NOTE: This document was updated on 13 May, 2013 with fixes to the substitution and application lemmas discovered while formalizing Progressive Types in the Coq Proof Assistant.

Definition[Active expressions]: An active expression ae is defined by
$ae := v_1 v_2 \mid c(v)$.

Lemma[AE]: For all closed expressions e, either

i. $e \in v$, or

ii. there exists E and ae such that $e = E[ae]$, or

iii. $e = E[\mathsf{err}\text{-}\omega]$.

Proof. By definition of e, v, and E. □

Lemma[Substitution]: If

1. $\Omega'; \Gamma, x : \tau' \vdash e : \tau$,
2. $\langle\rangle; \bullet \vdash v : \tau'$, and
3. $\Omega' \subseteq \Omega$,

then $\Omega; \Gamma \vdash e[x/v] : \tau$,

Proof. By induction on e, followed by induction on the typing derivation in each case. □

Lemma[Canonical Forms]: For $\Omega; \Gamma \vdash v : \tau$, if

- $\tau = \mathsf{Z}$, then $v = 0$.
- $\tau = \mathsf{N}$, then $v = \mathsf{n}$, as defined in Fig. 1.
- $\tau = \tau_1 \xrightarrow{\Omega'} \tau_2$, then $v = \lambda x : (\tau_1; \Omega').e$.

Proof. By induction on the typing derivation.

- Only T-Zero and T-Sub apply. T-Zero is immediate and T-Sub follows by induction.
- Only T-Num and T-Sub apply. T-Num is immediate and T-Sub follows by induction.
- Only T-Fun and T-Sub apply. T-Fun is immediate and T-Sub follows by induction.

□

Lemma[Transitivity of Subtyping]: If $\tau_1 \leqslant \tau_2$ and $\tau_2 \leqslant \tau_3$ then $\tau_1 \leqslant \tau_3$.

Proof. Suppose that $\tau_1 \leqslant \tau_2$ by S-12 and $\tau_2 \leqslant \tau_3$ by S-23.
We proceed by cases over S-12 and S-23 and by induction over the size of terms.

- Let S-12 = S-Bottom.
  Then $\tau_1 = \bot$ and $\tau_1 \leqslant \tau_3$ by S-Bottom.
- Let S-12 = S-Refl.
  Then $\tau_1 = \tau_2$, and so $\tau_1 \leqslant \tau_3$ by S-23.
- Let S-23 = S-Refl.
  Then $\tau_2 = \tau_3$, and so $\tau_1 \leqslant \tau_3$ by S-12.
- Let S-12 = S-Union-L
  Then $\tau_2 = \sigma_1 \cup \sigma_2$ and $\tau_1 \leqslant \sigma_1$.

- Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Bottom}$
  This case is not possible, as it requires $\tau_2 = \bot$, which is contradictory to the above assertion that $\tau_2 = \sigma_1 \cup \sigma_2$.
- Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Union\text{-}L}$
  Then $\tau_3 = \sigma_3 \cup \sigma_4$ and $\tau_2 \leqslant \sigma_3$.
  Then $\tau_1 \leqslant \tau_2$ and $\tau_2 \leqslant \sigma_3$. By the induction hypothesis $\tau_1 \leqslant \sigma_3$. So $\tau_1 \leqslant \tau_3$ by S-Union-L.
- Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Union\text{-}R}$
  This is similar to the case above.
- Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Union\text{-}Join}$
  Then $\tau_1 \leqslant \sigma_1$ and $\sigma_1 \leqslant \tau_3$. By induction $\tau_1 \leqslant \tau_3$.
- Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Arrow}$
  This case is not possible, as it requires $\tau_2 = \sigma_3 \xrightarrow{\Omega} \sigma_4$, which is contradictory to the above assertion that $\tau_2 = \sigma_1 \cup \sigma_2$.

- Let $\mathsf{S\text{-}12} = \mathsf{S\text{-}Union\text{-}R}$

  This is symmetric to the above case.

- Let $\mathsf{S\text{-}12} = \mathsf{S\text{-}Union\text{-}Join}$

  Then $\tau_1 = \sigma_1 \cup \sigma_2$ and $\sigma_1 \leqslant \tau_2$ and $\sigma_2 \leqslant \tau_2$.

  - Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Bottom}$
    Then $\tau_1 \leqslant \bot$. By inspection of the subtyping rules we can see that $\tau_1 = \rho_1 \cup \rho_2$, defined by

    $$\rho = \bot \mid \rho \cup \rho.$$

    We proceed by induction over the number n of $\cup$'s in τ_1.

