In this document, we formally prove the following Theorem 1:

Theorem 1. *If* $\Gamma \vdash p$ *then* p *is* NI^{Γ}_{τ}.

In Section 1, we present *semantics rules* and the *typing rules* of the language. In Section 2, we present the definitions and lemmas used to prove the theorem. In Section 3, we prove the Lemmas and Theorem 1.

1 Semantics and Type Rules

1.1 Semantic Rules

BASE

$$\mu \vdash v \Rightarrow v$$

VAR

$$\frac{\mu(x) = v}{\mu \vdash x \Rightarrow v}$$

OP

$$\frac{\mu \vdash e \Rightarrow v,\ \mu \vdash e' \Rightarrow v'}{\mu \vdash e \circ e' \Rightarrow^{[]} v \circ v'}$$

UPDATE

$$\frac{\mu \vdash e \Rightarrow v,\ m \in dom(\mu)}{\mu \vdash m := e \Rightarrow^{[]} \mu[m := v]}$$

SEQUENCE

$$\frac{\mu \vdash c \Rightarrow^{t'} \mu',\ \mu' \vdash c' \Rightarrow^{t''} \mu''}{\mu \vdash c; c' \Rightarrow^{t' \cdot t''} \mu''}$$

BRANCH-TRUE

$$\frac{\mu \vdash e \Rightarrow true,\ \ \mu \vdash c \Rightarrow^{t} \mu'}{\mu \vdash \mathsf{if\ e\ then\ c\ else\ c'} \Rightarrow^{\mathsf{t}} \mu'}$$

BRANCH-FALSE

$$\frac{\mu \vdash e \Rightarrow false,\ \ \mu \vdash c \Rightarrow^{t} \mu'}{\mu \vdash \mathsf{if\ e\ then\ c\ else\ c'} \Rightarrow^{\mathsf{t}} \mu'}$$

LOOP-TRUE

$$\frac{\begin{array}{l}\mu \vdash e \Rightarrow true, \qquad \mu \vdash c \Rightarrow^{t} \mu', \\ \mu' \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t'}} \mu''\end{array}}{\mu \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t \cdot t'}} \mu''}$$

LOOP-FALSE

$$\frac{\mu \vdash e \Rightarrow false}{\mu \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{[]} \mu}$$

SECURE BROADCAST

$$\frac{\mu \vdash k \Rightarrow vk \qquad \mu \vdash e \Rightarrow v}{\mu \vdash \mathsf{sbroadcast}(\{\mathsf{n_1} \ldots \mathsf{n_i}\}, \mathsf{k}, \mathsf{e}) \Rightarrow^{\mathsf{BEnc}(\{\mathsf{n_1} \ldots \mathsf{n_i}\}, \mathsf{vk}, \mathsf{v})} \mu}$$

KEY GENERATION

$$\frac{\mu \vdash KG(\{n_1, \ldots, n_i\}, v) \Rightarrow vk \qquad \mu \vdash k' \Rightarrow v \qquad \mu(n_j) = v_j\ j \in \{1..i\}}{\mu \vdash k := KG(\{v_1, \ldots, v_i\}, k') \Rightarrow^{[]} \mu[k := vk]}$$

1.2 Type Rules

$$\textsc{Lit}\ \frac{}{\Gamma \vdash n : \tau} \qquad \textsc{Var}\ \frac{\Gamma(x) = \tau\ var}{\Gamma \vdash x : \tau\ var} \qquad \textsc{NVar}\ \frac{\Gamma(n) = \tau\ Nvar(n)}{\Gamma \vdash n : \tau\ Nvar(n)} \qquad \textsc{MKvar}\ \frac{\Gamma(k) = \top\ MKvar}{\Gamma \vdash k : \top\ MKvar}$$

$$\textsc{KVar-Assign}\ \frac{\Gamma(k) = \tau\ Kvar(n_1 \ldots n_i) \qquad \Gamma \vdash n_j : \tau\ Nvar(n_j)\ j \in \{1...i\} \qquad \Gamma \vdash k' : \top\ MKvar}{\Gamma \vdash k := KG(\{n_1 \ldots n_i\}, k') : \tau\ cmd}$$

$$\textsc{Op}\ \frac{\Gamma \vdash e : \tau \qquad \Gamma \vdash e' : \tau}{\Gamma \vdash e \circ e' : \tau} \qquad \textsc{Assign}\ \frac{\Gamma \vdash x : \tau\ var \qquad \Gamma \vdash e : \tau}{\Gamma \vdash x := e : \tau\ cmd}$$

$$\textsc{Sequence}\ \frac{\Gamma \vdash c : \tau\ cmd \qquad \Gamma \vdash c' : \tau\ cmd}{\Gamma \vdash c; c' : \tau\ cmd}$$

$$\textsc{If}\ \frac{\Gamma \vdash e : \tau \qquad \Gamma \vdash c : \tau\ cmd \qquad \Gamma \vdash c' : \tau\ cmd}{\Gamma \vdash \mathsf{if\ e\ then\ c\ else\ c'} : \tau\ \mathsf{cmd}} \qquad \textsc{While}\ \frac{\Gamma \vdash e : \tau \qquad \Gamma \vdash c : \tau\ cmd}{\Gamma \vdash \mathsf{while\ e\ do\ c} : \tau\ \mathsf{cmd}}$$

$$\textsc{SBroadcast}\ \frac{\Gamma \vdash e : \tau \qquad \Gamma \vdash n_i :\ \tau\ Nvar(n_i)\ i \in \{1...j\} \qquad \Gamma \vdash k : \tau\ Kvar\ (n_1 \ldots n_j)}{\Gamma \vdash \mathsf{sbroadcast}(\{\mathsf{n}_1, ..., \mathsf{n}_\mathsf{j}\}, \mathsf{k}, \mathsf{e}) : \tau\ \mathsf{cmd}}$$

Subtyping rules

$$\textsc{Base}\ \frac{\tau \leq \tau'}{\vdash \tau \subseteq \tau'} \qquad \textsc{S-Var}\ \frac{\Gamma \vdash x : \tau\ var}{\Gamma \vdash x : \tau} \qquad \textsc{S-NVar}\ \frac{\vdash \tau \subseteq \tau'}{\Gamma \vdash \tau'\ NVar(n) \subseteq \tau\ NVar(n)}$$

$$\textsc{CMD}\ \frac{\vdash \tau \subseteq \tau'}{\Gamma \vdash \tau'\ cmd \subseteq \tau\ cmd} \qquad \textsc{Subtype}\ \frac{\Gamma \vdash p : \rho \qquad \vdash \rho \subseteq \rho'}{\Gamma \vdash p : \rho'}$$

2 Definitions and Lemmas

Definition 1 ($\tau - Equal$ Memories)**.** *Two memories* μ_0*,* μ_1 *are* $\tau - Equal$ *for* Γ*, written* $\mu_0 =^{\Gamma}_{\tau} \mu_1$, iff $dom(\mu_0) = dom(\mu_1) \wedge \forall x \in \mu_0$ *such that* if $\Gamma(\mathsf{x}) = \tau' \ \mathsf{var} \wedge \tau' \leq \tau$, *then* $\mu_0(x) = \mu_1(x)$.

