1 Generalised Modus Ponens

Part 1: Convert the following sentences to first-order logic formulae suitable for use with Generalised Modus Ponens.

1. Horses, cows and pigs are mammals.
2. An offspring of a horse is a horse.
3. Bluebeard is a horse.
4. Bluebeard is Charlie’s parent.
5. Offspring and parent are inverse relations.

Part 2: Use the sentences to answer a query using a backward-chaining algorithm.

• Draw the proof tree generated by an exhaustive backward-chaining algorithm for the query $\text{Horse}(h)$, where clauses are matched in the order given.

• How many solutions are a logical consequence of your knowledge base?

• How could we solve this problem?

Answer

Part 1:

1. $\text{Horse}(x) \Rightarrow \text{Mammal}(x)$
   $\text{Cow}(x) \Rightarrow \text{Mammal}(x)$
   $\text{Pig}(x) \Rightarrow \text{Mammal}(x)$

2. $\text{Offspring}(y, x) \land \text{Horse}(x) \Rightarrow \text{Horse}(y)$ (y is offspring of x)

3. $\text{Horse}(\text{Bluebeard})$

4. $\text{Parent}(\text{Bluebeard}, \text{Charlie})$ (x is parent of y)
5. Offspring\((x, y)\) ⇒ Parent\((y, x)\)  
    Parent\((x, y)\) ⇒ Offspring\((y, x)\)

Part 2:

Figure 1: Solution to the Generalised Modus Ponens problem.

This question deals with the problem of looping in backward-chaining proofs. The proof tree is shown in Figure 1.

The branch with Offspring\((Bluebeard, y)\) and Parent\((y, Bluebeard)\) repeats indefinitely, so the rest of the proof is never reached.

We get an infinite loop because of rule 2, Offspring\((x, y)\) ∧ Horse\((y)\) ⇒ Horse\((x)\).

The specific loop appearing in the figure arises because of the ordering of the clauses. We could be order Horse\((Bluebeard)\) before rule 2, which solve the problem of finding Horse\((Bluebeard)\) as a possible answer to the query.

2 Resolution

From “Horses are animals” it follows that “The head of a horse is the head of an animal”. Demonstrate that this inference is valid by carrying out the following steps:
1. Translate the premise and the conclusion into the language of First-Order Logic. Use three predicates: \textit{HeadOf}(h, x) (meaning “h is the head of x”), \textit{Horse}(x), and \textit{Animal}(x).

2. Negate the conclusion, and convert the premise and the negated conclusion into Conjunctive Normal Form.

3. Use resolution to show that the conclusion follows from the premises.

\textbf{Answer}

(1):
\begin{align*}
\forall x. & \text{Horse}(x) \Rightarrow \text{Animal}(x) \\
\forall x, h. & \text{Horse}(x) \land \text{HeadOf}(h, x) \Rightarrow \exists y. \text{Animal}(y) \land \text{HeadOf}(h, y)
\end{align*}

(2):
\begin{align*}
\text{A:} & \neg \text{Horse}(x) \lor \text{Animal}(x) \\
\text{B:} & \text{Horse}(G) \\
\text{C:} & \text{HeadOf}(H, G) \\
\text{D:} & \neg \text{Animal}(y) \lor \neg \text{HeadOf}(H, y)
\end{align*}

Here A comes from the first sentence in (1), while the others come from the second. H and G are Skolem constants.

(3):
Resolve D and C to yield \neg \text{Animal}(G). Resolve this with A (in (2) above to give \neg \text{Horse}(G). Resolve this with B to obtain a contradiction.

3 \hspace{1em} \textbf{Situation Calculus}

Last week you learnt about the frame problem and you were shown how it can be fixed by adding frame axioms.

