Proofs of section 4.3.1

While both semantics are defined on standard vectors, we need to consider partial vectors for compositional reasoning. Unfortunately, the operational semantics lacks a concept of partiality. To address this issue, we introduce a dummy statement and a dummy store to fill the missing components. For readability, we denote standard vectors as Σ and partial vectors as Ω. We use $\Omega_1 \equiv \Omega_2$ to indicate that Ω_1 and Ω_2 hide the same components. We note *empty*(Ω) (resp. *full*(Ω)) if all components are hidden (resp. visible). We note *partial*(Ω) is Ω is neither empty nor full. Finally, we note $\mathcal{L}(\Sigma)$ for the casting of a standard vector to a full partial vector. Recall the definition of replicate which we denote as follows: $(\!|\cdot,\cdot|\!)$ for short

$$(\!|s, \langle \sigma_0, \ldots, \sigma_{p-1} \rangle|\!) = \langle (s, \sigma_0), \ldots, (s, \sigma_{p-1}) \rangle$$

We define dummy replication, noted $(\!|\cdot,\cdot|\!)^\sharp$, which fills empty positions with dummy pairs.

$$(\!|s, \langle \omega_0, \ldots, \omega_{p-1} \rangle|\!)^\sharp = \langle \gamma_0, \ldots, \gamma_{p-1} \rangle$$

where $\gamma_i = (s, \omega_i)$ if $\omega_i \neq \mathbf{0}$ and $\gamma_i = (\texttt{skip}, [\,])$ otherwise. Similarly, we note $\mathcal{C}(\Omega)$ the standard vector obtain by replacing hidden components with the dummy store $[\,]$. Now, recall the definition of the operation semantics

$$[\![s]\!]_{op} = \{(\Sigma_1, \Sigma_2) \mid (\!|s, \Sigma_1|\!) \Rightarrow \Gamma \wedge \Gamma \rightarrow \Sigma_2\}$$

We define an alternative semantics $[\![\cdot]\!]^\sharp_{op}$ mapping partial vectors to standard vectors.

$$[\![s]\!]^\sharp_{op} = \{(\Omega_1, \Sigma_2) \mid (\!|s, \Omega_1|\!)^\sharp \Rightarrow \Gamma \wedge \Gamma \rightarrow \Sigma_2\}$$

Note that all semantics $[\![\cdot]\!]$, $[\![\cdot]\!]_{op}$ and $[\![]\!]^\sharp_{op}$ are deterministic. We introduce a property *alignedfor* which states that a statement is textually aligned for a given state vector.

$$\begin{aligned} \mathit{alignedfor}_\Sigma\ (s)\ \ = \\ \forall \Gamma.\ \mathit{reachable}((\!|s, \Sigma|\!), \Gamma) \Rightarrow \\ \forall i, j < p.\ \left(p, i \vdash \pi_i(\Gamma) \xrightarrow{\delta_i}_{\alpha_i} \gamma_i \wedge p, j \vdash \pi_j(\Gamma) \xrightarrow{\delta_j}_{\alpha_j} \gamma_j \right) \Rightarrow \delta_i = \delta_j \end{aligned}$$

The definition of *aligned*(s) is then simply $\forall \Sigma.\ \mathit{alignedfor}_\Sigma\ (s)$.

Our results rely on auxiliary lemmas, proven in the Coq development from the previous section. These lemmas assert that textually aligned programs enjoy proper composition properties and that both replication and textual alignment are maintained throughout computations. See lemma 1, lemma 2, proposition 1, proposition 2 and proposition 3.

Lemma 1. *For all s_1, s_2, Σ_1 and Σ_2 such that $aligned for_{\Sigma_1}\ (s_1; s_2)$, we have*

$$\exists \Sigma_3,\ (\Sigma_1, \Sigma_3) \in [\![s_1]\!]_{op} \wedge (\Sigma_3, \Sigma_2) \in [\![s_2]\!]_{op} \qquad \textit{if and only if} \qquad (\Sigma_1, \Sigma_2) \in [\![s_1; s_2]\!]_{op}$$