    - $n = 1$
      Then $\tau_1 = \bot \cup \bot$, and so $\tau_1 \leqslant \tau_3$ by S-Union-Join.
    - $n > 1$.
      Then $\tau_1 = \rho_1 \cup \rho_2$. Clearly ρ_1 and ρ_2 contain fewer than n $\cup$'s. By our induction hypothesis, $\rho_1 \leqslant \tau_3$ and $\rho_2 \leqslant \tau_3$. Then $\tau_1 \leqslant \tau_3$ by S-Union-Join.
  - Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Union\text{-}L}$
    Then $\tau_3 = \sigma_3 \cup \sigma_4$ and $\tau_2 \leqslant \sigma_3$.
    Then $\tau_1 \leqslant \tau_2$ and $\tau_2 \leqslant \sigma_3$. By the induction hypothesis $\tau_1 \leqslant \sigma_3$. So $\tau_1 \leqslant \tau_3$ by S-Union-L.
  - Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Union\text{-}R}$
    This is similar to the case above.
  - Let $\mathsf{S\text{-}23} = \mathsf{S\text{-}Union\text{-}Join}$
    In this case $\tau_2 = \sigma_3 \cup \sigma_4$, with $\sigma_3 \leqslant \tau_3$ and $\sigma_4 \leqslant \tau_3$.
    We know that $\sigma_1 \leqslant \tau_2$ by some rule, call it S-Sig1, and that $\sigma_2 \leqslant \tau_2$ by S-Sig2. We will proceed by induction over the height h of the derivation for $\sigma_1 \leqslant \tau_2$ and $\sigma_2 \leqslant \tau_2$. Without loss of generality, we examine the derivation for $\sigma_1 \leqslant \tau_2$

  Case: $h = 1$
    In this case either S-Sig1 = S-Bottom or S-Sig1 = S-Refl.
    · If S-Sig1 is S-Bottom, then $\sigma_1 = \bot$ and $\sigma_1 \leqslant \tau_3$ by S-Bottom.
    · In the case of S-Refl, then $\sigma_1 = \tau_2$, and so $\sigma_{\leqslant}\tau_3$ by S-Union-Join.

  Case: $h > 1$
    In this case S-Sig1 can be S-Union-L, S-Union-R, S-Union-Join, or S-Arrow.
    · If S-Sig1 = S-Union-L, then $\sigma_1 \leqslant \sigma_3$ and $\sigma_3 \leqslant \tau_3$. We know that the height of the derivation to show $\sigma_1 \leqslant \sigma_3$ is $h - 1$. By the induction hypothesis we conclude that $\sigma_1 \leqslant \tau_3$.
    · The case for S-Union-R is symmetric to the above case.

- Consider the case where S-Sig1 is S-Union-Join. Then $\sigma_1 = \sigma' \cup \sigma''$ with $\sigma' \leqslant \tau_2$ and $\sigma'' \leqslant \tau_2$. The derivations to show that $\sigma' \leqslant \tau_2$ and $\sigma'' \leqslant \tau_2$ have height $\leq h-1$. So by the induction hypothesis $\sigma' \leqslant \tau_3$ and $\sigma'' \leqslant \tau_3$. Then by S-Union-Join $\sigma_1 \leqslant \tau_3$.
- The case where S-Sig1 = S-Arrow is impossible, because it would require $\tau_2 = \sigma' \xrightarrow{\Omega'} \sigma''$, which is contradictory to the above statement that $\tau_2 = \sigma_3 \cup \sigma_4$.

It is symmetric to show that $\sigma_2 \leqslant \tau_3$. Thus we conclude that by S-Union-Join $\tau_1 \leqslant \tau_3$.

- Let S-23 = S-Arrow

In this case, $\tau_2 = \sigma_3 \xrightarrow{\Omega} \sigma_4$ and $\tau_3 = \sigma_5 \xrightarrow{\Omega'} \sigma_6$ with $\sigma_5 \leqslant \sigma_3$, $\Omega \subseteq \Omega'$, and $\sigma_4 \leqslant \sigma_6$.

Recall that $\tau_1 = \sigma_1 \cup \sigma_2$. Because we know that $\sigma_1 \leqslant \tau_2$ and $\sigma_2 \leqslant \tau_2$, we can see by inspection of the subtyping rules that $\sigma_1, \sigma_2 = \rho$ with ρ defined by

$$\rho = \bot \mid \sigma' \xrightarrow{\Omega} \sigma'' \mid \rho \cup \rho.$$

We proceed by induction over the number n of $\cup$'s in τ_1.

- $n = 1$

There are three possible cases for the shape of σ_1 and σ_2. We address each case for σ_1 and show that $\sigma_1 \leqslant \tau_3$:

- $\sigma_1 = \bot$. By S-Bottom $\sigma_1 \leqslant \tau_3$.
- $\sigma_1 = \tau_2$ (so $\sigma_1 \leqslant \tau_2$ by S-Refl). Then because $\tau_2 \leqslant \tau_3$ clearly $\sigma_1 \leqslant \tau_3$.
- $\sigma_1 = \sigma_1' \xrightarrow{\Omega_1} \sigma_1''$, and $\sigma_1 \leqslant \tau_2$ by S-Arrow. This is similar to the case below where S-12 and S-23 are S-Arrow. So we conclude that $\sigma_1 \leqslant \tau_3$.

The cases to show that $\sigma_2 \leqslant \tau_3$ are similar. Having shown that $\sigma_1 \leqslant \tau_3$ and $\sigma_2 \leqslant \tau_3$ we conclude that $\tau_1 \leqslant \tau_3$.

- $n > 1$.

