Appendix A

Proofs of Lemmas in Chapter 3

Proof of Lemma 3.2.9

Let $F$ be a function on sets of models defined by $F(X) = SEQ(M(G), X)$. We prove $M(*G) \neq \emptyset$ and $F(M(*G)) = M(*G)$. Note that this implies $M(*G) = M(G; *G) = SEQ(M(G), M(*G))$. In the proof we use sets of models $Y_1$ and $Y_2$, given by

$Y_1 = \{ \sigma \mid \text{there exists a } k \in \mathbb{N}, k \geq 1 \text{ and models } \sigma_1, \ldots, \sigma_k \text{ such that} \}
\begin{align*}
    \sigma &= \sigma_1 \cdots \sigma_k, \text{ with } \sigma_i \in M(G), \text{ for } i \in \{1, \ldots, k\}, \\
    |\sigma_i| &< \infty, \text{ for } i \in \{1, \ldots, k-1\}, \text{ and } |\sigma_k| = \infty \}
\end{align*}$

and

$Y_2 = \{ \sigma \mid \text{there exists an infinite sequence of models } \sigma_1, \sigma_2, \ldots \text{ such that} \}
\begin{align*}
    \sigma &= \sigma_1 \sigma_2 \cdots, \text{ with } \sigma_i \in M(G) \text{ and } |\sigma_i| < \infty, \text{ for } i \geq 1 \}
\end{align*}$

Then, by definition, $M(*G) = Y_1 \cup Y_2$. First we prove that $M(*G) \neq \emptyset$. Observe that $M(G) \neq \emptyset$. Let $\sigma \in M(G)$. If $|\sigma| < \infty$ then $\sigma \sigma \sigma \cdots \in Y_2$. If $|\sigma| = \infty$ then $\sigma \in Y_1$. Next we show that $F(M(*G)) = M(*G)$ by proving $M(*G) \subseteq F(M(*G))$ and $F(M(*G)) \subseteq M(*G)$.

To prove $M(*G) \subseteq F(M(*G))$, consider $\sigma \in M(*G)$.

- If $\sigma \in Y_1$, then $\sigma = \sigma_1 \cdots \sigma_k$.
  - If $k = 1$ then $\sigma \in M(G)$ and $|\sigma| = \infty$, thus $\sigma \in SEQ(M(G), M(*G)) = F(M(*G))$.
  - If $k > 1$ then $\sigma_2 \cdots \sigma_k \in Y_1 \subseteq M(*G)$, $\sigma_1 \in M(G)$ and $|\sigma| < \infty$, thus $\sigma = \sigma_1 \sigma_2 \cdots \sigma_k \in SEQ(M(G), M(*G)) = F(M(*G))$.

- If $\sigma \in Y_2$ then $\sigma_2 \sigma_3 \cdots \in Y_2 \subseteq M(*G)$, $\sigma_1 \in M(G)$ and $|\sigma| < \infty$, thus $\sigma = \sigma_1 \sigma_2 \sigma_3 \cdots \in SEQ(M(G), M(*G)) = F(M(*G))$. 
To prove \( F(\mathcal{M}(\star G)) \subseteq \mathcal{M}(\star G) \), consider \( \sigma \in F(\mathcal{M}(\star G)) \). Then \( \sigma \) is well-formed and \( \sigma \in SEQ(\mathcal{M}(G), \mathcal{M}(\star G)) \). Thus, either

1. \( \sigma \in \mathcal{M}(G) \) and \( |\sigma| = \infty \), and thus \( \sigma \in Y_1 \subseteq \mathcal{M}(\star G) \), or
2. \( \sigma = \sigma_1 \sigma_2 \) with \( \sigma_1 \in \mathcal{M}(G) \), \( |\sigma_1| < \infty \), and \( \sigma_2 \in \mathcal{M}(\star G) \). Then \( \sigma_1 \sigma_2 \in \mathcal{M}(\star G) \).

**Proof of Lemma 3.2.11**

We prove that parallel composition is commutative and associative, that is, \( \mathcal{M}(S_1 || S_2) = \mathcal{M}(S_2 || S_1) \), and \( \mathcal{M}((S_1 || S_2) || S_3) = \mathcal{M}(S_1 || (S_2 || S_3)) \). Commutativity follows easily from the definition, since \( \text{max} \) is commutative and the other clauses are symmetric. In the proof of associativity we use the following lemma which follows easily from the definitions.

**Lemma A.0.1** For all models \( \sigma, \sigma_1, \sigma_2 \), for all \( cset, cset_1, cset_2 \subseteq DCHAN \), and for all \( \tau \in TIME \),

1. \( [[\sigma]_{cset_1}]_{cset_2} = [\sigma]_{cset_1 \cap cset_2} \)
2. \( dch([\sigma]_{cset}) \subseteq cset \)
3. \( [\sigma]_{cset_1 \cup cset_2}(\tau) = [\sigma]_{cset_1}(\tau) \cup [\sigma]_{cset_2}(\tau). \)

To prove associativity, consider \( \sigma \in \mathcal{M}((S_1 || S_2) || S_3) \).
Then \( dch(\sigma) \subseteq dch(S_1 || S_2) \cup dch(S_3) \), and thus

\[
dch(\sigma) \subseteq dch(S_1) \cup dch(S_2) \cup dch(S_3) \quad (A.1)
\]

Furthermore, there exist \( \sigma_{12} \) and \( \sigma_3 \) such that

\[
\sigma_{12} \in \mathcal{M}(S_1 || S_2) \quad (A.2)
\]
\[
\sigma_3 \in \mathcal{M}(S_3) \quad (A.3)
\]
\[
[\sigma]_{dch(S_1) \cup dch(S_2)}(\tau) = \begin{cases} 
\sigma_{12}(\tau) & \text{for all } \tau < |\sigma_{12}| \\
\emptyset & \text{for all } \tau, |\sigma_{12}| \leq \tau < |\sigma|
\end{cases} \quad (A.4)
\]
\[
[\sigma]_{dch(S_3)}(\tau) = \begin{cases} 
\sigma_3(\tau) & \text{for all } \tau < |\sigma_3| \\
\emptyset & \text{for all } \tau, |\sigma_3| \leq \tau < |\sigma|
\end{cases} \quad (A.5)
\]
\[
|\sigma| = \text{max}(|\sigma_{12}|, |\sigma_3|) \quad (A.6)
\]
\[
c' \notin \sigma(\tau) \lor c'' \notin \sigma(\tau), \text{ for all } \tau < |\sigma| \quad (A.7)
\]
From (A.2) we obtain that there exist \( \sigma_1 \) and \( \sigma_2 \) such that
\[
\begin{align*}
\sigma_1 &\in \mathcal{M}(S_1) \\
\sigma_2 &\in \mathcal{M}(S_2)
\end{align*}
\] (A.8) (A.9)
and, for \( i \in \{1, 2\} \),
\[
[\sigma]_{dch(S_i)}(\tau) = \begin{cases} 
\sigma_i(\tau) & \text{for all } \tau < |\sigma_i| \\
\emptyset & \text{for all } \tau, |\sigma_i| \leq \tau < |\sigma|
\end{cases}
\] (A.10)
Further, \(|\sigma_{12}| = \max(|\sigma_1|, |\sigma_2|)\), and thus, by (A.6) and associativity of \( \max \),
\[
|\sigma| = \max(|\sigma_1|, |\sigma_2|, |\sigma_3|)
\] (A.11)
Let \( i \in \{1, 2\} \). From Lemma A.0.1 (point 1), \([\sigma]_{dch(S_i)}(\tau) = ([\sigma]_{dch(S_1) \cup dch(S_2)} \cup dch(S_i))(\tau)\).
Then, by (A.4), \([\sigma]_{dch(S_i)}(\tau) = \begin{cases} [\sigma]_{dch(S_i)}(\tau) & \text{for all } \tau < |\sigma_{12}| \\
\emptyset & \text{for all } \tau, |\sigma_{12}| \leq \tau < |\sigma|
\end{cases}\)
Thus, using (A.10) and (A.5), for \( i \in \{1, 2, 3\} \),
\[
[\sigma]_{dch(S_i)}(\tau) = \begin{cases} \sigma_i(\tau) & \text{for all } \tau < |\sigma_i| \\
\emptyset & \text{for all } \tau, |\sigma_i| \leq \tau < |\sigma|
\end{cases}
\] (A.12)
From (A.1), by Lemma 3.2.5, \( \sigma = [\sigma]_{dch(S_1) \cup dch(S_2) \cup dch(S_3)} \).
From (A.7) we obtain, for all \( \tau < |\sigma| \),
\[
c! \not\in [\sigma]_{dch(S_1) \cup dch(S_2) \cup dch(S_3)}(\tau) \lor c? \not\in [\sigma]_{dch(S_1) \cup dch(S_2) \cup dch(S_3)}(\tau)
\] (A.13)
Next, let \( \sigma_{23} \) be such that
\[
|\sigma_{23}| = \max(|\sigma_2|, |\sigma_3|)
\] (A.14)
\[
\sigma_{23}(\tau) = \begin{cases} [\sigma]_{dch(S_2) \cup dch(S_3)}(\tau) & \text{for all } \tau < |\sigma_{23}| \\
\emptyset & \text{for all } \tau, |\sigma_{23}| \leq \tau < |\sigma|
\end{cases}
\] (A.15)
Then, clearly,
\[
dch(\sigma_{23}) \subseteq dch(S_2) \cup dch(S_3)
\] (A.16)
By (A.13), \( c! \not\in [\sigma]_{dch(S_2) \cup dch(S_3)}(\tau) \lor c? \not\in [\sigma]_{dch(S_2) \cup dch(S_3)}(\tau) \), for all \( \tau < |\sigma| \).
Since \(|\sigma_{23}| \leq |\sigma|\), this leads by (A.15) to
\[
c! \not\in \sigma_{23}(\tau) \lor c? \not\in \sigma_{23}(\tau), \text{ for all } \tau < |\sigma_{23}|
\] (A.17)
Let \( i \in \{2, 3\} \). For \( \tau < |\sigma_{23}| \), by (A.15),
\[
[\sigma_{23}]_{dch(S_i)}(\tau) = ([\sigma]_{dch(S_2) \cup dch(S_3)} \cup dch(S_i))(\tau), \text{ and thus by Lemma A.0.1 (point 1),}
[\sigma_{23}]_{dch(S_i)}(\tau) = [\sigma]_{dch(S_i)}(\tau). \text{ Hence, using (A.12) and } |\sigma_{23}| \leq |\sigma|,
\]
Together with (A.16), (A.9), (A.3), (A.14), and (A.17) this leads to
\[ \sigma_{23} \in \mathcal{M}(S_2 \| S_3) \] (A.19)

Finally, we show that \( \sigma \in \mathcal{M}(S_1 \| (S_2 \| S_3)) \).

Therefore, observe that from (A.11), by associativity of max,
\[ |\sigma| = \text{max}(|\sigma_1|, \text{max}(|\sigma_2|, |\sigma_3|)) \] (A.19)

Then, from (A.12), \([\sigma]_{dch(S_1)}(\tau) = \begin{cases} \sigma_1(\tau) & \text{for all } \tau < |\sigma_1| \\ \emptyset & \text{for all } \tau, |\sigma_1| \leq \tau < |\sigma_2| \end{cases} \]

By (A.15), for all \( \tau < |\sigma_{23}|, [\sigma]_{dch(S_2) \cup dch(S_3)}(\tau) = \sigma_{23}(\tau) \). From (A.12), Lemma A.0.1 (point 3), and (A.14), for all \( \tau, |\sigma_{23}| \leq \tau < |\sigma|, [\sigma]_{dch(S_2) \cup dch(S_3)}(\tau) = \emptyset \). Together with (A.1), (A.8), (A.18), (A.19), and (A.7) this leads to \( \sigma \in \mathcal{M}(S_1 \| (S_2 \| S_3)) \).

**Proof of Lemma 3.2.13**

We show that for any program \( S, \mathcal{M}(S) \neq \emptyset \) and that for any \( \sigma \in \mathcal{M}(S) \):

1. \( dch(\sigma) \subseteq dch(S) \),
2. \( \sigma \) is well-formed.

The proof of \( \mathcal{M}(S) \neq \emptyset \) follows directly, by induction on the structure of \( S \), from the definition of the semantics. The other two points of this lemma are proved by induction on the structure of \( S \). Consider any \( \sigma \in \mathcal{M}(S) \). We give the most interesting cases.

* If \( S \equiv \text{delay } d \) then, by the definition of \( \mathcal{M}(\text{delay } d) \), \( dch(\sigma) = \emptyset = dch(\text{delay } d) \). Since, for all \( \tau < |\sigma|, \sigma(\tau) = \emptyset, \sigma \) is well-formed.

* Consider a guarded command \( G \equiv \left[ \bigcup_{i=1}^{n} c_i ? \rightarrow S_i \big| \text{delay } d \rightarrow S \right] \). Then either \( \sigma = \sigma_1 \sigma_2 \sigma_3 \) with \( \sigma_1 \in \text{LimitedWait}(G), \sigma_2 \in \text{Comm}(c_k), \sigma_3 \in \mathcal{M}(S_k) \), for some \( k \in \{1, \ldots, n\}, \) or \( \sigma = \sigma_4 \sigma_5 \) with \( \sigma_4 \in \text{TimeOut}(G), \sigma_5 \in \mathcal{M}(S) \). Further, note that \( dch(G) = \bigcup_{i=1}^{n} dch(c_i ?) \cup \bigcup_{i=1}^{n} dch(S_i) \).

1. Using the induction hypothesis, \( dch(\sigma) \subseteq (dch(\sigma_1) \cup dch(\sigma_2) \cup dch(\sigma_3) \cup dch(\sigma_4) \cup dch(\sigma_5)) \subseteq (\{c_i? | 1 \leq i \leq n\} \cup \{c_i | 1 \leq i \leq n\} \cup \bigcup_{i=1}^{n} dch(S_i) \cup dch(S)) \subseteq (\bigcup_{i=1}^{n} dch(c_i ?) \cup \bigcup_{i=1}^{n} dch(S_i) \cup dch(S)) = dch(G) \).
2. By the induction hypothesis, \( \sigma_3 \) and \( \sigma_5 \) are well-formed. From \( dch(\sigma_1) \subseteq \{c_i? | 1 \leq i \leq n\}, dch(\sigma_2) \subseteq \{c_i | 1 \leq i \leq n\}, \) and \( dch(\sigma_4) \subseteq \{c_i? | 1 \leq i \leq n\}, \) we obtain that \( \sigma_1, \sigma_2, \) and \( \sigma_5 \) are well-formed. Hence \( \sigma = \sigma_1 \sigma_2 \sigma_3 \) and \( \sigma = \sigma_4 \sigma_5 \) are well-formed.
Consider $S \equiv \star G$. By Corollary 3.2.10, $\sigma = \sigma_1 \sigma_2 \cdots$ with $\sigma_i \in \mathcal{M}(G)$, for $i \geq 1$. Then by the induction hypothesis, $dch(\sigma_i) \subseteq dch(G)$ and $\sigma_i$ well-formed, for $i \geq 1$. Hence $dch(\sigma) = dch(\sigma_1) \cup dch(\sigma_2) \cup \cdots \subseteq dch(G) = dch(\star G)$, and $\sigma$ is well-formed.

For $S \equiv S_1 \| S_2$, note that $dch(\sigma) \subseteq dch(S_1 \| S_2)$ is explicitly mentioned in the definition of the semantics. We prove that $\sigma$ is well-formed. The minimal waiting requirement follows directly from the definition. It remains to prove exclusion; for all $\tau < |\sigma|$, $\neg(c \in \sigma(\tau) \land c! \in \sigma(\tau))$ and $\neg(c \in \sigma(\tau) \land c? \in \sigma(\tau))$.

Consider $\tau < |\sigma|$. Assume $c \in \sigma(\tau)$. We prove that $c! \notin \sigma(\tau) \land c? \notin \sigma(\tau)$. Note that $c \in dch(\sigma) \subseteq dch(S_1) \cup dch(S_2)$.

Suppose $c \in dch(S_1)$. ($c \in dch(S_2)$ is similarly proved.) Then $c \in [\sigma]_{dch(S_1)}(\tau)$.

From the definition of the semantics, for $i \in \{1,2\}$, there exist $\sigma_i \in \mathcal{M}(S_i)$ such that $[\sigma]_{dch(S_i)}(\tau) = \begin{cases} \sigma_i(\tau) & \text{for all } \tau < |\sigma_i| \\ \emptyset & \text{for all } \tau, |\sigma_i| \leq \tau < |\sigma| \end{cases}$

Thus $c \in \sigma_i(\tau)$. Since, by the induction hypothesis, $\sigma_i$ is well-formed, we obtain

$$c! \notin \sigma_1(\tau) \lor c? \notin \sigma_1(\tau) \quad (A.20)$$

Next we prove

$$c! \notin \sigma_2(\tau) \lor c? \notin \sigma_2(\tau) \quad (A.21)$$

- If $c \notin dch(S_2)$ then first observe that $c \in dch(S_2)$ iff $c! \in dch(S_2)$ or $c? \in dch(S_2)$, and thus $dch(S_2) \cap \{c, c!, c?\} = \emptyset$. Since by the induction hypothesis, $dch(\sigma_2) \subseteq dch(S_2)$, we obtain $dch(\sigma_2) \cap \{c, c!, c?\} = \emptyset$. Hence (A.21) holds.

- If $c \in dch(S_2)$ then, using $c \in \sigma(\tau)$, $c \in [\sigma]_{dch(S_2)}(\tau)$. Hence $c \in \sigma_2(\tau)$, and thus (A.21) holds, since, by the induction hypothesis, $\sigma_2$ is well-formed.

Finally, observe that if $c! \in \sigma(\tau)$ and $c! \in dch(S_1)$ then $c! \in [\sigma]_{dch(S_1)}(\tau)$, and thus $c! \in \sigma_1(\tau)$. Similarly, if $c! \in \sigma(\tau)$ and $c! \in dch(S_2)$ then $c! \in \sigma_2(\tau)$. Since $c! \in \sigma(\tau)$ implies $c! \in dch(\sigma) \subseteq dch(S_1) \cup dch(S_2)$, $c! \in \sigma(\tau)$ leads to $c! \in (\sigma_1(\tau) \cup \sigma_2(\tau))$. Similarly, $c? \in \sigma(\tau)$ implies $c? \in (\sigma_1(\tau) \cup \sigma_2(\tau))$. Hence (A.20) and (A.21) lead to

$$c! \notin \sigma(\tau) \land c? \notin \sigma(\tau).$$

**Proof of Lemma 3.3.4**

We prove that $\{ \varphi \models S_1 \text{ sat } \varphi \} = \{ \varphi \models S_2 \text{ sat } \varphi \}$ if $\mathcal{M}(S_1) = \mathcal{M}(S_2)$.

Note that if $\mathcal{M}(S_1) = \mathcal{M}(S_2)$ then clearly $\{ \varphi \models S_1 \text{ sat } \varphi \} = \{ \varphi \models S_2 \text{ sat } \varphi \}$. It remains to prove that $\{ \varphi \models S_1 \text{ sat } \varphi \} = \{ \varphi \models S_2 \text{ sat } \varphi \}$ implies $\mathcal{M}(S_1) = \mathcal{M}(S_2)$.

By Lemma 3.3.22 we can derive $S_i \text{ sat } \psi_i$, where $\psi_i$ is precise for $S_i$, $i \in \{1,2\}$. First we give the main steps of the proof, justifying these steps later.
\( \{ \varphi \mid S_1 \text{ sat } \varphi \text{ is valid } \} = \{ \varphi \mid S_2 \text{ sat } \varphi \text{ is valid } \} \)

which implies

1. \( \{ \varphi \mid \models \psi_1 \land WF_{dcch(\psi_1)} \land \text{noact}(dcch(\varphi) - dcch(\psi_1)) \land \text{done} \rightarrow \varphi \} = \{ \varphi \mid \models \psi_2 \land WF_{dcch(\psi_2)} \land \text{noact}(dcch(\varphi) - dcch(\psi_2)) \land \text{done} \rightarrow \varphi \} \)

which implies

2. \( \models \psi_1 \land WF_{dcch(\psi_1)} \land \text{noact}(dcch(\varphi) - dcch(\psi_1)) \land \text{done} \leftrightarrow \psi_2 \land WF_{dcch(\psi_2)} \land \text{noact}(dcch(\varphi) - dcch(\psi_2)) \land \text{done} \), for all \( \varphi \)

which implies

3. \( \{ \sigma \mid \sigma \models \psi_1 \land WF_{dcch(\psi_1)} \land \text{noact}(dcch(\varphi) - dcch(\psi_1)) \land \text{done} \} = \{ \sigma \mid \sigma \models \psi_2 \land WF_{dcch(\psi_2)} \land \text{noact}(dcch(\varphi) - dcch(\psi_2)) \land \text{done} \} \), for all \( \varphi \)

which implies

4. \( \{ \sigma \mid [\sigma]_{dcch(\psi_1)} \models \psi_1 \land WF_{dcch(\psi_1)} \land \text{noact}(dcch(\varphi) - dcch(\psi_1)) \land \text{done} \} = \{ \sigma \mid [\sigma]_{dcch(\psi_2)} \models \psi_2 \land WF_{dcch(\psi_2)} \land \text{noact}(dcch(\varphi) - dcch(\psi_2)) \land \text{done} \} \)

for all \( \varphi \), which implies

5. \( \{ \sigma \mid [\sigma]_{dcch(\psi_1)} \models \psi_1 \land WF_{dcch(\psi_1)} \} = \{ \sigma \mid [\sigma]_{dcch(\psi_2)} \models \psi_2 \land WF_{dcch(\psi_2)} \} \)

which implies

6. \( \{ \sigma \mid [\sigma]_{dcch(\psi_1)} \models \psi_1 \land WF_{dcch(\psi_1)} \} = \{ \sigma \mid [\sigma]_{dcch(\psi_2)} \models \psi_2 \land WF_{dcch(\psi_2)} \} \)

which implies

7. \( \{ \sigma \mid [\sigma]_{dcch(\psi_1)} \text{ is well-formed, } [\sigma]_{dcch(\psi_1)} \models \psi_1 \} = \{ \sigma \mid [\sigma]_{dcch(\psi_2)} \text{ is well-formed, } [\sigma]_{dcch(\psi_2)} \models \psi_2 \} \)

which implies

8. \( \{ \sigma \mid [\sigma]_{dcch(\psi_1)} \text{ is well-formed, } \text{dcch}([\sigma]_{dcch(\psi_1)}) \subseteq \text{dcch}(S_1), [\sigma]_{dcch(\psi_1)} \models \psi_1 \} = \{ \sigma \mid [\sigma]_{dcch(\psi_2)} \text{ is well-formed, } \text{dcch}([\sigma]_{dcch(\psi_2)}) \subseteq \text{dcch}(S_2), [\sigma]_{dcch(\psi_2)} \models \psi_2 \} \)

which implies

9. \( \mathcal{M}(S_1) = \mathcal{M}(S_2) \).

Each step is justified as follows. Let \( i \in \{1, 2\} \).

1. If \( S_i \text{ sat } \varphi \text{ is valid } \), then by Lemma 3.3.24,
\( \psi_i \land WF_{dcch(\psi_i)} \land \text{noact}(dcch(\varphi) - dcch(\psi_i)) \land \text{done} \rightarrow \varphi \text{ is valid } \).

