Appendix B Proofs of Example Bisimulations

Example 26. In Example 4, for any terms s, t, r, $\mathsf{seq}(\mathsf{seq}(s,t),r) \approx_{vc} \mathsf{seq}(s, \mathsf{seq}(t,r))$.

Proof. Let R denote the reflexive symmetric closure of

$$\{(\mathsf{seq}(\mathsf{seq}(s,t),r), \mathsf{seq}(s,\mathsf{seq}(t,r)))\} \cup \{(\mathsf{seq}(\mathsf{seq}(s,\mathsf{skip}),r), \mathsf{seq}(s,r))\}.$$

We claim that R is a vc-bisimulation. We show conditions (1) and (2) by joint induction on the proof tree.

1. Suppose pRq and $p \xrightarrow{a} p'$.
   In the case that the final rule was saturation, if $p \Rightarrow p_1 \xrightarrow{a} p_2 \Rightarrow p'$ then by inductive hypothesis $q \Rightarrow q_1$ with $p_1 R q_1$, so $q_1 \xrightarrow{a} q_2$ with $p_2 R q_2$ and $q_2 \Rightarrow q'$ with $q' R q'$. By applying the saturation rule, $q \xrightarrow{a} q'$, as required.
   Otherwise, if $p = q$ then the case holds trivially, setting $q' = p'$.
   Otherwise, suppose $p = \mathsf{seq}(\mathsf{seq}(s,t),r)$ and $q = \mathsf{seq}(s,\mathsf{seq}(t,r))$. Then the only remaining applicable rule for $\rightarrow$ is the seq-defining rule for $\rightarrow$, we must have $\mathsf{seq}(s,t) \xrightarrow{a} u$ and $p' = \mathsf{seq}(u,r)$. Now consider the proof that $\mathsf{seq}(s,t) \xrightarrow{a} u$. If the final rule is through saturation, we may proceed inductively as in the case above. Otherwise, the last rule applied must have been the seq-defining rule for $\rightarrow$ and we must have $s \xrightarrow{a} s'$ and $u = \mathsf{seq}(s',t)$ so $p' = \mathsf{seq}(\mathsf{seq}(s',t),r)$. Now, since $s \xrightarrow{a} s'$ we may apply the seq-defining $\rightarrow$ rule and conclude that $q \rightarrow \mathsf{seq}(s', \mathsf{seq}(t,r))$. Setting $q' = \mathsf{seq}(s',\mathsf{seq}(t,r))$ concludes the case, since then $p' R q'$.
   Otherwise, suppose $p = \mathsf{seq}(s,\mathsf{seq}(t,r))$ and $q = \mathsf{seq}(\mathsf{seq}(s,t),r)$. Then the only remaining applicable rule for $\rightarrow$ is the seq-defining rule for $\rightarrow$, we must have $s \xrightarrow{a} s'$ and $p' = \mathsf{seq}(s',\mathsf{seq}(t,r))$. Now, we may apply the seq-defining $\rightarrow$ rule and conclude that $\mathsf{seq}(s,t) \xrightarrow{a} \mathsf{seq}(s',t)$ and again to conclude that $q \rightarrow q'$ for $q' = \mathsf{seq}(\mathsf{seq}(s',t),r)$. But then $p' R q'$, and so we are done.
   Otherwise, suppose $p = \mathsf{seq}(\mathsf{seq}(s,\mathsf{skip}),r)$ and $q = \mathsf{seq}(s,r)$. Then the only remaining applicable rule for $\rightarrow$ is the seq-defining rule for $\rightarrow$, we must have $\mathsf{seq}(s,\mathsf{skip}) \xrightarrow{a} u$ and $p' = \mathsf{seq}(u,r)$. The only possibility is saturation with $\mathsf{seq}(s,\mathsf{skip}) \Rightarrow s_1 \xrightarrow{a} s'$. By induction on the $\Rightarrow$ derivation it is easy to see that $s_1 = s$. Now, since $s \xrightarrow{a} s'$ we may apply the seq-defining $\rightarrow$ rule and conclude that $q \rightarrow \mathsf{seq}(s',r)$. We are done by reflexivity of R.
   Finally, suppose $p = \mathsf{seq}(s,r)$ and $q = \mathsf{seq}(\mathsf{seq}(s,\mathsf{skip}),r)$. Then the only remaining applicable rule for $\rightarrow$ is the seq-defining rule for $\rightarrow$, we must have $s \xrightarrow{a} s'$ and $p' = \mathsf{seq}(s',r)$. Then $\mathsf{seq}(s,\mathsf{skip}) \xrightarrow{a} \mathsf{seq}(s',\mathsf{skip})$ and $q \xrightarrow{a} q' = \mathsf{seq}(\mathsf{seq}(s',\mathsf{skip}),r)$. But then $p' R q'$, as required.
2. Suppose pRq and $p \Rightarrow p'$.
   If the last rule used is the reflexivity rule for $\Rightarrow$, we have $p = p'$ and we can take $q = q'$, as then $q \Rightarrow q'$ and $p' R q'$.
   If the last rule used is the transitivity rule for $\Rightarrow$, we proceed as for the saturation case for $\rightarrow$ using the inductive hypothesis.
   If the last rule used is for precongruence, then there are a few possibilities. If $p = q$ then we are done trivially taking $q' = p'$. If $p = \mathsf{seq}(\mathsf{seq}(s,t),r)$