Lemma 2. *For all e, s_1, s_2, Σ and Σ_a such that $aligned for_{\Sigma}$ (`if` e `then` s_1 `else` s_2 `end`), and e evaluates to true at all position in Σ, we have*

$$(\Sigma, \Sigma_a) \in [\![\texttt{if}\ e\ \texttt{then}\ s_1\ \texttt{else}\ s_2\ \texttt{end}]\!]_{op} \qquad \textit{if and only if} \qquad (\Sigma, \Sigma_a) \in [\![s_1]\!]_{op}$$

Proposition 1. *For all s, s' and Σ, Σ' such that $aligned for_{\Sigma}\ (s)$ if $(\!|s, \Sigma|\!) \Rightarrow (\!|s', \Sigma'|\!)$ then we have $aligned for_{\Sigma'}\ (s')$.*

Proposition 2. *For all s, Σ and C such that $aligned for_{\Sigma}\ (s)$, if $(\!|s, \Sigma|\!) \Rightarrow C$ then exists s' such that $C = (\!|s', map\ snd\ C|\!)$.*

Proposition 3. *For all s and Σ such that $aligned for_{\Sigma}\ (s)$,*

- *if $s = s_1; s_2$ then $aligned for_{\Sigma}\ (s_1)$*
- *if $s =$ `if` e `then` s_1 `else` s_2 `end` then $aligned for_{\Sigma}\ (s_1)$ and $aligned for_{\Sigma}\ (s_2)$*
- *if $s =$ `while` e `do` s `end` then $aligned for_{\Sigma}\ (s)$*

Before proving our main theorems, we introduce a few auxiliary results. The lemma 3 states that the denotational semantics maps partial vectors to compatible partial vectors. The proposition proposition 4 states that the denotational semantics is defined only for textually aligned programs. The lemma 5 states that computations on partial vectors must complete in one step.

Lemma 3. *For all s, Ω_1 and Ω_2, if $(\Omega_1, \Omega_2) \in [\![s]\!]$ then $\Omega_1 \equiv \Omega_2$.*

Proof sketch. Simple induction on the derivation of $(\Omega_1, \Omega_2) \in [\![s]\!]$ by the semantics rules. □

Proposition 4. *For all s, Σ and Ω, if $(\mathcal{L}(\Sigma), \Omega) \in [\![s]\!]$ then $aligned for_{\Sigma}\ (s)$.*

Proof sketch. By induction on the derivation of $(\Omega_1, \Omega_2) \in [\![s]\!]$. Note that by definition of the denotational semantics, which by construction must follows the global control flow of the program when synchronizations occur, synchronization are limited to empty or full vectors. Thus supposing that $aligned for_{\Sigma}\ (s)$ does not hold leads to a contradiction, the denotational semantics cannot be defined. □

Lemma 4. *For all s, $(\Omega_{\mathbf{0}}, \Omega_{\mathbf{0}}) \in [\![s]\!]^{\sharp}_{op}$.*

Proof sketch. Obvious as we start the execution from $(\!|s, \Omega_{\mathbf{0}}|\!)^{\sharp} = (\!|\texttt{skip}, \langle [\,], \ldots, [\,] \rangle|\!)$ □

Lemma 5. *For all s, Ω and Σ such that $partial(\Omega)$, if $(\Omega, \Sigma) \in [\![s]\!]^{\sharp}_{op}$ then $(\!|s, \Omega|\!)^{\sharp} \to \Sigma$.*

Proof sketch. As the vector is partial, at least one ofthe components is a `skip` command. Thus no global synchronization can occur. □

We now turn our attention to our main result, which assert that the operational and denotational semantics coincide for textually aligned programs.

Lemma 6. *For all s, Σ and Σ' we have $(\Sigma, \Sigma') \in [\![s]\!]_{op}$ if and only if $(\mathcal{L}(\Sigma), \Sigma') \in [\![s]\!]^\sharp_{op}$.*

Proof sketch. Obvious by definition. □

Lemma 7. *For all s_1, s_2, Ω_1, Ω_2 and Ω_3 such that*

- *if full(Ω_1) then alignedfor$_{\Omega_1}$ $(s_1; s_2)$*
- *$\Omega_1 \equiv \Omega_2$ and $\Omega_2 \equiv \Omega_3$*
- *$(\Omega_1, \mathcal{C}(\Omega_2)) \in [\![s_1]\!]^\sharp_{op}$ and $(\Omega_2, \mathcal{C}(\Omega_3)) \in [\![s_2]\!]^\sharp_{op}$*

we have $(\Omega_1, \mathcal{C}(\Omega_3)) \in [\![s_1; s_2]\!]^\sharp_{op}$.