Definition 2 (Filtering)**.** *Let* $\mathsf{filter}_{\tau}(\mathsf{t}) = \mathsf{filter}'_{\tau}(\mathsf{t}, [\,])$ *and* $\mathsf{filter}'_{\tau}([\,], \mathsf{t}) = \mathsf{t}$.

$$\mathsf{filter}'_{\tau}(\mathsf{BEnc}(\{\mathsf{n}_1, ..., \mathsf{n}_\mathsf{k}\}, \mathsf{vk}, \mathsf{v}') \cdot \mathsf{t}', \mathsf{t}) = \begin{cases} \mathsf{filter}'_{\tau}(\mathsf{t}', \mathsf{t} \cdot (\{\mathsf{n}_1, ..., \mathsf{n}_\mathsf{k}\}, \mathsf{v}')) & \mathsf{if}\ \Gamma(\mathsf{n}_1) \sqcap \Gamma(\mathsf{n}_2) ... \sqcap \Gamma(\mathsf{n}_\mathsf{k}) \leq \tau \\ \mathsf{filter}'_{\tau}(\mathsf{t}', \mathsf{t}) & \textit{Otherwise} \end{cases}$$

Definition 3 (NonInterference)**.** *A server program p is NI at* τ *for* Γ*, written* $NI^{\Gamma}_{\tau}(p)$, iff $\forall \mu_0, \mu_1$ *such that* $\mu_0 =^{\Gamma}_{\tau} \mu_1 \wedge \mu_0 \vdash p \Rightarrow^{t_0} \mu'_0 \wedge \mu_1 \vdash p \Rightarrow^{t_1} \mu'_1$*, then* $\mu'_0 =^{\Gamma}_{\tau} \mu'_1 \wedge \mathsf{filter}(\mathsf{t}_0) = \mathsf{filter}(\mathsf{t}_1)$.

Lemma 1. *(LowExpression)* $\forall \tau, \tau', \mu_0, \mu_1$*, if* $\mu_0 =^{\Gamma}_{\tau} \mu_1$ *and* $\Gamma \vdash e : \tau'$ *and* $\tau' \leq \tau$ *and* $\mu_0 \vdash e \Rightarrow v_0$ *and* $\mu_1 \vdash e \Rightarrow v_1$*, then* $v_0 = v_1$.

Lemma 2. *(HighCommand)* $\forall \tau, \tau', \mu, \mu', t$*, if* $\Gamma \vdash c : \tau'\ cmd$ *and* $\tau' \not\leq \tau$ *and* $\mu \vdash c \Rightarrow^{t} \mu'$*, then* $\mu =^{\Gamma}_{\tau} \mu'$ *and* $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

3 Proof

3.1 Proof of Lemma 1

Proof. We prove Lemma1 by structural induction on e with $P_{Struct}(e) = n$. According to our type system, specifically the subtyping rule *S-Var*, only variables of type τ *var* can be typed as τ. That is the reason we omit the cases of *NVar* and *MKVar*.

It follows that:

- $P_{Struct}(v) = 1$.
- $P_{Struct}(x) = 1$.
- $P_{Struct}(e \circ e') = P_{Struct}(e) + P_{Struct}(e')$.

From the hypothesis of Lemma 1, we have that:

(H_1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(H_2) $\Gamma \vdash e : \tau'$.

(H_3) $\tau' \leq \tau$.

(H_4) $\mu_0 \vdash e \Rightarrow v_0$.

(H_5) $\mu_1 \vdash e \Rightarrow v_1$.

And we need to prove that:

(G_1) $v_0 = v_1$.

Case 1 (Base Case)**.**

Subcase 1.1 (Lit: v)**.** *By (H_2) and the typing rule Lit, we have that:*

(H_6) $\Gamma \vdash v : \tau'$.

By (H_4) and the semantics rule Base, we have that:

(H_7) $\mu_0 \vdash v \Rightarrow v$.

By (H_5) and the semantics rule Base, we have that:

(H_8) $\mu_1 \vdash v \Rightarrow v$.

From (H_7) and (H_8), (G_1) is trivially true.

Subcase 1.2 (Var: x)**.** *By (H_2) and the subtyping rule S-Var (which is the only rule that allows variables of type τ' var to be typed as variables of type τ'), we have that:*

(H_6) $\Gamma \vdash x : \tau'$ *var*.

By (H_6) and the typing rule Var, we have that:

(H_7) $\Gamma(x) = \tau'$ *var.*

By (H_4) and the semantics rule Var, we have that:

(H_8) $\mu_0(x) = v_0$.

By (H_5) and the semantics rule Var, we have that:

(H_9) $\mu_1(x) = v_1$.

By Definition 1 and (H_1) of Lemma 1, we have that:

$$\forall y, \mu_0(y) = \mu_1(y),\ if \Gamma(y) = \tau'\ var \wedge \tau' \leq \tau$$

From this, and by (H_6) and (H_3), we have that :

(H_{10}) $\mu_0(x) = \mu_1(x)$.

By (H_{10}), (H_8) and (H_9), we conclude that $v_0 = v_1$, which is (G_1).

Case 2 (Op: $e' \circ e''$)**.** ***Inductive hypothesis for*** e
For $P_{Struct}(e) = n$, we have that:

(IH_1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(IH_2) $\Gamma \vdash e : \tau'$.

(IH_3) $\tau' \leq \tau$.

(IH_4) $\mu_0 \vdash e \Rightarrow v_0$.

(IH_5) $\mu_1 \vdash e \Rightarrow v_1$.

Then we conclude that:

(IH_6) $v_0 = v_1$.

Inductive case: height $= P_{Struct}(e) = n+1$.

Op: $e' \circ e''$**.** *For the typing rule Op, we have that:*

(H_1) $\Gamma \vdash e' : \tau'$.

(H_2) $\Gamma \vdash e'' : \tau'$.

By (IH_4) and the semantics rule Op, we have that:

(H_3) $\mu_0 \vdash e' \Rightarrow v_0$.

(H_4) $\mu_0 \vdash e'' \Rightarrow v'_0$.

(H_5) $\mu_0 \vdash e' \circ e'' \Rightarrow v_0 \circ v'_0$.

By (IH_5) and the semantics rule Op, we have that:

(H_6) $\mu_1 \vdash e' \Rightarrow v_1$.

