Exercise 6: Comprehensive CPS Correctness

1 Uniform substitution

Problem 1. *Give the result of applying the uniform substitution rule US with substitution*

$$\sigma = \{a \mapsto \{x'' = -g \& x \geq 0\}, b \mapsto ?x = 0; v := -cv, p(\bar{x}) \mapsto 2gx \leq 2gH - v^2\}$$

to

1. $[a \cup b]p(\bar{x}) \leftrightarrow [a]p(\bar{x}) \land [b]p(\bar{x})$
2. $[a; b]p(\bar{x}) \leftrightarrow [a][b]p(\bar{x})$
3. $[a^*]p(\bar{x}) \leftrightarrow p(\bar{x}) \land [a][a^*]p(\bar{x})$
4. $[a^*]p(\bar{x}) \leftrightarrow p(\bar{x}) \land [a^*](p(\bar{x}) \to [a]p(\bar{x}))$

About admissible substitutions that bind free-ish variables: In items (2) to (4) the given uniform substitution binds free variables. Indeed applying σ to $[a][b]p(\bar{x})$ results in

$$[x'' = -g \& x \geq 0][?x = 0; v := -cv]2gx \leq 2gH - v^2.$$

Here the x introduced in $?x$ is free in $\sigma(b)$. Since we are applying the substitution σ in a context where x was bound (by $x'' = \ldots$), a free variable is introduced in a context where it is bound. What is more x is also free in $\sigma(p(\bar{x}))$, despite x becoming bound in a context in which $p(\bar{x})$ appears. Yet the uniform substitution is still admissible! This is worrying when you remember, that 'if you bind a free variable, you go to logic jail'. But what we this slogan means by free variables is not quite what we might guess. A good intuition is that the free variables of a substitution are the free variables of the right-hand side minus the free variables of the left-hand side. For instance, for $\sigma = \{p \mapsto x \geq 0\}$, we have $FV(\sigma) = FV(x \geq 0) \setminus FV(p) = \{x\} \setminus \emptyset$. Indeed, there are two mechanisms that allow a uniform substitution to bind (what looks like) free variables, which we explain in the following.

First recall the formal definition of an admissible substitution. A substitution σ is U-admissible for φ if $U \cap FV(\sigma_{\Sigma(\varphi)}) = \emptyset$. Perhaps counter-intuitively, the free variables of a substitution are defined as $FV(\sigma) = \bigcup_{f \in \sigma} FV(\sigma f(\cdot)) \cup \bigcup_{p \in \sigma} FV(\sigma p(\cdot))$. In particular in our example $FV(\sigma_a) = \emptyset$. Regarding our intuition, this is because $FV(a) = \mathcal{V}$ meaning all variables. Hence σ is vacuously

$\{x\}$-admissible for $?x = 0; v := -cv$ and the substitution does not clash at this point. This is one mechanism by which a substitution may introduce (something like) free variables in a context in which they are bound. A slogan might be: 'only variables introduced by substituting predicate or function symbols can cause a clash'.

This does not explain why the substitution does not clash when substituting $p(\bar{x})$. (Although this is probably the more familiar case.) As the notation $\bar{x}$ is an abbreviation for the list of all relevant variables, $FV(p(\bar{x})) = \mathcal{V}$, so our intuition says such a substitution has no free variable. Indeed, this is acceptable, since what σ really contains is the map $p(\tilde{x}, \tilde{g}, \tilde{H}, \tilde{v}) \mapsto 2\tilde{g}\tilde{x} \leq 2\tilde{g}\tilde{H}\tilde{v}^2$.[1] The actual variables x, g, H and v are then reintroduced when applying σ to $p(\bar{x})$, since

$$\begin{aligned}\sigma(p(\bar{x})) &= \sigma(p(x, g, H, v)) \\ &= \{\tilde{x} \mapsto \sigma(x), \tilde{g} \mapsto \sigma(g), \tilde{H} \mapsto \sigma(H), \tilde{v} \mapsto \sigma(v)\}(\sigma p(\tilde{x}, \tilde{g}, \tilde{H}, \tilde{v})).\end{aligned}$$

Hence $FV(\sigma_p) = \emptyset$. In particular x is not free here. So this substitution will not clash if applied to a formula in a context where x is bound. A slogan might be: 'variables mentioned as arguments of predicate and function symbols can not cause a clash'.

Problem 2. *Let p an arity 0 predicate symbol. Give a uniform substitution σ for which it is necessary for soundness that US clashes when being applied to*

$$p \to [a]p$$

A uniform substitution may be $\sigma = \{p \mapsto x \geq 0, a \mapsto x := -1\}$. It is soundness critical that this substitution clashes, for otherwise (since $p \to [a]p$ is an axiom) we could prove $x \geq 0 \to [x := -1]x > 0$. But this is invalid.

Problem 3. *Give the result of applying uniform substitution rule US with substitution $\sigma = \{c() \mapsto x - y, p(\cdot) \mapsto (\cdot \geq yz)\}$ on the following formulas or explain why and how US clashes:*

1. $[x := c()]p(x) \leftrightarrow p(c())$
2. $[z := c()]p(u) \leftrightarrow p(u)$
3. $[y := c()]p(y) \leftrightarrow p(c())$

For each formula, there is one admissibility condition which happens when substituting the box. This means σ should be $\{x\}$-admissible (resp. $\{z\}, \{y\}$-admissible) for $p(x)$ (resp. $p(u), p(y)$). In all cases, we only need to look at the free variables introduced by $\sigma(p(\cdot))$, that is $\{y, z\}$. Thus the substitution succeeds in the first case, and clashes in the latter two.

