what wfps are **theorems** of the logic.

$$\vdash P$$

- Depends on the notion of **proof**.

Hilbert-Style Syntactic Inference

- Set of **Axioms**.
- Small set of **Rules of Inference**.

Hilbert-Style Proof

- A **proof** of a theorem P is
  - An ordered list of wfps ending with P
  - Each wfp on the list is
    - Either an axiom
    - or follows from previous wfps in the list by one of the rules of inference.

Hilbert-Style Derivation

- A **derivation** of P from $A_1, \ldots, A_n$ is
  - A list of wfps ending with P
  - Each wfp on the list is
    - Either an axiom
    - or some A_i
    - or follows from previous wfps in the list by one of the rules of inference.

Example Hilbert-Style Axioms*

All instances of:

(A1). $(\mathcal{A} \Rightarrow (\mathcal{B} \Rightarrow \mathcal{A}))$

(A2). $((\mathcal{A} \Rightarrow (\mathcal{B} \Rightarrow \mathcal{C})) \Rightarrow ((\mathcal{A} \Rightarrow \mathcal{B}) \Rightarrow (\mathcal{A} \Rightarrow \mathcal{C})))$

(A3). $((\neg\mathcal{B} \Rightarrow \neg\mathcal{A}) \Rightarrow ((\neg\mathcal{B} \Rightarrow \mathcal{A}) \Rightarrow \mathcal{B}))$

*From E. Mendelson, *Introduction to Mathematical Logic*, (Princeton: D. Van Nostrand) 1964, pp. 31–32.

Hilbert-Style Rule of Inference

Modus Ponens

$$\frac{\mathcal{A}, \mathcal{A} \Rightarrow \mathcal{B}}{\mathcal{B}}$$

Example Hilbert-Style Proof

(1)	$(A \Rightarrow ((A \Rightarrow A) \Rightarrow A))$	
	$\Rightarrow ((A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A))$	Instance of A2
(2)	$A \Rightarrow ((A \Rightarrow A) \Rightarrow A)$	Instance of A1
(3)	$(A \Rightarrow (A \Rightarrow A)) \Rightarrow (A \Rightarrow A)$	From 1, 2 by MP
(4)	$A \Rightarrow (A \Rightarrow A)$	Instance of A1
(5)	$A \Rightarrow A$	From 3, 4 by MP

Example Hilbert-Style Derivation

Of	$\neg Betty\ is\ the\ driver$
from	$John\ is\ the\ driver \Rightarrow Betty\ is\ the\ passenger$,
	$Betty\ is\ the\ passenger \Rightarrow \neg Betty\ is\ the\ driver$,
and	$John\ is\ the\ driver$

(1)	$John\ is\ the\ driver$	Assumption
(2)	$John\ is\ the\ driver \Rightarrow Betty\ is\ the\ passenger$	Assumption
(3)	$Betty\ is\ the\ passenger$	From 1, 2 by MP
(4)	$Betty\ is\ the\ passenger \Rightarrow \neg Betty\ is\ the\ driver$	Assumption
(5)	$\neg Betty\ is\ the\ driver$	From 3, 4 by MP

Some AI Connections

AI	Logic
domain knowledge or domain rules	assumptions or non-logical axioms
inference engine procedures	rules of inference
knowledge base	proof

Natural Deduction
(Style of Syntactic Inference)

- **No** Axioms.

- **Large** set of Rules of Inference.

  - A few **structural rules** of inference.

  - An **introduction rule** and an **elimination rule** for each connective.

Fitch-Style Proof*

- A **proof** of a theorem P is
  - An ordered list of wfps and subproofs ending with P
  - Each wfp or subproof on the list must be justified by a rule of inference.

*Based on F. B. Fitch, *Symbolic Logic: An Introduction*, (New York: Ronald Press), 1952.

Fitch-Style Derivation

- A **derivation** of a wfp P from an assumption A is a hypothetical subproof whose hypothesis is A and which contains
  - An ordered list of wfps and inner subproofs ending with P
  - Each wfp or inner subproof on the list must be justified by a rule of inference.

Structural Rules of Inference

$$\begin{array}{r|l} i. & A \quad Hyp \\ \hline & \end{array}$$

$$\begin{array}{r|l} i. & A \\ & \vdots \\ j. & A \quad Rep, i \end{array}$$

$$\begin{array}{r|l|l} i. & \multicolumn{2}{l}{A} \\ & \multicolumn{2}{l}{\vdots} \\ & & \vdots \\ j. & & A \quad Reit, i \end{array}$$

Rules for $\neg$

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

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

Rules for $\wedge$

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

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

Rules for $\vee$

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

$$
\begin{array}{r|l}
i. & A \vee B \\
i+1. & \quad \begin{array}{|l} A \quad Hyp \\ \hline \vdots \\ C \end{array} \\
j & \\
j+1. & \quad \begin{array}{|l} B \quad Hyp \\ \hline \vdots \\ C \end{array} \\
k & \\
k+1. & C \qquad \vee E, i-k
\end{array}
$$