Proof. We distinguish three cases. If *empty*(Ω_1) then the result is immediate by compatibility and by lemma 4. If *full*(Ω_1) then by compatibility and by definition of *full* there exists Σ_1, Σ_2 and Σ_3 such that $\Omega_1 = \mathcal{L}(\Sigma_1)$, $\Omega_2 = \mathcal{L}(\Sigma_2)$ and $\Omega_3 = \mathcal{L}(\Sigma_3)$. As $\mathcal{C}(\cdot) \circ \mathcal{L}(\cdot)$ is the identity function, and by application of lemma 6 we have $(\Sigma_1, \Sigma_2) \in [\![s_1]\!]_{op}$ and $(\Sigma_2, \Sigma_3) \in [\![s_1]\!]_{op}$. By lemma 1 we have $(\Sigma_1, \Sigma_3) \in [\![s_1; s_2]\!]_{op}$. We conclude by lemma 6. Finally, suppose that *partial*(Ω_1). By hypothesis and by lemma 5 we have $(\!|s_1, \Omega_1|\!)^\sharp \to \mathcal{C}(\Omega_2)$ and $(\!|s_2, \Omega_2|\!)^\sharp \to \mathcal{C}(\Omega_3)$. By definition of the semantics and observation of pointwise reductions we have $(\!|s_1; s_2, \Omega_1|\!)^\sharp \to \mathcal{C}(\Omega_3)$ and then $(\Omega_1, \mathcal{C}(\Omega_3)) \in [\![s_1; s_2]\!]^\sharp_{op}$. □

Lemma 8. *For all e, s_1, s_2, Ω, Ω_a and Ω_b and such that*

- *if full(Ω) then alignedfor$_\Omega$ (`if` e `then` s_1 `else` s_2 `end`)*
- *$\delta_e(\Omega) \equiv \Omega_a$ and $\delta_{!e}(\Omega) \equiv \Omega_b$*
- *$(\delta_e(\Omega), \mathcal{C}(\Omega_a)) \in [\![s_1]\!]^\sharp_{op}$ and $(\delta_{!e}(\Omega), \mathcal{C}(\Omega_b)) \in [\![s_2]\!]^\sharp_{op}$*

we have $(\Omega,\ \mathcal{C}(\Omega_a \parallel \Omega_b)) \in [\![$`if` e `then` s_1 `else` s_2 `end`$]\!]^\sharp_{op}$

Proof. We distinguish six cases.

1. If *empty*(Ω) then the result is immediate by compatibility and by lemma 4.
2. Otherwise, if *full*$(\delta_e(\Omega))$ then there exists Σ and Σ_a such that $\Omega = \delta_e(\Omega) = \mathcal{L}(\Sigma)$, $\Omega_a = \mathcal{L}(\Sigma_a)$ and $\Omega_b = \Omega_{\mathbf{0}}$. By lemma 6 we have $(\Sigma, \Sigma_a) \in [\![s_1]\!]_{op}$ and then, by lemma 2, it comes $(\Sigma, \Sigma_a) \in [\![$`if` e `then` s_1 `else` s_2 `end`$]\!]_{op}$. By applying lemma 6 again, we obtain $(\mathcal{L}(\Sigma), \Sigma_a) \in [\![$`if` e `then` s_1 `else` s_2 `end`$]\!]^\sharp_{op}$. Now, by $\Omega_a = \mathcal{L}(\Sigma_a)$ and $\Omega_b = \Omega_{\mathbf{0}}$, we have $\mathcal{C}(\Omega_a \parallel \Omega_b) = \mathcal{C}(\Omega_a) = \Sigma_a$. Finally, we have $(\Omega, \mathcal{C}(\Omega_a \parallel \Omega_b)) \in [\![$`if` e `then` s_1 `else` s_2 `end`$]\!]^\sharp_{op}$.