2. Since
\[ \models [\psi_1 \land WF_{dcch(\psi_1)} \land \text{noact}(dcch(\varphi) - dcch(\psi_1)) \land \text{done}] \rightarrow [\psi_1 \land WF_{dcch(\psi_1)} \land \text{noact}(dcch(\varphi) - dcch(\psi_1)) \land \text{done}] \],
we obtain
\[\models [\psi_2 \land WF_{dch(\psi_2)} \land (\text{noact}(dch(\varphi) - dch(\psi_2)) \cup \text{done})] \rightarrow [\psi_1 \land WF_{dch(\psi_1)} \land (\text{noact}(dch(\varphi) - dch(\psi_1)) \cup \text{done})].\]
Similarly,
\[\models [\psi_1 \land WF_{dch(\psi_1)} \land (\text{noact}(dch(\varphi) - dch(\psi_1)) \cup \text{done})] \rightarrow [\psi_2 \land WF_{dch(\psi_2)} \land (\text{noact}(dch(\varphi) - dch(\psi_2)) \cup \text{done})].\]
3. By the definition of validity of assertions.
4. Since \(dch(\psi_1 \land WF_{dch(\psi_1)} \land \text{noact}(dch(\varphi) - dch(\psi_1)) \cup \text{done}) \subseteq dch(\psi_1) \cup dch(\varphi)\) we can use Lemma 3.3.19.
5. Consider \(\varphi\) such that \(dch(\varphi) \subseteq dch(\psi_1) \cup dch(\psi_2)\).
6. By Lemma 3.3.19.
7. Using the correspondence between \(WF\) and well-formedness as expressed by Lemma 3.3.23.
8. By preciseness of \(\psi_i\) for \(S_i\), \(dch(\psi_i) \subseteq dch(S_i)\).
9. From preciseness of \(\psi_i\) for \(S_i\).

**Proof of Lemma 3.3.19**

Consider any \(cset \subseteq DCHAN\) and MTL assertion \(\varphi\). We prove that if \(dch(\varphi) \subseteq cset\) then, for all \(\sigma, \sigma \models \varphi\) iff \([\sigma]_{cset} \models \varphi\).

The proof is given by induction on the structure of \(\varphi\).

- If \(\varphi \equiv \text{comm}(c)\) then \(\{c\} = dch(\varphi) \subseteq cset\), and thus \(c \in cset\). Then, \(\sigma \models \text{comm}(c)\) iff \(|\sigma| > 0\) and \(c \in \sigma(0)\) iff \(0 < |\sigma| = |\sigma|_{cset}\) and \(c \in \sigma(0) \cap cset\) iff \(|\sigma|_{cset} > 0\) and \(c \in [\sigma]_{cset}(0)\) iff \([\sigma]_{cset} \models \text{comm}(c)\).

- If \(\varphi \equiv \text{wait}(c!)\) then \(\{c!\} = dch(\varphi) \subseteq cset\), and thus \(c! \in cset\). Then, \(\sigma \models \text{wait}(c!)\) iff \(|\sigma| > 0\) and \(c! \in \sigma(0)\) iff \(0 < |\sigma| = |\sigma|_{cset}\) and \(c! \in \sigma(0)\) iff \(\tau < |\sigma|_{cset}\) and \(c! \in [\sigma]_{cset}(0)\) iff \([\sigma]_{cset} \models \text{wait}(c!)\).

Similarly, for \(\varphi \equiv \text{wait}(c?).\)

- \(\sigma \models \text{done}\) iff \(|\sigma| = 0\) iff \(|\sigma|_{cset} = 0\) iff \([\sigma]_{cset} \models \text{done}\).

- Consider \(\varphi \equiv \varphi_1 \lor \varphi_2\). Then, for \(i \in \{1, 2\}, \ dch(\varphi_i) \subseteq dch(\varphi_1) \cup dch(\varphi_2) = dch(\varphi) \subseteq cset\). Hence, \(\sigma \models \varphi_1 \lor \varphi_2\) iff \(\sigma \models \varphi_1\) or \(\sigma \models \varphi_2\) iff, using the induction hypothesis, \([\sigma]_{cset} \models \varphi_1\) or \([\sigma]_{cset} \models \varphi_2\) iff \([\sigma]_{cset} \models \varphi_1 \lor \varphi_2\).

Similarly, for \(\varphi_1 \cup_{\leq} \varphi_2\) and \(\varphi_1 \cup_{=} \varphi_2\).

- If \(\varphi \equiv \neg \varphi_0\) then \(dch(\varphi_0) = dch(\neg \varphi_0) = dch(\varphi) \subseteq cset\). Thus, \(\sigma \models \neg \varphi_0\) iff \(\sigma \not\models \varphi_0\) iff, by the induction hypothesis, \([\sigma]_{cset} \not\models \varphi_0\) iff \([\sigma]_{cset} \models \neg \varphi_0\).
• Consider $\varphi \equiv \varphi_1 \land \varphi_2$. Then, for $i \in \{1, 2\}$, $dch(\varphi_i) \subseteq dch(\varphi_1) \cup dch(\varphi_2) = dch(\varphi) \subseteq \textit{cset}$. Hence, $\sigma \models \varphi_1 \land \varphi_2$ iff there exist models $\sigma_1$ and $\sigma_2$ such that $\sigma = \sigma_1 \sigma_2$, $\sigma_1 \models \varphi_1$, and $\sigma_2 \models \varphi_2$.

iff, using the induction hypothesis,

there exist models $\sigma_1$ and $\sigma_2$ such that $\sigma = \sigma_1 \sigma_2$, $[\sigma_1]_{\textit{cset}} \models \varphi_1$, and $[\sigma_2]_{\textit{cset}} \models \varphi_2$.

iff ($\ast$) (this step is justified below)

there exist models $\sigma_3$ and $\sigma_4$ such that $[\sigma]_{\textit{cset}} = \sigma_3 \sigma_4$, $\sigma_3 \models \varphi_1$, and $\sigma_4 \models \varphi_2$.

iff $[\sigma]_{\textit{cset}} \models \varphi_1 \land \varphi_2$.

Step ($\ast$) is proved as follows:

\begin{itemize}
  \item Only if Consider $\sigma_3 = [\sigma_1]_{\textit{cset}}$ and $\sigma_4 = [\sigma_2]_{\textit{cset}}$.
  \item Then $[\sigma]_{\textit{cset}} = [\sigma_1 \sigma_2]_{\textit{cset}} = [\sigma_1]_{\textit{cset}}[\sigma_2]_{\textit{cset}} = \sigma_3 \sigma_4$.
\end{itemize}

if Given $\sigma_3$ and $\sigma_4$, define $\sigma_1$ and $\sigma_2$ as follows:

$\sigma_1 = \sigma \upharpoonright |\sigma_3|$, $\sigma_2 = \sigma \upharpoonright |\sigma_3|$. Then $\sigma = \sigma_1 \sigma_2$. Since $|\sigma_3| = |[\sigma]_{\textit{cset}}| = |\sigma|$, we have $|\sigma_1| = |\sigma_3|$, and hence, by $|\sigma_1| + |\sigma_2| = |\sigma| = |[\sigma]_{\textit{cset}}| = |\sigma_3| + |\sigma_4|$, also $|\sigma_2| = |\sigma_4|$. Using $[\sigma_1 \sigma_2]_{\textit{cset}} = [\sigma]_{\textit{cset}} = \sigma_3 \sigma_4$ this leads to $[\sigma_1]_{\textit{cset}} = \sigma_3$ and $[\sigma_2]_{\textit{cset}} = \sigma_4$.

Similarly, for $\varphi \equiv C^\infty \varphi_1$.

### Proof of Lemma 3.3.23

We prove that if $dch(\sigma) \subseteq \textit{cset}$ and $\sigma \models \textit{WF}_{\textit{cset}}$ then $\sigma$ is well-formed.

Assume $\sigma \models \textit{WF}_{\textit{cset}}$. Then $\sigma \models \Box (\textit{MaxPar}_{\textit{cset}} \land \textit{Exclusion}_{\textit{cset}})$, and thus $\sigma \models (\textit{MaxPar}_{\textit{cset}} \land \textit{Exclusion}_{\textit{cset}}) \cup \text{done}$. Hence, for all $\tau < |\sigma|,$

1. $\sigma \upharpoonright \tau \models \neg(\text{wait}(c!) \land \text{wait}(c?))$, for all $\{c?, c!\} \subseteq \textit{cset}$;

2. $\sigma \upharpoonright \tau \models \neg(\text{comm}(c) \land \text{wait}(c!))$, for all $\{c, c!\} \subseteq \textit{cset}$, and
   $\sigma \upharpoonright \tau \models \neg(\text{comm}(c) \land \text{wait}(c?))$, for all $\{c, c?\} \subseteq \textit{cset}$.

Given our interpretation of assertions (Section 3.3.1), this implies, for all $\tau < |\sigma|,$

1. $\neg(c! \in \sigma(\tau) \land c? \in \sigma(\tau))$, for all $\{c?, c!\} \subseteq \textit{cset}$;

2. $\neg(c \in \sigma(\tau) \land c \in \sigma(\tau))$, for all $\{c, c!\} \subseteq \textit{cset}$, and
   $\neg(c \in \sigma(\tau) \land c? \in \sigma(\tau))$, for all $\{c, c?\} \subseteq \textit{cset}$.

Note that for all $\tau < |\sigma|$: if $c! \not\in \textit{cset}$ then, by $dch(\sigma) \subseteq \textit{cset}$, $c! \not\in dch(\sigma)$, and thus $c! \not\in \sigma(\tau)$. Similarly, if $c? \not\in \textit{cset}$ then $c? \not\in \sigma(\tau)$, and if $c \not\in \textit{cset}$ then $c \not\in \sigma(\tau)$. Thus, for all $\tau < |\sigma|$ and for all $c,$
1. \( \neg (c! \in \sigma(\tau) \land c? \in \sigma(\tau)) \);

2. \( \neg (c \in \sigma(\tau) \land c! \in \sigma(\tau)) \), and \( \neg (c \in \sigma(\tau) \land c? \in \sigma(\tau)) \).

Hence \( \sigma \) is well-formed.

**Proof of Lemma 3.4.1**

Consider any \( \gamma, \sigma_1, \) and \( \sigma_2 \). We prove \( \forall (\text{exp})(\gamma, \sigma_1) = \forall (\text{exp}[[\sigma_1]/\text{time}])(\gamma, \sigma_2) \).

The proof proceeds by induction on the structure of expression \( \text{exp} \):

- \( \forall (\tau)(\gamma, \sigma_1) = \tau = \forall (\tau)(\gamma, \sigma_2) = \forall (\tau[[\sigma_1]/\text{time}])(\gamma, \sigma_2) \).
- \( \forall (t)(\gamma, \sigma_1) = \gamma(t) = \forall (t)(\gamma, \sigma_2) = \forall (t[[\sigma_1]/\text{time}])(\gamma, \sigma_2) \).
- \( \forall (\text{time})(\gamma, \sigma_1) = |\sigma_1| = \forall (|\sigma_1|)(\gamma, \sigma_2) = \forall (\text{time}[[\sigma_1]/\text{time}])(\gamma, \sigma_2) \).
- Using the induction hypothesis, we obtain
  \[ \forall (\text{exp}_1 + \text{exp}_2)(\gamma, \sigma_1) = \forall (\text{exp}_1)(\gamma, \sigma_1) + \forall (\text{exp}_2)(\gamma, \sigma_1) = \forall (\text{exp}_1[[\sigma_1]/\text{time}])(\gamma, \sigma_2) + \forall (\text{exp}_2[[\sigma_1]/\text{time}])(\gamma, \sigma_2) = \forall ((\text{exp}_1 + \text{exp}_2)[[\sigma_1]/\text{time}])(\gamma, \sigma_2) \]

Similarly, for \( \text{exp}_1 \times \text{exp}_2 \).

**Proof of Lemma 3.4.3**

Consider any \( \gamma \) and \( \sigma_1 \). We prove that \( \llbracket p \rrbracket \gamma \sigma_1 \) iff for all \( \sigma_2 \), \( \llbracket p[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \sigma_2 \).

Observe that if, for all \( \sigma_2 \), \( \llbracket p[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \sigma_2 \), then, using \( \sigma_2 \) with \( |\sigma_2| = 0 \),

\( \llbracket p[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \), and hence \( \llbracket p \rrbracket \gamma \sigma_1 \).

It remains to prove that, for all \( \sigma_2 \), \( \llbracket p \rrbracket \gamma \sigma_1 = \text{true} \) implies \( \llbracket p[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \sigma_2 = \text{true} \).

Let \( \sigma_2 \) be any arbitrary model. Then we prove the following stronger property:

\( \llbracket p \rrbracket \gamma \sigma_1 = \text{true} \) implies \( \llbracket p[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \sigma_2 = \text{true} \), and

\( \llbracket p \rrbracket \gamma \sigma_1 = \text{false} \) implies \( \llbracket p[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \sigma_2 = \text{false} \).

The proof proceeds by induction on the structure of \( p \):

- \( \llbracket \text{comm via } c \text{ at } \text{exp} \rrbracket \gamma \sigma_1 = \text{true} \) implies, by Lemma 3.4.1,

  \( \forall (\text{exp})(\gamma, \sigma_1) < |\sigma_1| \) and \( c \in \sigma_1(\forall (\text{exp})(\gamma, \sigma_1)) \) which implies

  \( \forall (\text{exp}[[\sigma_1]/\text{time}])(\gamma, \sigma_1 \sigma_2) < |\sigma_1| < |\sigma_1 \sigma_2| \) and \( c \in \sigma_1 \sigma_2(\forall (\text{exp}[[\sigma_1]/\text{time}])(\gamma, \sigma_1 \sigma_2)) \) which implies

  \( \llbracket (\text{comm via } c \text{ at } \text{exp})[[\sigma_1]/\text{time}] \rrbracket \gamma \sigma_1 \sigma_2 = \text{true} \).

Similarly, for \( \text{wait to } c! \text{ at } \text{exp} \) and \( \text{wait to } c? \text{ at } \text{exp} \).

- \( \llbracket \text{comm via } c \text{ at } \text{exp} \rrbracket \gamma \sigma_1 = \text{false} \) implies, by Lemma 3.4.1,

  \( \forall (\text{exp})(\gamma, \sigma_1) < |\sigma_1| \) and \( c \notin \sigma_1(\forall (\text{exp})(\gamma, \sigma_1)) \) which implies
\[ \mathcal{V}(\exp[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2) < |\sigma_1| < |\sigma_1 \sigma_2| \] and \( c \notin \sigma_1 \sigma_2(\mathcal{V}(\exp[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2)) \) which implies

\[ \text{[(comm via } c \text{ at } \exp)[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = \text{false}. \]

Similarly, for \( \text{wait to } c! \text{ at } \exp \) and \( \text{wait to } c? \text{ at } \exp. \)

- \[ \exp_1 = \exp_2 \times \gamma_1 \sigma_1 = \text{true implies} \]

\[ \mathcal{V}(\exp_1)(\gamma, \sigma_1) = \mathcal{V}(\exp_2)(\gamma, \sigma_1) \] which implies, by Lemma 3.4.1,
\[ \mathcal{V}(\exp_1[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2) = \mathcal{V}(\exp_2[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2) \] which implies
\[ \text{[(exp_1 = exp_2)[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true.} \]

Similarly, for \( \exp_1 < \exp_2. \)

- \[ \exp_1 = \exp_2 \times \gamma_1 \sigma_1 = \text{false implies} \]

\[ \mathcal{V}(\exp_1)(\gamma, \sigma_1) \neq \mathcal{V}(\exp_2)(\gamma, \sigma_1) \] which implies, by Lemma 3.4.1,
\[ \mathcal{V}(\exp_1[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2) \neq \mathcal{V}(\exp_2[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2) \] which implies
\[ \text{[(exp_1 = exp_2)[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = false which implies} \]
\[ \text{[(exp_1 = exp_2)[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = false.} \]

Similarly, for \( \exp_1 < \exp_2. \)

- \[ \exp \in \mathbb{N} \gamma_1 \sigma_1 = \text{true implies} \]

\[ \mathcal{V}(\exp)(\gamma, \sigma) \in \mathbb{N} \] which implies, using Lemma 3.4.1,
\[ \mathcal{V}(\exp[|\sigma_1|/\text{time}])(\gamma, \sigma_1 \sigma_2) \in \mathbb{N} \] which implies
\[ \text{[(exp \in \mathbb{N})[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true.} \]

Similarly, for \( \text{[exp \in \mathbb{N}]\gamma_1 = false.} \)

- \[ \neg p \gamma_1 \sigma_1 = \text{true implies} \]

\[ \text{NOT}_3[\neg p] \gamma_1 \sigma_1 = \text{true which implies} \]
\[ \neg p \gamma_1 = \text{false which implies, by the induction hypothesis,} \]
\[ \text{[p[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = false which implies} \]
\[ NOT_3 [p[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true which implies} \]
\[ \text{[\neg p[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true.} \]

Similarly, for \( \text{[\neg p]\gamma_1 = false.} \)

- \[ p_1 \lor p_2 \gamma_1 \sigma_1 = \text{true implies} \]

\[ \text{[p_1] \gamma_1 \sigma_1 OR_3 [p_2] \gamma_1 = true which implies, by the definition of } OR_3, \]
\[ \text{[p_1] \gamma_1 \sigma_1 = true or [p_2] \gamma_1 = true which implies, by the induction hypothesis,} \]
\[ \text{[p_1[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true or [p_2[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true which implies} \]
\[ \text{[(p_1 \lor p_2)[|\sigma_1|/\text{time}]\gamma_1 \sigma_2 = true.} \]
Proof of Lemma 3.4.6

Consider \( cset \subseteq DCHAN \) and assertion \( p \) such that \( dch(p) \subseteq cset \). Consider any environment \( \gamma \) and model \( \sigma \). To show that \( [p] \gamma \sigma \iff [p][\sigma]_{cset} \), we prove the following, stronger, property:

- \( [p] \gamma \sigma = true \iff [p][\sigma]_{cset} = true \), and
- \( [p] \gamma \sigma = false \iff [p][\sigma]_{cset} = false \).

Note that \( V(exp)(\gamma, \sigma) = V(exp)(\gamma, [\sigma]_{cset}) \), since \( |\sigma| = |[\sigma]_{cset}| \).

The proof proceeds by induction on the structure of \( p \).

- If \( p \equiv \text{comm via } c \text{ at } exp \), then \( c \in dch(p) \subseteq cset \).

  - \( [\text{comm via } c \text{ at } exp] \gamma \sigma = true \iff \)
    
    \( V(exp)(\gamma, \sigma) < |\sigma| \text{ and } c \in \sigma(V(exp)(\gamma, \sigma)) \) iff
    
    \( V(exp)(\gamma, \sigma) < |\sigma| = |[\sigma]_{cset}| \) and \( c \in \sigma(V(exp)(\gamma, \sigma)) \cap cset \) iff
    
    \( V(exp)(\gamma, \sigma) < |[\sigma]_{cset}| \) and \( c \in [\sigma]_{cset}(V(exp)(\gamma, \sigma)) \) iff
    
    \( [\text{comm via } c \text{ at } exp] \gamma[\sigma]_{cset} = true \).

  - \( [\text{comm via } c \text{ at } exp] \gamma \sigma = false \iff \)
    
    \( V(exp)(\gamma, \sigma) < |\sigma| \text{ and } c \notin \sigma(V(exp)(\gamma, \sigma)) \) iff
    
    \( V(exp)(\gamma, \sigma) < |\sigma| = |[\sigma]_{cset}| \) and \( c \notin \sigma(V(exp)(\gamma, \sigma)) \cap cset \) iff
    
    \( V(exp)(\gamma, \sigma) < |[\sigma]_{cset}| \) and \( c \notin [\sigma]_{cset}(V(exp)(\gamma, \sigma)) \) iff
    
    \( [\text{comm via } c \text{ at } exp] \gamma[\sigma]_{cset} = false \).

Similarly, for \( p \equiv \text{wait to } c! \text{ at } exp \) and \( p \equiv \text{wait to } c? \text{ at } exp \).
\[ [\text{exp}_1 = \text{exp}_2] \land \sigma = \text{true} \iff \forall (\text{exp}_1)(\gamma, \sigma) = \forall (\text{exp}_2)(\gamma, \sigma) \iff \forall (\text{exp}_1)(\gamma, [\sigma]_{\text{reset}}) = \forall (\text{exp}_2)(\gamma, [\sigma]_{\text{reset}}) \iff \forall (\text{exp}_1 = \text{exp}_2)(\gamma, [\sigma]_{\text{reset}}) \iff [\text{exp}_1 = \text{exp}_2] \land [\sigma]_{\text{reset}} = \text{true}. \]

Similarly, for \([\text{exp}_1 = \text{exp}_2] \land \sigma = \text{false}, p \equiv (\text{exp}_1 < \text{exp}_2), \) and \(p \equiv (\text{exp} \in \mathbb{N}).\)

\[ [\neg p] \land \sigma = \text{true} \iff \neg \exists T \land p \sigma = \text{true} \iff [p] \land \sigma = \text{false} \iff [p] [\sigma]_{\text{reset}} = \text{false} \iff [\neg p] [\sigma]_{\text{reset}} = \text{true}. \]

Similarly, for \([\neg p] \land \sigma = \text{false}.\)

\[ [p_1 \lor p_2] \land \sigma = \text{true} \iff [p_1] \land \sigma \lor [p_2] \land \sigma = \text{true} \iff [p_1] \land \sigma = \text{true} \lor [p_2] \land \sigma = \text{true} \iff [p_1] [\sigma]_{\text{reset}} = \text{true} \lor [p_2] [\sigma]_{\text{reset}} = \text{true} \iff ([p_1] [\sigma]_{\text{reset}} \lor [p_2] [\sigma]_{\text{reset}}) = \text{true} \iff [p_1 \lor p_2] [\sigma]_{\text{reset}} = \text{true}. \]

Similarly, for \([p_1 \lor p_2] \land \sigma = \text{false}.\)

\[ \exists t : p \land \sigma = \text{true} \iff \text{there exists a } \tau \in \text{TIME} \cup \{\infty\} \text{ such that } [p](\gamma : t \mapsto \tau) \sigma = \text{true} \iff \text{there exists a } \tau \in \text{TIME} \cup \{\infty\} \text{ such that } [p](\gamma : t \mapsto \tau) [\sigma]_{\text{reset}} = \text{true} \iff \exists t : p \land [\sigma]_{\text{reset}} = \text{true}. \]

Similarly, for \(\exists t : p \land \sigma = \text{false}.\)
Appendix B

Soundness and Completeness of the Proof System in Section 3.3

B.1 Soundness of the Proof System in Section 3.3

We prove the soundness of the proof system from Section 3.3 (Theorem 3.3.20) by showing that the axioms are valid and that the inference rules preserve validity. In the proofs below, we often rely on the following lemma which is easily verified.

Lemma B.1.1 \( \sigma \models \varphi_1 \cup \varphi_2 \) iff either

1. \( \sigma \models \square \varphi_1 \), or
2. there exists a \( \tau \in \text{TIME} \), such that \( \sigma \models \square_{< \tau} \varphi_1 \) and \( \sigma \models \diamond_{= \tau} \varphi_2 \).

To prove that \( S \text{ sat } \varphi \) is valid for some \( S \) and \( \varphi \), we have to verify that \( \sigma \models \varphi \) for all \( \sigma \in \mathcal{M}(S) \).