Rules for $\Rightarrow$

$$
\begin{array}{r|l}
i. & \begin{array}{|ll} \quad\quad A & Hyp \\ \hline \quad\quad \vdots & \\ \quad\quad B & \end{array} \\
j. & \\
k & A \Rightarrow B \quad\quad \Rightarrow I, i\text{--}j
\end{array}
\qquad\qquad
\begin{array}{r|ll}
i. & A & \\
 & \vdots & \\
j. & A \Rightarrow B & \\
k. & B & \Rightarrow E, i, j
\end{array}
$$

Rules for $\Leftrightarrow$

$$
\begin{array}{r|l}
i. & \quad \begin{array}{|ll} A & Hyp \\ \hline \vdots & \\ B & \end{array} \\
j. & \\
j+1. & \quad \begin{array}{|ll} B & Hyp \\ \hline \vdots & \\ A & \end{array} \\
k. & \\
k+1. & A \Leftrightarrow B \qquad \Leftrightarrow I, i\text{–}k
\end{array}
$$

$$
\begin{array}{r|ll}
i. & A & \\
 & \vdots & \\
j. & A \Leftrightarrow B & \\
k. & B & \Leftrightarrow E, i, j
\end{array}
\qquad\qquad
\begin{array}{r|ll}
i. & B & \\
 & \vdots & \\
j. & A \Leftrightarrow B & \\
k. & A & \Leftrightarrow E, i, j
\end{array}
$$

Example Fitch-Style Proof

$$
\begin{array}{ll}
\begin{array}{r|ll}
1. & A & Hyp \\ \hline
2. & A & Rep, 1
\end{array} & \\
3.\quad A \Rightarrow A & \Rightarrow I, 1\text{–}2
\end{array}
$$

Example Fitch-Style Derivation

1.	*Tom is the driver* $\wedge$(*Tom is the driver* $\Rightarrow$ *Betty is the passenger*) $\wedge$(*Betty is the passenger* $\Rightarrow \neg$*Betty is the driver*)	Hyp
2.	*Tom is the driver*	$\wedge E, 1$
3.	*Tom is the driver* $\Rightarrow$ *Betty is the passenger*	$\wedge E, 1$
4.	*Betty is the passenger*	$\Rightarrow E, 2, 3$
5.	*Betty is the passenger* $\Rightarrow \neg$*Betty is the driver*	$\wedge E, 1$
6.	$\neg$*Betty is the driver*	$\Rightarrow E, 4, 5$

Fitch-Style Proof of Axiom A1

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

CarPool World Derivation

1.	$(Tom\ is\ the\ driver \Leftrightarrow \neg Tom\ is\ the\ passenger)$ $\wedge\ (Tom\ is\ the\ passenger \Leftrightarrow Betty\ is\ the\ driver)$ $\wedge\ (Betty\ is\ the\ driver \Leftrightarrow \neg Betty\ is\ the\ passenger)$ $\wedge\ Tom\ is\ the\ driver$	Hyp
2.	$Tom\ is\ the\ driver$	$\wedge E$, 1
3.	$Tom\ is\ the\ driver \Leftrightarrow \neg Tom\ is\ the\ passenger$	$\wedge E$, 1
4.	$\neg Tom\ is\ the\ passenger$	$\Leftrightarrow E$, 3
5.	$\neg Betty\ is\ the\ passenger$	Hyp
6.	$Betty\ is\ the\ driver \Leftrightarrow \neg Betty\ is\ the\ passenger$	$\wedge E$, 1
7.	$Betty\ is\ the\ driver$	$\Leftrightarrow E$, 5, 6
8.	$Tom\ is\ the\ passenger \Leftrightarrow Betty\ is\ the\ driver$	$\wedge E$, 1
9.	$Tom\ is\ the\ passenger$	$\Leftrightarrow E$, 7, 8
10.	$\neg Tom\ is\ the\ passenger$	$Reit$, 4
11.	$\neg\neg Betty\ is\ the\ passenger$	$\neg I$, 5–10
12.	$Betty\ is\ the\ passenger$	$\neg E$, 11

More Connections

- $A \vdash P$ if and only if $\vdash A \Rightarrow P$.
- So proving theorems and deriving conclusions from assumptions are equivalent.
- But no atomic proposition is a theorem.
- So theorem proving makes more use of Introduction Rules than most AI reasoning systems.

3 Important Properties of Logical Systems

Soundness: $\vdash P$ implies $\models P$

Consistency: not both $\vdash P$ and $\vdash \neg P$

Completeness: $\models P$ implies $\vdash P$

Connections

- If at most 1 of $\models P$ and $\models \neg P$ then soundness implies consistency.

- Soundness is the essence of "correct reasoning."

- Completeness less important for running systems since a proof may take too long to wait for.

- The Propositional Logics we have been looking at are complete.

- Gödel's Incompleteness Theorem: A logic strong enough to formalize arithmetic is *either* inconsistent *or* incomplete.

More Connections

$$A_1, \ldots, A_n \vdash P \qquad \Leftrightarrow \qquad \vdash A_1 \wedge \ldots \wedge A_n \Rightarrow P$$

soundness $\Downarrow\Uparrow$ completeness soundness $\Downarrow\Uparrow$ completeness

$$A_1, \ldots, A_n \models P \qquad \Leftrightarrow \qquad \models A_1 \wedge \ldots \wedge A_n \Rightarrow P$$