Then $\tau_1 = \rho_1 \cup \rho_2$ and ρ_1 and ρ_2 contain fewer than n $\cup$'s. By our induction hypothesis, $\rho_1 \leqslant \tau_3$ and $\rho_2 \leqslant \tau_3$. Then $\tau_1 \leqslant \tau_3$ by S-Union-Join.

- Let S-12 = S-Arrow

Then $\tau_1 = \sigma_1 \xrightarrow{\Omega} \sigma_2$ and $\tau_2 = \sigma_3 \xrightarrow{\Omega'} \sigma_4$ with $\sigma_3 \leqslant \sigma_1$, $\Omega \subseteq \Omega'$, and $\sigma_2 \leqslant \sigma_4$.

- Let S-23 = S-Bottom

This case is not possible, as it requires $\tau_2 = \bot$, which is contradictory to the above assertion that $\tau_2 = \sigma_3 \xrightarrow{\Omega} \sigma_4$.

- Let S-23 = S-Union-L

Then $\tau_3 = \sigma_3 \cup \sigma_4$ and $\tau_2 \leqslant \sigma_3$.

Then $\tau_1 \leqslant \tau_2$ and $\tau_2 \leqslant \sigma_3$. By the induction hypothesis $\tau_1 \leqslant \sigma_3$. So $\tau_1 \leqslant \tau_3$ by S-Union-L.

- Let S-23 = S-Union-R

This is similar to the case above.

- Let S-23 = S-Union-Join

This case is not possible, as it requires $\tau_2 = \sigma_5 \cup \sigma_6$, which is contradictory to the above assertion that $\tau_2 = \sigma_3 \xrightarrow{\Omega} \sigma_4$.

- Let S-23 = S-Arrow

Then $\tau_3 = \sigma_5 \xrightarrow{\Omega''} \sigma_6$, with $\sigma_5 \leqslant \sigma_3$, $\Omega' \subseteq \Omega''$, and $\sigma_4 \leqslant \sigma_6$.

So $\sigma_5 \leqslant \sigma_3$ and $\sigma_3 \leqslant \sigma_1$ so by the induction hypothesis, $\sigma_5 \leqslant \sigma_1$. Similarly $\sigma_2 \leqslant \sigma_4$ and $\sigma_4 \leqslant \sigma_6$ implies $\sigma_2 \leqslant \sigma_6$. By transitivity of the subset relation $\Omega \subseteq \Omega''$. Thus $\tau_1 \leqslant \tau_3$ by S-Arrow.

$\square$

Lemma[Application]: If $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \sigma$, $apply(\sigma, \tau', \Omega') = \tau''$, and $\tau' \not\leqslant \bot$, then $\tau_2 \leqslant \tau''$, $\tau' \leqslant \tau_1$ and $\Omega \subseteq \Omega'$.

Proof. By inspection of the subtyping rules, we can see that subtype relationship between $\tau_1 \xrightarrow{\Omega} \tau_2$ and σ must take one of the following forms:

- $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_1 \xrightarrow{\Omega} \tau_2$, by S-Refl.
- $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_3$ by S-Union-L.
- $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_4$ by S-Union-R.
- $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_1' \xrightarrow{\Omega'} \tau_2'$, and $\tau_1' \leqslant \tau_1$, $\Omega \leqslant \Omega'$, $\tau_2 \leqslant \tau_2'$ by S-Arrow.

We proceed by case analysis over these relationships.

- We can see that $\sigma = \tau_1 \xrightarrow{\Omega} \tau_2$. By inspection of the *apply* function we can see that because $apply(\sigma, \tau', \Omega') = apply(\sigma, \tau', \Omega') = \tau''$, it must be the case that $\tau' = \tau_1$, that $\Omega \subseteq \Omega'$ and that $\tau_2 = \tau''$. Then by S-Refl, $\tau' \leqslant \tau_1$ and $\tau_2 \leqslant \tau''$.
- Here $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_3$. We know $apply(\tau_3 \cup \tau_4, \tau', \Omega') = \tau''$, so $\tau'' = apply(\tau_3, \tau', \Omega') \cup apply(\tau_4, \tau', \Omega')$. Let $\tau_{a3} = apply(\tau_3, \tau', \Omega')$ and $\tau_{a4} = apply(\tau_4, \tau', \Omega')$. By structural induction over σ, $\tau_2 \leqslant \tau_{a3}$, $\tau' \leqslant \tau_1$, and $\Omega \subseteq \Omega'$. Then, $\tau_2 \leqslant \tau''$ by S-Union-Left and $\tau_2 \leqslant \tau_{a3}$.
- The case where $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_4$ is similar to the previous case.
- Consider the case where $\tau_1 \xrightarrow{\Omega} \tau_2 \leqslant \tau_1' \xrightarrow{\Omega'} \tau_2'$, and $\tau_1' \leqslant \tau_1$, $\Omega \leqslant \Omega'$, $\tau_2 \leqslant \tau_2'$. Then $apply(\sigma, \tau', \Omega') = apply(\tau_1' \xrightarrow{\Omega'} \tau_2', \tau', \Omega')$, so by the conditions on *apply* we know that $\tau_1' = \tau'$, that $\Omega'' \subseteq \Omega'$, and that $\tau_2 = \tau''$. Then $\tau_1' = \tau' \leqslant \tau_1$, and $\tau_2 \leqslant \tau'' = \tau_2'$, and $\Omega \subseteq \Omega'' \subseteq \Omega'$ so $\Omega \subseteq \Omega'$.