3. Otherwise, if $full(\delta_{!e}(\Omega))$ the proof is similar to the previous case.

4. Otherwise, if $empty(\delta_{!e}(\Omega))$ then $\delta_e(\Omega)$ is partial and by lemma 5 we have $(\!|s_1, \delta_e(\Omega)|\!)^\sharp \to \mathcal{C}(\Omega_a)$. As e evaluates to true on all visible components of $\delta_e(\Omega)$, and by applying usual properties of conditionals pointwise, it comes $(\!|\texttt{if } e \texttt{ then } s_1 \texttt{ else } s_2 \texttt{ end}, \delta_e(\Omega)|\!)^\sharp \to \mathcal{C}(\Omega_a)$. Obviously, we also have $(\!|\texttt{if } e \texttt{ then } s_1 \texttt{ else } s_2 \texttt{ end}, \delta_{!e}(\Omega)|\!)^\sharp \to \mathcal{C}(\Omega_{\mathbf{0}})$ and $\mathcal{C}(\Omega_a \parallel \Omega_{\mathbf{0}}) = \mathcal{C}(\Omega_a)$. It is then immediate, by observation of point wise reductions, that $(\!|\texttt{if } e \texttt{ then } s_1 \texttt{ else } s_2 \texttt{ end}, \Omega|\!)^\sharp \to \mathcal{C}(\Omega_a \parallel \Omega_b)$ and then $(\Omega, \mathcal{C}(\Omega_a \parallel \Omega_b)) \in [\![\texttt{if } e \texttt{ then } s_1 \texttt{ else } s_2 \texttt{ end}]\!]^\sharp_{op}$.

5. Otherwise, if $empty(\delta_e(\Omega))$ the proof is similar to the previous case.

6. Otherwise we have $partial(\delta_e(\Omega))$ and $partial(\delta_{!e}(\Omega))$. This case is similar to the last two, it relies on the fact that both branches being partial none of them can perform a synchronization. Again the result is obtained by combining pointwise reductions which is possible thanks to compatibility.

□

Lemma 9. *For all b and s we have* $(\!|\texttt{while } b \texttt{ do } s \texttt{ end}, \Omega_{\mathbf{0}}|\!)^\sharp \to \mathcal{C}(\Omega_{\mathbf{0}})$

Proof sketch. Obvious. □

Lemma 10. *Forall b, s, Ω_1 and Ω_2 such that*

- *if $full(\Omega_1)$ then $aligned for_{\Omega_1}$ $(\texttt{while } b \texttt{ do } s \texttt{ end})$*
- $\delta_b(\Omega_1) \equiv \Omega_2$
- $(\delta_b(\Omega_1), \mathcal{C}(\Omega_2)) \in [\![s; \texttt{while } b \texttt{ do } s \texttt{ end}]\!]^\sharp_{op}$

we have $(\delta_b(\Omega_1), \mathcal{C}(\Omega_2)) \in [\![\texttt{while } b \texttt{ do } s \texttt{ end}]\!]^\sharp_{op}$

Proof sketch. The proof is similar to proofs of lemma 7 and lemma 8. □

Lemma 11. *For all b, s, Ω_1 , Ω_2 such that*

- *if $full(\Omega_1)$ then $aligned for_{\Omega_1}$ $(\texttt{while } b \texttt{ do } s \texttt{ end})$*
- $\delta_b(\Omega_1) \equiv \Omega_2$
- $(\delta_b(\Omega_1), \mathcal{C}(\Omega_2)) \in [\![\texttt{while } b \texttt{ do } s \texttt{ end}]\!]^\sharp_{op}$

we have $(\Omega_1, \mathcal{C}(\Omega_2 \parallel \delta_{!b}(\Omega_1))) \in [\![\texttt{while } b \texttt{ do } s \texttt{ end}]\!]^\sharp_{op}$.

Proof sketch. By induction on the derivation of $(\delta_b(\Omega_1), \mathcal{C}(\Omega)) \in [\![\texttt{while } b \texttt{ do } s \texttt{ end}]\!]^\sharp_{op}$. □

Proposition 5. *For all s, Ω_1 and Ω_2 we have* $(\Omega_1, \Omega_2) \in [\![s]\!] \Rightarrow (\Omega_1, \mathcal{C}(\Omega_2)) \in [\![s]\!]^\sharp_{op}$

Proof. The proof is by structural induction on s.

