AI-KR Home Assignment II: Penguins and Paint

Solutions to Lars Karlsson.

January 23, 1998

These are a number of exercises on circumscription. before trying them out, read the article on circumscription. There will also be a lecture including lecture notes on circumscription. You should also take a look at the DLS software at `http://www.ida.liu.se/labs/kplab/projects/dls/`, which takes a circumscriptive theory (or an arbitrary second-order formula) and reduces it to an equivalent first-order formula if possible. DLS can be very useful in verifying your results, and to test different choices of what to minimize, vary and fix.

1 Example

To illustrate how you should go about, we use a theory that tries to express that "dogs normally don't climb trees".

$$
\begin{aligned}
&\Gamma(Ab, Climbs) = \\
&\quad \{\ \forall x. Dog(x) \wedge \neg Ab(x) \Rightarrow \neg Climbs(x), \\
&\quad Dog(Rabalder), \\
&\quad Climbs(Rabalder), \\
&\quad Dog(Pluto), \\
&\quad \forall x. x = Rabalder \vee x = Pluto, \\
&\quad Rabalder \neq Pluto\ \}
\end{aligned}
\tag{1}
$$

Eventually, we should be able to deduce that $\neg Climbs(Pluto)$ and $Climbs(Rabalder)$.

1.1 Minimizing, varying and fixing

First, we have to determine what predicates to miminize and vary. Obviously, we want to minimize abnormality (Ab). The predicate $Climbs$ is dependent of Ab (abnormal dogs don't climb trees) so it should be varied. Regarding Dog, that should not depend on Ab, so that should be fixed. Thus, we determine to minimize Ab, vary $Climbs$ and hold everything else fixed.

1.2 Applying circumscription schema

Apply the circumscription schema to the theory:

$$CIRC(\Gamma; Ab; Climbs) = \tag{2}$$

$$\begin{aligned}
&\Gamma(Ab, Climbs) \wedge \\
&\forall Ab', Climbs'.\neg[\Gamma(Ab', Climbs') \wedge Ab' < Ab]
\end{aligned}$$

We use $P' < P$ as an abbreviation for $\forall x[P'(x) \Rightarrow P(x)] \wedge \neg\forall x[P(x) \Rightarrow P'(x)]$.
Writing this out, we get:

$$CIRC(\Gamma; Ab; Climbs) = \tag{3}$$

$$\begin{aligned}
&\forall x.Dog(x) \wedge \neg Ab(x) \Rightarrow \neg Climbs(x) \wedge \\
&Dog(Rabalder) \wedge Climbs(Rabalder) \wedge \\
&Dog(Pluto) \wedge \\
&\forall x.x = Rabalder \vee x = Pluto \wedge \\
&Rabalder \neq Pluto \wedge \\
&\forall Ab', Climbs'. \qquad (4)\\
&\qquad [\forall x.Dog(x) \wedge \neg Ab'(x) \\
&\qquad\ \ \Rightarrow \neg Climbs'(x) \wedge \\
&\qquad Dog(Rabalder) \wedge Climbs'(Rabalder) \wedge \\
&\qquad Dog(Pluto) \wedge \\
&\qquad \forall x.x = Rabalder \vee x = Pluto \wedge \\
&\qquad Rabalder \neq Pluto \wedge \\
&\qquad \forall x.Ab'(x) \Rightarrow Ab(x)] \Rightarrow \\
&\qquad \forall x.Ab(x) \equiv Ab'(x)
\end{aligned}$$

(See equation (16) in *Lecture Notes 7*)

1.3 Find substitution

Next, we should try to find proper predicate expressions to substitute for our predicate variables, thus getting back to a first-order theory. When doing this manually, our common sense in order to come up with a good solution. Actually, we cannot always be sure that there is a first-order equivalent! The minimized predicates should be as small as possible:

$$Ab'(x) = (\lambda x.x = Rabalder) \tag{5}$$

The varied predicates should satisfy the theory:

$$Climbs'(x) = (\lambda x.\neg Dog(x) \vee x = Rabalder) \tag{6}$$

(Pedagogical choice. This is actually equivalent to $(\lambda x.x = Rabalder)$ due to the domain closure.)
Notice that with DLS, we can sometimes automatically obtain the first-order equvalent theory.

1.4 Insert substitution

Do the substitutions and simplify.

$$
\begin{aligned}
&CIRC(\Gamma; Ab; Climbs) = \\
&\quad \Gamma(Ab, Climbs) \wedge \\
&\quad [\forall x.Dog(x) \wedge \neg(x = Rabalder) \\
&\quad\quad \Rightarrow \neg(\neg Dog(x) \vee x = Rabalder) \wedge \\
&\quad Dog(Rabalder) \wedge \\
&\quad (\neg Dog(Rabalder) \vee Rabalder = Rabalder) \wedge \\
&\quad Dog(Pluto) \wedge \\
&\quad \forall x.x = Rabalder \vee x = Pluto \wedge \\
&\quad Rabalder \neq Pluto \wedge \\
&\quad \forall x.(x = Rabalder) \Rightarrow Ab(x)] \Rightarrow \\
&\quad \forall x.Ab(x) \equiv (x = Rabalder)
\end{aligned} \tag{7}
$$

$$
\begin{aligned}
= \;& \Gamma(Ab, Climbs) \wedge \\
& [\; T \wedge Dog(Rabalder) \wedge \\
& \;\; T \wedge Dog(Pluto) \wedge \\
& \;\; \forall x.x = Rabalder \vee x = Pluto \wedge \\
& \;\; Rabalder \neq Pluto \wedge \\
& \;\; Ab(Rabalder)] \Rightarrow \\
& \;\; \forall x.Ab(x) \equiv (x = Rabalder)
\end{aligned} \tag{8}
$$

$$
= \; \Gamma(Ab, Climbs) \wedge \forall x.Ab(x) \equiv (x = Rabalder)
$$

(Everything else is already in $\Gamma(Ab, Climbs)$)

Now we have a first-order version of our circumscribed theory, from which we can deduce $\neg Climbs(Pluto)$ and $Climbs(Rabalder)$.