(H_7) $\mu_1 \vdash e'' \Rightarrow v'_1$.

(H_8) $\mu_1 \vdash e' \circ e'' \Rightarrow v_1 \circ v'_1$.

Concerning e', from the inductive hypothesis on e, it follows that:

(H_6) $v_0 = v_1$.

Concerning e", from the inductive hypothesis on e, it follows that:

(H_7) $v'_0 = v'_1$.

By the transitive property of $=$ *and since* $v_0 = v_1$ *and* $v'_0 = v'_1$*, we conclude that* $v_0 \cdot v'_0 = v_1 \cdot v'_1$*, which is* (G_1).

□

3.2 Proof of Lemma 2

Proof. We prove lemma2 by structural induction on c with $P_{Struct}(c) = n$. It follows that:

- $P_{Struct}(x := e') = 1$.
- $P_{Struct}(k := KG(\{n_1, ..., n_i\}, k')) = 1$.
- $P_{Struct}(\mathsf{sbroadcast}(\{\mathsf{n_1}, ..., \mathsf{n_j}\}, \mathsf{k}, \mathsf{e})) = 1$.
- $P_{Struct}(c'; c'') = P_{Struct}(c') + P_{Struct}(c'')$.
- $P_{Struct}(\mathsf{while\ e\ do\ c}) = \mathsf{P_{Struct}(c)} + 1$.
- $P_{Struct}(\mathsf{if\ e\ then\ c'\ else\ c''}) = \mathsf{MAX}(\mathsf{P_{Struct}(c')}, \mathsf{P_{Struct}(c'')}) + 1$.

From the hypothesis of Lemma 2, we have that:

(H_1) $\Gamma \vdash c : \tau'\ cmd$.

(H_2) $\tau' \not\leq \tau$.

(H_3) $\mu \vdash c \Rightarrow^t \mu'$.

And we need to prove that:

(G_1) $\mu =^{\Gamma}_{\tau} \mu'$.

(G_2) $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

Case 1 (Base case)**.**

Subcase 1.1 (Assign: $x := e'$.)**.** *By the hypothesis of the lemma, we have that:*

(H₄) $\Gamma \vdash x := e' : \tau'\ cmd$.

By (H₄) and the typing rule Assign, we have that:

(H₅) $\Gamma \vdash x : \tau'\ var$.

(H₆) $\Gamma \vdash e : \tau'$.

By (H₅) and the typing rule Var, we have that:

(H₇) $\Gamma(x) = \tau'\ var$.

By (H₃) and the semantics rule Update, we have that:

(H₈) $\mu \vdash e \Rightarrow v$.

(H₉) $\mu \vdash x := e \Rightarrow^t \mu[x := v]$*, then* $\mu' = \mu[\mu[x := v]$.

By the semantics rule Update, we have that:

(H_{10}) $t = [\,]$.

For (H_2), we have that $\tau' \not\leq \tau$. Since the only variable in which μ and μ' are different is x by (H_9) and the security level of the variable x is $\tau' var$, we can conclude that $\mu =^{\Gamma}_{\tau} \mu'$ which is (G_1).

To prove (G_2), by (H_{10}), $t = [\,]$. Therefore $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

Subcase 1.2 (KVar-Assign: $k := KG(\{n_1,...,n_i\},k')$)**.** *By the hypothesis of Lemma 2 , we have that:*

(H_4) $\Gamma \vdash k := KG(\{n_1,...,n_i\},k') : \tau'\ cmd$.

By (H_4) and the typing rule KVar-Assign, we have that:

(H_5) $\Gamma(k) = \tau\ Kvar(n_i,...,n_i)$.

(H_6) $\Gamma \vdash n_j : \tau'\ Nvar(n_j)$.

(H_7) $\Gamma \vdash k' : \top\ MKvar$.

By (H_3) and the semantics rule Update, we have that:

(H_8) $\mu \vdash KG(\{n_1,...,n_i\},v) \Rightarrow vk$.

(H_9) $\mu \vdash k' \Rightarrow v$.

(H_{10}) $\mu(n_j) = v_j$.

(H_{11}) $\mu \vdash k := KG(\{v_1,...,v_i\},k') \Rightarrow^{t} \mu[k := vk]$, *then* $\mu' = \mu[k := vk]$.

By the semantics rule Update, we have that:

(H_{12}) $t = [\,]$.

For (H_2), we have that $\tau' \not\leq \tau$. Since the only variable in which μ and μ' are different is k by (H_{11}) and the type of k is $\tau' Kvar(n_1...,n_i)$ by (H_5), we can conclude that $\mu =^{\Gamma}_{\tau} \mu'$ which is (G_1).

To prove (G_2), by (H_{12}), $t = [\,]$. Therefore $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$, *which is ($G_2$)*

Subcase 1.3 (SBroadcast: $\mathsf{sbroadcast}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{j}\},\mathsf{k},\mathsf{e})$)**.** *By the hypothesis of the lemma, we have that:*

(H_4) $\Gamma \vdash \mathsf{sbroadcast}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{j}\},\mathsf{k},\mathsf{e}) : \tau'\ \mathsf{cmd}$.

By (H_4) and the typing rule SBroadcast, we have that:

(H_5) $\Gamma \vdash e : \tau'$.

(H_6) $\Gamma \vdash n_i :\ \tau\ Nvar(n_i)$.

(H_7) $\Gamma \vdash k : \tau\, Kvar\ (n_1 \ldots n_j)$.

By (H_3) and the semantics rule Secure Broadcast, we have that:

(H_8) $\mu \vdash \mathsf{sbroadcast}(\{\mathsf{n_1}, ..., \mathsf{n_j}\}, \mathsf{k}, \mathsf{e}) \Rightarrow^{\mathsf{t}} \mu$.

By the semantics rule Secure Broadcast, we have that:

(H_9) $t = \mathsf{BEnc}(\{\mathsf{n_1}, ..., \mathsf{n_k}\}, \mathsf{vk}, \mathsf{v'})$.

For (H_2), we have that $\tau' \not\leq \tau$. Since the initial memory μ and the final memory μ are equal by (H_8), we conclude that $\mu =^{\Gamma}_{\tau} \mu'$ which is (G_1).

To prove (G_2), by (H_9), $t = \mathsf{BEnc}(\{\mathsf{n_1}, ..., \mathsf{n_k}\}, \mathsf{vk}, \mathsf{v'})$. By Definition 2, and by (H_2) and (H_6), $\Gamma(n_1) \sqcap \Gamma(n_2) ... \sqcap \Gamma(n_k) \not\leq \tau$, therefore $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$; which proves ($G_2$)

Inductive hypothesis for c *For $P_{Struct}(c) = n$, we have that:*

(IH_1) $\Gamma \vdash c : \tau'\ cmd$.