Problem 4. *Prove the following using uniform substitutions*

$$x = x_0 \land y = y_0 \to [x := x + y][y := x - y][x := x - y]x = y_0 \land y = x_0$$

[1] This is what the $\bar{x}$ notation means. Here the $\tilde{}$-variables are really 0-ary function symbols.

First we might try to cut in the following formula

$$[x := x + y][y := x - y][x := x - y]x = y_0 \wedge y = x_0$$
$$\leftrightarrow [y := x + y - y][x := x + y - y]x = y_0 \wedge y = x_0$$

and attempt to show the formula by deriving it from the assignment axiom

$$[x := c()]p(\bar{x}) \leftrightarrow p(c())$$

with the uniform substitution

$$\sigma = \{c() \mapsto x + y, p(\bar{x}) \mapsto [y := x - y][x := x - y]x = y_0 \wedge y = x_0\}.$$

However this can not work. The substitution will clash here, since we will need to replace the free occurrence x in $x := x - y$ by $x + y$. Since we are substituting a *function* symbol that mentions a free variable (x) that is bound in this position, this causes a problem. Instead what we need to do is cut in the instances of the assignment axiom going from the inside out. Then we can use the contextual equivalence proof rule to replace for example $[x := x - y]x = y_0 \wedge y = x_0$ by $x - y = y_0 \wedge y = x_0$ in context.

2 ModelPlex

Problem 5. *Consider a water tank with current water level x and maximum water level m. The controller is triggered after ε time units and can choose any flow $-1 \leq f \leq \frac{m-x}{\varepsilon}$.*

1. *Prove that the tank never overflows.*
2. *Find the monitor specification conjecture, i.e. what the monitor should be equivalent to.*
3. *Find the monitor specification for the water tank.*

When looking for a monitor, it is critical to consider what assumptions you make and what variables you choose to monitor. For a system on Earth, it is reasonable to assume that the gravitational constant g is positive. But remember that the more assumptions you add, the weaker the resulting proof is.

Monitoring intuitively answers the question: "Is the system's data consistant with my model?". If true, then the proof of safety for the model entails that the system is currently safe. Comparing the current value x^+ with the previous value x, this corresponds to the d$\mathcal{L}$ formula $\langle(\mathrm{ctrl};\mathrm{plant})^*\rangle x = x^+$ if considering multiple loops or $\langle\mathrm{ctrl};\mathrm{plant}\rangle x = x^+$ if you are monitoring every controller step. We call this formula the (model) monitor conjecture. It is true when the relation between x and x^+ can be explained with one iteration of the model, thus that the system is in a safe state.

For the water tank, we make the assumption that the time ϵ is non-zero to handle the division. The model monitor specification conjecture is then as follows:

```
<
  f:=*;?(-1<=f&f<=(m-x)/eps);
  t:=0;{x'=f,t'=1&x>=0&t<=eps}
>(x=xpost & t=tpost & f=fpost)
```

Once simplified to a first order formula (so that it can be checked in real time), we obtain the following monitor specification:

```
xpost = fpost*tpost + x & m >= x + fpost*eps & x >= 0 &
 fpost >= -1 & xpost >= 0 & tpost >= 0 & eps >= tpost
```

If we did not add the assumption about ϵ, we would need to change the specification, adding … `& eps != 0`. KeYmaera X is able to prove that the conjecture is "almost" equivalent to the stronger monitor. More precisely, the only struggle is with the division, i.e. proving `f<=(m-x)/eps -> eps != 0`. So the monitor is still sound.

Remark: A stronger controller can also be defined to check whether the relation between x and x^+ can be explained with one iteration of the controller, called *controller monitor*. Thus we might want to check the formula $\langle \text{ctrl} \rangle x = x^+$. However, this is not sound. Consider the following provable formula $x \geq 0 \to [(v := *; \{x' = v \& v \geq 0\})^*]x \geq 0$. The conjecture above is not sufficient to ensure safety as it is the domain of the ODE that prevents any negative value of v. A sound *controller* monitor conjecture for this model would be $\langle \text{ctrl;dom} \rangle x = x^+$ where "dom" is the domain of the plant, so the controller would be $\langle v := *; v \geq 0 \rangle (x = x^+ \wedge v = v^+)$.

3 Virtual substitution

Problem 6. *Normalize the following formulas*

1. $\exists x\, x^2 = a \vee b > 0$
2. $\exists x\, x^2 = a \vee x^2 > b$
3. $\exists x\, ((\exists y\, x^5 > y^4) \wedge \forall z\, (z + a \leq b \to z^5 \geq a))$

Problem 7. *Substitute the square root expression ε into the formula F, i.e. compute F_x^{ε}. Then find an equivalent formula in the language of real arithmetic.*

1. $F \equiv 3x^2 > 2x \wedge x \neq 3$ *and* $\varepsilon \equiv \sqrt{2}/2$
2. $F \equiv 4x^3 < a \wedge ax \neq 0$ *and* $\varepsilon = a + \sqrt{a}$

Problem 8. *Consider polynomials $p = ax^2 + bx + c$ and $q = dx + e$. When do $p = 0$ and $q = 0$ have simultaneous solutions?*

Problem 9. *Let p, q be as before. When does the system $p = 0, q > 0$ have a solution?*

Problem 10. *Use Virtual substitution to eliminate the quantifier in*

$$\exists x\, (ax + b \leq 0 \wedge cx - d < 0)$$