1.5 Summary of how to do it!

1. Select what predicates to minimize, vary and hold fixed.

2. Apply the circumscription scheme:

$$
\begin{aligned}
&CIRC(\Gamma; P; S) = \\
&\quad \Gamma(P, S) \wedge \\
&\quad \forall P', S'.\neg[\Gamma(P', S') \wedge P' < P]
\end{aligned} \tag{9}
$$

3. Find proper substitutions:

$$P' = (\lambda x.\alpha(x)) \tag{10}$$

$$S' = (\lambda x.\beta(x)) \tag{11}$$

4. Insert substitutions and simplify! If you are lucky, you end up with something like $\forall x.P(x) \equiv (x = a_1 \vee \ldots \vee x = a_n)$.

1.6 Using DLS

We can use the DLS algorithm (on the WWW at `http://www.ida.liu.se/labs/kplab/projects/dls/circ.html`) to verify our results. Insert the following formula:

```
all x ( (Dog(x) & -Ab(x)) -> -Climbs(x)) &
Dog(rabalder) &
Climbs(rabalder) &
Dog(pluto) &
all x (x=rabadler | x=pluto) &
rabalder = pluto
```

Chose `Ab` to be minimized and `Climbs` to vary, and press the **Submit** button. The following first-order formula pops out.

```
all y
      (
        -Ab(y)
        |
        -Ab(rabalder)
        |
          (Dog(y) & y = rabalder)
        )
```

This can be simplified to $\forall y[\neg Ab(y) \vee y = rabalder]$, and as we can also infer $Ab(Rabalder)$ from the original theory, using proof by contradiction, we have $\forall y[Ab(y) \equiv y = rabalder]$.

2 Exercises

When solving the exercises, write out each step clearly. You don't have to do complete natural deduction proofs, but don't be too short either.

1. Let the theory Γ consist of:

$$\forall x.P(x) \equiv \neg Q(x) \tag{12}$$
$$P(a) \tag{13}$$

   Use parallel circumscription to minimize P while varying Q.

2. Let the theory Γ consist of:

$$Bird(Tweety) \tag{14}$$
$$\forall x.Penguin(x) \Rightarrow Bird(x) \wedge \neg Flies(x) \tag{15}$$
$$\forall x.Bird(x) \wedge \neg Ab(x) \Rightarrow Flies(x) \tag{16}$$

   Show that using parallel circumscription to minimize Ab, we can infer that

$$Flies(Tweety). \tag{17}$$

A key decision in this problem is whether the *Penguin* predicate should be varied or fixed. Actually, the sentence above is correct only in one of these two cases. Motivate your decision. Hint: think in terms of models, and what models are compared when minimizing.

Having made the correct decision, you will also be able to show that

$$\forall x . \neg Penguin(x). \tag{18}$$

3. This is a peek at the PMON logic, which you will see more of in the later modules. We use two predicates: $Holds(t, f)$ denoting that a feature f holds at a time-point t; and $Occlude(t, f)$ which is an abnormality predicate. *Occlude* represents that the default behavior of f (that is keep its truth value from the previous time-point) does not apply to time-point t. Instead, f might get a new truth value. Thus, if we want to say that a feature f becomes true at time-point t due to some action, then we write $Holds(t, f) \wedge Occlude(t, f)$.

   Now assume that an agent called Fred is painting a box green from time-point 1 to 2. Thus, if Fred has some green color in his possesion, then the box will become green. We can formulate this as follows.

$$\begin{array}{l} Holds(1, \textit{has-green-color}) \Rightarrow \\ \quad Holds(2, \textit{box-is-green}) \wedge Occlude(2, \textit{box-is-green}) \end{array} \tag{19}$$

   What is the result of using circumscription to minimize *Occlude* while keeping *Holds* fixed? Try to simplify as much as possible.

   After minimizing *Occlude*, we add the so-called no-change premiss:

$$\forall t, f[\neg Occlude(t+1, f) \Rightarrow (Holds(t, f) \equiv Holds(t+1, f))] \tag{20}$$

   This premiss states that is f is not occluded at time-point $t+1$, then f must have the same truth value there as at time-point t. That is to say, unoccluded features persist.

   We will also add the following observation *after* cirumscribing.

$$\begin{array}{l} \neg Holds(0, \textit{box-is-green}) \wedge Holds(0, \textit{has-green-color}) \wedge \\ Holds(0, \textit{sun-is-red}) \end{array} \tag{21}$$

   Taking the circumscribed version of (19) and then adding (20) and (21), what can we prove regarding what holds and doesn't hold at time-point 3?

   To summarize, you should

   (a) do the circumscription of (19), and

   (b) use that (including the original theory) and (20) and (21) to infer what holds at 3.