Well-Formedness

Consider a program \( S \) and a finite set \( cset \subseteq DCHAN \). We prove that \( S \text{ sat } WF_{cset} \) is valid. Consider any \( \sigma \in \mathcal{M}(S) \). Then, by Lemma 3.2.13, \( \sigma \) is well-formed, that is, for all \( \tau < |\sigma| \),

1. \( \neg(c! \in \sigma(\tau) \land c? \in \sigma(\tau)) \), for all \( c \);
2. \( \neg(c \in \sigma(\tau) \land c! \in \sigma(\tau)) \) and \( \neg(c \in \sigma(\tau) \land c? \in \sigma(\tau)) \), for all \( c \).

Hence, for all \( \tau < |\sigma| \),

1. \( \neg(c! \in \sigma(\tau) \land c? \in \sigma(\tau)) \), for all \( c \) with \( \{c!, c?\} \subseteq cset \);
2. \( \neg(c \in \sigma(\tau) \land c! \in \sigma(\tau)) \), for all \( c \) with \( \{c, c!\} \subseteq cset \), and
   \( \neg(c \in \sigma(\tau) \land c? \in \sigma(\tau)) \), for all \( c \) with \( \{c, c?\} \subseteq cset \).

Given our interpretation of assertions (Section 3.3.1) this implies, for all \( \tau < |\sigma| \),
1. \( \sigma \uparrow \tau \models \land_{(c, e) \subseteq \text{cset}} \neg (\text{wait}(e!) \land \text{wait}(e?)) \);

2. \( \sigma \uparrow \tau \models \land_{(c, e) \subseteq \text{cset}} \neg (\text{comm}(c) \land \neg \text{wait}(e!)) \land \land_{(c, e) \subseteq \text{cset}} \neg (\text{comm}(c) \land \neg \text{wait}(e?)) \).

Furthermore, for all \( \tau \geq |\sigma| \), we have \( \sigma \uparrow \tau \models \neg \text{wait}(e!) \land \neg \text{wait}(e?) \land \neg \text{comm}(c) \), for any channel \( c \). Thus, for all \( \tau \in \text{TIME} \),

1. \( \sigma \uparrow \tau \models \land_{(c, e) \subseteq \text{cset}} \neg (\text{wait}(e!) \land \text{wait}(e?)) \);

2. \( \sigma \uparrow \tau \models \land_{(c, e) \subseteq \text{cset}} \neg (\text{comm}(c) \land \neg \text{wait}(e!)) \land \land_{(c, e) \subseteq \text{cset}} \neg (\text{comm}(c) \land \neg \text{wait}(e?)) \).

Hence, by definition, \( \sigma \models \Box (\text{MinWait}_{\text{cset}} \land \text{Exclusion}_{\text{cset}}) \), and thus \( \sigma \models WF_{\text{cset}} \).

**Communication Invariance**

Consider a program \( S \) and a set \( \text{cset} \subseteq DCHAN \) such that \( \text{cset} \cap \text{dch}(S) = \emptyset \). We prove that \( S \models \Box \text{noact}(\text{cset}) \) is valid. Consider any \( \sigma \in \mathcal{M}(S) \). From Lemma 3.2.13, we obtain \( \text{dch}(\sigma) \subseteq \text{dch}(S) \), and hence \( \text{cset} \cap \text{dch}(\sigma) = \emptyset \). Thus, by definition of \( \text{dch}(\sigma) \), for all \( \tau < |\sigma| \), \( \sigma(\tau) \cap \text{cset} = \emptyset \). Hence, for all \( \tau < |\sigma| \),

1. if \( c \in \text{cset} \) then \( c \notin \sigma(\tau) \),

2. if \( e! \in \text{cset} \) then \( e! \notin \sigma(\tau) \), and

3. if \( e? \in \text{cset} \) then \( e? \notin \sigma(\tau) \).

Thus, for all \( \tau < |\sigma| \),

1. \( \sigma \uparrow \tau \models \neg \text{comm}(c) \), for \( c \in \text{cset} \),

2. \( \sigma \uparrow \tau \models \neg \text{wait}(e!) \), for \( e! \in \text{cset} \), and

3. \( \sigma \uparrow \tau \models \neg \text{wait}(e?) \), for \( e? \in \text{cset} \).

Since, for all \( \tau \geq |\sigma| \), we have \( \sigma \uparrow \tau \models \neg \text{wait}(e!) \land \neg \text{wait}(e?) \land \neg \text{comm}(c) \), for any channel \( c \), this leads to \( \sigma \models \Box \text{noact}(\text{cset}) \), and thus \( S \models \Box \text{noact}(\text{cset}) \).

**Conjunction**

It is easy to see that if \( S \models \varphi_1 \) and \( S \models \varphi_2 \) are valid, then \( S \models \varphi_1 \land \varphi_2 \) is valid.

**Consequence**

Assume validity of \( S \models \varphi_1 \) and \( \varphi_1 \rightarrow \varphi_2 \). Consider \( \sigma \in \mathcal{M}(S) \). Then \( \sigma \models \varphi_1 \). Hence, by the implication, \( \sigma \models \varphi_2 \). Thus \( S \models \varphi_2 \) is valid.

**Skip and Delay**

It follows directly from the definitions that \( \text{skip} \models \text{done} \) and \( \text{delay } d \models \square_{=d} \text{done} \) are valid, thus the Axioms 3.3.9 and 3.3.10 are sound.
**Send and Receive**

To prove the soundness of Axiom 3.3.11, we must show validity of the formula $c! \text{sat} \ wait(c!) \ U \ (\text{comm}(c) \text{U}_{=K_c} \text{done})$. Consider any $\sigma \in M(c!)$. Then

1. $\sigma$ is a non-terminating model from $\text{WaitSend}(c)$, that is, for all $\tau \in \text{TIME}$, $\sigma(\tau).\text{wts} = \{c\}$, or
2. there exists a $\tau \in \text{TIME}$ such that for all $\tau_1 < \tau$, $\sigma(\tau_1).\text{wts} = \{c\}$, for all $\tau_1, \tau \leq \tau_1 < \tau + K_c$, $\sigma(\tau_1) = \{c\}$, and $|\sigma| = \tau + K_c$.

This implies that

1. for all $\tau \in \text{TIME}$, $\sigma \uparrow \tau \models \text{wait}(c!)$, or
2. there exists a $\tau \in \text{TIME}$ such that for all $\tau_1 < \tau$, $\sigma \uparrow \tau_1 \models \text{wait}(c!)$, and $\sigma \uparrow \tau \models \text{comm}(c) \text{U}_{=K_c} \text{done}$.

Thus

1. $\sigma \models \Box \text{wait}(c!)$, or
2. there exists a $\tau \in \text{TIME}$ such that $\sigma \models \Box_{<\tau} \text{wait}(c!)$, and $\sigma \models \Diamond_{\geq \tau} (\text{comm}(c) \text{U}_{=K_c} \text{done})$.

Hence, by Lemma B.1.1, $\sigma \models \text{wait}(c!) \ U \ (\text{comm}(c) \text{U}_{=K_c} \text{done})$.

The soundness of Axiom 3.3.12 is proved similarly.

**Sequential Composition**

Assume $S_1$ sat $\varphi_1$ and $S_2$ sat $\varphi_2$ are valid. We show $S_1; S_2$ sat $\varphi_1 \ C \ \varphi_2$. Consider any $\sigma \in M(S_1; S_2)$. Then there exist $\sigma_1 \in M(S_1)$ and $\sigma_2 \in M(S_2)$ such that $\sigma = \sigma_1 \sigma_2$. From $S_1$ sat $\varphi_1$, we obtain $\sigma_1 \models \varphi_1$. Similarly, $\sigma_2 \models \varphi_2$. Then the definition of the $C$ operator leads to $\sigma \models \varphi_1 \ C \ \varphi_2$.

**Guarded Command without Delay**

First consider $G \equiv \[\prod_{i=1}^{n} c_i \rightarrow S_i\]$. Assume $c_i; S_i$ sat $\varphi_i$ is valid, for $i \in \{1, \ldots, n\}$. Consider any $\sigma \in M([\prod_{i=1}^{n} c_i \rightarrow S_i])$. Then $\sigma \in SEQ(\text{Wait}(G), \text{Comm}(G))$. Thus either

1. $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$, thus for all $\tau \in \text{TIME}$, $\sigma(\tau) = \{c_1, \ldots, c_n\}$, or
2. $\sigma = \sigma_1 \sigma_2 \sigma_3$, with
   $\sigma_1 \in \text{Wait}(G)$, $\sigma_2 \in \text{Comm}(c_k)$, and $\sigma_3 \in M(S_k)$ for some $k \in \{1, \ldots, n\}$; then there exists a $\tau \in \text{TIME}$ such that $|\sigma_1| = \tau$ and, for all $\tau_1 < \tau$, $\sigma_1(\tau_1) = \{c_1, \ldots, c_n\}$, there exists a $k \in \{1, \ldots, n\}$ such that for all $\tau_1 < K_c$, $\sigma_2(\tau_1) = \{c_k\}$, $|\sigma_2| = K_c$, and $\sigma_3 \in M(S_k)$.
Hence either

1. for all \( \tau \in \text{TIME} \), \( \sigma(\tau) = \{ c_1?, \ldots, c_n? \} \), or

2. \( \sigma = \sigma_1 \sigma_4 \), there exists a \( \tau \in \text{TIME} \) such that \( |\sigma_1| = \tau \) and, for all \( \tau_1 < \tau \),
   \( \sigma_1(\tau_1) = \{ c_1?, \ldots, c_n? \} \), there exists a \( k \in \{ 1, \ldots, n \} \) such that
   \( \sigma_4 \in \mathcal{M}(c_k?; S_k) \) with \( \sigma_4(0) = \{ c_k \} \).

For \( \text{cset} \subseteq \text{DCHAN} \), \( \text{noact}(\text{cset}) \) has been defined as follows:
\[ \Lambda_{c \in \text{cset}} \neg \text{wait}(c!) \land \Lambda_{c \in \text{cset}} \neg \text{wait}(c?) \land \Lambda_{c \in \text{cset}} \neg \text{comm}(c). \]
This leads to

1. for all \( \tau \in \text{TIME} \), \( \sigma \uparrow \tau, \sigma \uparrow \tau \models \text{noact}(\text{dch}(G) - \{ c_1?, \ldots, c_n? \}) \), or

2. \( \sigma = \sigma_1 \sigma_4 \), there exists a \( \tau \in \text{TIME} \) such that for all \( \tau_1 < \tau \), \( \sigma_1(\tau_1) \models \Lambda_i \text{wait}(c_i?) \),
   \( \sigma_1(\tau_1) \models \text{noact}(\text{dch}(G) - \{ c_1?, \ldots, c_n? \}) \), \( |\sigma_1| = \tau \), and there exists a \( k \in \{ 1, \ldots, n \} \)
   such that \( \sigma_4 \models \varphi_k \) and \( \sigma_4 \models \text{comm}(c_k) \).

Recall that \( \text{wait}_G \) is defined as \( \Lambda_i \text{wait}(c_i?) \land \text{noact}(\text{dch}(G) - \{ c_1?, \ldots, c_n? \}) \). Then

1. \( \sigma \models \Box \text{wait}_G \), or

2. there exists a \( \tau \in \text{TIME} \) such that \( \sigma \models \Box_\tau \text{wait}_G \) and
   \[ \sigma \models \Box_\tau \bigvee_i (\varphi_i \land \text{comm}(c_i)) \).

By Lemma B.1.1 this implies \( \sigma \models \text{wait}_G \cup \bigvee_i (\varphi_i \land \text{comm}(c_i)) \), which proves the soundness of Rule 3.3.14.

The soundness of the rule for Guarded Command with Delay (Rule 3.3.15) is proved similarly.

**Iteration**

Assume \( G \models \varphi \) is valid; we prove \( *G \models C^\infty \varphi \). Consider any \( \sigma \in \mathcal{M}(*G) \). Then, either

- \( \sigma = \sigma_1 \cdots \sigma_k \), with \( \sigma_i \in \mathcal{M}(G) \), for \( i \in \{ 1, \ldots, k \} \), \( |\sigma_i| < \infty \), for \( i \in \{ 1, \ldots, k - 1 \} \), and
  \( |\sigma_k| = \infty \), or

- \( \sigma = \sigma_1 \sigma_2 \cdots \), with \( \sigma_i \in \mathcal{M}(G) \) and \( |\sigma_i| < \infty \), for \( i \geq 1 \).

In both cases we can find an infinite sequence of models \( \sigma_1, \sigma_2, \ldots \) such that \( \sigma = \sigma_1 \sigma_2 \sigma_3 \cdots \), with \( \sigma_i \in \mathcal{M}(G) \) for \( i \geq 1 \). (For the first case, define \( \sigma_i \equiv \sigma_k \), for \( i > k \).) From \( G \models \varphi \)
we obtain \( \sigma_i \models \varphi \) for \( i \geq 1 \). Then the definition of \( C^\infty \) leads to \( \sigma \models C^\infty \varphi \).
Parallel Composition

We prove the soundness of the General Parallel Composition Rule. Assume $S_1 \text{ sat } \varphi_1$ and $S_2 \text{ sat } \varphi_2$ are valid, $\text{dch}(\varphi_1) \subseteq \text{dch}(S_1)$, and $\text{dch}(\varphi_2) \subseteq \text{dch}(S_2)$.

We show the validity of

$$S_1 \parallel S_2 \text{ sat } (\varphi_1 \land [\varphi_2 C \Box \text{noact}(...)] \lor (\varphi_2 \land [\varphi_1 C \Box \text{noact}(...)]).$$

Consider any $\sigma \in M(S_1 \parallel S_2)$. Then $\text{dch}(\sigma) \subseteq \text{dch}(S_1) \cup \text{dch}(S_2)$, and for all $i \in \{1, 2\}$ there exist $\sigma_i \in M(S_i)$ such that $|\sigma| = \max(|\sigma_1|, |\sigma_2|)$,

$c \notin \sigma(\tau) \lor c \notin \sigma(\tau)$, for all $\tau < |\sigma|$, and

$$[\sigma]_{\text{dch}(S_i)}(\tau) = \begin{cases} \sigma_i(\tau) & \text{for all } \tau < |\sigma_i| \\ \varnothing & \text{for all } \tau, |\sigma_i| \leq \tau < |\sigma| \end{cases}$$

Suppose $|\sigma_1| \leq |\sigma_2|$. Then $|\sigma| = |\sigma_2|$. We prove $\sigma \models \varphi_2 \land [\varphi_1 C \Box \text{noact}(...)]$.

- First we show $\sigma \models \varphi_2$. Since $|[\sigma]_{\text{dch}(S_2)}| = |\sigma| = |\sigma_2|$ and $|[\sigma]_{\text{dch}(S_2)}(\tau) = \sigma_2(\tau)|$, for all $\tau < |\sigma|$, we have $[\sigma]_{\text{dch}(S_2)} = \sigma_2$. Hence, from $\sigma_2 \in M(S_2)$ we obtain $[\sigma]_{\text{dch}(S_2)} \in M(S_2)$, and thus $S_2 \text{ sat } \varphi_2$ leads to $[\sigma]_{\text{dch}(S_2)} \models \varphi_2$.

Since $\text{dch}(\varphi_2) \subseteq \text{dch}(S_2)$, Lemma 3.3.19 leads to $\sigma \models \varphi_2$.

- Next we prove $\sigma \models [\varphi_1 C \Box \text{noact}(...)]$.

From $\sigma_1 \in M(S_1)$ and $S_1 \text{ sat } \varphi_1$ we obtain $\sigma_1 \models \varphi_1$.

Now we define a model $\sigma_3$ that satisfies $\Box \text{noact}(\text{dch}(S_1))$.

Let $\sigma_3$ be such that $|\sigma_3| = |\sigma| - |\sigma_1|$, and, for all $\tau < |\sigma_3|$, $\sigma_3(\tau) = \sigma(\tau + |\sigma_1|)$.

Since $[\sigma]_{\text{dch}(S_1)}(\tau) = \varnothing$, for all $\tau, |\sigma_1| \leq \tau < |\sigma|$, we obtain $[\sigma]_{\text{dch}(S_1)}(\tau + |\sigma_1|) = \varnothing$, for all $\tau < |\sigma| - |\sigma_1|$. Thus, for all $\tau < |\sigma_3|$, $[\sigma]_{\text{dch}(S_1)}(\tau) = [\sigma]_{\text{dch}(S_1)}(\tau + |\sigma_1|) = \varnothing$.

Since, for $\tau \geq |\sigma_3|$, $\sigma_3 \uparrow \tau \models \text{noact}(\text{dch}(S_1))$, we obtain $\sigma_3 \models \Box \text{noact}(\text{dch}(S_1))$, and thus $\sigma_1 \sigma_3 \models \varphi_1 C \Box \text{noact}(\text{dch}(S_1))$.

Since $[\sigma]_{\text{dch}(S_1)}(\tau) = \begin{cases} \sigma_1(\tau) & \text{for all } \tau < |\sigma_1| \\ \varnothing & \text{for all } \tau, |\sigma_1| \leq \tau < |\sigma| \end{cases}$, we have $[\sigma]_{\text{dch}(S_1)} = \sigma_1 \sigma_3$.

This leads to $[\sigma]_{\text{dch}(S_1)} \models \varphi_1 C \Box \text{noact}(\text{dch}(S_1))$. Since $\text{dch}(\varphi_1) \subseteq \text{dch}(S_1)$, we have $\text{dch}(\varphi_1 C \Box \text{noact}(\text{dch}(S_1))) \subseteq \text{dch}(S_1)$ and Lemma 3.3.19 leads to $\sigma \models \varphi_1 C \Box \text{noact}(\text{dch}(S_1))$.

Similarly, for $|\sigma_2| \leq |\sigma_1|$ we can prove $\sigma \models \varphi_1 \land [\varphi_2 C \Box \text{noact}(...)]$, which proves the soundness of the General Parallel Composition Rule.
B.2 Preciseness of the Proof System in Section 3.3

To prove Lemma 3.3.22 (Preciseness) we show that for every program \( S \) we can prove \( S \text{ sat } \varphi \) where \( \varphi \) is precise for \( S \). To show that an assertion \( \varphi \) is precise for a program \( S \), we have to prove

1. \( S \) satisfies \( \varphi \), i.e. \( \sigma \models \varphi \) for all \( \sigma \in \mathcal{M}(S) \),

2. if \( \sigma \) is well-formed, \( \text{dch}(\sigma) \subseteq \text{dch}(S) \), and \( \sigma \models \varphi \), then \( \sigma \in \mathcal{M}(S) \), and

3. \( \text{dch}(\varphi) = \text{dch}(S) \).

Part 1 follows from the soundness of the proof system (Theorem 3.3.20) and part 3 can be verified easily; here we prove the part 2. We show, by induction on the structure of \( S \), that we can derive \( S \text{ sat } \varphi \) with \( \varphi \) precise for \( S \).

Skip

If \( S \equiv \text{skip} \) then by the Skip Axiom we obtain \( \text{skip } \text{sat } \text{done} \).

We show that assertion \( \text{done} \) is a precise assertion for statement \( \text{skip} \).

Consider any well-formed model \( \sigma \) with \( \text{dch}(\sigma) \subseteq \text{dch}(\text{skip}) \), and thus \( \text{dch}(\sigma) = \emptyset \).

Assume \( \sigma \models \text{done} \). Then \( |\sigma| = 0 \), and hence \( \sigma \in \mathcal{M}(\text{skip}) \).

Delay

If \( S \equiv \text{delay } d \), then we obtain by the Delay Axiom, \( \text{delay } d \text{ sat } \Diamond_{=d} \text{done} \).

We show that \( \Diamond_{=d} \text{done} \) is precise for \( \text{delay } d \).

Consider any well-formed model \( \sigma \) with \( \text{dch}(\sigma) \subseteq \text{dch}(\text{delay } d) = \emptyset \).

Then \( \text{dch}(\sigma) = \emptyset \), and thus for all \( \tau < |\sigma| \), \( \sigma(\tau) = \emptyset \). Assume \( \sigma \models \Diamond_{=d} \text{done} \).

Then \( |\sigma| = d \), and hence \( \sigma \in \mathcal{M}(\text{delay } d) \).

Send and Receive

Consider \( S \equiv c! \). By the Send Axiom we can derive the formula

\( c! \text{ sat } \text{wait}(c!) \cup (\text{comm}(c) \cup_{=K_c} \text{done}) \).

We show that \( \text{wait}(c!) \cup (\text{comm}(c) \cup_{=K_c} \text{done}) \) is precise for \( c! \). Let \( \sigma \) be a well-formed model such that \( \text{dch}(\sigma) \subseteq \text{dch}(c!) = \{c, c!\} \).

Assume \( \sigma \models \text{wait}(c!) \cup (\text{comm}(c) \cup_{=K_c} \text{done}) \). Then, by Lemma B.1.1,

1. for all \( \tau \in \text{TIME} \), \( \sigma \uparrow \tau \models \text{wait}(c!) \), or

2. there exists a \( \tau \in \text{TIME} \) such that for all \( \tau < \tau', \sigma \uparrow \tau \models \text{wait}(c!) \) and \( \sigma \uparrow \tau \models \text{comm}(c) \cup_{=K_c} \text{done} \).
Thus

1. for all $\tau \in \text{TIME}$, $|\sigma| > \tau$ and $c! \in \sigma(\tau)$, or

2. there exists a $\tau \in \text{TIME}$, such that for all $\tau_1 < \tau$, $c! \in \sigma(\tau_1)$,
   for all $\tau_2, \tau_2 \leq \tau_2 < \tau + K_c$, $c \in \sigma(\tau_2)$, and $|\sigma| = \tau + K_c$.

Since $dch(\sigma) \subseteq \{c, c!\}$ and $\sigma$ a well-formed model,

1. for all $\tau \in \text{TIME}$, $\sigma(\tau) = \{c!\}$ and $|\sigma| = \infty$, or

2. $\sigma = \sigma_1 \sigma_2$ with $|\sigma_1| < \infty$, for all $\tau < |\sigma_1|$: $\sigma_1(\tau) = \{c!\}$,
   for all $\tau < K_c$: $\sigma_2(\tau) = \{c\}$, and $|\sigma_2| = K_c$.

This implies

1. $\sigma \in \text{WaitSend}(c)$ and $|\sigma| = \infty$, or

2. $\sigma = \sigma_1 \sigma_2$ with $|\sigma_1| < \infty$, $\sigma_1 \in \text{WaitSend}(c)$ and $\sigma_2 \in \text{Comm}(c)$.

Hence $\sigma \in \text{SEQ}(\text{WaitSend}(c), \text{Comm}(c)) = M(c!)$.

Similarly, $\text{wait}(c?) \cup (\text{comm}(c) \cup \text{done})$ is precise for $c?$.  

**Sequential Composition**

Consider $S \equiv S_1; S_2$. By the induction hypothesis we can derive $S_1 \text{ sat } \varphi_1$ and $S_2 \text{ sat } \varphi_2$ where $\varphi_1$ and $\varphi_2$ are precise for $S_1$ and $S_2$, respectively. By the Invariance Axiom we obtain $S_1 \text{ sat } \Box \neg\text{act}(dch(S_2) - dch(S_1))$ and $S_2 \text{ sat } \Box \neg\text{act}(dch(S_1) - dch(S_2))$.

Thus, using the Conjunction Rule,

$S_1 \text{ sat } \varphi_1 \land \Box \neg\text{act}(dch(S_2) - dch(S_1))$ and $S_2 \text{ sat } \varphi_2 \land \Box \neg\text{act}(dch(S_1) - dch(S_2))$.

Hence, by the Sequential Composition Rule, we obtain $S_1; S_2 \text{ sat } \varphi$ with

$\varphi \equiv (\varphi_1 \land \Box \neg\text{act}(dch(S_2) - dch(S_1))) \land (\varphi_2 \land \Box \neg\text{act}(dch(S_1) - dch(S_2)))$.