- If s is `skip` then, by examination of the definition of $[\![\cdot]\!]$, we have $\Omega_2 = \Omega_1$. By reflexivity of $\Rightarrow$, we have (i) $(\!|\texttt{skip}, \Omega_1|\!)^\sharp \Rightarrow (\!|\texttt{skip}, \Omega_1|\!)^\sharp$. By observation of pointwise reductions it is immediate that (ii) $(\!|\texttt{skip}, \Omega_1|\!)^\sharp \to \Omega_1$. By (i), (ii) and by definition of $[\![\cdot]\!]^\sharp_{op}$ we have $(\Omega_1, \Omega_1) \in [\![\texttt{skip}]\!]^\sharp_{op}$. The proof is similar if s is $x := e$.

- If s is `sync` then, by examination of the definition of $[\![\cdot]\!]$, we have $\neg \mathit{partial}(\Omega_1)$ and $\Omega_2 = \Omega_1$. We distinguish two cases.

  - If $\mathit{empty}(\Omega_1)$ then, by definition of $(\!|\cdot, \cdot|\!)^\sharp$, it is immediate that $(\!|\texttt{sync}, \Omega_1|\!)^\sharp = (\!|\texttt{skip}, \Omega_1|\!)^\sharp$. Thus, by reflexivity of $\Rightarrow$ we have (i) $(\!|\texttt{sync}, \Omega_1|\!)^\sharp \Rightarrow (\!|\texttt{skip}, \Omega_1|\!)^\sharp$ and it is immediate that (ii) $(\!|\texttt{skip}, \Omega_1|\!)^\sharp \to \Omega_1$. By (i), (ii) and by definition of $[\![\cdot]\!]^\sharp_{op}$ we have $(\Omega_1, \Omega_1) \in [\![\texttt{sync}]\!]^\sharp_{op}$.
  - If $\mathit{full}(\Omega_1)$ then there exists Σ such $\Omega_1 = \mathcal{L}(\Sigma)$. From this equality, we have We have $(\!|\texttt{sync}, \Omega_1|\!)^\sharp = (\!|\texttt{sync}, \Sigma|\!)$ and $(\!|\texttt{skip}, \Omega_1|\!)^\sharp = (\!|\texttt{skip}, \Sigma|\!)$. By observation of pointwise reductions we have (i) $(\!|\texttt{sync}, \Omega_1|\!)^\sharp \Rightarrow (\!|\texttt{skip}, \Omega_1|\!)^\sharp$ and (ii) $(\!|\texttt{skip}, \Omega_1|\!)^\sharp \to \Omega_1$. By (i), (ii) and by definition of $[\![\cdot]\!]^\sharp_{op}$ we have $(\Omega_1, \Omega_1) \in [\![\texttt{sync}]\!]^\sharp_{op}$.

- If s is $s_1; s_2$ then there exists Ω such that $(\Omega_1, \Omega) \in [\![s_1]\!]$ and $(\Omega, \Omega_2) \in [\![s_2]\!]$. By proposition 3, we have $\mathit{alignedfor}_\Omega\ (s_1)$. By induction hypothesis, we have $(\Omega_1, \mathcal{C}(\Omega)) \in [\![s_1]\!]^\sharp_{op}$ and $(\Omega, \mathcal{C}(\Omega_2)) \in [\![s_2]\!]^\sharp_{op}$. By lemma 3, we have $\Omega_1 \equiv \Omega$ and $\Omega \equiv \Omega_2$. By lemma 7 it comes $(\Omega_1, \Omega_2) \in [\![s1; s_2]\!]^\sharp_{op}$