Consider the following predicates and functions:

1. \textit{At}(sq, s) means that the agent is at square \textit{sq} in situation \textit{s}.
2. \textit{Heading}(dir, s) means that the agent is facing in direction \textit{dir} in situation \textit{s}.
3. \textit{Next}(sq1, dir, sq2) means that square \textit{sq2} is adjacent to square \textit{sq1} in direction \textit{dir}.
4. \textit{Result}(act, s) is the situation resulting from executing the action \textit{act} in situation \textit{s}.
5. \textit{Turn}(x) is the action of turning \textit{x} where \textit{x} \in \{\textit{left}, \textit{right}\}.
6. \textit{Shoot} is the action of shooting once forward.
7. $\text{Newdir}(\text{dir}_1, x, \text{dir}_2)$ means that $\text{dir}_2$ is the new direction the agent will face if it is facing in direction $\text{dir}_1$ and turns $x \in \{\text{left}, \text{right}\}$.

8. $\text{Wumpus}(\text{sq}, s)$ means that the Wumpus is in square $\text{sq}$ in situation $s$.

In the following we assume that the action $\text{Shoot}$ only has an effect in directly adjacent squares.

a) Formalise a precondition and an effect axiom for the Wumpus World that best describes the action $\text{Turn}(x)$.

b) Formalise a precondition and an effect axiom that best describes the $\text{Shoot}$ action in the Wumpus World.

c) Formalise a frame axiom that best describes the $\text{Shoot}$ action in the Wumpus World. You only need to do this for the $\text{Wumpus}$ fluent.

**Answer**

Note that, the formalisation of actions using precondition and effect axioms differs from the lecture notes, where only one axiom is used. Alternative effect axioms, in the form used in the lecture notes, are given below.

- Preconditions describe the fluents that must hold for an action to be possible.
- Effects describe the fluents that will hold as a result of taking the action.
- Frame axioms state what doesn’t change as a result of taking an action.

In the following axioms universal quantifiers (whose scope is the entire sentence) are omitted.

a) If the agent is heading in direction $\text{dir}_1$ and the result of turning $x$ is $\text{dir}_2$ then the agent is heading in direction $\text{dir}_2$ in the situation following after turning $x$.

- **Precondition:** $\text{Heading}(\text{dir}_1, s) \land \text{Newdir}(\text{dir}_1, x, \text{dir}_2) \Rightarrow \text{Poss}(\text{Turn}(x), s)$
- **Effect:** $\text{Poss}(\text{Turn}(x), s) \Rightarrow \text{Heading}(\text{dir}_2, \text{Result}(\text{Turn}(x), s))$

Alternatively:

$\text{Heading}(\text{dir}_1, s) \land \text{Newdir}(\text{dir}_1, x, \text{dir}_2) \Rightarrow \text{Heading}(\text{dir}_2, \text{Result}(\text{Turn}(x), s))$

b) If the agent is at square $\text{sq}_1$ and heading in direction $\text{dir}$ and the next square in direction $\text{dir}$ is $\text{sq}_2$ then the result of shooting will be that the wumpus is not in square $\text{sq}_2$ (if it was there then it’s dead).

- **Precondition:** $\text{At}(\text{sq}_1, s) \land \text{Heading}(\text{dir}, s) \land \text{Next}(\text{sq}_1, \text{dir}, \text{sq}_2) \Rightarrow \text{Poss}(\text{Shoot}, s)$
• Effect: $\text{Poss}(\text{Shoot}, s) \Rightarrow \neg \text{Wumpus}(sq2, \text{Result}(\text{Shoot}, s))$

Alternatively:

$\text{At}(sq1, s) \land \text{Heading}(dir, s) \land \text{Next}(sq1, dir, sq2) \Rightarrow \neg \text{Wumpus}(sq2, \text{Result}(\text{Shoot}, s))$

c) If the agent is at square $sq1$ and heading in direction $dir$ and the next square in direction $dir$ is $sq2$ and the wumpus is in square $sq3$ and square $sq2$ doesn’t equal $sq3$ then the Wumpus is still in square $sq3$ after shooting.

$\text{At}(sq1, s) \land \text{Heading}(dir, s) \land \text{Next}(sq1, dir, sq2) \land \text{Wumpus}(sq3, s) \land sq2 \neq sq3 \Rightarrow \text{Wumpus}(sq3, \text{result}(\text{Shoot}, s))$