(IH_2) $\mu \vdash c \Rightarrow^{t} \mu'$.

(IH_3) $\tau' \not\leq \tau$.

Then we conclude that:

(IH_4) $\mu =^{\Gamma}_{\tau} \mu'$.

(IH_5) $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

Inductive case: height $= P_{Struct}(c) = n + 1$.

Sequence: $c'; c''$**.** *For the typing rule Sequence, we have that:*

(H_1) $\Gamma \vdash c' : \tau'\ cmd$.

(H_2) $\Gamma \vdash c'' : \tau'\ cmd$.

By (IH_2) and the semantics rule Sequence, we have that:

(H_3) $\mu \vdash c' \Rightarrow^{t'} \mu'$.

(H_4) $\mu' \vdash c'' \Rightarrow^{t''} \mu''$.

(H_5) $\mu \vdash c'; c'' \Rightarrow^{t' \cdot t''} \mu''$.

Concerning c', from the inductive hypothesis on c, it follows that:

(H_6) $\mu =^{\Gamma}_{\tau} \mu'$.

(H_7) $\mathsf{filter}_{\tau}(\mathsf{t'}) = [\,]$.

Concerning c", from the inductive hypothesis on c, it follows that:

(H_8) $\mu' =^{\Gamma}_{\tau} \mu''$.

(H_9) $\mathsf{filter}_{\tau}(\mathsf{t}') = [\,]$.

By the transitive property of $=^{\Gamma}_{\tau}$ *and since* $\mu =^{\Gamma}_{\tau} \mu'$ *and* $\mu' =^{\Gamma}_{\tau} \mu''$*, we can conclude* $\mu =^{\Gamma}_{\tau} \mu''$.

In the semantics rule Sequence, $t' \cdot t''$ *denotes the concatenation of two traces* t' *and* t''*. Since* $\mathsf{filter}_{\tau}(\mathsf{t}') = [\,]$ *by (H_7) and* $\mathsf{filter}_{\tau}(\mathsf{t}'') = [\,]$ *by (H_9), we conclude that* $\mathsf{filter}_{\tau}(\mathsf{t}' \cdot \mathsf{t}'') = [\,]$.

While: while e do c**.** *For the typing rule While, we have that:*

(H_1) $\Gamma \vdash c : \tau'$ *cmd*.

(H_2) $\Gamma \vdash e : \tau'$.

By (H_1) and because the height of $\Gamma \vdash$ while e do c *is* $n+1$*, we know that the height of the typing derivation tree of (H_1) is* $\leq n$*. By applying the inductive hypothesis on c, we get that:*

(H_3) $\mu =^{\Gamma}_{\tau} \mu'$.

(H_4) $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

We want to prove that the inductive hypothesis works for while e do c *also. By the hypothesis of Lemma 2, we have that if:*

(H'_0) $\Gamma \vdash$ while e do c : τ' cmd.

(H'_1) $\mu \vdash$ while e do c $\Rightarrow^{\mathsf{t}\cdot\mathsf{t}'} \mu''$.

(H'_2) $\tau' \not\leq \tau$.

Then we want to prove that:

(G_1) $\mu =^{\Gamma}_{\tau} \mu''$.

(G_2) $\mathsf{filter}_{\tau}(\mathsf{t} \cdot \mathsf{t}') = [\,]$.

By the semantics rule of Loop, we have that:

(H'_3) $\mu \vdash e \Rightarrow v$.

(H'_4) $\mu \vdash c \Rightarrow^{t} \mu'$.

(H'_5) $\mu' \vdash$ while e do c $\Rightarrow^{\mathsf{t}'} \mu''$.

Base case. *The only possibility for the semantics tree to be of height $= 2$ is when $v = False$.*

$$\frac{\mu \vdash e \Rightarrow False}{\mu \vdash \textsf{while e do c} \Rightarrow^{[\,]} \mu}$$

By (H_3) we conclude that $\mu =^{\Gamma}_{\tau} \mu$. We also conclude that $\textsf{filter}_{\tau}([\,]) = [\,]$.

Assuming that our inductive hypothesis holds for the case of while when evaluating the height of the semantics tree $\leq m$, we want to prove the case of While with height $= m+1$.

$$\frac{\mu \vdash e \Rightarrow True \qquad (1)\ \mu \vdash c \Rightarrow^{t} \mu'' \qquad (H_6')\ \mu'' \vdash \textsf{while e do c}' \Rightarrow^{\textsf{t}'} \mu'}{\mu \vdash \textsf{while e do c}' \Rightarrow^{\textsf{t}\cdot\textsf{t}'} \mu'}$$

By applying the structural induction to (1), *we have that $\mu =^{\Gamma}_{\tau} \mu''$ and* $\textsf{filter}_{\tau}(\textsf{t}) = [\,]$. *By applying the inductive hypothesis on* while e do c *by (H_6'), we can conclude that $\mu'' =^{\Gamma}_{\tau} \mu'$ and* $\textsf{filter}_{\tau}(\textsf{t}') = [\,]$. *By applying the transitive property on $=^{\Gamma}_{\tau}$, we can conclude that $\mu'' =^{\Gamma}_{\tau} \mu'$ which is (G_1). Moreover, since* $\textsf{filter}_{\tau}(\textsf{t}) = [\,]$ *and* $\textsf{filter}_{\tau}(\textsf{t}') = [\,]$, *and $t \cdot t'$ is a concatenation, then* $\textsf{filter}_{\tau}(\textsf{t} \cdot \textsf{t}') = [\,]$ *which is (G_2).*

If: if e then c$'$ else c$''$**.** *For the typing rule If, we have that:*

(H_1) $\Gamma \vdash c' : \tau'$ *cmd.*

(H_2) $\Gamma \vdash c'' : \tau'$ *cmd.*

(H_3) $\Gamma \vdash e : \tau'$.

By (H_1) and the inductive hypothesis on c$'$, *we get that:*

(H_4) $\mu =^{\Gamma}_{\tau} \mu'$.

(H_5) $\textsf{filter}_{\tau}(\textsf{t}) = [\,]$.

By (H_1) and the inductive hypothesis on c$''$, *we get that:*

(H_6) $\mu =^{\Gamma}_{\tau} \mu'$.

(H_7) $\textsf{filter}_{\tau}(\textsf{t}') = [\,]$.

Concerning if e then c$'$ else c$''$, *by the hypothesis of Lemma 2, we have that if:*

(H_8) $\Gamma \vdash$ if e then c$'$ else c$''$: τ' cmd.

(H_9) $\mu \vdash$ if e then c$'$ else c$'' \Rightarrow^{\textsf{t}} \mu'$.

(H_{10}) $\mu \vdash$ if e then c$'$ else c$'' \Rightarrow^{\textsf{t}} \mu'$.