- If s is `if` e `then` s_1 `else` s_2 `end` then there exists Σ_a and Σ_b such that $(\delta_b(\Omega_1), \Omega_a) \in [\![s_1]\!]$, $(\delta_{!b}(\Omega_1), \Omega_b) \in [\![s_2]\!]$ and $\Omega_2 = \Omega_a \parallel \Omega_b$. By proposition 3 we have $\mathit{alignedfor}_\Omega\ (s_1)$ and $\mathit{alignedfor}_\Omega\ (s_2)$. By induction hypothesis, we have $(\delta_b(\Omega_1), \mathcal{C}(\Omega_a)) \in [\![s_1]\!]^\sharp_{op}$ and $(\delta_{!b}(\Omega_1), \mathcal{C}(\Omega_b)) \in [\![s_2]\!]^\sharp_{op}$. By lemma 3, we have $\delta_b(\Omega_1) \equiv \Omega_a$ and $\delta_{!b}(\Omega_1) \equiv \Omega_b$. Thus by lemma 8, we have $(\Omega_1, \mathcal{C}(\Omega_2)) \in [\![\texttt{if}\ e\ \texttt{then}\ s_1\ \texttt{else}\ s_2\ \texttt{end}]\!]^\sharp_{op}$.

- If s is `while` e `do` s_0 `end` then we proceed by induction on the derivation of $\mathit{loop}\ b\ \Omega_1\ \Omega_2$. Suppose that $\neg \mathit{empty}(\delta_e(\Omega_1))$. Otherwise the result is immediate by lemma 9. There exists Ω_a and Ω_b such that $(\delta_e(\Omega_1), \Omega_a) \in [\![s]\!]$ and $\mathit{loop}\ b\ s\ \Omega_a\ \Omega_b$. By lemma 3, we have $\delta_e(\Omega_1) \equiv \Omega_a$ and $\Omega_a \equiv \Omega_b\Omega_2$. By the two induction hypothesis, we have $(\delta_e(\Omega_1), \mathcal{C}(\Omega_a)) \in [\![s]\!]^\sharp_{op}$ and $(\Omega_a, \mathcal{C}(\Omega_b)) \in [\![\texttt{while}\ e\ \texttt{do}\ s\ \texttt{end}]\!]^\sharp_{op}$. Then by lemma 7 we have $(\delta_e(\Omega_1), \mathcal{C}(\Omega_b)) \in [\![s; \texttt{while}\ e\ \texttt{do}\ s\ \texttt{end}]\!]^\sharp_{op}$. By lemma 10 it comes $(\delta_e(\Omega_1), \mathcal{C}(\Omega_b)) \in [\![\texttt{while}\ e\ \texttt{do}\ s\ \texttt{end}]\!]^\sharp_{op}$. We conclude by lemma 11.

□

Proposition 6. *For all s, Σ_1 and Σ_2 we have $(\mathcal{L}(\Sigma_1), \Omega_2) \in [\![s]\!]^\sharp_{op} \Rightarrow (\Sigma_1, \Sigma_2) \in [\![s]\!]_{op}$*

Proof. Trivial, using the equality $(\!|s, \mathcal{L}(\Sigma)|\!)^\sharp = (\!|s, \Sigma|\!)$. □

Theorem 1. *For all s, Σ_1 and Σ_2, if $(\mathcal{L}(\Sigma_1), \mathcal{L}(\Sigma_2)) \in [\![s]\!]$ then $(\Sigma_1, \Sigma_2) \in [\![s]\!]_{op}$*

Proof. Suppose that $(\mathcal{L}(\Sigma_1), \mathcal{L}(\Sigma_2)) \in [\![s]\!]$. By proposition 5 we have $(\mathcal{L}(\Sigma_1), \mathcal{C}(\mathcal{L}(\Sigma_2))) \in [\![s]\!]^\sharp_{op}$. But $\mathcal{C}(\mathcal{L}(\Sigma_2)) = \mathcal{L}(\Sigma_2)$ and then by proposition 6 we have $(\Sigma_1, \Sigma_2) \in [\![s]\!]_{op}$. □

The second simulation result is stated by the following theorem. We omit the proof which, as the proof of the previous theorem, relies on the compositionality properties of textually aligned programs stated at the beginning of this section.

Theorem 2. *For all* s, Σ_1 *and* Σ_2, *if* $aligned for_{\Sigma_1}(s)$ *and* $(\Sigma_1, \Sigma_2) \in [\![s]\!]_{op}$ *then* $(\mathcal{L}(\Sigma_1), \mathcal{L}(\Sigma_2)) \in [\![s]\!]$