We prove that $\varphi$ is precise for $S_1; S_2$.

Consider a well-formed model $\sigma$ such that $dch(\sigma) \subseteq dch(S_1; S_2)$. Assume $\sigma \models \varphi$.

By definition of the $\Box$ operator, there exist models $\sigma_1$ and $\sigma_2$ such that $\sigma = \sigma_1 \sigma_2$, $\sigma_1 \models \varphi_1 \land \Box \neg\text{act}(dch(S_2) - dch(S_1))$, and $\sigma_2 \models \varphi_2 \land \Box \neg\text{act}(dch(S_1) - dch(S_2))$.

Using $\sigma_1 \models \Box \neg\text{act}(dch(S_2) - dch(S_1))$, Lemma 3.3.1 leads to $[\sigma_1]_{dch(S_1) \cup dch(S_2)} = [\sigma_1]_{dch(S_1)}$.

From $dch(\sigma) \subseteq dch(S_1; S_2) = dch(S_1) \cup dch(S_2)$ and $\sigma = \sigma_1 \sigma_2$ we obtain

$dch(\sigma_1) \subseteq dch(S_1) \cup dch(S_2)$. Thus, by Lemma 3.2.5, $\sigma_1 = [\sigma_1]_{dch(S_1) \cup dch(S_2)} = [\sigma_1]_{dch(S_1)}$.

Hence, by Lemma 3.2.5, $dch(\sigma_1) \subseteq dch(S_1)$. Together with $\sigma_1 \models \varphi_1$, preciseness of $\varphi_1$ for $S_1$ leads to $\sigma_1 \in M(S_1)$. Similarly, $\sigma_2 \in M(S_2)$. By $\sigma = \sigma_1 \sigma_2$ this leads to $\sigma \in M(S_1; S_2)$. 

Guarded Command without Delay

Consider $S \equiv G \equiv \left[ \left[ \bigwedge_{i=1}^{n} c_i ? \rightarrow S_i \right] \land \text{ delay } d \rightarrow S \right]$. First assume $d = \infty$, thus $S \equiv G \equiv \left[ \left[ \bigwedge_{i=1}^{n} c_i ? \rightarrow S_i \right] \right]$. Assume, by the induction hypothesis, that we can derive $c_i ?; S_i \models \varphi_i$ with $\varphi_i$ precise for $c_i ?, S_i$, for $i \in \{1, \ldots, n\}$.

By the Invariance Axiom and the Conjunction Rule we can derive $c_i ?, S_i \models \varphi_i \land \left[ \text{noact}(dch(G) - dch(c_i ?, S_i)) \land \text{ done} \right]$, for $i \in \{1, \ldots, n\}$.

Applying the Rule for Guarded Command without Delay, we obtain
$$\left[ \bigwedge_{i=1}^{n} c_i ? \rightarrow S_i \right] \models \varphi$$

where $\varphi \equiv \text{wait}_G \equiv \bigvee_i (\varphi_i \land \text{comm}(c_i) \land \left[ \text{noact}(dch(G) - dch(c_i ?, S_i)) \land \text{ done} \right])$

with $\text{wait}_G \equiv \bigwedge_i \text{wait}(c_i ?) \land \text{noact}(dch(G) - \{c_1 ?, \ldots, c_n ?\})$.

We prove that $\varphi$ is precise for $G$. Let $\sigma$ be a well-formed model such that $dch(\sigma) \subseteq dch(\left[ \bigwedge_{i=1}^{n} c_i ? \rightarrow S_i \right])$. Assume $\sigma \models \varphi$. Using Lemma B.1.1, this implies

1. $\sigma \models \Box \text{wait}_G$, or
2. there exists a $\tau \in \text{TIME}$, such that $\sigma \models \Box_{\leq \tau} \text{wait}_G$ and $\sigma \models \Box_{= \tau} \bigvee_i (\varphi_i \land \text{comm}(c_i) \land \left[ \text{noact}(dch(G) - dch(c_i ?, S_i)) \land \text{ done} \right])$.

By the well-formedness of $\sigma$, $dch(\sigma) \subseteq dch(G)$, and the definition of $\text{wait}_G$,

1. $|\sigma| = \infty$, and for all $\tau \in \text{TIME}$, $\sigma(\tau) = \{c_1 ?, \ldots, c_n ?\}$, or
2. there exists a $\tau \in \text{TIME}$ such that for all $\tau_1 < \tau$, $\sigma(\tau_1) = \{c_1 ?, \ldots, c_n ?\}$, there exists a $k \in \{1, \ldots, n\}$ such that $\sigma \uparrow \tau \models \varphi_k \land \text{comm}(c_k) \land \left[ \text{noact}(dch(G) - dch(c_k ?, S_k)) \land \text{ done} \right]$.

By the definition of $\text{Wait}(G)$ and using well-formedness of $\sigma$,

1. $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$, or
2. $\sigma = \sigma_1 \sigma_2$ with $\sigma_1 \in \text{Wait}(G)$ and there exists a $k \in \{1, \ldots, n\}$ such that $\sigma_2 \models \varphi_k \land \text{comm}(c_k)$, $\sigma_2$ is well-formed, and $\sigma_2 \models \text{noact}(dch(G) - dch(c_k ?, S_k)) \land \text{ done}$.

From $\sigma_2 \models \text{noact}(dch(G) - dch(c_k ?, S_k)) \land \text{ done}$, by Lemma 3.3.1,

$$[\sigma_2]_{dch(c_k ?, S_k) \cup dch(G)} = [\sigma_2]_{dch(c_k ?, S_k)}.$$ Using $dch(\sigma_2) \subseteq dch(\sigma) \subseteq dch(G)$, Lemma 3.2.5 leads to $\sigma_2 = [\sigma_2]_{dch(G)}$. Since $dch(c_k ?, S_k) \subseteq dch(G)$,

$$[\sigma_2]_{dch(G)} = [\sigma_2]_{dch(c_k ?, S_k) \cup dch(G)} = [\sigma_2]_{dch(c_k ?, S_k)}.$$ and thus $\sigma_2 = [\sigma_2]_{dch(c_k ?, S_k)}$.

Hence, by Lemma 3.2.5, $dch(\sigma_2) \subseteq dch(c_k ?, S_k)$. Thus, either

1. $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$, or
2. $\sigma = \sigma_1 \sigma_2$ with $\sigma_1 \in \text{Wait}(G)$ and there exists a $k \in \{1, \ldots, n\}$ such that $\sigma_2 \models \varphi_k \land \text{comm}(c_k)$, $\sigma_2$ is well-formed, and $dch(\sigma_2) \subseteq dch(c_k ?, S_k)$. 

Since $\varphi_k$ is precise for $c_k?; S_k$, we obtain

1. $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$, or
2. $\sigma = \sigma_1\sigma_2$ with $\sigma_1 \in \text{Wait}(G)$ and there exists a $k \in \{1, \ldots, n\}$ such that $\sigma_2 \in \mathcal{M}(c_k?; S_k)$ and $c_k \in \sigma_2(0)$.

Hence

1. $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$, or
2. $\sigma = \sigma_1\sigma_2$ with $\sigma_1 \in \text{Wait}(G)$ and there exists a $k \in \{1, \ldots, n\}$ such that $\sigma_2 \in \text{SEQ}(\text{Comm}(c_k), \mathcal{M}(S_k))$.

Thus

1. $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$, or
2. $\sigma = \sigma_1\sigma_2$ with $\sigma_1 \in \text{Wait}(G)$ and $\sigma_2 \in \text{Comm}(G)$.

Hence $\sigma \in \mathcal{M}(\{[\times_{i=1}^n c_i? \rightarrow S_i]\})$.

For $S \equiv G \equiv \{[\times_{i=1}^n c_i? \rightarrow S_i \mid \text{delay } d \rightarrow S]\}$ with $d < \infty$ we can similarly obtain a precise specification by means of the rule for Guarded Command with Delay.

**Iteration**

Consider $S \equiv \ast G$. Assume by the induction hypothesis that we can derive $G \text{ sat } \varphi$ with $\varphi$ precise for $G$. We prove that $C^\infty \varphi$ is precise for $\ast G$. Let $\sigma$ be a well-formed model such that $dch(\sigma) \subseteq dch(\ast G)$. (Thus $dch(\sigma) \subseteq dch(G)$.) Suppose $\sigma \models C^\infty \varphi$. From the definition of $C^\infty$, there exist models $\sigma_1, \sigma_2, \sigma_3, \ldots$ such that $\sigma = \sigma_1\sigma_2\sigma_3 \cdots$ and $\sigma_i \models \varphi$ for $i \geq 1$. Then, for $i \geq 1$, $dch(\sigma_i) \subseteq dch(G)$, so by preciseness of $\varphi$ for $G$ we have $\sigma_i \in \mathcal{M}(G)$. Hence, using Corollary 3.2.10, $\sigma \in \mathcal{M}(\ast G)$.

**Parallel Composition**

Consider $S \equiv S_1 \| S_2$. Assume, by the induction hypothesis, that we can derive $S_1 \text{ sat } \varphi_1$ and $S_2 \text{ sat } \varphi_2$ with $\varphi_1$ and $\varphi_2$ precise for, respectively, $S_1$ and $S_2$. From preciseness, $dch(\varphi_1) \subseteq dch(S_1)$, and $dch(\varphi_2) \subseteq dch(S_2)$. Thus we can apply the General Parallel Composition Rule, obtaining

$$S_1 \| S_2 \text{ sat } (\varphi_1 \sqcap [\varphi_2 C \square \text{noact}(dch(S_2))]) \vee (\varphi_2 \sqcap [\varphi_1 C \square \text{noact}(dch(S_1))]).$$

We prove that $(\varphi_1 \sqcap [\varphi_2 C \square \text{noact}(dch(S_2))]) \vee (\varphi_2 \sqcap [\varphi_1 C \square \text{noact}(dch(S_1))])$ is precise for $S$. Let $\sigma$ be well-formed such that $dch(\sigma) \subseteq dch(S_1 \| S_2)$. Suppose $\sigma \models (\varphi_1 \sqcap [\varphi_2 C \square \text{noact}(dch(S_2))]) \vee (\varphi_2 \sqcap [\varphi_1 C \square \text{noact}(dch(S_1))]).$
We show that $\sigma \in M(S_1 \| S_2)$.

By well-formedness of $\sigma$, we obtain

$\neg(c! \in \sigma(\tau) \land c? \in \sigma(\tau) \land c! \notin \sigma(\tau) \lor c? \notin \sigma(\tau)$, for all $\tau < |\sigma|$.

It remains to prove, for $i \in \{1, 2\}$, that there exist $\sigma_i \in M(S_i)$ such that

$|\sigma| = \max(|\sigma_1|, |\sigma_2|)$, and $[\sigma]_{dch(S_i)}(\tau) = \begin{cases} \sigma_i(\tau) & \text{for all } \tau < |\sigma_i| \\ \emptyset & \text{for all } \tau, |\sigma_i| \leq \tau < |\sigma| \end{cases}$

Assume $\sigma \models \varphi_1 \land [\varphi_2 \square \text{noact}(dch(S_2))]$.

Define $\sigma_1$ as $[\sigma]_{dch(S_1)}$. Then $\sigma_1$ is well-formed and $dch(\sigma) \subseteq dch(S_1)$. From $\sigma \models \varphi_1$ we obtain, by Lemma 3.3.19, using $dch(\varphi_1) \subseteq dch(S_1)$, that $[\sigma]_{dch(S_1)} \models \varphi_1$, and thus $\sigma_1 \models \varphi_1$. Hence, by the preciseness of $\varphi_1$ for $S_1$, $\sigma_1 \in M(S_1)$.

From $\sigma \models \varphi_2 \square \text{noact}(dch(S_2))$, by Lemma 3.3.19, $[\sigma]_{dch(S_2)} \models \varphi_2 \square \text{noact}(dch(S_2))$.

Then there are $\sigma_2$ and $\sigma_3$ such that $[\sigma]_{dch(S_2)} = \sigma_2 \sigma_3$, $\sigma_2 \models \varphi_2$, and $\sigma_3 \models \square \text{noact}(dch(S_2))$.

Since $\sigma$ is well-formed, $\sigma_2$ is well-formed. Together with $dch(\sigma_2) \subseteq dch(S_2)$, by the preciseness of $\varphi_2$ for $S_2$, we obtain $\sigma_2 \in M(S_2)$.

Note that $|\sigma| = |\sigma_2 \sigma_3| \geq |\sigma_2|$ and $|\sigma| = |\sigma_1|$. Hence $|\sigma| = \max(|\sigma_1|, |\sigma_2|) = |\sigma_1|$ and thus,

using the definition of $\sigma_1$, $[\sigma]_{dch(S_1)}(\tau) = \begin{cases} \sigma_1(\tau) & \text{for all } \tau < |\sigma_1| \\ \emptyset & \text{for all } \tau, |\sigma_1| \leq \tau < |\sigma| \end{cases}$

From $\sigma_3 \models \square \text{noact}(dch(S_2))$, by the definition of $\text{noact}$,

$[\sigma_3]_{dch(S_2)}(\tau) = \emptyset$, for all $\tau < |\sigma_3|$. Since $[\sigma]_{dch(S_2)} = \sigma_2 \sigma_3$, we obtain

$[\sigma]_{dch(S_2)}(\tau) = \begin{cases} \sigma_2(\tau) & \text{for all } \tau < |\sigma_2| \\ \emptyset & \text{for all } \tau, |\sigma_2| \leq \tau < |\sigma| \end{cases}$

Hence $\sigma \in M(S_1 \| S_2)$.

Interchanging the indices 1 and 2 in the proof above yields a proof for the case that $\sigma \models \varphi_2 \land [\varphi_1 \square \text{noact}(dch(S_1))]$. 
Appendix C

Soundness and Completeness of the Proof System in Section 3.4

C.1 Soundness of the Proof System in Section 3.4

In this appendix we prove soundness of the proof system given in Section 3.4. In order to verify that $C : \{p\} \rightarrow S \{q\}$ is valid, we have to prove for any environment $\gamma$, for any well-formed, nonterminating, model $\delta$, and for any $\sigma \in M(S)$:

if $[p]_{\gamma} \delta \sigma$ then $[C]_{\gamma} \delta \sigma$ and if $|\delta \sigma| < \infty$ then $[q]_{\gamma} \delta \sigma$.

Well-Formedness Axiom

First, we prove that if $\sigma$ is well-formed, then $[WellForm_{cset}]_{\gamma} \sigma$ for any environment $\gamma$ and any finite set $cset \subseteq DCHAN$. If $\sigma$ is well-formed then, for all $\tau < |\sigma|$, 

1. $\neg(c! \in \sigma(\tau) \land c? \in \sigma(\tau))$;
2. $\neg(c \in \sigma(\tau) \land c! \in \sigma(\tau))$ and $\neg(c \in \sigma(\tau) \land c? \in \sigma(\tau))$.

Hence, for all $\tau < |\sigma|$, 

1. $\neg(c! \in \sigma(\tau) \land c? \in \sigma(\tau))$, for all $c$ with $\{c!, c?\} \subseteq cset$;
2. $\neg(c \in \sigma(\tau) \land c! \in \sigma(\tau))$, for all $c$ with $\{c, c!\} \subseteq cset$, and $\neg(c \in \sigma(\tau) \land c? \in \sigma(\tau))$, for all $c$ with $\{c, c?\} \subseteq cset$.

Given our interpretation of assertions, this implies 

1. $[\Lambda_{(c!,c?)} \subseteq cset \neg(wait \ to \ c? \ at \ t \land \ wait \ to \ c! \ at \ t)]_{\gamma} \sigma$;
2. $[(\Lambda_{(c!,c)} \subseteq cset \neg(wait \ to \ c! \ at \ t \land \ comm \ via \ c \ at \ t)) \land$

$(\Lambda_{(c,c?)} \subseteq cset \neg(wait \ to \ c? \ at \ t \land \ comm \ via \ c \ at \ t))]_{\gamma} \sigma$.

Hence, by definition, $[\forall t < time : MW_{cset}(t) \land Excl_{cset}(t)]_{\gamma} \sigma$, and thus $[WellForm_{cset}]_{\gamma} \sigma$. 
Next we show that $WellForm_{cset} : \{true\} S \{WellForm_{cset}\}$ is valid for any program $S$ any finite set $cset \subseteq DCHAN$. Let $\gamma$ be an arbitrary environment, and $\hat{\sigma}$ a well-formed model such that $|\hat{\sigma}| < \infty$. Consider any $\sigma \in M(S)$. Then, by Lemma 3.2.13, $\sigma$ is well-formed, and, since $\hat{\sigma}$ is well-formed, $\hat{\sigma}\sigma$ is also well-formed. Hence $[WellForm_{cset}]\gamma\hat{\sigma}\sigma$.

Consequence

To prove soundness of the Consequence Rule, assume $\models C_0 : \{p_0\} S \{q_0\}$,
$p \land time < \infty \rightarrow p_0, C_0 \rightarrow C$, and $q_0 \rightarrow q$.
We show that $C : \{p\} S \{q\}$ is valid.
Let $\gamma$ be an arbitrary environment, and $\hat{\sigma}$ a well-formed model such that $|\hat{\sigma}| < \infty$.
Consider any $\sigma \in M(S)$. Assume $[p]\gamma\hat{\sigma}$.
Then $[p \land time < \infty]\gamma\hat{\sigma}$ and hence, by $p \land time < \infty \rightarrow p_0$, we obtain $[p_0]\gamma\hat{\sigma}$.
Then $\models C_0 : \{p_0\} S \{q_0\}$ leads to $[C_0]\gamma\hat{\sigma}\sigma$ and if $|\hat{\sigma}\sigma| < \infty$ then $[q_0]\gamma\hat{\sigma}\sigma$.
Thus, from $C_0 \rightarrow C$ and $q_0 \rightarrow q$, we obtain $[C]\gamma\hat{\sigma}\sigma$ and if $|\hat{\sigma}\sigma| < \infty$ then $[q]\gamma\hat{\sigma}\sigma$.

Initial Invariance

Suppose $time$ does not occur in $p$. We prove $\models p : \{p\} S \{p\}$. Consider an environment $\gamma$ and a terminating model $\hat{\sigma}$. Assume $[p]\gamma\hat{\sigma}$. Consider $\sigma \in M(S)$. By Lemma 3.4.3, $[p]\gamma\hat{\sigma}$ leads to $[p[\hat{\sigma}/time]]\gamma\hat{\sigma}\sigma$. Since $time$ does not occur in $p$ this leads to $[p]\gamma\hat{\sigma}\sigma$.

Channel Invariance

Let $cset$ be a finite subset of $DCHAN$ such that $cset \cap dch(S) = \emptyset$.
We prove $\models \neg cset \text{ during } [t_0, time) : \{time = t_0\} S \{\neg cset \text{ during } [t_0, time)\}$.
Consider an environment $\gamma$ and a terminating model $\hat{\sigma}$. Assume $[time = t_0]\gamma\hat{\sigma}$.
Consider $\sigma \in M(S)$. Then, by Lemma 3.2.13, $dch(\sigma) \subseteq dch(S)$ and hence
$[\neg cset \text{ during } [t_0, time)]\gamma\hat{\sigma}\sigma$.
The soundness of the Conjunction Rule and the Quantification Rule can be easily proved.

Substitution

Suppose $time$ does not occur in expression $exp$, and $\models C : \{p\} S \{q\}$.
We prove $\models C[exp/t] : \{p[exp/t]\} S \{q[exp/t]\}$.
Consider an environment $\gamma$ and a terminating model $\hat{\sigma}$. Let $\sigma \in M(S)$.
Assume $[p[exp/time]]\gamma\hat{\sigma}$. Then $[p](\gamma : t \mapsto \mathcal{V}(exp)(\gamma, \hat{\sigma}))\hat{\sigma}$.
Since $time$ does not occur in $exp$, Lemma 3.4.1 leads to $[p](\gamma : t \mapsto \mathcal{V}(exp)(\gamma, \hat{\sigma}))\hat{\sigma}$.
From $\models C : \{p\} S \{q\}$ we obtain $[q](\gamma : t \mapsto \mathcal{V}(exp)(\gamma, \hat{\sigma}))\hat{\sigma}$ and
if $|\sigma| < \infty$ then $[[C](\gamma : t \mapsto \nu(\exp)(\gamma, \delta \sigma))\delta \sigma]$. 
Thus $[[q[\exp/l]]\gamma \delta \sigma$ and if $|\sigma| < \infty$ then $[[C[\exp/l]]\gamma \delta \sigma$.

**Skip**

We prove that $time = t_0 : \{time = t_0\} \text{ skip } \{time = t_0\}$ is valid.
Let $\gamma$ be an arbitrary environment, and $\sigma$ a well-formed model such that $|\sigma| < \infty$.
Consider any $\sigma \in M(\text{skip})$. Then $|\sigma| = 0$, thus $\delta \sigma = \delta$, and hence $|\delta \sigma| = |\delta|$.
Assume $[time = t_0]\gamma \delta \sigma$. Then clearly $\gamma(t_0) = |\delta|$ and thus $\gamma(t_0) = |\delta \sigma|$.
Hence $[time = t_0]\gamma \delta \sigma$.

**Delay**

We show that $time = t_0 + d : \{time = t_0\} \text{ delay } d \{time = t_0 + d\}$ is valid.
Let $\gamma$ be an arbitrary environment, and $\sigma$ a well-formed model such that $|\sigma| < \infty$.
Assume $[time = t_0]\gamma \delta \sigma$. Thus $\gamma(t_0) = |\delta|$. Consider $\sigma \in M(\text{delay } d)$. Then $|\sigma| = d$.
Hence $|\delta \sigma| = |\delta| + |\sigma| = \gamma(t_0) + d$. This implies $[time = t_0 + d]\gamma \delta \sigma$.

**Send**

Assume

$$(\exists t \geq t_0 : \text{wait to c! at } t_0 \text{ until comm at } t \land time = t + K_c) \rightarrow C$$

(C.1)

We prove that $C : \{time = t_0\} c! \{C \land time < \infty\}$ is valid.
Let $\gamma$ be an arbitrary environment, and $\sigma$ a well-formed model such that $|\sigma| < \infty$.
Consider any $\sigma \in M(c!)$. Then $\sigma = \sigma_1 \sigma_2$ with $\sigma_1 \in \text{WaitSend}(c)$ and $\sigma_2 \in \text{Comm}(c)$.
Assume $[time = t_0]\gamma \delta \sigma$. Then $\gamma(t_0) = |\delta|$.

1. If $|\sigma_1| = \infty$ then $\sigma = \sigma_1 \in \text{WaitSend}(c)$ and thus
   $[[\text{wait to c! during } [t_0, \infty) \land time = \infty] \gamma \delta \sigma$, since $\gamma(t_0) = |\delta|$. Then
   $[[\text{wait to c! during } [t_0, \infty) \land time = \infty] \gamma \delta \sigma$, and thus
   $[\exists t \geq t_0 \text{ wait to c! at } t_0 \text{ until comm at } t \land time = t + K_c] \gamma \delta \sigma$.
   Hence, by (C.1), $[[C] \gamma \delta \sigma$. Further, observe that $|\delta \sigma| = \infty$.