(H_{11}) $\tau' \not\leq \tau$.

Then we want to prove that:

(G_1) $\mu =^{\Gamma}_{\tau} \mu'$.

(G_2) $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

By the semantics rule of Branch- True, we have that:

(H_{12}) $\mu \vdash e \Rightarrow v$.

(H_{13}) $\mu \vdash c' \Rightarrow^{t} \mu'$.

(H_{14}) $t = [\,]$.

By (H_3) and (H_4) we can conclude that $\mu =^{\Gamma}_{\tau} \mu'$. *By ($H_5$) and ($H_{14}$) we conclude that* $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$.

By the semantics rule of Branch- False, we have that:

(H_{15}) $\mu \vdash e \Rightarrow v$.

(H_{16}) $\mu \vdash c'' \Rightarrow^{t} \mu'$.

(H_{17}) $t = [\,]$.

By (H_3) and (H_5) we can conclude that $\mu =^{\Gamma}_{\tau} \mu'$. *By ($H_6$) and ($H_{17}$) we conclude that* $\mathsf{filter}_{\tau}(\mathsf{t}) = [\,]$. □

3.3 Proof of Theorem 1

Proof. By Definition 3, for p to be NI^{Γ}_{τ}, we need to prove that:

(G_1) $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

(G_2) $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

We prove this theorem by induction on the height of typing derivation tree $\Gamma \vdash p$.

Case 1. *Base case: height = 2*

Subcase 1.1. $p \triangleq x := e'$.

Concerning (G_1) and (G_2), by Definition 3, we have that:

(H_1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(H_2) $\mu_i \vdash x := e' \Rightarrow^{t_i} \mu'_i$.

Moreover, for the semantics rule Update we have that:

(H_3) $t_i = [\,]$.

Hence, by the semantics rule Update and (H_2), we have that:

(H_4) $\mu_i \vdash e' \Rightarrow v_i$.

(H_5) $\mu'_i = \mu_i[x := v_i]$.

By the hypothesis of Theorem 1, we have:

(H_6) $\Gamma \vdash x := e' : \tau'$ *cmd.*

By (H_6) and the typing rule Assign, we have that:

(H_7) $\Gamma \vdash x : \tau'$ *var.*

(H_8) $\Gamma \vdash e' : \tau'$.

By (H_7) and the typing rule Var, we have:

(H_9) $\Gamma(x) = \tau'$ *var.*

Depending on τ', we have two cases:

(H_{10}) $\tau' \leq \tau$.

By Lemma 1 that can be applied due to (H_1), (H_8), (H_{10}) and (H_4) then (H_{11}): $v_0 = v_1$.
To prove (G_1), we rely on the Definition 1 that states that $\mu_0 =^{\Gamma}_{\tau} \mu_1$, if $\forall x \in \mu_0$ such that $\Gamma(x) = \tau'$ var and $\tau' \leq \tau$ then $\mu_0(x) = \mu_1(x)$.
Since (H_1) holds and the only variable in which μ_i and μ'_i are different is x by (H_5), then we need to prove that $\mu_0(x) = \mu_1(x)$ which holds by (H_{11}).
To prove (G_2) we rely on the Definition 2 and (H_3). Since $t_0 = t_1 = [\,]$ then $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

Since we proved (G_1) and (G_2), then p is NI^{Γ}_{τ}.

(H_{12}) $\tau' \not\leq \tau$.

To prove (G_1), we rely on the Definition 1 that states that $\mu_0 =^{\Gamma}_{\tau} \mu_1$, if $\forall y \in \mu_0$ such that $\Gamma(y) = \tau'$ var and $\tau' \leq \tau$ then $\mu_0(s) = \mu_1(y)$. For (H_{12}), we have that $\tau' \not\leq \tau$. Since we are only interested by variables with security level less or equal than τ, we can conclude by (H_5) and (H_1) that $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

To prove (G_2) we rely on the Definition 2 and (H_3). Since $t_0 = t_1 = [\,]$ then $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

Since we proved (G_1) and (G_2), then p is NI^{Γ}_{τ}.

Subcase 1.2. $p \overset{\Delta}{=} k := KG(\{n_1 \ldots n_i\}, k')$.

Concerning (G_1) and (G_2), by Definition 3, we have that:

(H_1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(H_2) $\mu_i \vdash k := KG(\{n_1 \ldots n_i\}, k') \Rightarrow^{t_i} \mu'_i$.

By the semantics rule Update we have that:

(H_3) $t_i = [\,]$.

Moreover, by the semantics rule Update and (H_2), we have that:

(H_4) $\mu_i \vdash KG(\{n_1 \ldots n_i\}, k') \Rightarrow v_i$.

(H_5) $\mu'_i = \mu_i[x := v_i]$.

By the hypothesis of Theorem 1, we have:

(H_6) $\Gamma \vdash k := KG(\{n_1 \ldots n_i\}, k') : \tau'$ *cmd*.

By the typing rule Kvar-Assign, we have that:

(H_7) $\Gamma(k) = \tau'$ $Kvar(n_1, \ldots, n_i)$.

(H_8) $\Gamma \vdash n_j : \tau'$ $Nvar(n_j)$.

(H_9) $\Gamma k' : \top$ *MKvar*.

To prove (G_1), we rely on the Definition 1 that states that $\mu_0 =^{\Gamma}_{\tau} \mu_1$, if $\forall y \in \mu_0$ such that $\Gamma(y) = \tau'$ var and $\tau' \leq \tau$ then $\mu_0(y) = \mu_1(y)$. For (H_7), $\Gamma(k) = \tau'$ $Kvar(n_1, \ldots, n_i)$. Since we are only interested in variables of type τ' var, we conclude by (H_5) and (H_1) that $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

To prove (G_2) we rely on the Definition 2 and (H_3). Since $t_0 = t_1 = [\,]$ then $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

Since we proved (G_1) and (G_2), then p is NI^{Γ}_{τ}.

Subcase 1.3. $p \overset{\Delta}{=} \mathsf{sbroadcast}(\{\mathsf{n}_1, \ldots, \mathsf{n}_\mathsf{j}\}, \mathsf{k}, \mathsf{e})$.

Concerning (G_1) and (G_2), by Definition 3, we have that:

(H_1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(H_2) $\mu_i \vdash \mathsf{sbroadcast}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{j}\},\mathsf{k},\mathsf{e}) \Rightarrow^{\mathsf{t}_\mathsf{i}} \mu_\mathsf{i}$.

By the semantics rule Secure Broadcast, we have:

(H_3) $t_i = \mathsf{BEnc}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{vk},\mathsf{v}')$.

(H_4) $\mu_i \vdash k \Rightarrow vk_i$.

(H_5) $\mu_i \vdash e \Rightarrow$ "m_i".