$\square$

Lemma[Delta]: If $\tau_1 \leqslant \tau_2$ and $\delta_\tau(c, \tau_1, \Omega) = \tau'$ and $\delta_\tau(c, \tau_2, \Omega) = \tau''$ then $\tau' \leqslant \tau''$.

Proof. We can see by observation of the subtyping rules that the subtype relationship between τ_1 and τ_2 must take one of the following forms:

- $\bot \leqslant \tau_2$
- $\tau_1 \leqslant \tau_1$
- $\tau_1 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \leqslant \tau_3$
- $\tau_1 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \leqslant \tau_4$
- $\tau_3 \cup \tau_4 \leqslant \tau_2$ with $\tau_3 \leqslant \tau_2$ and $\tau_4 \leqslant \tau_2$.
- $\tau_1 = \sigma_1 \xrightarrow{\Omega'} \sigma_2 \leqslant \sigma_3 \xrightarrow{\Omega''} \sigma_4 = \tau_2$ and $\sigma_3 \leqslant \sigma_1$, $\Omega' \subseteq \Omega''$, and $\sigma_2 \leqslant \sigma_4$.

We proceed by case analysis over these relationships.

- Here, $\delta_\tau(c, \bot, \Omega) = \bot$, so by S-Bot, $\bot \leqslant \tau''$.
- In this case, $\delta_\tau(c, \tau_1, \Omega) = \tau' = \tau'' = \delta_\tau(c, \tau_2, \Omega)$. By S-Refl, $\tau' \leqslant \tau''$.
- Here, $\tau_1 \leqslant \tau_3 \cup \tau_4$ and $\tau_1 \leqslant \tau_3$. By observation of δ_τ, we see that $\delta_\tau(c, \tau_3 \cup \tau_4, \Omega) = \delta_\tau(c, \tau_3, \Omega) \cup \delta_\tau(c, \tau_4, \Omega)$. Let $\tau_3' = \delta_\tau(c, \tau_3, \Omega)$. Then we have $\tau_1 \leqslant \tau_3$, $\delta_\tau(c, \tau_1, \Omega) = \tau'$ and $\tau_3' = \delta_\tau(c, \tau_3, \Omega)$. So by structural induction over τ_2, $\tau' \leqslant \tau_3'$ and by S-Union-Left, $\tau' \leqslant \tau''$.
- This is similar to the case above.

- Here, $\tau_3 \cup \tau_4 \leqslant \tau_2$ with $\tau_3 \leqslant \tau_2$ and $\tau_4 \leqslant \tau_2$. By observation of δ_τ, we can see that $\delta_\tau(c, \tau_3 \cup \tau_4, \Omega) = \delta_\tau(c, \tau_3, \Omega) \cup \delta_\tau(c, \tau_4, \Omega)$. Let $\tau_3' = \delta_\tau(c, \tau_3, \Omega)$ and $\tau_4' = \delta_\tau(c, \tau_4, \Omega)$. Then $\tau_3 \leqslant \tau_2$, $\delta_\tau(c, \tau_3, \Omega) = \tau_3'$ and $\tau_4' = \delta_\tau(c, \tau_4, \Omega)$, and $\tau'' = \delta_\tau(c, \tau_2, \Omega)$. By structural induction of τ_1, $\tau_3' \leqslant \tau''$ and $\tau_4' \leqslant \tau''$. Then by S-Union-Join $\tau' \leqslant \tau''$.

- Let us say without loss of generality that $c = \div$. Here, we can see that $\delta_\tau(\div, \tau_1, \Omega) = \delta_\tau(\div, \sigma_1 \xrightarrow{\Omega'} \sigma_2, \Omega) = \bot$ and that $\delta_\tau(\div, \tau_2, \Omega) = \delta_\tau(\div, \sigma_3 \xrightarrow{\Omega'} \sigma_4, \Omega) = \bot$ and that div-$\lambda \in \Omega$. So $\tau' = \tau'' = \bot$ so by S-Refl $\tau' \leqslant \tau''$.

$\square$

Lemma[Inversion]: If

- $\Omega; \Gamma \vdash v_1(v_2) : \tau$, then
  - $\Omega; \Gamma \vdash v_1 : \tau_1$,
  - $\Omega; \Gamma \vdash v_2 : \tau_2$, and
  - *apply*$(\tau_1, \tau_2, \Omega) : \tau_3$
  - $\tau_3 \leqslant \tau$

  And one of the following holds by case analysis on values:

  - $v_1 = \lambda x : (\tau'; \Omega').e'$ and
    1. $\Omega; \Gamma \vdash v_1 : \tau' \xrightarrow{\Omega'} \tau''$,
    2. $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau'$
    3. $\Omega'; \Gamma[x : \tau'] \vdash e' : \tau''$
  - $v_1 = 0$, $\Omega; \Gamma \vdash v_1 : \tau_1$ with $\mathsf{Z} \leqslant \tau_1$, and app-0 $\in \Omega$.
  - $v_1 = \mathsf{n}$, $\Omega; \Gamma \vdash v_1 : \tau_1$ with $\mathsf{N} \leqslant \tau_1$, and app-n $\in \Omega$.