precongruence in the second argument is used, then $r \Rightarrow r'$ we may take $q' = \mathsf{seq}(\mathsf{seq}(s,t),r')$. Then $p'Rq'$ and $q \Rightarrow q'$ using precongruence. If precongruence in the first argument is used, then $\mathsf{seq}(s,t) \Rightarrow u$. If this is due to reflexivity or transitivity we may proceed by induction as above, and if it is due to precongruence in s or t we may proceed as in the previous subcase, mirroring the transition in q. The final possibility is that $s = \mathsf{skip}$ and $p' = \mathsf{seq}(t,r)$. In this case $q \Rightarrow q' = \mathsf{seq}(t,r)$ by applying the seq-defining rule for $\Rightarrow$. Then $p'Rq'$ by reflexivity of R.
If $p = \mathsf{seq}(s, \mathsf{seq}(t,r))$ and precongruence is used with $s \Rightarrow s'$, $t \Rightarrow t'$ or $r \Rightarrow r'$ then we can proceed as in the previous case. Finally, if $\mathsf{seq}(t,r) \Rightarrow u$ by the seq-defining $\Rightarrow$ rule, we have $t = \mathsf{skip}$ and $u = r$ and $p' = \mathsf{seq}(s,r)$. Then $q \Rightarrow q$ by reflexivity, and $p'Rq$.
If $p = \mathsf{seq}(s,r)$ and $q = \mathsf{seq}(\mathsf{seq}(s,\mathsf{skip}),r)$ and precongruence is used with $s \Rightarrow s'$ or $r \Rightarrow r'$ we can proceed as above. If $s = \mathsf{skip}$ and $p' = r$ then $q \Rightarrow r$ also, and rRr by reflexivity of R. If $p = \mathsf{seq}(\mathsf{seq}(s,\mathsf{skip}),r)$ and $q = \mathsf{seq}(s,r)$ and precongruence is used with $s \Rightarrow s'$ or $r \Rightarrow r'$ we can proceed as above. If $s = \mathsf{skip}$ and $p' = \mathsf{seq}(\mathsf{skip},r)$ then $q \Rightarrow q$ and $p'Rq$ by reflexivity.
The final case is that $p \Rightarrow q$ by a seq-defining $\Rightarrow$ rule. If $p = q$ then we are done trivially taking $q' = p'$. If $p = \mathsf{seq}(s,\mathsf{seq}(t,r))$ then we must have $s = \mathsf{skip}$ and $p' = \mathsf{seq}(t,r)$ and then $q = q' = \mathsf{seq}(\mathsf{seq}(skip,t),r) \Rightarrow p'$ with $p'Rq'$ by reflexivity. If $p = \mathsf{seq}(s,r)$ then $s = \mathsf{skip}$ and $p = r$. Then $q = \mathsf{seq}(\mathsf{seq}(s,\mathsf{skip}),r) \Rightarrow r$ using precongruence, transitivity and the seq-defining $\Rightarrow$ rule. Finally, rRr by reflexivity. The other cases cannot apply, as in such cases p does not match the source of the conclusion of the seq-defining $\Rightarrow$ rule.

3. The third condition never arises, since no entries in R are values. □

From this we can use the congruence result to show that $\mathsf{thunk}(\mathsf{seq}(\mathsf{seq}(s,t),r)) \approx_{vc} \mathsf{thunk}(\mathsf{seq}(s,\mathsf{seq}(t,r)))$, although it can be shown directly by adding appropriate entries to our proposed bisimulation relation.

Example 27. In Example 4, $\mathsf{seq}(\mathsf{skip},s) \approx_{vc} s$ for each s.

Proof. Let R denote the reflexive symmetric closure of $\{(\mathsf{seq}(\mathsf{skip},s),s)\}$. We show that R is a vc-bisimulation.