By the hypothesis of Theorem 1, we have:

(H_6) $\Gamma \vdash \mathsf{sbroadcast}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{j}\},\mathsf{k},\mathsf{e}) : \tau'\ \mathsf{cmd}$.

By (H_6) and the typing rule SBroadcast, we have:

(H_7) $\Gamma \vdash e : \tau'$.

(H_8) $\Gamma \vdash n_i :\ \tau\, Nvar(n_i)$.

(H_9) $\Gamma \vdash K : \tau\, Kvar(n_1,...,n_j)$.

To prove (G_1), we rely on the Definition 1 that states that $\mu_0 =^\Gamma_\tau \mu_1$, if $\forall y \in \mu_0$ such that $\Gamma(y) = \tau'$ var and $\tau' \leq \tau$ then $\mu_0(y) = \mu_1(y)$. For ($H_7 - H_9$), all the variables types are different than τ' var. Since we are only interested in variables of type τ' var, (G_1) follows by (H_1) and (H_2).

To prove (G_2), we have two cases:

(H_{10}) $\Gamma(n_1) \sqcap \Gamma(n_2) ... \sqcap \Gamma(n_k) \leq \tau'$

Being $\mu'_0 =^\Gamma_\tau \mu'_1$ and by (H_2) and(H_3) we have that $t_0 = t_1$. For an execution that starts with μ_0, $t_0 = \mathsf{BEnc}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{vk}_0,\mathsf{v}')$ and for an execution that starts with μ_1, $t_1 = \mathsf{BEnc}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{vk}_1,\mathsf{v}')$. By ($H_{11}$), $\Gamma(n_1) \sqcap \Gamma(n_2) ... \sqcap \Gamma(n_k) \leq \tau'$ then $\mathsf{filter}(\mathsf{t}_0) = [\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{v}']$ and $\mathsf{filter}(\mathsf{t}_1) = [\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{v}']$. It results that $\mathsf{filter}(\mathsf{t}_0) = \mathsf{filter}(\mathsf{t}_1)$ and therefore we prove (G_2).

Since we proved (G_1) and (G_2), then p is NI^Γ_τ.

(H_{11}) $\Gamma(n_1) \sqcap \Gamma(n_2) ... \sqcap \Gamma(n_k) \not\leq \tau'$

Being $\mu'_0 =^\Gamma_\tau \mu'_1$ and by (H_2) and(H_3) we have that $t_0 = t_1$. For an execution that starts with μ_0, $t_0 = \mathsf{BEnc}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{vk}_0,\mathsf{v}')$ and for an execution that starts with μ_1, $t_1 = \mathsf{BEnc}(\{\mathsf{n}_1,...,\mathsf{n}_\mathsf{k}\},\mathsf{vk}_1,\mathsf{v}')$. By ($H_{11}$), $\Gamma(n_1) \sqcap \Gamma(n_2) ... \sqcap \Gamma(n_k) \not\leq \tau'$, then $\mathsf{filter}_\tau(\mathsf{t}_0) = [\,]$ and $\mathsf{filter}_\tau(\mathsf{t}_1) = [\,]$. It results that $\mathsf{filter}_\tau(\mathsf{t}_0) = \mathsf{filter}_\tau(\mathsf{t}_1)$ and therefore we prove (G_2).

Since we proved (G_1) and (G_2), then p is NI^Γ_τ.

Case 2. Inductive case: height$= n+1$

Subcase 2.1. $p \stackrel{\Delta}{=} c';c''$.

We want to prove that $\Gamma \vdash c';c''$ is NI^Γ_τ. We assume that c' and c'' are of height $\leq n$, and $c';c''$ of height $n+1$.

For the typing rule Sequence, we have that:

(H$_1$*)* $\Gamma \vdash c' : \tau'$ *cmd.*

(H$_2$*)* $\Gamma \vdash c'' : \tau'$ *cmd.*

For the semantics rule Sequence, we have that:

(H$_3$*)* $\mu_i \vdash c' \Rightarrow^{t'_i} \mu'_i$.

(H$_4$*)* $\mu'_i \vdash c'' \Rightarrow^{t''_i} \mu''_i$.

Concerning c'*, from the inductive hypotheses, it follows that:*

(H$_5$*)* $\mu_{c'_0} =^{\Gamma}_{\tau} \mu_{c'_1}$.

(H$_6$*)* $\mu_{c'_0} \vdash c \Rightarrow^{t'_0} \mu'_{c'_0}$

(H$_7$*)* $\mu_{c'_1} \vdash c \Rightarrow^{t'_1} \mu'_{c'_1}$

(H$_8$*)* $\mu'_{c'_0} =^{\Gamma}_{\tau} \mu'_{c'_1}$.

(H$_9$*)* $\mathsf{filter}_\tau(\mathsf{t}'_0) = \mathsf{filter}_\tau(\mathsf{t}'_1)$.

It follows that c' *is* NI^{Γ}_{τ}*.*
Concerning c''*, from the inductive hypotheses, it follows that:*

(H$_{10}$*)* $\mu_{c''_0} =^{\Gamma}_{\tau} \mu_{c''_1}$.

(H$_{11}$*)* $\mu_{c''_0} \vdash c \Rightarrow^{t''_0} \mu'_{c''_0}$

(H$_{12}$*)* $\mu_{c''_1} \vdash c \Rightarrow^{t''_1} \mu'_{c''_1}$

(H$_{13}$*)* $\mu'_{c''_0} =^{\Gamma}_{\tau} \mu'_{c''_1}$.

(H$_{14}$*)* $\mathsf{filter}_\tau(\mathsf{t}''_0) = \mathsf{filter}_\tau(\mathsf{t}''_1)$.

It follows that c'' *is* NI^{Γ}_{τ}*.*
In the semantics rule Sequence, we use $t' \cdot t''$ *to denote the concatenation of two traces* t' *and* t''*. By (H*$_9$*) and (H*$_{14}$*) we have that* $t'_0 = t'_1$ *and* $t''_0 = t''_1$*, which implies that* $t'_0 \cdot t''_0 = t'_1 \cdot t''_1$*. We conclude that* $\mathsf{filter}_\tau(\mathsf{t}'_0 \cdot \mathsf{t}''_0) = \mathsf{filter}_\tau(\mathsf{t}'_1 \cdot \mathsf{t}''_1)$.
Since c'*,* c'' *are* NI^{Γ}_{τ} *and* $\mathsf{filter}_\tau(\mathsf{t}'_0 \cdot \mathsf{t}''_0) = \mathsf{filter}_\tau(\mathsf{t}'_1 \cdot \mathsf{t}''_1)$*, we conclude that* $\Gamma \vdash c';c''$ *is* NI^{Γ}_{τ}*.*

Subcase 2.2. $p \overset{\Delta}{=} \mathsf{while\ e\ do\ c}$.
For the typing rule While, we have that:

(H$_1$*)* $\Gamma \vdash c : \tau'$ *cmd.*

(H$_2$*)* $\Gamma \vdash e : \tau'$.