2. If $|\sigma_1| < \infty$ then there exists a $\tau \in \text{TIME}$, such that for all $\tau_1 < \tau$: $\sigma(\tau_1) = \{c\}$,
   for all $\tau_2, \tau \leq \tau_2 < \tau + K_c : \sigma(\tau_2) = \{c\}$, and $|\sigma| = \tau + K_c$.
   Thus $[[\exists t \geq t_0 \text{ wait to c! during } [t_0, t) \land \text{comm via c during } [t, t + K_c) \land
   \text{time} = t + K_c] \gamma \delta \sigma$, and $[[\text{time} < \infty] \gamma \delta \sigma$. Hence
   $[[\exists t \geq t_0 \text{ wait to c! at } t_0 \text{ until comm at } t \land time = t + K_c \land time < \infty] \gamma \delta \sigma$.
   By (C.1) this leads to $[[C \land time < \infty] \gamma \delta \sigma$.

The soundness of the Receive Rule can be proved similar.
Sequential Composition

Assume $\models C_1 : \{p\}S_1\{r\}$ and $\models C_2 : \{r\}S_2\{q\}$.
We show that $(C_1 \land \text{time} = \infty) \lor C_2 : \{p\}S_1; S_2\{q\}$ is valid.
Let $\gamma$ be an arbitrary environment, and $\sigma$ a well-formed model such that $|\sigma| < \infty$.
Consider any $\sigma \in \mathcal{M}(S_1; S_2) = \text{SEQ}(\mathcal{M}(S_1), \mathcal{M}(S_2))$.
Then $\sigma = \sigma_1\sigma_2$ with $\sigma_1 \in \mathcal{M}(S_1)$ and $\sigma_2 \in \mathcal{M}(S_2)$). Assume $[p]^{\gamma}\sigma$.
Then, by $\models C_1 : \{p\}S_1\{r\}$, we obtain $[C_1]^{\gamma}\sigma_1$ and if $|\sigma_1| < \infty$ then $[r]^{\gamma}\sigma_1$.

1. If $|\sigma_1| = \infty$ then $\sigma = \sigma_1$ and $|\sigma\sigma| = \infty$. Thus $[C_1]^{\gamma}\sigma$ and $[\text{time} = \infty]^{\gamma}\sigma$.

2. If $|\sigma_1| < \infty$ then $[r]^{\gamma}\sigma_1$. Hence, by $\models C_2 : \{r\}S_2\{q\}$, we obtain $[C_2]^{\gamma}\sigma_1\sigma_2$, and if $|\sigma_1\sigma_2| < \infty$ then $[q]^{\gamma}\sigma_1\sigma_2$. Thus $[C_2]^{\gamma}\sigma$, and if $|\sigma| < \infty$ then $[q]^{\gamma}\sigma$.

Guarded Command

First we prove the soundness of the Rule for Guarded Command without Delay. Assume

\[(\text{wait in } G \text{ during } [t_0, \infty) \land \text{time} = \infty) \rightarrow C_{\text{nonterm}}\]  \hfill (C.2)

\[(\exists t, t_0 \leq t < \infty : \text{wait in } G \text{ during } [t_0, t) \land \text{comm } c_i \text{ in } G \text{ from } t) \rightarrow p_i, \text{ for all } i \in \{1, \ldots, n\}\]  \hfill (C.3)

\[\models C_i : \{p_i\} S_i\{q_i\}, \text{ for all } i \in \{1, \ldots, n\}\]  \hfill (C.4)

We prove that $C_{\text{nonterm}} \lor \bigvee_{i=1}^n C_i : \{\text{time} = t_0\} \left[\bigvee_{i=1}^n c_i ? \rightarrow S_i\right] \{\bigvee_{i=1}^n q_i\}$ is valid.
Let $\gamma$ be an arbitrary environment, and $\sigma$ a well-formed, nonterminating model.
Assume $[\text{time} = t_0]^{\gamma}\sigma$. Thus $\gamma(t_0) = |\sigma|$. Consider any $\sigma \in \mathcal{M}(\bigvee_{i=1}^n c_i ? \rightarrow S_i) = \text{SEQ}(\text{Wait}(G), \text{Comm}(G))$. Then there are two possibilities:

- Either $\sigma \in \text{Wait}(G)$ and $|\sigma| = \infty$. Then, by definition of $\text{Wait}(G)$,
  $[\text{wait in } G \text{ during } [t_0, \infty)]^{\gamma}\sigma$, and thus, by (C.3), $[C_{\text{nonterm}}]^{\gamma}\sigma$.

- Or $\sigma = \sigma_1\sigma_2\sigma_3$ with $\sigma_1 \in \text{Wait}(G)$ and $|\sigma| < \infty$, $\sigma_2 \in \text{Comm}(c_k)$, and
  $\sigma_3 \in \mathcal{M}(S_k)$ for some $k$. Then there exists a $\tau \in \text{TIME}$, $\tau \geq |\sigma| = \gamma(t_0)$ such that
  $[\text{wait in } G \text{ from } t_0 \text{ till } \tau]^{\gamma}\sigma_1\sigma_2$, and there exists a $k$ such that
  $[\text{comm } c_k \text{ in } G \text{ from } \tau]^{\gamma}\sigma_1\sigma_2$. Hence
  $[\exists t, t_0 \leq t < \infty : \text{wait in } G \text{ during } [t_0, t) \land \text{comm } c_k \text{ in } G \text{ from } t]^{\gamma}\sigma_1\sigma_2$. Thus,
  by (C.4), $[p_k]^{\gamma}\sigma_1\sigma_2$. Since $\sigma_3 \in \mathcal{M}(S_k)$, we obtain from (C.4); $[C_k]^{\gamma}\sigma_1\sigma_2\sigma_3$.
By $\sigma = \sigma_1\sigma_2\sigma_3$, $[C_k]^{\gamma}\sigma$. Similarly, if $|\sigma| < \infty$ then $[q_k]^{\gamma}\sigma$.

The soundness of the rule for Guarded Command with Delay is proved similarly.
Iteration

Assume

$$\models C : \{C\} G \{C\}$$  \hspace{2cm} (C.5)

$$(\forall t_1 < \infty \exists t_2 > t_1 : C[t_2/time]) \rightarrow C_{\text{nonterm}}$$  \hspace{2cm} (C.6)

We show that $C_{\text{nonterm}} \land \text{time} = \infty : \{C\} \ast G \{false\}$ is valid. Let $\gamma$ be an arbitrary environment, and $\hat{\sigma}$ a well-formed model such that $|\hat{\sigma}| < \infty$. Assume $[C]_T \gamma \hat{\sigma}$. Consider any $\sigma \in \mathcal{M}(G)$, then

$$\sigma \in \{\sigma \mid \text{there exists a } k \in \mathbb{N}, k \geq 1 \text{ and models } \sigma_1, \ldots, \sigma_k \text{ such that}$$

$$\sigma = \sigma_1 \cdots \sigma_k, \text{ with } \sigma_i \in \mathcal{M}(G), \text{ for } i \in \{1, \ldots, k\},$$

$$|\sigma_i| < \infty, \text{ for } i \in \{1, \ldots, k - 1\}, \text{ and } |\sigma_k| = \infty \} \cup \{\sigma \mid \text{there exists an infinite sequence of models } \sigma_1, \sigma_2, \ldots \text{ such that}$$

$$\sigma = \sigma_1 \sigma_2 \cdots, \text{ with } \sigma_i \in \mathcal{M}(G) \text{ and } |\sigma_i| < \infty, \text{ for } i \geq 1 \}.$$

Hence $|\sigma| = \infty$, and thus $[\text{time} = \infty]_T \gamma \hat{\sigma} \sigma$. Remains to prove $[C_{\text{nonterm}} \lor C]_T \gamma \hat{\sigma} \sigma$.

There are two possibilities:

1. There exists a $k \in \mathbb{N}, k \geq 1$ and models $\sigma_1, \ldots, \sigma_k$ such that $\sigma = \sigma_1 \cdots \sigma_k$, with $\sigma_i \in \mathcal{M}(G)$, for $i \in \{1, \ldots, k\}$, $|\sigma_i| < \infty$, for $i \in \{1, \ldots, k - 1\}$, and $|\sigma_k| = \infty$.

Then we prove, by induction on $i$, that $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_i$, for $i \in \{0, \ldots, k - 1\}$.

**Basic** For $i = 0$ we have, by our assumption, $[C]_T \gamma \hat{\sigma}$.

**Induction** Consider $i$ with $0 < i < k$.

With $i = k - 1$ we obtain $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_{k-1}$. Since $\sigma_k \in \mathcal{M}(G)$, (C.5) leads to $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_k$ and thus $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_k$. Since $|\hat{\sigma}| = \infty$ we have $[C[\infty/time]]_T \gamma \hat{\sigma}_1 \cdots \sigma_k$.

Hence $[\forall t_1 < \infty \exists t_2 > t_1 : C[t_2/time]]_T \gamma \hat{\sigma}_1 \cdots \sigma_k$, and thus, by (C.6), $[C_{\text{nonterm}}]_T \gamma \hat{\sigma}_1 \cdots \sigma_k$.

2. There exists an infinite sequence of models $\sigma_1, \sigma_2, \ldots$ such that $\sigma = \sigma_1 \sigma_2 \cdots$, with $\sigma_i \in \mathcal{M}(G)$ and $|\sigma_i| < \infty$, for $i \geq 1$.

We prove, by induction on $i$ that, for all $i \geq 0$, $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_i$.

**Basic** For $i = 0$ we have, by our assumption, $[C]_T \gamma \hat{\sigma}$.

**Induction** Let $i > 0$. By the induction hypothesis, $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_{i-1}$.

Then $\sigma_i \in \mathcal{M}(G)$, and $|\sigma_i| < \infty$ lead, by (C.5), to $[C]_T \gamma \hat{\sigma}_1 \cdots \sigma_i$.

Now, using Lemma 3.4.3, $[C[\hat{\sigma}_1 \cdots \sigma_i/time]]_T \gamma \hat{\sigma}_1 \cdots \sigma_i \sigma_{i+1} \cdots$, for $i \geq 0$.

Thus $[C[\hat{\sigma}_1 \cdots \sigma_i/time]]_T \gamma \hat{\sigma}_1 \sigma_{i+1} \cdots$, for $i \geq 0$. Observe that for all $\tau_1 \in \text{TIME}$ there exists a $i$ such that $|\hat{\sigma}_1 \cdots \sigma_i| > \tau_1$. Hence, for all $\tau_1 \in \text{TIME}$ there exists a $\tau_2 > \tau_1$.
such that $[[C[t_2/time]]\gamma\dot{\sigma}]$. This leads to $\forall t_1 < \infty \exists t_2 > t_1 : C[t_2/time]$$\gamma\dot{\sigma}$, and thus, by (C.6), to $[[C_{nonterm}]\gamma\dot{\sigma}]$.

**Parallel Composition**

Assume

$$\models C_i : \{p_i\} S_1 \{q_i\}, \text{ for } i = 1, 2 \quad (C.7)$$

$$\exists t_1, t_2 : time = max(t_1, t_2) \land \bigwedge_{i=1}^{2} C_i[t_i/time] \land \text{no dch}(S_i) \text{ during } [t_i, time] \rightarrow C \quad (C.8)$$

$$\exists t_1, t_2 : time = max(t_1, t_2) \land \bigwedge_{i=1}^{2} q_i[t_i/time] \land \text{no dch}(S_i) \text{ during } [t_i, time] \rightarrow q \quad (C.9)$$

$$\text{dch}(C_i, q_i) \subseteq \text{dch}(S_i), \text{ for } i = 1, 2 \quad (C.10)$$

$$t_1 \text{ and } t_2 \text{ are fresh logical variables} \quad (C.11)$$

We show that $C : \{p_1 \land p_2\} S_1 S_2 \{q\}$ is valid.

Let $\gamma$ be an arbitrary environment, and $\dot{\sigma}$ a well-formed model such that $|\dot{\sigma}| < \infty$.

Assume $[[p_1 \land p_2]\gamma\dot{\sigma}]$. Consider any $\sigma \in M(S_1\|S_2)$.

Then $\text{dch}(\sigma) \subseteq \text{dch}(S_1) \cup \text{dch}(S_2)$, and for $i = 1, 2$ there exist $\sigma_i \in M(S_i)$ such that

$$|\sigma| = max(|\sigma_1|, |\sigma_2|), \text{ and } [\sigma]_{\text{dch}(S_i)}(\tau) = \begin{cases} \sigma_i(\tau) & \text{for all } \tau < |\sigma_i| \\ \emptyset & \text{for all } \tau, |\sigma_i| \leq \tau < |\sigma| \end{cases}$$

By (C.7) we obtain $[[C_i]y\dot{\sigma}i]$; and if $|\dot{\sigma}_i| < \infty$ then $[[q_i]y\dot{\sigma}i]$. Define $\tilde{\gamma} = (\gamma : t_1 \mapsto |\dot{\sigma}| + |\sigma_1|, t_2 \mapsto |\dot{\sigma}| + |\sigma_2|)$. Then, using (C.11), $[[C_i[t_i/time]]y\dot{\sigma}i]$; and if $|\dot{\sigma}_i| < \infty$ then $[[q_i[t_i/time]]y\dot{\sigma}i]$. Using Lemma 3.4.3, we obtain $[[C_i[t_i/time]]y\dot{\sigma}|_{\text{dch}(S_i)}]$, and if $|\dot{\sigma}_i| < \infty$ then $[[q_i[t_i/time]]y\dot{\sigma}|_{\text{dch}(S_i)}]$. By (C.10) and Lemma 3.4.6 this leads to $[[C_i[t_i/time]]y\dot{\sigma}$ and, since $|\sigma| < \infty$ implies $|\sigma_i| < \infty$, if $|\dot{\sigma}| < \infty$ then $[[q_i[t_i/time]]y\dot{\sigma}$.

Further, $[[time = max(t_1, t_2)]y\dot{\sigma}$ and $[[\bigwedge_{i=1}^{2} \text{no dch}(S_i) \text{ during } [t_i, time]]y\dot{\sigma}$.

Thus $[[\exists t_1, t_2 : time = max(t_1, t_2) \land \bigwedge_{i=1}^{2} C_i[t_i/time] \land \text{no dch}(S_i) \text{ during } [t_i, time]]y\dot{\sigma}$ and if $|\dot{\sigma}| < \infty$ then

$[[\exists t_1, t_2 : time = max(t_1, t_2) \land \bigwedge_{i=1}^{2} q_i[t_i/time] \land \text{no dch}(S_i) \text{ during } [t_i, time]]y\dot{\sigma}$.

Hence, by (C.8) and (C.9), $[[C]y\dot{\sigma}$ and if $|\dot{\sigma}| < \infty$ then $[[q]y\dot{\sigma}$.

**C.2 Completeness of the Proof System in Section 3.4**

In this section we show that the proof system of Section 3.4 is relatively complete. Similar to the notion of precise assertions, which is used to prove the completeness of the proof
system in Section 3.3, we define the notion of a characteristic assertion. Let \( FV(p) \) denote the set of free logical variables occurring in assertion \( p \).

**Definition C.2.1** An assertion \( C \) is characteristic for a program \( S \) with respect to a logical variable \( t_0 \) iff the following points hold:

1. \( \vdash C \vdash \{ \text{time} = t_0 \} \vdash S \vdash \{ C \land \text{time} < \infty \} \), that is, for all \( \gamma, \hat{\delta} \) and \( \sigma \):
   
   if \( \gamma(t_0) = |\hat{\delta}| < \infty \) and \( \sigma \in \mathcal{M}(S) \), then \( [C]^{\gamma\hat{\delta}\sigma} \).

2. For all \( \gamma, \hat{\delta} \) and \( \sigma \): if \( \sigma \) is well-formed, \( dch(\sigma) \subseteq dch(S) \), \( \gamma(t_0) = |\hat{\delta}| < \infty \) and \( [C]^{\gamma\hat{\delta}\sigma} \) then \( \sigma \in \mathcal{M}(S) \).

3. \( dch(C) = dch(S) \) and \( FV(C) = \{ t_0 \} \).

The main idea of the completeness proof is to show first that we can define a characteristic assertion \( C \) for each program \( S \) in our assertion language. Next we prove that we can derive a characteristic specification \( C : \{ \text{time} = t_0 \} \vdash S \vdash \{ C \land \text{time} < \infty \} \) in our proof system.

Then we show that with an arbitrary precondition we can derive any valid commitment and postcondition from the characteristic specification. Although this approach is similar to the completeness proof for the proof system of Section 3.3, the following example illustrates that there is a small complication. Consider

\[ \vdash \text{time} \geq 8 : \{ \text{comm via c at 5} \} \vdash \text{delay 3} \{ \text{time} \geq 8 \} \],

which is valid since \( \text{comm via c at 5} \) implies \( \text{time} \geq 5 \). In order to derive this formula in the proof system we would like to use the following characteristic specification:

\[ \vdash \text{time} = t_0 + 3 : \{ \text{time} = t_0 \} \vdash \text{delay 3} \{ \text{time} = t_0 + 3 \} \],

and derive \( \text{time} \geq 8 \) by \( (\text{comm via c at 5})[t_0/\text{time}] \) and \( \text{time} = t_0 + 3 \). Note, however, that \( \text{comm via c at 5} \land \text{time} = t_0 + 3 \rightarrow \text{time} \geq 8 \) is not valid. The problem is that we cannot use the fact that \( \text{comm via c at 5} \) implies \( \text{time} \geq 5 \), and thus after the substitution \( [t_0/\text{time}] \) implies \( t_0 \geq 5 \). Therefore the precondition is transformed such that this implicit information is made explicit. This transformation is given by the following operation on assertions. For an assertion \( p \) we define \( p^+ \) inductively as follows:

\[
p^+ \equiv \begin{cases} 
\text{comm via c at exp} \land \text{exp} \prec \text{time} \quad & \text{if } p \equiv \text{comm via c at exp} \\
\text{wait to c}! \text{ at exp} \land \text{exp} \prec \text{time} \quad & \text{if } p \equiv \text{wait to c}! \text{ at exp} \\
\text{wait to c}? \text{ at exp} \land \text{exp} \prec \text{time} \quad & \text{if } p \equiv \text{wait to c}? \text{ at exp} \\
\neg p^+ \quad & \text{if } p \equiv \neg p \\
\neg p_1 \lor \neg p_2 \quad & \text{if } p \equiv p_1 \lor p_2 \\
\exists t : p^+ \quad & \text{if } p \equiv \exists t : p
\end{cases}
\]

Then we have the following lemma:

**Lemma C.2.2** For all assertions \( p \), \( \vdash p \leftrightarrow p^+ \).
Now we can give the completeness proof, assuming the following lemma which will be proved later.

**Lemma C.2.3** For any program $S$ and any logical variable $t_0$ there exists an assertion $C$ such that

1. $C$ is characteristic for $S$ w.r.t. $t_0$, and
2. $\vdash C : \{\text{time} = t_0\} S \{C \land \text{time} < \infty\}$.

**Proof:** See Section C.2.1.

**Lemma C.2.4** If $\dot{C}$ is characteristic for $S$ w.r.t. $t_0$ and $\models C : \{p\} S \{q\}$, then, with $\text{cset} = \text{dch}(C) - \text{dch}(\dot{C})$;

1. $\models \dot{C} \land p^+[t_0/time] \land t_0 < \infty \land \text{no cset during } [t_0, \text{time}) \land \text{WellForm}_{\text{dch}(\dot{C})} \rightarrow C$
2. $\models \dot{C} \land p^+[t_0/time] \land t_0 < \infty \land \text{no cset during } [t_0, \text{time}) \land \text{WellForm}_{\text{dch}(\dot{C})} \land \text{time} < \infty \rightarrow q$

**Proof:** Assume $\dot{C}$ is characteristic for $S$ w.r.t. $t_0$, $\models C : \{p\} S \{q\}$, and $\text{cset} = \text{dch}(C) - \text{dch}(\dot{C})$. Suppose 

$\left[\dot{C} \land p^+[t_0/time] \land t_0 < \infty \land \text{no cset during } [t_0, \text{time}) \land \text{WellForm}_{\text{dch}(\dot{C})}\right] \gamma \sigma$.

Define $\dot{\sigma} = \sigma \downarrow \gamma(t_0)$ and $\sigma_1 = \sigma \uparrow \gamma(t_0)$. Then $\sigma = \dot{\sigma} \sigma_1$ and, since $\left[t_0 < \infty\right] \gamma \sigma$, $|\dot{\sigma}| < \infty$.

From $\left[\text{WellForm}_{\text{dch}(\dot{C})}\right] \gamma \sigma$, we obtain by Lemma 3.4.6, $\left[\text{WellForm}_{\text{dch}(\dot{C})}\right] \gamma \sigma_{\text{dch}(\dot{C})}$. Thus $\left[\sigma_{\text{dch}(\dot{C})}\right]$ is well-formed. Hence, using $\sigma = \dot{\sigma} \sigma_1$ and $|\dot{\sigma}| < \infty$, $\left[\sigma_1_{\text{dch}(\dot{C})}\right]$ is well-formed. Furthermore, $\left[\dot{C}\right] \gamma \dot{\sigma} \sigma_1$ leads by Lemma 3.4.6 to $\left[\dot{C}\right] \gamma \dot{\sigma} \sigma_1_{\text{dch}(\dot{C})}$, and thus $\left[\dot{C}\right] \gamma \dot{\sigma} \sigma_1_{\text{dch}(\dot{C})}$. Again using Lemma 3.4.6 this leads to $\left[\dot{C}\right] \gamma \dot{\sigma} \sigma_1_{\text{dch}(\dot{C})}$.

Observe that $\text{dch}(\left[\sigma_1_{\text{dch}(\dot{C})}\right]) \subseteq \text{dch}(\dot{C}) = \text{dch}(S)$. Since $\dot{C}$ is characteristic for $S$ w.r.t. $t_0$, this implies $\left[\sigma_1_{\text{dch}(\dot{C})}\right] \in \mathcal{M}(S)$.

From $\left[\text{no cset during } [t_0, \text{time})\right] \gamma \dot{\sigma} \sigma_1$ we can derive $\left[\sigma_1_{\text{dch}(\dot{C})}\right] = \left[\sigma_1_{\text{dch}(\dot{C})}\right]_{\text{dch}(\dot{C})}$ (recall that $\text{cset} = \text{dch}(C) - \text{dch}(\dot{C})$). Hence $\left[\sigma_1_{\text{dch}(\dot{C})}\right] \in \mathcal{M}(S)$. From $\left[p^+[t_0/time]\right] \gamma \dot{\sigma} \sigma_1$ we obtain $\left[p^+[\dot{\sigma}/\text{time}\right] \gamma \dot{\sigma} \sigma_1$.

By Lemma 3.4.3, $\left[p^+\right] \gamma (\dot{\sigma} \sigma_1) \downarrow \dot{\sigma}$. Thus, using $\models p \leftrightarrow p^+$, $\left[p\right] \gamma \dot{\sigma}$. Since $\dot{\sigma}$ is well-formed and $|\dot{\sigma}| < \infty$, we can now use $\models C : \{p\} S \{q\}$ to derive that

1. $\left[C\right] \gamma \dot{\sigma} \sigma_1_{\text{dch}(\dot{C})}$ and
2. if $|\sigma_1| < \infty$ then $\left[q\right] \gamma \dot{\sigma} \sigma_1_{\text{dch}(\dot{C})}$.

Hence, by Lemma 3.4.6,

1. $\left[C\right] \gamma \dot{\sigma} \sigma_1$, that is, $\left[C\right] \gamma \sigma$, and
2. if $\left[\text{time} < \infty\right] \gamma \sigma$ then $|\sigma_1| < \infty$, and thus $\left[q\right] \gamma \sigma$.