1. Suppose sRt and $s \xrightarrow{a} s'$.
The case that $s = t$ is trivial.
Next, suppose $t = \mathsf{seq}(\mathsf{skip},s)$. Then $t \Rightarrow s \xrightarrow{a} s'$. By saturation, $t \xrightarrow{a} s'$. Since $s'Rs'$ we are done.
Finally, suppose $s = \mathsf{seq}(\mathsf{skip},s)$. The final rule in $s \xrightarrow{a} s'$ must be saturation, which can be dealt with generically.
2. Suppose $s \Rightarrow s'$ and sRt.
If $s = t$ the case is trivial.
If $t = \mathsf{seq}(\mathsf{skip},s)$ then $t \Rightarrow s \Rightarrow s'$ so take $t' = s'$ and we are done by reflexivity of $\Rightarrow$.
If $s = \mathsf{seq}(\mathsf{skip},t)$ then we consider the proof $s \Rightarrow s'$. Cases of transitivity, reflexivity and precongruence are dealt generically. The final case is that $s' = t$, in which case take $t' = t$ and use reflexivity of $\Rightarrow$.

3. Suppose sRt where s is a value. Then either $t = s$ (in which case we are done trivially using reflexivity of R and $\Rightarrow$) or $t = \mathsf{seq}(\mathsf{skip}, s)$, since $\mathsf{seq}(\mathsf{skip}, t)$ is not a value. Then $t \Rightarrow s$, and we are done by reflexivity of R and $\Rightarrow$. □

The above arguments can also show the corresponding modular bisimulations in the MSOS setting. We have formalised them in Coq along with the right unit law for seq, at `cs.swan.ac.uk/~csmdc/bisim-examples.v`. In the remaining examples, we formulate and prove modular MSOS bisimulations.

Example 28. In any disjoint extension of $\mathbf{E}_{\mathsf{catch},\mathsf{print}}$, the following equation holds:

$$\mathsf{catch}(\mathsf{print}(v), x) \approx_{msos} \mathsf{print}(v)$$

Proof. Let R be the reflexive symmetric closure of $\{(\mathsf{catch}(\mathsf{print}(v), x), \mathsf{print}(v))\} \cup \{(\mathsf{catch}(\mathsf{skip}, x), \mathsf{skip})\}$. We show that R is an MSOS bisimulation:

1. Suppose $s \xrightarrow{L} s'$ and $s\ R\ t$. Reflexivity of R can be dealt with in a generic manner.
   If sRt and $s \xrightarrow{L} s'$ using a built-in rule, we can proceed as in Example 26 formally using an induction over the proof derivation. The only remaining cases are the new saturation rules in Definition 11.
   In the case of the read-saturation rule, suppose $L = \{\mathbf{x} = x, \ldots\}$ and $x \Rightarrow z$ and $s \xrightarrow{\{\mathbf{x}=z,\ldots\}} s'$. Then by induction $t \xrightarrow{\{\mathbf{x}=z,\ldots_1\}} t'$ with $s'Rt'$ and $\mathsf{writes}(\ldots)\ R\ \mathsf{writes}(\ldots_1)$ and so by applying the same read-saturation rule $t \xrightarrow{\{\mathbf{x}=x,\ldots_1\}} t'$ as required.
   In the case of the write-saturation rule, suppose $L = \{\mathbf{x}' = x, \ldots\}$ and $s \xrightarrow{\{\mathbf{x}'=z,\ldots\}} s'$ and $z \Rightarrow x$. Then by induction $t \xrightarrow{\{\mathbf{x}'=z_1,\ldots_1\}} t'$ with $s'\ R\ t'$, $z\ R\ z_1$ and $\mathsf{writes}(\ldots)\ R\ \mathsf{writes}(\ldots_1)$. Since $z\ R\ z_1$ and $z \Rightarrow x$, by induction $z_1 \Rightarrow x_1$ with $x\ R\ x_1$. By applying the write-saturation rule, $t \xrightarrow{\{\mathbf{x}'=x_1,\ldots_1\}} t'$. Since $\mathsf{writes}(L)\ R\ \mathsf{writes}(\{\mathbf{x}' = x_1, \ldots_1\})$, we are done.
   If s proceeds using a construct-specific rule and $s = \mathsf{print}(v)$ then $s' = \mathsf{skip}$ and L is unobservable other than $L(\mathsf{output}') = \mathsf{cons}(v, \mathsf{nil})$. Then $t = \mathsf{catch}(\mathsf{print}(v), x)$ and $t \xrightarrow{L} \mathsf{catch}(\mathsf{skip}, x) = t'$, and $\mathsf{skip}\ R\ t'$.
   If s proceeds using a construct-specific rule and $s = \mathsf{catch}(\mathsf{print}(v), x)$ then L is unobservable other than $L(\mathsf{output}') = \mathsf{cons}(v, \mathsf{nil})$ and s' is $s' = \mathsf{catch}(\mathsf{skip}, x)$ or $s' = \mathsf{skip}$ (using saturation). Then $t = \mathsf{print}(v) \xrightarrow{L} \mathsf{skip} = t'$. Then $s'Rt'$.
2. If $s\ R\ t$ and $s \Rightarrow s'$ then we consider $\Rightarrow$. Instances of precongruence (in v or x) can be matched in t, and reflexivity and transitivity can be dealt with in the standard manner. Otherwise, we must have $s = \mathsf{catch}(\mathsf{skip}, x)$ and $s' = \mathsf{skip}$. But then $t = \mathsf{skip}$, and we are done using reflexivity of R.
3. If $v\ R\ s$ then $v = \mathsf{skip}$, $s = \mathsf{catch}(\mathsf{skip}, x)$ and so $s \Rightarrow v$ as required. □