In the following, we show by induction that c is NI^{Γ}_{τ}.

By (H_1) *and because the height of* $\Gamma \vdash \mathsf{while\ e\ do\ c}$ *is* $n+1$, *we know that the height of the typing derivation tree of* (H_1) *is* $\leq n$. *Hence, we can apply the inductive hypothesis and get:*

(H₃) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(H₄) $\mu_0 \vdash c \Rightarrow^{t_0} \mu'_0$.

(H₅) $\mu_1 \vdash c' \Rightarrow^{t_1} \mu'_1$.

Then,

(H₆) $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

(H₇) $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

We want to prove that while e do c *is* NI^{Γ}_{τ}. *By the hypothesis of the theorem, we have that if:*

(H'_0) $\Gamma \vdash \mathsf{while\ e\ do\ c} : \tau'\ \mathsf{cmd}$.

(H'_1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.

(H'_2) $\mu_0 \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t}_0} \mu'_0$.

(H'_3) $\mu_1 \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t}_1} \mu'_1$.

Then, we want to prove that:

(G_0) $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

(G_1) $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

By (H'_2) *and the semantics rule of Loop, we have that:*

(H'_4) $\mu_0 \vdash e \Rightarrow v_0$.

By (H'_3) *and the semantics rule of Loop, we have that:*

(H'_5) $\mu_1 \vdash e \Rightarrow v_1$.

Depending on τ', *we have two cases:*

Subcase 2.2.1. $\Gamma \vdash e : \tau'$, $\tau' \leq \tau$.

(H'_6) $\tau' \leq \tau$.

In the case of (H'_6), *we check if we can apply the Subtype rule on While command. We would like to check if the command While of type* τ' *cmd can be typable as* τ'' *cmd, where* $\tau'' \not\leq \tau'$. *If the Subtype rule can be applied, the the proof of this case is analogous to Subcase 2.2.2.*

Otherwise, if the Subtype rule cannot be applied, then by Lemma 1 that can be applied on (H'_1), (H_2), (H'_6), (H'_4) *and* (H'_5), *we conclude that* $v_0 = v_1$.

We prove this case by induction on the height of the semantics tree of (H'_2). *(We do not show this formally, but we rely on the fact that the height of* (H'_2) *is equal to the height of* (H'_3) *by Lemma 1).*

Base case: height = 2. *The only possibility for the semantics tree to be of height* $= 2$ *is:*

- $v_0 = v_1 = False.$

$$\frac{\mu_0 \vdash e \Rightarrow False}{\mu_0 \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_0}} \mu_0}$$

$$\frac{\mu_1 \vdash e \Rightarrow False}{\mu_1 \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_1}} \mu_1}$$

We have that $\mu_0' =_\tau^\Gamma \mu_0$ *and* $\mu_1' =_\tau^\Gamma \mu_1$. *We conclude by* (H_1') *that* $\mu_0' =_\tau^\Gamma \mu_1'$. *Moreover, since* $t_0 = t_1 = [\,]$ *then* $\mathsf{filter}_\tau(\mathsf{t_0}) = \mathsf{filter}_\tau(\mathsf{t_1})$.

Assuming that our inductive hypothesis holds for the case of While when evaluating the height of semantics tree $\leq m$ *with* $\Gamma \vdash e : \tau', \tau' \leq \tau$, *let us prove the case of While with height* $= m+1$.

Inductive case: height = m+1.

$$\frac{\mu_0 \vdash e \Rightarrow True \qquad \mu_0 \vdash c \Rightarrow^{t_0} \mu_0'' \qquad (H_7')\ \mu_0'' \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_0'}} \mu_0'}{\mu_0 \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_0 \cdot t_0'}} \mu_0'}$$

The height of (H_2') *is* $m+1$.

$$\frac{\mu_1 \vdash e \Rightarrow True \qquad \mu_1 \vdash c \Rightarrow^{t_1} \mu_1'' \qquad (H_8')\ \mu_1'' \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_1'}} \mu_1'}{\mu_1 \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_1 \cdot t_1'}} \mu_1'}$$

The height of (H_3') *is* $m+1$.

By the previous induction on c, we have that $\mu_0'' =_\tau^\Gamma \mu_1''$ *and* $\mathsf{filter}_\tau(\mathsf{t_0}) = \mathsf{filter}_\tau(\mathsf{t_1})$, *through* $\mu_0 =_\tau^\Gamma \mu_1$, $\mu_0 \vdash c' \Rightarrow^{t_0} \mu_0''$ *and* $\mu_1 \vdash c' \Rightarrow^{t_1} \mu_1''$.

Moreover, by induction on while e do c *by* (H_7') *and* (H_8'), *we can conclude that* $\mu_0' =_\tau^\Gamma \mu_1'$ *which is already our goal* (G_1) *and* $\mathsf{filter}_\tau(\mathsf{t_0'}) = \mathsf{filter}_\tau(\mathsf{t_1'})$. *This is because we have that* $\mu_0'' =_\tau^\Gamma \mu_1''$ *and* $\mu_0'' \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_0'}} \mu_0'$ *and* $\mu_1'' \vdash \mathsf{while\ e\ do\ c} \Rightarrow^{\mathsf{t_1'}} \mu_1'$.

Since $t_i \cdot t_i'$ *is the concatenation of two traces* t_i *and* t_i', *and since* $\mathsf{filter}_\tau(\mathsf{t_0}) = \mathsf{filter}_\tau(\mathsf{t_1})$ *and* $\mathsf{filter}_\tau(\mathsf{t_0'}) = \mathsf{filter}_\tau(\mathsf{t_1'})$, *we conclude that* $\mathsf{filter}_\tau(\mathsf{t_0 \cdot t_0'}) = \mathsf{filter}_\tau(\mathsf{t_1 \cdot t_1'})$ *from which* (G_2) *follows.*

Since (G_1) *and* (G_2) *are satisfied, then* while e do c *is* NI_τ^Γ.

Subcase 2.2.2. $\Gamma \vdash e : \tau',\ \tau' \not\leq \tau.$

(H_9') $\tau' \not\leq \tau.$

By Lemma 2 that can be applied on (H_1), (H_4), (H_5) *and* (H'_9), *we conclude that* $\mu_i =^{\Gamma}_{\tau} \mu'_i$.