Given the two lemmas above, we can now prove (relative) completeness:
Theorem C.2.5 If $\models C : \{p\} S \{q\}$ then $\vdash C : \{p\} S \{q\}$.

Proof: Assume $\models C : \{p\} S \{q\}$. Let $t_0$ be fresh, that is, $t_0 \notin FV(C, p, q)$.

By Lemma C.2.3, there exists an assertion $\hat{C}$ such that $\hat{C}$ is characteristic for $S$ w.r.t. $t_0$, and $\vdash \hat{C} : \{\text{time} = t_0\} S \{\hat{C} \land \text{time} < \infty\}$.

Let $cset = dch(C) - dch(\hat{C})$. Then $cset = dch(C) - dch(S)$, since $\hat{C}$ is characteristic for $S$. This implies that $cset \cap dch(S) = \emptyset$. Since $\text{time}$ does not occur in $p[t_0/time]$, we can then derive by the Initial Invariance Axiom, the Channel Invariance Axiom and the Conjunction Rule,

$\vdash p[t_0/time] \land t_0 < \infty \land \neg \text{cset during [t}_0, \text{time})$:

$\{p[t_0/time] \land \text{time} = t_0 < \infty\} S \{p[t_0/time] \land t_0 < \infty \land \neg \text{cset during [t}_0, \text{time})\}.$

By the Well-formedness Axiom we can derive

$\vdash \text{WellForm}_{dch(\hat{C})} : \{\text{true}\} S \{\text{WellForm}_{dch(\hat{C})}\}.$

The Conjunction Rule (twice) then leads to

$\vdash \hat{C} \land p[t_0/time] \land t_0 < \infty \land \text{WellForm}_{dch(\hat{C})} \land \neg \text{cset during [t}_0, \text{time})$:

$\{\hat{C} \land p[t_0/time] \land t_0 < \infty \land \text{WellForm}_{dch(\hat{C})} \land \neg \text{cset during [t}_0, \text{time}) \land \text{time} < \infty\}.$

Using Lemma C.2.2 and the Consequence Rule we obtain

$\vdash \hat{C} \land p^+[t_0/time] \land t_0 < \infty \land \text{WellForm}_{dch(\hat{C})} \land \neg \text{cset during [t}_0, \text{time})$:

$\{\hat{C} \land p^+[t_0/time] \land t_0 < \infty \land \text{WellForm}_{dch(\hat{C})} \land \neg \text{cset during [t}_0, \text{time}) \land \text{time} < \infty\}.$

By Lemma C.2.4, the relative completeness assumption, and the Consequence Rule this leads to

$\vdash C : \{p[t_0/time] \land \text{time} = t_0\} S \{q\}.$

Since $t_0 \notin FV(C, p, q)$, we can derive by the Quantification Rule

$\vdash C : \{\exists t_0 : p[t_0/time] \land \text{time} = t_0\} S \{q\}.$

Finally, since $\models p \rightarrow \exists t_0 : p[t_0/time] \land \text{time} = t_0$, the relative completeness assumption and the Consequence Rule lead to

$\vdash C : \{p\} S \{q\}. \quad \square$

C.2.1 Proof of Lemma C.2.3

First a few definitions:

- $I_{t_0} = \{(\gamma, \hat{\sigma}) \mid \hat{\sigma} \text{ is a well-formed model and } \gamma(t_0) = |\hat{\sigma}| < \infty\}$
- $\text{After}_{t_0}(p)$ iff for all $\sigma$, if $[p]_{\gamma_0 \sigma_0 \sigma}$, for some $(\gamma_0, \sigma_0) \in I_{t_0}$, then for all $(\gamma, \hat{\sigma}) \in I_{t_0}$:
  $[p]_{\gamma \hat{\sigma} \sigma}$. (The predicate $\text{After}_{t_0}(p)$ expresses that assertion $p$ does not refer to point of time before $t_0$.)
From the definition of characteristic assertions we obtain

**Lemma C.2.6** $C$ is characteristic for $S$ w.r.t. $t_0$ iff

$$dch(C) = dch(S), \ FV(C) = \{t_0\} \text{ and } [C]_{t_0} = \mathcal{M}(S).$$

The following lemma expresses that, under certain conditions, the $\text{Concat}$ operator corresponds to sequential composition.

**Lemma C.2.7** If $After_{t_0}(p_1), After_{t_0}(p_2)$, and $dch(p_1) = dch(p_2)$ then

$$[\text{Concat}_{t_0}(p_1, p_2)]_{t_0} = \text{SEQ}([p_1]_{t_0}, [p_2]_{t_0}).$$

**Proof:** Assume $After_{t_0}(p_1), After_{t_0}(p_2)$, and $dch(p_1) = dch(p_2)$. Then

$\sigma \in [\text{Concat}_{t_0}(p_1, p_2)]_{t_0}$

iff

$$\sigma \in [\exists t, t_0 \leq t \leq \text{time} : p_1[t/time] \land (t < \infty \rightarrow p_2[t/t_0])]_{t_0}$$

iff

$$dch(\sigma) \subseteq dch(p_1 \land p_2), \ \sigma \text{ well-formed, and for all } (\gamma, \hat{\sigma}) \in I_{t_0}:$$

$$[\exists t, t_0 \leq t \leq \text{time} : p_1[t/time] \land (t < \infty \rightarrow p_2[t/t_0])]_{t_0} \gamma \hat{\sigma}$$

iff

$$dch(\sigma) \subseteq dch(p_1) = dch(p_2), \ \sigma \text{ well-formed, and for all } (\gamma, \hat{\sigma}) \in I_{t_0} \text{ there exists a } \tau \in \text{TIME} \cup \{\infty\}, \ \gamma(t_0) \leq \tau \leq |\hat{\sigma}|, \ [p_1[\tau/time]]_{t_0} \gamma \hat{\sigma}, \ \text{and if } \tau < \infty \text{ then } [p_2[\tau/t_0]]_{t_0} \gamma \hat{\sigma}$$

iff

$$dch(\sigma) \subseteq dch(p_1) = dch(p_2), \ \sigma \text{ well-formed, and for all } (\gamma, \hat{\sigma}) \in I_{t_0} \text{ there exists a } \tau \in \text{TIME} \cup \{\infty\}, \ |\hat{\sigma}| \leq \tau \leq |\hat{\sigma}|, \ [p_1[\tau/time]]_{t_0} \gamma \hat{\sigma}, \ \text{and if } \tau < \infty \text{ then } [p_2[\tau/t_0]]_{t_0} \gamma \hat{\sigma}$$

iff (using Lemma 3.4.3)

$$dch(\sigma) \subseteq dch(p_1) = dch(p_2), \ \sigma \text{ well-formed, and for all } (\gamma, \hat{\sigma}) \in I_{t_0} \text{ there exists a } \tau_0 \in \text{TIME} \cup \{\infty\}, \ 0 \leq \tau_0 \leq |\sigma|, \ [p_1[|\hat{\sigma}| + \tau_0/time]]_{t_0} \gamma \hat{\sigma}, \ \text{and if } \tau_0 < \infty \text{ then } [p_2[|\hat{\sigma}| + \tau_0/t_0]]_{t_0} \gamma \hat{\sigma}$$

iff (using Lemma 3.4.3)
if $\tau_0 < \infty$ then $\llbracket p_2 \rrbracket (\gamma : t_0 \mapsto |\check{\sigma}(\sigma \downarrow \tau_0)|) \check{\sigma}(\sigma \downarrow \tau)(\sigma \uparrow \tau)$

iff (using $After_{t_0}(p_1)$ and $After_{t_0}(p_2)$)

$dch(\sigma) \subseteq dch(p_1) = dch(p_2)$, $\sigma$ well-formed, and there exists a $\tau_0 \in TIME \cup \{\infty\}$, $\tau_0 \leq |\sigma|$, such that for all $(\gamma, \check{\sigma}) \in I_{t_0}$, $\llbracket p_1 \rrbracket \gamma \check{\sigma}(\sigma \downarrow \tau_0)$, and

if $\tau_0 < \infty$ then for all $(\gamma, \check{\sigma}(\sigma \downarrow \tau_0)) \in I_{t_0}$, $\llbracket p_2 \rrbracket \gamma \check{\sigma}(\sigma \downarrow \tau)(\sigma \uparrow \tau)$

iff (using the assumption $dch(p_1) = dch(p_2)$)

$\sigma$ well-formed, and there exists a $\tau_0 \in TIME \cup \{\infty\}$, $\tau_0 \leq |\sigma|$, such that for all $(\gamma, \check{\sigma}) \in I_{t_0}$, 

$(\sigma \downarrow \tau_0) \in \llbracket p_1 \rrbracket_{t_0}$, and if $\tau_0 < \infty$ then $(\sigma \uparrow \tau) \in \llbracket p_2 \rrbracket_{t_0}$

iff

there exists a $\tau_0 \in TIME \cup \{\infty\}$ such that $(\sigma \downarrow \tau_0) \in \llbracket p_1 \rrbracket_{t_0}$ and $(\sigma \uparrow \tau_0) \in \llbracket p_2 \rrbracket_{t_0}$

iff

$\sigma \in SEQ(\llbracket p_1 \rrbracket_{t_0}, \llbracket p_2 \rrbracket_{t_0})$.

We show that Lemma C.2.3 holds by proving the following lemma.

**Lemma C.2.8** For every program $S$ and logical variable $t_0$ there exists an assertion $C$ such that

1. $\llbracket C \rrbracket_{t_0} \subseteq M(S)$,
2. $\vdash C : \{time = t_0\} S \{C \land time < \infty\},$
3. $dch(C) = dch(S)$, $FV(C) = \{t_0\}$, and
4. $After_{t_0}(C)$, $C \rightarrow time \geq t_0$.

First observe that this lemma implies Lemma C.2.3, since the second point and soundness of the proof system implies $\vdash C : \{time = t_0\} S \{C \land time < \infty\}$. Thus for all $\check{\sigma}$ and $\gamma$ with $|\check{\sigma}| < \infty$, $\check{\gamma}$ well-formed and $\llbracket time = t_0\rrbracket_\gamma \check{\sigma}$, and for all $\sigma \in M(S)$, we have $\llbracket C \rrbracket_\gamma \check{\sigma}$. Hence, $\llbracket C \rrbracket_\gamma \check{\sigma}$ for all $(\gamma, \check{\sigma}) \in I_{t_0}$ and $\sigma \in M(S)$. Thus $\sigma \in M(S)$ implies $\sigma \in \llbracket C \rrbracket_{t_0}$. This proves $M(S) \subseteq \llbracket C \rrbracket_{t_0}$, and thus, by the first point, $\llbracket C \rrbracket_{t_0} = M(S)$. Together with the third point we obtain, by Lemma C.2.6, that $C$ is characteristic for $S$ w.r.t. $t_0$.

We prove Lemma C.2.8 by induction on the structure of $S$, and the last point has been added for this inductive proof. The most difficult case in the proof is the iteration construct. Therefore we first prove the lemma for $S \equiv \ast G$. This requires a number of definitions and lemmas.

Define $M(G^k)$ for $k \geq 1$ by $M(G^1) = M(G)$ and $M(G^{k+1}) = SEQ(M(G), M(G^k))$.

**Lemma C.2.9** $M((G^k) =$

$\{\sigma |$ there exist $\sigma_0, \ldots, \sigma_{k-1}$ such that $\sigma = \sigma_0 \cdots \sigma_{k-1}$ and $\sigma_i \in M(G)$ for $i < k\}$

**Proof:** By induction on $k$. For $k = 1$ trivial. $M(G^{k+1}) = SEQ(M(G), M(G^k)) =$

$\{\sigma |$ there exist $\sigma^1$ and $\sigma^2$ such that $\sigma = \sigma^1 \sigma^2$, $\sigma^1 \in M(G)$, and $\sigma^2 \in M(G)\}$.

By the induction hypothesis we obtain
{σ} there exist σ^1 and σ^2 such that σ = σ^1σ^2, σ^1 ∈ \mathcal{M}(G), and there exist σ_0,...,σ_{k-1} such that σ^2 = σ_0⋯σ_{k-1} and σ_i ∈ \mathcal{M}(G) for i < k} = 

{σ} there exist σ^1 and σ_0,...,σ_{k-1} such that σ^1 ∈ \mathcal{M}(G), σ_i ∈ \mathcal{M}(G), for i < k, and σ = σ^1σ_0⋯σ_{k-1} \square

Next we define an assertion p(n) such that \mathcal{M}(G^k) = [p[k/n]]_{t_0}, for k ≥ 1.

Let p(k) denote p[k/n]. Assume C_0 is characteristic for G with respect to t_0.

Then p should be such that

p(k) ↔ \exists t_1,...,t_{k-1}: \forall i ∈ \mathbb{N}, i < k - 1 : t_i ≤ t_{i+1} ≤ \text{time}∧
(t_i < \infty \to C_0[t_i/t_0, t_{i+1}/\text{time}])∧ (t_{k-1} < \infty \to C_0[t_{k-1}/t_0]).

First we show that such an assertion p(n) can be defined in our assertion language. Note that we cannot use the expression above for p(k) with k replaced by variable n since then we obtain a a second-order formula p(n) = \exists t_1,...,t_{n-1} which is not a formula of our assertion language. We give, however, an equivalent expression in our assertion language by using the idea that finite sequences can be coded in first-order arithmetic. Therefore we need some predicates which are used to code and decode finite sequences of numbers from \text{TIME}∪ \{∞\}. In our assertion language we can define the predicates LIST(x), LEN(x, y) and PROJ(x, y, z). These assertions are defined such that

- \text{LIST}(x) holds iff x codes a finite sequence of elements from \text{TIME}∪ \{∞\}
- \text{LEN}(x, y) holds iff if \text{LIST}(x) then x codes a sequence of length y of elements from \text{TIME}∪ \{∞\}
- \text{PROJ}(x, y, z) holds iff if \text{LIST}(x) and \text{LEN}(x, v), for some v > y, then z is the (i + 1)^{th} number in the sequence coded by x. (Note that y ∈ \mathbb{N}.)

Then we can define p as follows:

p(n) = n ∈ \mathbb{N} ∧ n > 0 ∧ t_0 < \infty ∧ \exists t : \text{LIST}(t) ∧ \text{LEN}(t, n) ∧ \text{PROJ}(t, 0, t_0)∧
(\forall i ∈ \mathbb{N}, i < n - 1 \forall t_1, t_2 : \text{PROJ}(t, i, t_1) ∧ \text{PROJ}(t, i + 1, t_2) →
 t_1 ≤ t_2 ≤ \text{time} ∧ (t_1 < \infty \to C_0[t_1/t_0, t_2/\text{time}]))∧
(\forall t_1 : \text{PROJ}(t, n - 1, t_1) ∧ t_1 < \infty \to C_0[t_1/t_0])

Note that p(0) ↔ \text{false}.

Lemma C.2.10

1. \ p(1) ↔ (t_0 < \infty ∧ C_0)
2. \ p(k + 1) ↔ [t_0 < \infty ∧ \text{Concat}_{t_0}(C_0, p(k))], for all k ∈ \mathbb{N}, k ≥ 1.
3. \ If C_0 → \text{time} ≥ t_0 then p(k + 1) ↔ \text{Concat}_{t_0}(p(k), C_0), for all k ∈ \mathbb{N}, k ≥ 1.

Proof:
1. \( p(1) \leftrightarrow t_0 < \infty \land \exists t : \text{LIST}(t) \land \text{LEN}(t, 1) \land \text{PROJ}(t, 0, t_0) \land (\forall t_1 : \text{PROJ}(t, 0, t_1) \land t_1 < \infty \rightarrow C_0[t_1/t_0]) \)
   \[ \leftrightarrow (t_0 < \infty \land C_0) \]

2. \( p(k + 1) \leftrightarrow t_0 < \infty \land \exists t : \text{LIST}(t) \land \text{LEN}(t, k + 1) \land \text{PROJ}(t, 0, t_0) \land (\forall i \in \mathbb{N}, i < k \forall t_1, t_2 : \text{PROJ}(t, i, t_1) \land \text{PROJ}(t, i + 1, t_2) \rightarrow t_1 \leq t_2 \leq \text{time} \land (t_1 < \infty \rightarrow C_0[t_1/t_0, t_2/\text{time}]))) \land (\forall t_1 : \text{PROJ}(t, k, t_1) \land t_1 < \infty \rightarrow C_0[t_1/t_0]) \)
   \[ \leftrightarrow (t_0 < \infty \land \exists t_3 : t_0 \leq t_3 \leq \text{time} \land (t_0 < \infty \rightarrow p(k)[t_3/t_0]) \]
   \[ \leftrightarrow [t_0 < \infty \land \text{Concat}_{t_0}(C_0, p(k))] \]

3. \( p(k + 1) \leftrightarrow t_0 < \infty \land \exists t : \text{LIST}(t) \land \text{LEN}(t, k + 1) \land \text{PROJ}(t, 0, t_0) \land (\forall i \in \mathbb{N}, i < k - 1 \forall t_1, t_2 : \text{PROJ}(t, i, t_1) \land \text{PROJ}(t, i + 1, t_2) \rightarrow t_1 \leq t_2 \leq \text{time} \land (t_1 < \infty \rightarrow C_0[t_1/t_0, t_2/\text{time}]))) \land (\forall t_1 : \text{PROJ}(t, k, t_1) \land t_1 < \infty \rightarrow C_0[t_1/t_0]) \)
   \[ \leftrightarrow \exists t_3 : t_0 \leq t_3 \leq \text{time} \land t_0 < \infty \land \exists t : \text{LIST}(t) \land \text{LEN}(t, k) \land \text{PROJ}(t, 0, t_0) \land (\forall i \in \mathbb{N}, i < k - 1 \forall t_1, t_2 : \text{PROJ}(t, i, t_1) \land \text{PROJ}(t, i + 1, t_2) \rightarrow t_1 \leq t_2 \leq \text{time} \land (t_1 < \infty \rightarrow C_0[t_1/t_0, t_2/\text{time}]))) \land (t_3 < \infty \rightarrow C_0[t_3/t_0]) \)
   \[ (\text{use that } C_0[t_1/t_0, t_3/\text{time}] \text{ implies } t_0 \leq t_3) \]
   \[ \leftrightarrow \exists t_3, t_0 \leq t_3 \leq \text{time} : p(k)[t_3/\text{time}] \land (t_3 < \infty \rightarrow C_0[t_3/t_0]) \]
   \[ \leftrightarrow \text{Concat}_{t_0}(p(k), C_0) \]

Then we can easily prove the following lemma:

**Lemma C.2.11** If \( \text{After}_{t_0}(C_0) \), \( \text{After}_{t_0}(p(k)) \), and \( \mathcal{M}(G) = [C_0]_{t_0} \) then, for \( k \geq 1 \), \( \mathcal{M}(G^k) = [p(k)]_{t_0} \).

**Proof:** By induction on \( k \):
- \( \mathcal{M}(G) = [C_0]_{t_0} = [t_0 < \infty \land C_0]_{t_0} = [p(1)]_{t_0} \), using Lemma C.2.10.
- \( \mathcal{M}(G^{k+1}) = \text{SEQ}(\mathcal{M}(G), \mathcal{M}(G^k)) = \text{SEQ}([C_0]_{t_0}, [p(k)]_{t_0}) = [\text{Concat}_{t_0}(C_0, p(k))]_{t_0} \), by using Lemma C.2.10. Finally, observe that \( [\text{Concat}_{t_0}(C_0, p(k))]_{t_0} = [t_0 < \infty \land \text{Concat}_{t_0}(C_0, p(k))]_{t_0} = [p(k + 1)]_{t_0} \). \( \square \)
The following lemma asserts that if a model $\sigma$ coincides in every finite interval of time with some model from the semantics of a statement, then $\sigma$ must be an element of the semantics. Informally, it expresses that we have no fairness constraints or unbounded non-determinism in our semantic definition. We do not exclude an infinite model from the semantics of a statement if all finite approximations of this model are present in the semantics.

**Lemma C.2.12** $\sigma \in \mathcal{M}(S)$ iff for all $\tau \in \text{TIME}$, $\sigma \downarrow \tau \in \mathcal{M}(S) \downarrow \tau$.

**Proof:** First define for a set of models $\Sigma$ and a model $\sigma$ a predicate $\text{Approx}(\Sigma, \sigma)$ which is true if there exists a sequence $\sigma_{i_0}, \sigma_{i_1}, \ldots$ such that $i_0 < i_1 < \cdots$, $i_k \in \mathbb{N}$, $\sigma_{i_k} \in \mathcal{M}(S)$ and $\sigma \downarrow i_k = \sigma_{i_k} \downarrow i_k$, for all $k \in \mathbb{N}$. Then we prove the following stronger version of the lemma: If $\text{Approx}(\mathcal{M}(S), \sigma)$ then $\sigma \in \mathcal{M}(S)$. This can be proved by induction of the structure of $S$.

**Lemma C.2.13** $\sigma \in \mathcal{M}(\ast G)$ iff $|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$ there exists a $\tau_2 > \tau_1$, $\tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that $\sigma \downarrow \tau_2 \in \mathcal{M}(G^k)$.

**Proof:**

(if) Assume $|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$ there exists a $\tau_2 > \tau_1$, $\tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that $\sigma \downarrow \tau_2 \in \mathcal{M}(G^k)$. Thus for all $\tau_1 \in \text{TIME}$ there exists a $k \geq 1$ such that $\sigma \downarrow \tau_1 \in \mathcal{M}(G^k) \downarrow \tau_1$, since $\tau_1 < \tau_2$. Then, by Lemma C.2.9, there exist $\sigma_0, \ldots, \sigma_{k-1}$ with $\sigma_i \in \mathcal{M}(G)$ for $i < k$ such that $\sigma \downarrow \tau_1 = (\sigma_0 \cdots \sigma_{k-1}) \downarrow \tau_1$. Define, for $i \geq k$, $\sigma_i = \sigma_{k-1}$. Then there exist $\sigma_0, \sigma_1, \ldots$ with $\sigma_i \in \mathcal{M}(G)$ for $i \in \mathbb{N}$ such that $\sigma \downarrow \tau_1 = (\sigma_0 \sigma_1 \cdots) \downarrow \tau_1$. Using Corollary 3.2.10, we obtain, for all $\tau_1 \in \text{TIME}$, $\sigma \downarrow \tau_1 \in \mathcal{M}(\ast G) \downarrow \tau_1$. Hence, by Lemma C.2.12, $\sigma \in \mathcal{M}(\ast G)$.

(only if) Assume $\sigma \in \mathcal{M}(\ast G)$. Then, either

1. there exist $n \geq 0$ and $\sigma_0, \ldots, \sigma_n$ such that $\sigma = \sigma_0 \cdots \sigma_n$, $\sigma_i \in \mathcal{M}(G)$ for $i \leq n$, $|\sigma_i| < \infty$ for $i < n$ and $|\sigma_n| = \infty$, or
2. $|\sigma| = \infty$ and there exist $\sigma_0, \sigma_1, \ldots$ such that $\sigma = \sigma_0 \sigma_1 \cdots$, $\sigma_i \in \mathcal{M}(G)$, and $|\sigma_i| < \infty$, for all $i \in \mathbb{N}$.