- $\Omega; \Gamma \vdash \div(v) : \tau$ then either
  - $\Omega; \Gamma \vdash v : \mathsf{N}$ and $\mathsf{N} \leqslant \tau$, or
  - $\Omega; \Gamma \vdash v : \tau_1$ with $\mathsf{Z} \leqslant \tau_1$ and div-0 $\in \Omega$.
  - $\Omega; \Gamma \vdash v : \tau_1$ with $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau_1$ and div-$\lambda \in \Omega$.

- $\Omega; \Gamma \vdash$ *add1*$(v) : \tau$ then either
  - $\Omega; \Gamma \vdash v : \tau_1$ with $\tau_1 \leqslant \mathsf{N} \cup \mathsf{Z}$ and $\mathsf{N} \leqslant \tau$, or
  - $\Omega; \Gamma \vdash v : \tau_1$ with $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau_1$ and add1-$\lambda \in \Omega$.

- $\Omega; \Gamma \vdash$ err-$\omega : \tau$ then $\omega \in \Omega$.

Proof. We proceed by analysis over these cases.

- If $\Omega; \Gamma \vdash v_1(v_2) : \tau$, two rules apply, T-App and T-Sub.

  If T-App was used, by inspection of the *apply* function, it is immediate that $\Omega; \Gamma \vdash v_1 : \tau_1$, $\Omega; \Gamma \vdash v_2 : \tau_2$, and *apply*$(\tau_1, \tau_2, \Omega) = \tau$.

  It T-Sub was applied, then by induction on the typing derivation T-App applies to give us $\Omega; \Gamma \vdash v_1 : \tau_1$, $\Omega; \Gamma \vdash v_2 : \tau_2$, and $\Omega; \Gamma \vdash v_1(v_2) : \tau'$ with *apply*$(\tau_1, \tau_2, \Omega') = \tau_3$. By induction of the subtyping derivation and transitivity of subtyping, $\tau_3 \leqslant \tau$.

  Now we know that $\Omega; \Gamma \vdash v_1(v_2) : \tau$, $\Omega; \Gamma \vdash v_1 : \tau_1$, $\Omega; \Gamma \vdash v_2 : \tau_2$, and *apply*$(\tau_1, \tau_2, \Omega) = \tau_3 \leqslant \tau$. By inspection of the values presented in the semantics, we know that either $v_1 = \lambda x : (\tau'; \Omega').e$, $v_1 = 0$, or $v_1 = \mathsf{n}$. We address each of the cases.

- If $v_1 = \lambda x : (\tau'; \Omega').e$, then $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau_1$. This could have been show either by T-Fun or T-Sub.

  If T-Fun is used to show that $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau_1$, it is immediate that $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau' \xrightarrow{\Omega'} \tau'' = \tau_1$, and that $\Omega'; \Gamma[x : \tau'] \vdash e : \tau''$. By S-Refl $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau_1$.

  In the case where $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau_1$ is shown by T-Sub, then we can see by induction on the typing derivation that T-Fun was applied at some point in the derivation. This shows that $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau' \xrightarrow{\Omega'} \tau''$ and also that $\Omega'; \Gamma[x : \tau'] \vdash e : \tau''$. Also by induction over the typing derivation and by transitivity of subtyping, $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau_1$.

- If $v_1 = \mathsf{0}$ then T-Zero or T-Sub could apply for $\Omega; \Gamma \vdash v_1 : \tau_1$. If T-Zero applies, then $\Omega; \Gamma \vdash v_1 : \mathsf{Z} = \tau_1$, so by S-Refl, $\mathsf{Z} \leqslant \tau_1$. Furthermore, because $\mathit{apply}(\mathsf{Z}, \tau_2, \Omega) = \tau$, by inspection of *apply* it must be the case the $\mathsf{app\text{-}0} \in \Omega$.

  If T-Sub is used then by induction T-Zero is used at some point in the typing derivation. So $\Omega; \Gamma \vdash v_1 : \mathsf{Z}$ with $\mathsf{Z} \leqslant \tau_1$. By induction over the subtyping derivation $\tau_1 = \sigma_1 \cup \cdots \cup \mathsf{Z} \cup \cdots \cup \sigma_n$. Then by inspection of *apply*,

  $$\mathit{apply}(\tau_1, \tau_2, \Omega) = \mathit{apply}(\sigma_1, \tau_2, \Omega) \cup \cdots \cup \mathit{apply}(\mathsf{Z}, \tau_2, \Omega) \cup \cdots \cup \mathit{apply}(\sigma_n, \tau_2, \Omega).$$

  Because $\mathit{apply}(\mathsf{Z}, \tau_2, \Omega)$ must be defined we conclude that $\mathsf{app\text{-}0} \in \Omega$.