We prove that $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$ *by applying the transitive property on* $=^{\Gamma}_{\tau}$*:*

- *By* (H'_1) *we have that* (1) $\mu_0 =^{\Gamma}_{\tau} \mu_1$.
- *By Lemma 2 we have that* (2) $\mu_0 =^{\Gamma}_{\tau} \mu'_0$ *and* (3) $\mu_1 =^{\Gamma}_{\tau} \mu'_1$.
- *By* (1) *and* (2) *we have that* (4) $\mu_1 =^{\Gamma}_{\tau} \mu'_0$.
- *By* (1) *and* (3) *we have that* (5) $\mu_0 =^{\Gamma}_{\tau} \mu'_1$.
- *By* (4) *and* (5) *we conclude that* (6) $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

By applying structural induction, we have that $\mathsf{filter}_{\tau}(\mathsf{t}'_0) = \mathsf{filter}_{\tau}(\mathsf{t}'_1)$. *By previous induction on c we have that* $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$. *Since* $t_i \cdot t'_i$ *is the concatenation of two traces* t_i *and* t'_i, *and since* $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$ *and* $\mathsf{filter}_{\tau}(\mathsf{t}'_0) = \mathsf{filter}_{\tau}(\mathsf{t}'_1)$, *we conclude that* $\mathsf{filter}_{\tau}(\mathsf{t}_0 \cdot \mathsf{t}'_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1 \cdot \mathsf{t}'_1)$ *from which* (G_2) *follows.*

Since (G_1) *and* (G_2) *are satisfied, then* while e do c′ *is* NI^{Γ}_{τ}.

Subcase 2.3. $p \triangleq$ if e thenc′ else c″.

For the typing rule If, we have that:

(H_1) $\Gamma \vdash e : \tau'$.

(H_2) $\Gamma \vdash c' : \tau'$ *cmd.*

(H_3) $\Gamma \vdash c'' : \tau'$ *cmd.*

For the semantics rule Branch, we have that:

(H_4) $\mu_i \vdash e \Rightarrow v_i$.

(H_5) $\mu_i \vdash c' \Rightarrow_{t_i} \mu'_i$.

(H_6) $\mu_i \vdash c'' \Rightarrow_{t_i} \mu'_i$.

Concerning c′, by induction, we conclude that:

(H_7) $\mu'_{c'_0} =^{\Gamma}_{\tau} \mu'_{c'_1}$.

(H_8) $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

It follows that c′ is NI^{Γ}_{τ}.

Concerning c″, by induction, we conclude that:

(H_8) $\mu'_{c''_0} =^{\Gamma}_{\tau} \mu'_{c''_1}$.

(H_9) $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$.

It follows that c'' is NI^{Γ}_{τ}.

Depending on τ', we have two cases:

Subcase 2.3.1. $\Gamma \vdash e : \tau', \tau' \leq \tau$.

(H_{10}) $\tau' \leq \tau$.

In the case of (H_{10}), we check if we can apply the Subtype rule on If command. We would like to check if the command If of type τ' cmd can be typable as τ'' cmd, where $\tau'' \not\leq \tau'$. If the Subtype rule can be applied, the the proof of this case is analogous to Subcase 2.3.2.

Otherwise, if the Subtype rule cannot be applied, then by Lemma 1, we can conclude that:

(H_{11}) $v_0 = v_1$.

$$\frac{\mu_0 \vdash e \Rightarrow True \qquad \mu_0 \vdash c' \Rightarrow^{t_0} \mu_0'}{\mu_0 \vdash \qquad If\ e\ \mathsf{thenc'\ else\ c''} \Rightarrow^{\mathsf{t_0}} \mu_0'}$$

$$\frac{\mu_1 \vdash e \Rightarrow True \qquad \mu_1 \vdash c' \Rightarrow^{t_1} \mu_1'}{\mu_1 \vdash \qquad If\ e\ \mathsf{thenc'\ else\ c''} \Rightarrow^{\mathsf{t_1}} \mu_1'}$$

We have that $\mu_0' =^{\Gamma}_{\tau} \mu_0$ and $\mu_1' =^{\Gamma}_{\tau} \mu_1$. Since $\mu_0 =^{\Gamma}_{\tau} \mu_1$, we conclude that $\mu_0' =^{\Gamma}_{\tau} \mu_1'$. Since $t_0 = t_1 = [\,]$, then $\mathsf{filter}_{\tau}(\mathsf{t_0}) = \mathsf{filter}_{\tau}(\mathsf{t_1})$.

$$\frac{\mu_0 \vdash e \Rightarrow False \qquad \mu_0 \vdash c'' \Rightarrow^{t_0} \mu_0'}{\mu_0 \vdash \qquad If\ e\ \mathsf{thenc'\ else\ c''} \Rightarrow^{\mathsf{t_0}} \mu_0'}$$

$$\frac{\mu_1 \vdash e \Rightarrow false \qquad \mu_1 \vdash c'' \Rightarrow^{t_1} \mu_1'}{\mu_1 \vdash \qquad If\ e\ \mathsf{thenc'\ else\ c''} \Rightarrow^{\mathsf{t_1}} \mu_1'}$$

We have that $\mu_0' =^{\Gamma}_{\tau} \mu_0$ and $\mu_1' =^{\Gamma}_{\tau} \mu_1$. Since $\mu_0 =^{\Gamma}_{\tau} \mu_1$, we conclude that $\mu_0' =^{\Gamma}_{\tau} \mu_1'$. Since $t_0 = t_1 = [\,]$, then $\mathsf{filter}_{\tau}(\mathsf{t_0}) = \mathsf{filter}_{\tau}(\mathsf{t_1})$.

Subcase 2.3.2. $\Gamma \vdash e : \tau', \tau' \not\leq \tau$.

(H_{12}) $\tau' \not\leq \tau$.

By Lemma 2, we conclude that $\mu_i =^{\Gamma}_{\tau} \mu_i'$.

We prove that $\mu_0' =^{\Gamma}_{\tau} \mu_1'$ by applying the transitive property on $=^{\Gamma}_{\tau}$.

- *We have that (1) $\mu_0' =^{\Gamma}_{\tau} \mu_0$.*
- *We have that (2) $\mu_1' =^{\Gamma}_{\tau} \mu_1$.*

- *We have that (3)* $\mu_0 =^{\Gamma}_{\tau} \mu_1$.
- *By (1) and (3) we have that (4)* $\mu_1 =^{\Gamma}_{\tau} \mu'_0$.
- *By (2) and (3) we have that (5)* $\mu_0 =^{\Gamma}_{\tau} \mu'_1$.
- *By (4) and (5) we conclude that (6)* $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$.

Since $\mathsf{filter}_{\tau}(\mathsf{t}_0) = \mathsf{filter}_{\tau}(\mathsf{t}_1)$ *and* $\mu'_0 =^{\Gamma}_{\tau} \mu'_1$*, then* p *is* NI^{Γ}_{τ}.

□