1. Let $a = [x = 1.0, f = \zeta(s : A)\ln(s.x)]$ and $b = [x = -1.0, f = \zeta(s1 : B)s1.x]$, where $A \equiv [x : \text{PosReal}, f : \text{Real}]$, and $B \equiv [x : \text{Real}, f : \text{Real}]$. We can easily see that $a : A$ and $b : B$. Note that $(a.x := b.x).f$ reduces to $\ln(-1.0)$ and that this term is not typable. However, with a (Sub Object/contravariant) rule we can prove the following judgement:

$$\emptyset \vdash (a.x := b.x).f : \text{Real}$$

Thus, we have shown that the addition of a (Sub Object/contravariant) rule results in an unsound system since we can type a term which produces a run-time error.

2. Typing $o \triangleright [l = \zeta(s : A)s]$ in $\text{OB}_1$ with variance annotations.

(a) The term $o.l.l$ will be typable for $A \equiv [l : [l^+ : \cdot \cdot \cdot ]]$. 

$$\emptyset \vdash o.l.l : []$$ (Val. Select)

(b) The term $o.l \cdots l$ will be typable, with type $[]$, for a finite number of $l$'s, $n$, if $A \equiv [l : [l^+ : \cdot \cdot \cdot l^+ : \cdot \cdot \cdot ]]$. We will have a nesting of $n$ $l$'s with the all but the outermost $l$ annotated with a $\cdot$. The derivation will follow the same structure as in part (a).
3. Rewrite of Unify($E \cup \{t = \tau\}$) case.

Let $E' = \{t \to \text{nat} = \text{bool} \to s, t = \text{nat}\}$. This set of type equations is clearly not unifiable, but without applying the substitution to $E'$ we can have:

$$\text{Unify}(E') = \text{Unify}(\{t \to s = \text{bool} \to \text{nat}\} \cup \{t = \text{nat}\})$$

$$= \text{Unify}(\{t \to \text{nat} = \text{bool} \to s\}) \circ \{t \mapsto \text{nat}\}$$

$$= \text{Unify}(\{t = \text{bool}\} \cup \{s = \text{bool}\}) \circ \{t \mapsto \text{nat}\}$$

$$= \{t \mapsto \text{bool}\} \circ \{s \mapsto \text{bool}\} \circ \{t \mapsto \text{nat}\}$$

Thus, the Unify algorithm will not produce a unifying substitution if we do not apply $[t := \tau]$ to $E$ before the recursive call.

4. Unification:

(a) $S = \{v \mapsto t\} \circ \{w \mapsto u\} \circ \{r \mapsto v \to w\}$.

(b) If $b_1 \not\equiv b_2$ then fail. Otherwise, let $b \equiv b_1 \equiv b_2$ in $S = \{r \mapsto b\} \circ \{s \mapsto b\} \circ \{t \mapsto b\}$.

(c) Fails as $s \in \text{fv}(s \to s)$.

5. Type Inference:

(a) $\emptyset \vdash \lambda x : t_1.\lambda y : t_2.x : t_1 \to t_2 \to t_1$

(b) $\emptyset \vdash \lambda x : t_1 \to t_2 \to t_3.\lambda y : t_1 \to t_2.\lambda z : t_1.(xz)(yz) : (t_1 \to t_2 \to t_3) \to (t_1 \to t_2) \to t_1 \to t_3$

(c) $\emptyset \vdash \lambda f : t_1 \to t_1.\lambda x : t_1.f(fx) : (t_1 \to t_1) \to t_1 \to t_1$

Scott Russell, Santiago Pericas-G. Created 12.14.01