Then

1. either $\sigma \in \mathcal{M}(G^{n+1})$ and $|\sigma| = \infty$. Thus for all $\tau_1 \in \text{TIME}$ take $\tau_2 = \infty$ and $k = n + 1$. Then $\sigma \downarrow \tau_2 \in \mathcal{M}(G^k)$.
2. or $|\sigma| = \infty$ and there exist $\sigma_0, \sigma_1, \ldots$ such that $\sigma = \sigma_0 \sigma_1 \cdots$, $\sigma_i \in \mathcal{M}(G)$, for all $i \in \mathbb{N}$. Observe that there exists a constant $K > 0$ such that $\sigma \in \mathcal{M}(G)$ implies $|\sigma| > K$. Hence $|\sigma_i| > K$, for all $i \in \mathbb{N}$. Consider $\tau_1 \in \text{TIME}$. Let $k \geq 1$ be such that $k \times K > \tau_1$, and let $\tau_2 = |\sigma_0 \cdots \sigma_{k-1}|$. Then $\tau_2 > k \times K > \tau$ and $\sigma \downarrow \tau_2 = (\sigma_0 \sigma_1 \cdots) \downarrow \tau_2 = (\sigma_0 \cdots \sigma_{k-1}) \downarrow \tau_2$. Since $\sigma_0 \cdots \sigma_{k-1} \in \mathcal{M}(G^k)$, this leads to $\sigma \downarrow \tau_2 \in \mathcal{M}(G^k)$. □
Now we prove the main lemma of this section.

Lemma C.2.14 If there exists an assertion $C_0$ such that

1. $[C_0]_{t_0} \subseteq \mathcal{M}(G)$,
2. $\vdash C_0 : \{\text{time} = t_0\} G \{C_0 \land \text{time} < \infty\}$,
3. $dch(C_0) = dch(G)$, $FV(C_0) = \{t_0\}$, and
4. $\text{After}_{t_0}(C_0)$, $C_0 \rightarrow \text{time} \geq t_0$.

then there exists an assertion $C$ such that

1. $[C]_{t_0} \subseteq \mathcal{M}(\ast G)$,
2. $\vdash C : \{\text{time} = t_0\} \ast G \{C \land \text{time} < \infty\}$,
3. $dch(C) = dch(\ast G)$, $FV(C) = \{t_0\}$, and
4. $\text{After}_{t_0}(C)$, $C \rightarrow \text{time} \geq t_0$.

Proof: Define $C \equiv \forall t_1 < \infty \exists t_2 > t_1 \exists n : p[t_2/\text{time}] \land \text{time} = \infty$.

Proof of 1.
From the first and the second assumption we obtain $[C_0]_{t_0} = \mathcal{M}(G)$.

We prove $[C]_{t_0} \subseteq \mathcal{M}(\ast G)$ as follows. Assume $\sigma \in [C]_{t_0}$. Then $|\sigma| = \infty$ and

$\sigma \in \llbracket \forall t_1 < \infty \exists t_2 > t_1 \exists n : p[t_2/\text{time}] \rrbracket_{t_0}$. Hence, $\text{dch}(\sigma) \subseteq \text{dch}(C)$, $\sigma$ well-formed,

$|\sigma| = \infty$ and for all $(\gamma, \hat{\sigma}) \in I_{t_0}$, for all $\tau_1 \in \text{TIME}$, there exists a $\tau_2 > \tau_1, \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k$ such that $\llbracket p[k/n, \tau_2/\text{time}] \rrbracket \gamma \hat{\sigma}$. Since $p(n)$ contains $n \in \mathbb{N} \land n > 0$ we obtain $k \in \mathbb{N}, k \geq 1$. Furthermore, note that $\text{After}_{t_0}(C_0)$ implies $\text{After}_{t_0}(p[k/n, \tau_2/\text{time}])$, and that $dch(C) = dch(p)$. This leads to

$dch(\sigma) \subseteq dch(p)$, $\sigma$ well-formed, $|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$, there exists a $\tau_2 > \tau_1, \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that, for all $(\gamma, \hat{\sigma}) \in I_{t_0}, \llbracket p[k/n, \tau_2/\text{time}] \rrbracket \gamma \hat{\sigma}$.

Since $|\hat{\sigma}| < \infty$, this implies

$dch(\sigma) \subseteq dch(p)$, $\sigma$ well-formed, $|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$, there exists a $\tau_2 > (\tau_1 + |\hat{\sigma}|), \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that, for all $(\gamma, \hat{\sigma}) \in I_{t_0}, \llbracket p[k/n, \tau_2/\text{time}] \rrbracket \gamma \hat{\sigma}$.

Then, using Lemma 3.4.3,

$dch(\sigma) \subseteq dch(p)$, $\sigma$ well-formed, $|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$, there exists a $\tau_2 > (\tau_1 + |\hat{\sigma}|), \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that, for all $(\gamma, \hat{\sigma}) \in I_{t_0}, \llbracket p[k/n] \rrbracket \gamma \hat{\sigma}(\sigma \downarrow (\tau_2 - |\hat{\sigma}|)).$

Hence

$|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$, there exists a $\tau_2 > \tau_1, \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that, $(\sigma \downarrow (\tau_2 - |\hat{\sigma}|)) \in \llbracket p[k/n] \rrbracket_{t_0}$.

Thus

$|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$ there exists a $\tau_2 > \tau_1, \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that $\sigma \downarrow \tau_2 \in \llbracket p(k) \rrbracket_{t_0}$. 

From Lemma C.2.11, using $[C_0]_t = \mathcal{M}(G)$, $|\sigma| = \infty$ and for all $\tau_1 \in \text{TIME}$ there exists a $\tau_2 > \tau_1, \tau_2 \in \text{TIME} \cup \{\infty\}$ and $k \geq 1$ such that $\sigma \downarrow \tau_2 \in \mathcal{M}(G^k)$. Now Lemma C.2.13 leads to $\sigma \in \mathcal{M}(\pi G)$.

**Proof of 2.**

From our assumption:

$\vdash C_0 : \{\text{time} = t_0\} G \{C_0 \land \text{time} < \infty\}.$

Since $C_0 \rightarrow \text{time} \geq t_0$, the Consequence Rule leads to

$\vdash C_0 \land \text{time} \geq t_0 : \{\text{time} = t_0\} G \{C_0 \land \text{time} \leq \text{time} < \infty\}.$

By the Substitution Rule,

$\vdash C_0[t/t_0] \land \text{time} \geq t : \{\text{time} = t\} G \{C_0[t/t_0] \land \text{time} \leq \text{time} < \infty\} \quad (C.12)$

Define $\check{C} \equiv (\exists n : p) \lor (\text{time} = t_0 < \infty)$. The Initial Invariance Axiom leads to

$\vdash \check{C}[t/time] \land t < \infty : \{\check{C}[t/time] \land t < \infty\} G \{\check{C}[t/time] \land t < \infty\}.$

Since $C_0 \rightarrow \text{time} \geq t_0$, we obtain $p \rightarrow \text{time} \geq t_0$, and hence $\check{C} \rightarrow \text{time} \geq t_0$.

Thus $\check{C}[t/time] \rightarrow t \geq t_0$. By the Consequence Rule

$\vdash t_0 \leq t < \infty \land \check{C}[t/time] : \{t < \infty \land \check{C}[t/time]\} G \{t_0 \leq t < \infty \land \check{C}[t/time]\} \quad (C.13)$

Then (C.12) and (C.13) lead, by the Conjunction Rule to

$\vdash t_0 \leq t \leq \text{time} \land \check{C}[t/time] \land t < \infty \land C_0[t/t_0] :$

\[
\{\text{time} = t < \infty \land \check{C}[t/time]\} G
\{t_0 \leq t \leq \text{time} \land \check{C}[t/time] \land t < \infty \land C_0[t/t_0] \land \text{time} < \infty\}.
\]

Observe that, using the definition of $\check{C}$, $t_0 \leq t < \text{time} \land \check{C}[t/time] \land t < \infty \land C_0[t/t_0]$ implies $\exists t, t_0 \leq t < \text{time} \exists n : p[t/time] \land t < \infty \land C_0[t/t_0] \lor$

$\exists t, t_0 \leq t < \text{time} : (\text{time} = t_0 < \infty)[t/time] \land t < \infty \land C_0[t/t_0])$ which implies

$\exists n : \text{Concat}_{t_0}(p, C_0) \lor (\exists t : t = t_0 < \text{time} \land C_0[t/t_0])$ which implies

(use Lemma C.2.10, recall that $C_0 \rightarrow \text{time} \geq t_0$)

$\exists n : p[n + 1/n] \lor (C_0 \land t_0 < \text{time})$ which implies (by Lemma C.2.10)

$\exists n : p$, which implies $\check{C}$.

Further, note that $\text{time} = t \land \check{C}[t/time] \land \text{time} < \infty$ implies $\text{time} = t < \infty \land \check{C}[t/time]$.

Then the Consequence Rule leads to

$\vdash \check{C} : \{\text{time} = t \land \check{C}[t/time]\} G \{\check{C}\}.$

Using the Quantification Rule,

$\vdash \check{C} : \{\exists t : \text{time} = t \land \check{C}[t/time]\} G \{\check{C}\}.$

Since $\check{C} \rightarrow \exists t : \text{time} = t \land \check{C}[t/time]$, the Consequence Rule leads to

$\vdash \check{C} : \{\check{C}\} G \{\check{C}\} \quad (C.14)$

Define $C_{nt} \equiv \forall t_1 \lt \infty \exists t_2 > t_1 \exists n : p[t_2/time]$. Then

$\forall t_1 \lt \infty \exists t_2 > t_1 : \check{C}[t_2/time] \rightarrow C_{nt} \quad (C.15)$
By (C.14) and (C.15), the Iteration Rule leads to
\[ \vdash C_{nt} \land time = \infty : \{ \hat{C} \} \neq G \{ false \}. \]
Since \( time = t_0 \land time < \infty \) implies \( time = t_0 < \infty \), which implies \( \hat{C} \), and
\( false \rightarrow C \land time < \infty \), we obtain by the Consequence Rule
\[ \vdash C_{nt} \land time = \infty : \{ time = t_0 \} \neq G \{ C \land time < \infty \}. \]
Since \( C \equiv C_{nt} \land time = \infty \), this leads to \( \vdash C : \{ time = t_0 \} \neq G \{ C \land time < \infty \}. \)

**Proof of 3.**
Since \( \text{dch}(C) = \text{dch}(p) = \text{dch}(C_0) \) and \( \text{FV}(C) = \text{FV}(p) - \{ n \} = \text{FV}(C_0) \), we obtain by the first assumption \( \text{dch}(C) = \text{dch}(G) = \text{dch}(\star G) \) and \( \text{FV}(C) = \{ t_0 \}. \)

**Proof of 4.**
From \( \text{After}_{t_0}(C_0) \) we obtain \( \text{After}_{t_0}(p(k)) \), and hence \( \text{After}_{t_0}(C) \).
Similarly, \( C \rightarrow time \geq t_0. \)

Finally, we prove Lemma C.2.8. Consider a logical variable \( t_0 \). We prove by induction on the structure of \( S \) that there exists an assertion \( C \) such that

1. \( \| C \|_{t_0} \subseteq \mathcal{M}(S) \),
2. \( \vdash C : \{ time = t_0 \} S \{ C \land time < \infty \} \),
3. \( \text{dch}(C) = \text{dch}(S), \text{FV}(C) = \{ t_0 \} \), and
4. \( \text{After}_{t_0}(C), C \rightarrow time \geq t_0. \)

**Proof:**
For the atomic statements we only give the characteristic assertions. It is easy to see that the assertions below satisfy the requirements of the lemma.

**Skip** For \( \text{skip} \): \( C \equiv time = t_0 \)

**Delay** For \( \text{delay} d \): \( C \equiv time = t_0 + d \)

**Send** For \( e! \):
\( C \equiv (\text{wait to e! during } [t_0, \infty)) \lor (\exists t_1, t_0 \leq t_1 < \infty : \text{wait to e! at } t_0 \text{ until comm at } t_1 \land time = t_1 + K_e) \)

**Receive** For \( e? \):
\( C \equiv (\text{wait to e? during } [t_0, \infty)) \lor (\exists t_1, t_0 \leq t_1 < \infty : \text{wait to e? at } t_0 \text{ until comm at } t_1 \land time = t_1 + K_e) \)

**Sequential Composition**
For \( S_1, S_2 \): assume \( C_i \) is characteristic for \( S_i \) w.r.t. \( t_0 \), for \( i \in \{ 1, 2 \} \).
Define \( \text{cset}_{12} \equiv \text{dch}(S_1) - \text{dch}(S_2), \text{cset}_{21} \equiv \text{dch}(S_2) - \text{dch}(S_1), \)
\( \hat{C}_1 \equiv C_1 \land \text{no cset}_{21} \text{ during } [t_0, time), \) and \( \hat{C}_2 \equiv C_2 \land \text{no cset}_{12} \text{ during } [t_0, time). \)
Then define \( C \equiv \text{Concat}_{t_0} (\hat{C}_1, \hat{C}_2) \). From Lemma C.2.7 we obtain
\( \| C \|_{t_0} = \text{SEQ}(\| \hat{C}_1 \|_{t_0}, \| \hat{C}_2 \|_{t_0}) = \text{SEQ}(\mathcal{M}(S_1), \mathcal{M}(S_2)) = \mathcal{M}(S_1; S_2). \)
Next we show \( \vdash C : \{\text{time} = t_0\} \quad S_1; S_2 \quad \{C \land \text{time} < \infty\}. \)

By the induction hypothesis,

\[ \vdash C_1 : \{\text{time} = t_0\} \quad S_1 \quad \{C_1 \land \text{time} < \infty\}, \]

and

\[ \vdash C_2 : \{\text{time} = t_0\} \quad S_2 \quad \{C_2 \land \text{time} < \infty\}. \]

Since \( C_1 \rightarrow \text{time} \geq t_0 \) and \( C_2 \rightarrow \text{time} \geq t_0 \), we obtain by the Consequence Rule

\[ \vdash C_1 : \{\text{time} = t_0\} \quad S_1 \quad \{C_1 \land t_0 \leq \text{time} < \infty\}, \]

and

\[ \vdash C_2 \land \text{time} \geq t_0 : \{\text{time} = t_0\} \quad S_2 \quad \{C_2 \land t_0 \leq \text{time} < \infty\}. \]

The Channel Invariance Axiom leads to

\[ \vdash \text{no cset}_1 \text{ during } [t_0, \text{time}) : \{\text{time} = t_0\} \quad S_1 \quad \{\text{no cset}_1 \text{ during } [t_0, \text{time})\}, \]

and

\[ \vdash \text{no cset}_2 \text{ during } [t_0, \text{time}) : \{\text{time} = t_0\} \quad S_2 \quad \{\text{no cset}_2 \text{ during } [t_0, \text{time})\}. \]

Thus by the Conjunction Rule,

\[ \vdash \hat{C}_1 : \{\text{time} = t_0\} \quad S_1 \quad \{\hat{C}_1 \land t_0 \leq \text{time} < \infty\}, \]

and

\[ \vdash \hat{C}_2 \land \text{time} \geq t_0 : \{\text{time} = t_0\} \quad S_2 \quad \{\hat{C}_2 \land t_0 \leq \text{time} < \infty\}. \]

By the Substitution Rule the last formula leads to

\[ \vdash \hat{C}_2[t/t_0] \land \text{time} \geq t : \{\text{time} = t\} \quad S_2 \quad \{\hat{C}_2[t/t_0] \land t \leq \text{time} < \infty\}. \]

Then from the Sequential Composition Adaptation Rule we obtain

\[ \vdash (\hat{C}_1 \land \text{time} = \infty) \lor (\exists t : \hat{C}_1[t/time] \land t_0 \leq \text{time} < \infty \land \hat{C}_2[t/t_0] \land \text{time} < \infty) : \]

\[ \{\text{time} = t_0\} \quad S_1; S_2 \quad \exists t : \hat{C}_1[t/time] \land t_0 \leq \text{time} < \infty \land \hat{C}_2[t/t_0] \land \text{time} < \infty. \]

Thus by the Consequence Rule,

\[ \vdash (\hat{C}_1 \land \text{time} = \infty) \lor (\exists t, t_0 \leq \text{time} < \infty : \hat{C}_1[t/time] \land \hat{C}_2[t/t_0]) : \]

\[ \{\text{time} = t_0\} \quad S_1; S_2 \quad \exists t, t_0 \leq \text{time} < \infty : \hat{C}_1[t/time] \land \hat{C}_2[t/t_0] \land \text{time} < \infty. \]

Thus the Consequence Rule leads to

\[ \vdash \text{Concat}_0(\hat{C}_1, \hat{C}_2) : \{\text{time} = t_0\} \quad S_1; S_2 \quad \{\text{Concat}_0(\hat{C}_1, \hat{C}_2) \land \text{time} < \infty\}. \]

**Guarded Command** First consider a guarded command without delay.

Assume \( C_i \) is characteristic for \( S_i \) w.r.t. \( t_0 \), for \( i \in \{1, \ldots, n\} \).

Define \( \text{cset}_i \equiv \text{dch}(S_i) - \text{dch}(c_i ?; \ldots, c_n ?) \), \( \text{cset}_i \equiv \text{dch}(C) - \text{dch}(c_i ?; S_i) \), and

\[ C \equiv (\land \text{wait to c}_i ? \text{ during } [t_0, \infty) \land \text{no cset}_0 \text{ during } [t_0, \infty)) \lor \]

\[ (\exists t_1, t_0 \leq t_1 < \infty : \land \text{wait to c}_i ? \text{ during } [t_0, t_1) \land \text{no cset}_0 \text{ during } [t_0, t_1) \land \]

\[ \land \text{wait to c}_i ? \text{ at } t_1 \land \text{no cset}_0 \text{ during } [t_1, \text{time}) \]

Similarly, for a guarded command with delay.

**Iteration** See Lemma C.2.14.

**Parallel Composition**

For \( S_1 | S_2 \), assume \( C_i \) is characteristic for \( S_i \) w.r.t. \( t_0 \), for \( i = 1, 2 \).

Define \( C \equiv \exists t_1, t_2 : \text{time} = \max(t_1, t_2) \land \land \text{cset}_i \land \text{no dch}(S_i) \text{ during } [t_i, \text{time}) \)

The proof that \( C \) is characteristic for \( S_1 | S_2 \) w.r.t. \( t_0 \) proceeds similar to the proof of a precise specification for \( S_1 | S_2 \) in Appendix B.2. Since \( dch(C_i) = dch(S_i) \), for \( i = 1, 2 \), this assertion can be derived easily. \( \square \)
Appendix D

Soundness of the Proof Systems in Chapter 4

First we introduce a lemma that will be used in the proofs below.

**Lemma D.0.15** For any program $S$ and any $\sigma \in M(S)$:

if $|\sigma| < \infty$ then $\sigma.f\text{inal}(x) = \sigma.i\text{nit}(x)$ for $x \notin u\text{var}(S)$.

**Proof:** This property can be proved easily by induction on the structure of $S$. \hfill \Box

**D.1 Soundness of the Proof System in Section 4.3.2**

We start with a few lemmas:

**Lemma D.1.1** For any expression $e$ from the programming language and any model $\sigma$, $\mathcal{E}(e)(\sigma.\text{init}) = Va\text{l}(e)\sigma$.

**Proof:** Directly from the definitions, by induction on the structure of $e$. \hfill \Box

**Lemma D.1.2** For a boolean expression $b$ from the programming language and any model $\sigma$, $B(b)(\sigma.\text{init})$ iff $\sigma \models b$.

**Proof:** By induction on the structure of $b$.

- $B(e_1 = e_2)(\sigma.\text{init})$ iff $\mathcal{E}(e_1)(\sigma.\text{init}) = \mathcal{E}(e_2)(\sigma.\text{init})$ iff (using Lemma D.1.1)
  $Va\text{l}(e_1)\sigma = Va\text{l}(e_2)\sigma$ iff $\sigma \models e_1 = e_2$

- $B(e_1 < e_2)(\sigma.\text{init})$ iff $\mathcal{E}(e_1)(\sigma.\text{init}) < \mathcal{E}(e_2)(\sigma.\text{init})$ iff (using Lemma D.1.1)
  $Va\text{l}(e_1)\sigma < Va\text{l}(e_2)\sigma$ iff $\sigma \models e_1 < e_2$

- $B(\neg b)(\sigma.\text{init})$ iff not $B(b)(\sigma.\text{init})$ iff (by the induction hypothesis) $\sigma \not\models b$ iff $\sigma \models \neg b$
\[ B(b_1 \lor b_2)(\sigma.init) \text{ iff } B(b_1)(\sigma.init) \text{ or } B(b_2)(\sigma.init) \text{ iff (by the induction hypothesis)} \]
\[ \sigma \models b_1 \text{ or } \sigma \models b_2 \text{ iff } \sigma \models b_1 \lor b_2. \]

Similar to Lemma 3.3.19, we can easily prove the following lemma by induction on the structure of assertion \( \varphi \).

**Lemma D.1.3 (Projection)** Consider any cset \( \subseteq DCHAN \) and MTL assertion \( \varphi \).
If \( dch(\varphi) \subseteq \text{cset} \) then, for all \( \sigma, \sigma \models \varphi \iff [\sigma]_{\text{cset}} \models \varphi. \)

We prove soundness of the rules and axioms that are new in comparison with the proof system from Section 3.3.2. Soundness of the modified Well-Formedness Axiom follows directly from Lemma 4.2.9. The Variable Invariance Axiom can be proved sound easily by using Lemma D.0.15.

**Assignment**
Consider \( \sigma \in M(x := e) \). Then \( \sigma.final = (\sigma.init : x \mapsto \mathcal{E}(e)(\sigma.init)) \), and \( |\sigma| = K_\sigma \).
Thus \( \sigma.final(x) = \mathcal{E}(e)(\sigma.init) \). By Lemma D.1.1, \( \sigma.final(x) = \text{Val}(e)\sigma \) and hence \( \text{Val}(\text{fin}(x))\sigma = \text{Val}(e)\sigma \). From \( |\sigma| = K_\sigma \) we obtain \( \sigma \uparrow K_\sigma \vdash \text{done} \).
Thus \( \sigma \models \text{fin}(x) = e \land \diamondsuit_{\leq K_\sigma} \text{done} \). Hence, \( \models x := e \text{ sat } \text{fin}(x) = e \land \diamondsuit_{\leq K_\sigma} \text{done} \).

**Delay**
Consider \( \sigma \in M(\text{delay e}) \). Then \( \sigma.final = \sigma.init \) and \( |\sigma| = \max(0, \mathcal{E}(e)(\sigma.init)) \). Thus \( \max(|\sigma| - \max(0, \mathcal{E}(e)(\sigma.init)), 0) = 0 \), that is, \( |\sigma| = \max(0, \mathcal{E}(e)(\sigma.init)) = 0 \), and hence \( \sigma \uparrow \max(0, \mathcal{E}(e)(\sigma.init)) \vdash \text{done} \). By Lemma D.1.1, \( \sigma \uparrow \max(0, \text{Val}(e)\sigma) \vdash \text{done} \), and thus \( \sigma \models \Diamond_{\leq e} \text{done} \). Hence, \( \models \text{delay e sat } \Diamond_{\leq e} \text{done} \).

**Input and Output**
We prove the soundness of the Input Rule. The soundness of the Output Rule is proved similarly. Consider \( \sigma \in M(c?x) \). Then
- either \( |\sigma| = \infty \), and for all \( \tau_1 < |\sigma|: \sigma.comm(\tau_1) = \{c?) \),
- or there exists a \( \tau \in \text{TIME} \) such that for all \( \tau_1 < \tau: \sigma.comm(\tau_1) = \{c?) \), there exists a value \( \vartheta \) such that, for all \( \tau_2, \tau \leq \tau_2 < |\sigma|, \sigma.comm(\tau_2) = \{(c, \vartheta)\}, |\sigma| = \tau + K_c \) and \( \sigma.final = (\sigma.init : x \mapsto \vartheta) \).