- The case where $v_1 = \mathsf{n}$ is similar to the case where $v_1 = \mathsf{0}$.

- Now we consider $\Omega; \Gamma \vdash \div(v) : \tau$. Two rules may apply to $\Omega; \Gamma \vdash \div(v) : \tau$, T-Sub and T-Op. In the case of T-Op, we see that $\Omega; \Gamma \vdash v : \tau_1$, and that $\tau = \delta_\tau(\div, \tau_1, \Omega)$.

  In the case where T-Sub is used to type $\Omega; \Gamma \vdash \div(v) : \tau$, by induction on the typing derivation T-Op was used to produce $\Omega; \Gamma \vdash \div(v) : \tau_2$, which gives us that $\Omega; \Gamma \vdash v : \tau_1$ and $\tau_2 = \delta_\tau(\div, \tau_1, \Omega)$. Further by induction on the typing derivation, $\tau_2 \leqslant \tau$.

  By observation of the values in the semantics either

  - $v = \mathsf{n}$
  - $v = \mathsf{0}$
  - $v = \lambda x : (\tau'; \Omega').e$

  We address each case. If

  - $v = \mathsf{n}$, there are two rules that could have been used to type v. In the case where T-Num is used it is immediate that $\Omega; \Gamma \vdash v : \mathsf{N} = \tau_1$, so by S-Refl, $\mathsf{N} \leqslant \tau_1$. Futher $\tau = \delta_\tau(\div, \tau_1, \Omega) = \delta_\tau(\div, \mathsf{N}, \Omega) = \mathsf{N}$ so by S-Refl, $\mathsf{N} \leqslant \tau$.

    If T-Sub is used to type v, then $\Omega; \Gamma \vdash v : \tau_1$. By induction on the typing derivation, T-Num must have been applied to v, so $\Omega; \Gamma \vdash v : \mathsf{N}$ and by induction on the subtyping derivation, $\mathsf{N} \leqslant \tau_1$. By inspection of δ_τ we see that $\delta_\tau(\div, \mathsf{N}, \Omega) = \mathsf{N}$, so by **Lemma[Delta]**, $\mathsf{N} \leqslant \tau_2$ and by induction on subtyping $\mathsf{N} \leqslant \tau$.

  - $v = \mathsf{0}$ then either T-Zero or T-Sub was used as the final step of the typing derivation. In the case of T-Zero it is immediate that $\mathsf{Z} \leqslant \tau_1$. Additionally because $\tau_2 = \delta_\tau(\div, \mathsf{Z}, \Omega)$, we conclude that $\delta_\tau(\div, \mathsf{Z}, \Omega)$ is defined, and therefore by inspection of δ_τ, $\mathsf{div\text{-}0} \in \Omega$.

    If T-Sub was used to show that $\Omega; \Gamma \vdash v : \tau_1$ then by induction over the typing derivation, T-Zero must have been used to show $\Omega; \Gamma \vdash v : \mathsf{Z}$ and $\mathsf{Z} \leqslant \tau_1$. Also by induction over that derivation, we can see that

    $$\tau_1 = \sigma_1 \cup \cdots \cup \mathsf{Z} \cup \cdots \cup \sigma_n.$$

    So then

    $$\tau_2 = \delta_\tau(\div, \tau_1, \Omega) = \delta_\tau(\div, \sigma_1, \Omega) \cup \cdots \cup \delta_\tau(\div, \mathsf{Z}, \Omega) \cup \cdots \cup \delta_\tau(\div, \sigma_n, \Omega).$$

    From this we conclude that $\delta_\tau(\div, \mathsf{Z}, \Omega)$ is defined, and therefore $\mathsf{div\text{-}0} \in \Omega$.

  - $v = \lambda x : (\tau'; \Omega').e$ then either T-Fun or T-Sub was used to show $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau_1$. If T-Fun was used, then $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau' \xrightarrow{\Omega'} \tau''$. By S-Refl $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau_1$. Further, because $\tau_2 = \delta_\tau(\div, \tau' \xrightarrow{\Omega'} \tau'', \Omega)$ is defined, $\mathsf{div\text{-}\lambda} \in \Omega$.

If $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau_1$ was shown through use of $\mathsf{T\text{-}Sub}$ then by induction on the typing derivation $\mathsf{T\text{-}Fun}$ must be used to show that $\Omega; \Gamma \vdash \lambda x : (\tau'; \Omega').e : \tau' \xrightarrow{\Omega'} \tau''$ and $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau_1$. By observation of the subtyping rules, either

$$\tau_1 = \sigma_1 \cup \cdots \cup (\tau' \xrightarrow{\Omega'} \tau'') \cup \cdots \cup \sigma_n$$

or

$$\tau_1 = \tau_3 \xrightarrow{\Omega''} \tau_4 \text{ with } \tau_3 \leqslant \tau',\ \Omega' \subseteq \Omega'',\ \text{and } \tau'' \leqslant \tau_4.$$

In the first case because $\delta_\tau(\div, \tau_1, \Omega)$ is defined, $\delta_\tau(\div, \tau' \xrightarrow{\Omega'} \tau'', \Omega)$ must be defined. Therefore $\mathsf{div}\text{-}\lambda \in \Omega$.