Example 29. In any disjoint extension of $\mathbf{E}_{\mathsf{seq},\mathsf{atomic},\mathsf{assign},\mathsf{deref}}$, the following holds:

$$\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x}, v), \mathsf{deref}(\mathsf{x}))) \approx_{msos} \mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x}, v), v))$$

Proof. Let R denote the reflexive symmetric closure of

$$\{(\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),\mathsf{deref}(\mathsf{x}))),\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),v)))\}.$$

We show that R is an MSOS bisimulation.

1. If $s\xrightarrow{L}s'$ and $s=\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),\mathsf{deref}(\mathsf{x}))$ a proof search finds the following proof derivation:

$$\dfrac{\dfrac{\dfrac{\mathsf{assign}(\mathsf{x},\ v)\xrightarrow{L_1}\mathsf{skip}}{\mathsf{seq}(\mathsf{assign}(\mathsf{x},\ v),\ \mathsf{deref}(\mathsf{x}))\xrightarrow{L_1}\mathsf{seq}(\mathsf{skip},\ \mathsf{deref}(\mathsf{x}))}\qquad \mathsf{seq}(\mathsf{skip},\ \mathsf{deref}(\mathsf{x}))\Rightarrow\mathsf{deref}(\mathsf{x})}{\mathsf{seq}(\mathsf{assign}(\mathsf{x},\ v),\ \mathsf{deref}(\mathsf{x}))\xrightarrow{L_1}\mathsf{deref}(\mathsf{x})}\qquad \dfrac{\mathsf{deref}(\mathsf{x})\xrightarrow{L_2}v\qquad \mathsf{atomic}(v)\xrightarrow{-}v}{\mathsf{atomic}(\mathsf{deref}(\mathsf{x}))\xrightarrow{L_2}v}}{\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x},\ v),\ \mathsf{deref}(\mathsf{x})))\xrightarrow{L}v}$$

   where: $\mathsf{reads}(L_1)=\mathsf{reads}(L)$, L_1 is unobservable except $\mathsf{writes}(L_1)(\mathbf{store}')=\mathsf{update}(L(\mathsf{store}),\mathsf{x},v)$, L_2 is unobservable with with (L_1,L_2) composable and L their composition, so L is unobservable except $L(\mathbf{store}')=\mathsf{update}(L(\mathbf{store}),\mathsf{x},v)$ and $s'=v$. There are other derivations moving the skip rewrite, but these place the same restrictions on L and s'.

   The following proof shows that $\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),v))\xrightarrow{L}s'$ also:

$$\dfrac{\dfrac{\dfrac{\mathsf{assign}(\mathsf{x},v)\xrightarrow{L_1}\mathsf{skip}}{\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),v)\xrightarrow{L_1}\mathsf{seq}(\mathsf{skip},v)}\qquad \mathsf{seq}(\mathsf{skip},v)\Rightarrow v}{\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),v)\xrightarrow{L_1}v}\qquad \mathsf{atomic}(v)\xrightarrow{L_2}v}{\mathsf{atomic}(\mathsf{seq}(\mathsf{assign}(\mathsf{x},v),v))\xrightarrow{L}v}$$

   We are done by reflexivity of R.

   There are additional derivations that use the read- or write- saturation rule: entries in L can be rewritten. Each can be matched in the target derivation. Note that any rewrite in the **store** component is a pointwise rewrite for each variable. If the read/write-saturation rule rewrites the x entry between the assign and the deref, then $s'=v_1$ for some v_1 with $v\Rightarrow v_1$ and $L(\mathbf{store}')=\mathsf{update}(L(\mathbf{store}),\mathsf{x},v_1)$. Such behaviour can be simulated in the target derivation using (write-)saturation and congruence rules.

   The symmetric case proceeds similarly, by exhaustive analysis of the proofs of $s\xrightarrow{L}s'$.

   The reflexivity case is trivial.
2. The only $\Rightarrow$ case is reflexivity, which holds trivially.
3. The only value case is reflexivity, which holds trivially. □