In the second case $\delta_\tau(\div, \tau_1, \Omega) = \delta_\tau(\div, \tau_3 \xrightarrow{\Omega'} \tau_4, \Omega)$ and so $\mathsf{div}\text{-}\lambda \in \Omega$.

- The proof for $\Omega; \Gamma \vdash \mathit{add1}(v) : \tau$ is similar to the above case.
- Consider $\Omega; \Gamma \vdash \mathsf{err}\text{-}\omega : \tau$. Either $\mathsf{T\text{-}Err}$ or $\mathsf{T\text{-}Sub}$ was used to make this judgment. It follows immediately from $\mathsf{T\text{-}Err}$ that $\omega \in \Omega$.

  If $\mathsf{T\text{-}Sub}$ was used to show $\Omega; \Gamma \vdash \mathsf{err}\text{-}\omega : \tau$ we can induce over the typing derivation to show that $\mathsf{T\text{-}Err}$ must have been applied to show that $\Omega; \Gamma \vdash \mathsf{err}\text{-}\omega : \tau'$. It then follows that $\omega \in \Omega$.

$\square$

Lemma[Progress]: If $\Omega; \Gamma \vdash e : \tau$ then

1. $e \in v$, or
2. $e \neq \mathsf{err}\text{-}\omega$, and there exists e' such that $e \to e'$, or
3. $e = \mathsf{err}\text{-}\omega$ where $\omega \in \Omega$.

Proof. : By case analysis on e. By Lemma[AE] we know that e can take a finite number of forms, as follows:

i. $e \in v$, or

ii. there exists E and ae such that $e = E[ae]$, or

iii. $e = E[\mathsf{err}\text{-}\omega]$.

We consider these possibilities by cases.

Case: $e \in v$

Clearly, we are in 1.

Case: $e = E[e']$ for some $E, e' \in \{ae, \mathsf{err}\text{-}\omega\}$.

We proceed again by case analysis, this time on e'.

Case: $e' = \mathsf{n}(v)$ for $\mathsf{n} \neq 0$.

We can see by E-Apply-Num in Fig. 3 that $\mathsf{n}(v) \Rightarrow \mathsf{err\text{-}app\text{-}n}$. Then, again by Fig. 3, we see that $E[\mathsf{n}(v)] \to E[\mathsf{err\text{-}app\text{-}n}]$. Thus we are in 2.

Case: $e' = 0(v)$.

Similar to the case above.

Case: $e' = (\lambda x : (\tau; \Omega).e')(v)$

Then again by Fig. 3 we can see that $(\lambda x : (\tau; \Omega).e')(v) \Rightarrow e[x/v]$. So then $E[(\lambda x : (\tau; \Omega).e')(v)] \to E[e[x/v]]$, and we are in 2.

Case: $e' = div(v)$.

In this case, by Fig. 3, $div(v) \Rightarrow \delta(div, v)$. We can see by inspection of the δ function that $\delta(div, v)$ is defined for every value v. So then $E[div(v)] \to E[\delta(div, v)]$. So we are in 2.

Case: $e' = add1(v)$.

This case is similar to the case above.

Case: $e' = \mathsf{err}\text{-}\omega$.

By Fig. 3 $E[\mathsf{err}\text{-}\omega] \to \mathsf{err}\text{-}\omega$. So we are in 2.

Case: $e = \mathsf{err}\text{-}\omega$.

Because we know that $\Omega; \Gamma \vdash e : \tau$ we know that we must be able to construct a typing derivation to show this. We prove this case by induction on the height h of that typing derivation.

Case: $h = 2$

In this case, the only rule that can apply is T-Err, which gives us the following derivation:

$$\frac{\omega \in \Omega}{\Omega; \Gamma \vdash \mathsf{err}\text{-}\omega : \bot}$$

In order to achieve this derivation, it must be the case that $\mathsf{err}\text{-}\omega \in \Omega$, so we are in 3.

Case: $h > 2$

In this case, the rule that must apply is T-Sub. Then we have as the antecedent that $\Omega; \Gamma \vdash \mathsf{err}\text{-}\omega : \tau_1$ and $\tau_1 \leqslant \tau$. We know that the typing derivation for $\Omega; \Gamma \vdash \mathsf{err}\text{-}\omega : \tau_1$ has height $h - 1$, so by our induction hypothesis we conclude that $\mathsf{err}\text{-}\omega \in \Omega$, so we are in 3.

Having covered the possible cases for e, we conclude that the statement is true.

$\square$

Lemma[Preservation]: If $\Omega; \Gamma \vdash e_1 : \tau$ and $e_1 \rightarrow e_2$, then $\Omega; \Gamma \vdash e_2 : \tau$, and if $e_2 = E[\textsf{err-}\omega]$ then $\omega \in \Omega$.

Proof. By **Lemma[AE]** we can see that either $e_1 = v$, $e_1 = E[ae]$ for some E, ae, or $e_1 = E[\textsf{err-}\omega]$ for some E, ω. By inspection of the rules, we know that $e_1 \neq v$. So either $e_1 = E[e_1']$ for $e_1' = ae$ and $e_2 = E[e_2']$ for some e_2', or $e_1' = \textsf{err-}\omega$. Note that by induction over the derivation for e_1, it must be the case the $\Omega; \Gamma \vdash e_1' : \tau_1'$.

We proceed by cases over e_1'.

Case: $e_1' = (\lambda x : (\tau' : \Omega').e)(v)$

By inversion, we can see that $\Omega; \Gamma \vdash \lambda x : (\tau' : \Omega').e : \tau'''$, with $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau'''$, that $\Omega'; \Gamma[x : \tau'] \vdash e : \tau''$, and that $\Omega; \Gamma \vdash v : \sigma$. We can also see by inversion that $\mathit{apply}(\tau''', \sigma, \Omega) = \tau_1''$, with $\tau_1'' \leqslant \tau_1'$.

Because we know that $\tau' \xrightarrow{\Omega'} \tau'' \leqslant \tau'''$ and $\mathit{apply}(\tau''', \sigma, \Omega) = \tau_1''$, we can use the application lemma to show that $\tau'' \leqslant \tau_1''$, that $\sigma \leqslant \tau'$ and that $\Omega' \subseteq \Omega$.

By subsumption we can show that $\Omega; \Gamma \vdash v : \tau'$.

By inspection of the reduction rules, $e_2' = e[x/v]$.

Then because $\Omega'; \Gamma[x : \tau'] \vdash e : \tau''$, $\Omega; \Gamma \vdash v : \tau'$, and $\Omega' \subseteq \Omega$ we can show by substitution that $\Omega; \Gamma \vdash e[x/v] : \tau''$. Then by subsumption we can see that $\Omega; \Gamma \vdash e[x/v] : \tau_1'$.

Case: $e_1' = \textsf{0}(v)$

By inversion we know that $\Omega; \Gamma \vdash \textsf{0} : \tau_1$ such that $\textsf{Z} \leqslant \tau_1$, that $\Omega; \Gamma \vdash v : \tau_2$, and that $\textsf{app-0} \in \Omega$.

By inspection of the reduction rules $e_2' = \textsf{err-app-0}$. We have shown that $\textsf{app-0} \in \Omega$ and so T-Err $\Omega; \Gamma \vdash e_2' : \bot$. Because $\bot \leqslant \tau$ we can show by T-Sub that $\Omega; \Gamma \vdash e_2 : \tau$.

Case: $e_1' = \textsf{n}(v)$

This is similar to $e_1' = \textsf{0}(v)$

Case: $e_1' = \div(\textsf{n})$

By inversion we can see that $\Omega; \Gamma \vdash e_1' : \tau_1'$ and that $\textsf{N} \leqslant \tau_1'$.

By inspection of the reduction rules $e_2' = 1/\textsf{n}$. By T-Num, $\Omega; \Gamma \vdash 1/\textsf{n} : \textsf{N}$.

Because $\textsf{N} \leqslant \tau_1'$ by T-Sub $\Omega; \Gamma \vdash 1/\textsf{n} : \tau_1'$.

Case: $e_1' = \div(\textsf{0})$

By inversion we know that $\Omega; \Gamma \vdash \textsf{0} : \tau_1$ such that $\textsf{Z} \leqslant \tau_1$ and that $\textsf{div-0} \in \Omega$.

By inspection of the reduction rules $e_2' = \textsf{err-div-0}$. We have shown that $\textsf{div-0} \in \Omega$ and so T-Err $\Omega; \Gamma \vdash e_2' : \bot$. Because $\bot \leqslant \tau$ we can show by T-Sub that $\Omega; \Gamma \vdash e_2 : \tau$.

Case: $e_1' = \div(\lambda x : (\tau'; \Omega').e)$

This is similar to $e_1' = \div(\textsf{0})$

Case: $e_1' = \mathit{add1}(\textsf{n})$

By inversion we can see that $\Omega; \Gamma \vdash e_1' : \tau_1'$ and that $\textsf{N} \leqslant \tau_1'$.

By inspection of the reduction rules $e_2' = \textsf{n} + 1$. By T-Num, $\Omega; \Gamma \vdash \textsf{n} + 1 : \textsf{N}$.

Because $\textsf{N} \leqslant \tau_1'$ by T-Sub $\Omega; \Gamma \vdash \textsf{n} + 1 : \tau_1'$.

Case: $e_1' = \mathit{add1}(\lambda x : (\tau'; \Omega').e)$

This is similar to $e_1' = \div(\textsf{0})$

Case: $e_1' = \textsf{err-}\omega$

Because $\Omega; \Gamma \vdash \textsf{err-}\omega : \tau_1'$, the derivation must have used T-Err, and so we conclude that $\omega \in \Omega$.

Then by inspection of the reduction rules $e_2 = \textsf{err-}\omega$. By T-Err, $\Omega; \Gamma \vdash e_2 : \bot$. Because $\bot \leqslant \tau$ we can show by T-Sub that $\Omega; \Gamma \vdash e_2 : \tau$.

□