Hence
- either \( |\sigma| = \infty \), and for all \( \tau_1 < |\sigma|: \sigma \uparrow \tau_1 \models \text{wait}(c?) \),
• or there exists a $r \in \text{TIME}$ such that for all $\tau_1 < r$: $\sigma \uparrow \tau_1 \models \text{wait}(c?)$ and there exists a value $\theta$ such that, for all $\tau_2, \tau_1 \leq \tau_2 < \tau + K_c$, $\sigma \uparrow \tau_2 \models \text{comm}(c, \theta)$ and $\sigma.\text{final}(x) = \theta$.

Thus

• either $\sigma \models \Box \text{wait}(c?)$,

• or there exists a $r \in \text{TIME}$ such that $\sigma \models \Box_{<r} \text{wait}(c?)$ and there exists a value $\theta$ such that, $\sigma \models \Diamond_{=r} \left[ \text{comm}(c, \theta) \cup_{=K_c} \text{done} \right]$ and $\text{Val}(\text{fin}(x)) \sigma = \theta$.

This leads to

• either $\sigma \models \Box \text{wait}(c?)$,

• or there exists a $r \in \text{TIME}$ such that $\sigma \models \Box_{<r} \text{wait}(c?)$ and $\sigma \models \Diamond_{=r} \left[ \text{comm}(c, \text{fin}(x)) \cup_{=K_c} \text{done} \right]$.

Then by Lemma B.1.1 we obtain $\models \text{wait}(c?) \cup \left[ \text{comm}(c, \text{fin}(x)) \cup_{=K_c} \text{done} \right]$.

Hence, $\models c?x \models \text{wait}(c?) \cup \left[ \text{comm}(c, \text{fin}(x)) \cup_{=K_c} \text{done} \right]$.

**Sequential Composition Rule**

Assume $\models S_1 \models \phi_1$ and $\models S_2 \models \phi_2$. We prove $S_1; S_2 \models \phi_1 \cap \phi_2$.

Consider $\sigma \in \mathcal{M}(S_1; S_2) = \text{SEQV}(\mathcal{M}(S_1), \mathcal{M}(S_2))$. Then

• either $\sigma \in \mathcal{M}(S_1)$ and $|\sigma| = \infty$,

• or $\sigma = \sigma_1\sigma_2$ with $\sigma_1 \in \mathcal{M}(S_1)$, $|\sigma_1| < \infty$, $\sigma_2 \in \mathcal{M}(S_2)$, and $\sigma_1.\text{final} = \sigma_2.\text{init}$.

Since $\mathcal{M}(S_2) \neq \emptyset$, then there exist models $\sigma_1$ and $\sigma_2$ such that $\sigma = \sigma_1\sigma_2$, $\sigma_1 \in \mathcal{M}(S_1)$, $\sigma_2 \in \mathcal{M}(S_2)$, and if $|\sigma_1| < \infty$ then $\sigma_1.\text{final} = \sigma_2.\text{init}$. Using $S_1 \models \phi_1$ and $S_2 \models \phi_2$, this implies that there exist models $\sigma_1$ and $\sigma_2$ such that $\sigma = \sigma_1\sigma_2$, $\sigma_1 \models \phi_1$, $\sigma_2 \models \phi_2$, and if $|\sigma_1| < \infty$ then $\sigma_1.\text{final} = \sigma_2.\text{init}$. Hence, $\sigma \models \phi_1 \cap \phi_2$.

**Guarded Command Evaluation**

Consider $\sigma \in \mathcal{M}(G)$. Observe that then $\sigma \uparrow K_g \in \mathcal{M}(\text{delay } K_g)$. Hence $\sigma \models \text{noact}(\text{dch}(G)) \cup_{=K_g} \text{true}$. Further, if $B(\neg b_G)(\sigma.\text{init})$ then $\sigma \in \mathcal{M}(\text{delay } K_g)$.

Thus, by Lemma D.1.2, $\sigma \models \neg b_G$ implies $\sigma \uparrow K_g \models \text{done} \land \text{nochange}(\text{var}(G))$.

Using $(\sigma \uparrow K_g).\text{init} = \sigma.\text{init}$, we obtain $\sigma \uparrow K_g \models \neg b_G \rightarrow \text{done} \land \text{nochange}(\text{var}(G))$.

Hence $\sigma \models \text{noact}(\text{dch}(G)) \cup_{=K_g} [\neg b_G \rightarrow \text{done} \land \text{nochange}(\text{var}(G))]$, and thus $\models G \models \text{noact}(\text{dch}(G)) \cup_{=K_g} [\neg b_G \rightarrow \text{done} \land \text{nochange}(\text{var}(G))]$. 
Guarded Command with Purely Boolean Guards

Assume $\vdash S_i \text{sat } \varphi_i$, for all $i \in \{1, \ldots, n\}$. Consider $\sigma \in \mathcal{M}([\Pi_i^n b_i \rightarrow S_i])$. Then $B(b_G)(\sigma.\text{init})$ implies that there exists a $k \in \{1, \ldots, n\}$ such that $B(b_k)(\sigma.\text{init})$ and $\sigma \in \mathcal{M}(\text{delay } K_g; S_k)$. Therefore, $B(b_G)(\sigma.\text{init})$ implies that there exists a $k \in \{1, \ldots, n\}$ such that $B(b_k)(\sigma.\text{init})$ and $\sigma \in \mathcal{M}(\text{delay } K_g; S_k)$. Thus $B(b_G)(\sigma.\text{init})$ implies that there exists a $k \in \{1, \ldots, n\}$ such that $B(b_k)(\sigma.\text{init})$ and $\sigma \in \mathcal{M}(\text{delay } K_g; S_k)$. Since $\sigma \vdash K_g.\text{init} = \sigma.\text{init}$, $B(b_G)((\sigma \uparrow K_g).\text{init})$ implies that there exists a $k \in \{1, \ldots, n\}$ such that $B(b_k)((\sigma \uparrow K_g).\text{init})$ and $\sigma \vdash K_g \vdash \forall i \in \{1, \ldots, n\} (b_i \land \varphi_i)$, and hence $\sigma \vdash \diamond K_g (b_G \rightarrow \forall i^n (b_i \land \varphi_i))$. This leads to $\vdash [\Pi_i^n b_i \rightarrow S_i] \text{sat } \diamond K_g (b_G \rightarrow \forall i^n (b_i \land \varphi_i))$.

Guarded Command with IO-guards

Next we prove soundness of the rule for $G \equiv [\Pi_i^n b_i; c_i ? x_i \rightarrow S_i; [b; \text{delay } e \rightarrow S]]$. Consider $\sigma \in \mathcal{M}(G)$.

- If $B(\neg b_G)(\sigma.\text{init})$ then, by Lemma D.1.2, $\sigma \vdash \neg b_G$, and thus $\sigma \uparrow K_g \vdash \neg b_G$. Hence $\sigma \vdash \diamond K_g (b_G \land \neg b \rightarrow \text{wait}(G) \cup \forall i^n (b_i \land \varphi_i \land \text{comm}(c_i))) \land (b_G \land b \rightarrow [\text{wait}(G) \cup \neg_e \forall i^n (b_i \land \varphi_i \land \text{comm}(c_i))] \lor [\text{wait}(G) \cup \neg_e \varphi_i])$.

- Now assume $B(b_G)(\sigma.\text{init})$. Suppose $\vdash c_i ? x_i; S_i \text{sat } \varphi_i$, for all $i = 1, \ldots, n$, and $\vdash S \text{sat } \varphi$. If $B(\neg b)(\sigma.\text{init})$ then $\text{Timeout}(G) = \emptyset$ and $\text{LimitedWait}(G) = \emptyset$, and hence $\sigma \in \text{SEQV}(\mathcal{M}(\text{delay } K_g), \text{NotLimWait}(G), \text{Comm}(G))$. Then $\sigma \uparrow K_g = \sigma_1 \sigma_2$ with $\sigma_1 \in \text{Wait}(G)$ and $\sigma_2 \in \text{Comm}(G)$. Hence, using Lemma D.1.2, $\sigma \vdash \diamond K_g (b_G \land \neg b \rightarrow \text{wait}(G) \cup \forall i^n (b_i \land \varphi_i \land \text{comm}(c_i)))$. If $B(b)(\sigma.\text{init})$ then, similarly, $\sigma \vdash \diamond K_g (b_G \land b \rightarrow [\text{wait}(G) \cup \neg_e \forall i^n (b_i \land \varphi_i \land \text{comm}(c_i))] \lor [\text{wait}(G) \cup \neg_e \varphi_i])$.

Iteration

Assume $G \text{sat } \varphi$. Consider $\sigma \in \mathcal{M}(\ast G)$. Then either

- there exists an infinite sequence of models $\sigma_1, \sigma_2, \ldots$ such that $\sigma = \sigma_1 \sigma_2 \cdots$, with for all $i \geq 1$: $\sigma_i \in \mathcal{M}(G), \sigma_{i+1}.\text{init} = \sigma_i.\text{final}, |\sigma_i| < \infty$, and $B(b_G)(\sigma_i.\text{init})$. Then, using $G \text{sat } \varphi$, $\sigma_i \vdash \varphi$. By Lemma D.1.2, $\sigma_i \vdash b_G$. Hence $\sigma \vdash \Diamond (b_G \land \varphi)$.

- there exists a $k \in \mathbb{N}, k \geq 1$ and $\sigma_1, \ldots, \sigma_k$ such that $\sigma = \sigma_1 \cdots \sigma_k$ and, for all $i \in \{1, \ldots, k\}, \sigma_i \in \mathcal{M}(G)$. Hence $\sigma_i \vdash \varphi$ and for all $i \in \{1, \ldots, k-1\}$: $\sigma_{i+1}.\text{init} = \sigma_i.\text{final}, |\sigma_i| < \infty$, and $B(b_G)(\sigma_i.\text{init})$. Thus $\sigma_i \vdash b_G$. Furthermore, either $|\sigma_k| = \infty$ or $B(\neg b_G)(\sigma_k.\text{init})$. 


If $|\sigma_i| = \infty$ then, by the definition of $M(*G), B(b) (\sigma_i$$.init)$, Thus for all $i \in \{1, \ldots, k\}: \sigma_i \models \varphi \land b_\sigma$. Further, $\sigma = \sigma_1 \cdots \sigma_k \sigma_k \cdots$, and if $|\sigma_i| < \infty$ then $i < k$ and thus $\sigma_i$.init = $\sigma_i$.final. Hence $\sigma \models C^\infty (b_\sigma \land \varphi)$.

- If $B(-b_G)(\sigma_i$.init) then $\sigma_k \models -b_G$. Hence $\sigma \models (b_G \land \varphi) C^* (-b_G \land \varphi)$.

**Parallel Composition**

Assume $\models S_1$ sat $\varphi_1$ and $\models S_2$ sat $\varphi_2$ are valid, $dch(\varphi_1) \subseteq dch(S_1)$ and $\var{\varphi_1} \subseteq \var{S_1}$, for $i \in \{1, 2\}$. We show the validity of $S_1||S_2$ sat $(\varphi_1 \land (\varphi_2 C (\text{noact}(dch(S_2) \cup \text{done} \land \text{nochange}(\var{S_2})))))) \lor$

$(\varphi_2 \land (\varphi_1 C (\text{noact}(dch(S_1) \cup \text{done} \land \text{nochange}(\var{S_1}))))))$. Consider any $\sigma \in M(S_1||S_2)$. Then $dch(\sigma) \subseteq dch(S_1) \cup dch(S_2)$, and for $i = 1, 2$ there exist $\sigma_i \in M(S_i)$ such that $|\sigma| = \max(|\sigma_1|, |\sigma_2|), \sigma$.init = $\sigma_i$.init,

$[\sigma$.comm$]_{dch(S_i)}(\tau) = \{ \sigma_i$.comm$((\tau) \text{ for all } \tau < |\sigma|$

$)$for all $\tau, |\sigma_i| \leq \tau < |\sigma|$

and if $|\sigma| < \infty$ then $\sigma$.final($x)$ = $\{ \sigma_i$.final$((x) \text{ if } x \in \var{S_i}$

$)$= $\sigma_i$.init$((x) \text{ if } x \notin \var{S_i||S_2}$

Suppose $|\sigma_1| \leq |\sigma_2|$. Then $|\sigma| = |\sigma_2|$. We prove $\models \sigma \models \varphi_2 \land (\varphi_1 C (\text{noact}(dch(S_1) \cup \text{done} \land \text{nochange}(\var{S_1})))))$.

- First we show $\models \sigma \models \varphi_2$. Since $|\sigma|_{dch(S_2)} = |\sigma| = |\sigma_2|$, $[\sigma$.comm$]_{dch(S_2)}(\tau) = \sigma_2$.comm$((\tau), \text{ for all } \tau < |\sigma|$, and thus $[\sigma$.comm$]_{dch(S_2)} = \sigma_2$.comm$.$

Further, $[\sigma$.init$]_{dch(S_2)} = \sigma$.init$ = \sigma_2$.init$.$

- If $|\sigma| = \infty$ then $|\sigma_2| = \infty$ and then, by Lemma 4.2.9,

$\hat{\sigma_2} = (\sigma_2$.init, $\sigma_2$.comm, $\sigma_2$.final) $\in M(S_2)$.

Since $[\sigma]_{dch(S_2)} = \hat{\sigma}$ we obtain $[\sigma]_{dch(S_2)} \in M(S_2)$.

Thus $\models S_2$ sat $\varphi_2$ leads to $[\sigma]_{dch(S_2)} \models \varphi_2$.

Since $dch(\varphi_2) \subseteq dch(S_2)$, Lemma D.1.3 leads to $\models \sigma \models \varphi_2$.

- If $|\sigma| < \infty$, define $\hat{\sigma} = (\sigma$.init, $\sigma$.comm, $\sigma_2$.final).

Then $[\hat{\sigma}]_{dch(S_2)} = \sigma_2 \in M(S_2)$, and thus $[\hat{\sigma}]_{dch(S_2)} \models \varphi_2$.

Using $dch(\varphi_2) \subseteq dch(S_2)$, Lemma D.1.3 leads to $\models \hat{\sigma} \models \varphi_2$.

Since $\hat{\sigma}$.final$((x) = \sigma_2$.final$((x) = \sigma$.final$((x) \text{ for all } x \in \var{S_2}$ and

$\var{\varphi_2} \subseteq \var{S_2}$, this leads to $\models \sigma \models \varphi_2$.

- Next we prove $\models \varphi_1 C (\text{noact}(dch(S_1) \cup \text{done} \land \text{nochange}(\var{S_1}))))$.

From $\sigma_1 \in M(S_1)$ and $\models S_1$ sat $\varphi_1$, we obtain $\sigma_1 \models \varphi_1$.

We define a model $\sigma_3$ that satisfies $\text{noact}(dch(S_1)) \cup \text{done} \land \text{nochange}(\var{S_1})$.
Let $\sigma_3$ be such that $|\sigma_3| = |\sigma| - |\sigma_1|$, for all $\tau < |\sigma_3|$, $\sigma_3.\text{comm}(\tau) = \sigma.\text{comm}(\tau + |\sigma_1|)$, and $\sigma_3.\text{init} = \sigma_3.\text{final} = \sigma_1.\text{final}$.

Then clearly $\sigma_3 \models \Box \text{nochange}(\text{var}(S_1))$.

Since for all $\tau$, $|\sigma_1| \leq \tau < |\sigma|$, $[\sigma.\text{comm}]_{\text{dch}(S_1)}(\tau) = \emptyset$, we obtain for all $\tau < |\sigma| - |\sigma_1|$, $[\sigma.\text{comm}]_{\text{dch}(S_1)}(\tau + |\sigma_1|) = \emptyset$.

Thus, for all $\tau < |\sigma_3|$, $[\sigma_3.\text{comm}]_{\text{dch}(S_1)}(\tau) = [\sigma.\text{comm}]_{\text{dch}(S_1)}(\tau + |\sigma_1|) = \emptyset$, and hence $\sigma_3 \models \text{noact(dch}(S_1)) \cup [\text{done} \land \text{nochange(var}(S_1)))]$.

Thus $\sigma_1 \sigma_3 \models \varphi_1 \land \text{noact(dch}(S_1)) \cup [\text{done} \land \text{nochange(var}(S_1)))]$.

Since $[\sigma.\text{comm}]_{\text{dch}(S_1)}(\tau) = \begin{cases} \sigma_1.\text{comm}(\tau) & \text{for all } \tau < |\sigma_1| \\ \emptyset & \text{for all } \tau, |\sigma_1| \leq \tau < |\sigma| \end{cases}$

we have $[\sigma.\text{comm}]_{\text{dch}(S_1)} = (\sigma_1.\text{comm})(\sigma_3.\text{comm})$.

Further, $[\sigma.\text{init}]_{\text{dch}(S_1)} = \sigma.\text{init} = \sigma_1.\text{init}$.

Define $\hat{\sigma} = (\sigma.\text{init}, \sigma.\text{comm}, \sigma_1.\text{final})$. Then $[\hat{\sigma}]_{\text{dch}(S_1)} = \sigma_1 \sigma_3$. This leads to

$[\hat{\sigma}]_{\text{dch}(S_1)} \models \varphi_1 \land \text{noact(dch}(S_1)) \cup [\text{done} \land \text{nochange(var}(S_1)))]$.

Since $\text{dch}(\varphi_1) \subseteq \text{dch}(S_1)$, we have

$\text{dch}(\varphi_1) \subseteq \text{dch}(S_1) \cup [\text{done} \land \text{nochange(var}(S_1)))] \subseteq \text{dch}(S_1)$

and thus Lemma D.1.3 leads to

$\hat{\sigma} \models \varphi_1 \land \text{noact(dch}(S_1)) \cup [\text{done} \land \text{nochange(var}(S_1)))]$.

Since $\hat{\sigma}.\text{final}(x) = \sigma_1.\text{final}(x) = \sigma.\text{final}(x)$ for all $x \in \text{var}(S_1)$ and $\text{var}(\varphi_1) \subseteq \text{var}(S_1) \cup [\text{done} \land \text{nochange(var}(S_1)))] \subseteq \text{var}(S_1)$, this leads to $\sigma \models \varphi_1 \land \text{noact(dch}(S_1)) \cup [\text{done} \land \text{nochange(var}(S_1)))]$.

Similarly, for $|\sigma_2| \leq |\sigma_1|$ we can prove

$\sigma \models \varphi_1 \land (\varphi_2 \land \text{noact(dch}(S_2)) \cup [\text{done} \land \text{nochange(var}(S_2))])$,

which proves soundness of the Parallel Composition Rule.

### D.2 Soundness of the Proof System in Section 4.4.2

First a number of lemmas which can be proved easily by structural induction.

**Lemma D.2.1** For any boolean expression $b$ from the programming language, state $s$, environment $\gamma$, and communication function $cf$: $B(b)(s) \iff [b]_\gamma(s, cf)$.

**Lemma D.2.2** For any expression $e$ from the programming language, state $s$, environment $\gamma$, and communication function $cf$: $E(e)(s) = V(e)(\gamma, s, cf)$.

**Lemma D.2.3** For any assertion $p$, state $s$, environment $\gamma$, and communication function $cf$: $[p[exp/x]]_\gamma(s, cf) \iff [p]_\gamma((s : x \mapsto V(exp)(\gamma, s, cf)), cf)$. 


Similar to Lemma 3.4.3, we have

**Lemma D.2.4** For all $\gamma$, $s$, $c_{f_1}$: $[p] \gamma(s, c_{f_1})$ iff for all $c_{f_2}$, $[p[c_{f_1}/\text{time}]] \gamma(s, c_{f_1}^c c_{f_2})$.

From this lemma we obtain:

**Lemma D.2.5** For all $\gamma$, $s$, $c_{f_1}$, $c_{f_2}$: if $[p[c_{f_1}^c c_{f_2}/\text{time}]] \gamma(s, c_{f_1})$ then $[p] \gamma(s, c_{f_1}^c c_{f_2})$.

**Proof:** $[p[c_{f_1}^c c_{f_2}/\text{time}]] \gamma(s, c_{f_1})$ implies (by Lemma D.2.4) $[p[c_{f_1}^c c_{f_2}/\text{time}][c_{f_1}/\text{time}]] \gamma(s, c_{f_1}^c c_{f_2})$.

Thus $[p[c_{f_1}^c c_{f_2}/\text{time}]] \gamma(s, c_{f_1}^c c_{f_2})$, and hence $[p] \gamma(s, c_{f_1}^c c_{f_2})$. $\square$

In the proof below $\gamma$ is an arbitrary environment, $s_0 \in \text{STATE}$, and $c_{f_0} \in \text{CF}$ is a well-formed communication function with $|c_{f_0}| < \infty$. We prove $\models C : \{p\} S \{q\}$ as follows:

- if $[p] \gamma(s_0, c_{f_0})$ and $(s_0, c_{f_1}, s_1) \in M(S)$ then $[C] \gamma(s_1, c_{f_0}^c c_{f_1})$, and if $|c_{f_1}| < \infty$ then $[q] \gamma(s_1, c_{f_0}^c c_{f_1})$. (Observe that $\text{var}(C) = \emptyset$ has been required for all commitments.)

The soundness of the Well-Formedness Axiom follows easily from Lemma 4.2.9.

**Initial Invariance**

Assume $[[p \land C]] \gamma(s_0, c_{f_0})$. Consider $(s_0, c_{f_1}, s_1) \in M(S)$.

By Lemma D.2.4 we obtain $[[p \land C][c_{f_0}/\text{time}]] \gamma(s_0, c_{f_0}^c c_{f_1})$.

Since $\text{time}$ does not occur in $p \land C$, this leads to $[[p \land C]] \gamma(s_0, c_{f_0}^c c_{f_1})$.

By Lemma D.0.15, $s_1(x) = s_0(x)$ for $x \notin \text{var}(S)$.

Since $\text{var}(C) = \emptyset$ and $\text{var}(S) \cap \text{var}(p) = \emptyset$, we obtain $[[p \land C]] \gamma(s_1, c_{f_0}^c c_{f_1})$.

**Skip**

Assume $[[p \land C]] \gamma(s_0, c_{f_0})$. Consider $(s_0, c_{f_1}, s_1) \in M(\text{skip})$.

Then $s_1 = s_0$ and $|c_{f_1}| = 0$, and hence $[[p \land C]] \gamma(s_1, c_{f_0}^c c_{f_1})$.

**Assignment**

Assume $[[(q \land C)[\text{time} + K_a/\text{time}, e/x]] \gamma(s_0, c_{f_0})$. Consider $(s_0, c_{f_1}, s_1) \in M(x := e)$.

Then $s_1 = (s_0 : x \mapsto E(e)(s_0))$ and $|c_{f_1}| = K_a$. By Lemma D.2.3 this leads to $[[(q \land C)[\text{time} + K_a/\text{time}]] \gamma((s_0 : x \mapsto V(e)(\gamma(s_0, c_{f_0})), c_{f_0})$. Using Lemma D.2.2 we obtain $[[(q \land C)[\text{time} + K_a/\text{time}]] \gamma(s_1, c_{f_0})$. This is equivalent to $[[(q \land C)[c_{f_0} + |c_{f_1}/\text{time}]] \gamma(s_1, c_{f_0})$, and thus, by Lemma D.2.5, $[[q \land C]] \gamma(s_1, c_{f_0}^c c_{f_1})$. 

Delay

Assume \([q \land C][time + \max(0, e)/time]p\gamma(s_0, c_{f_0})\). Consider \((s_0, c_{f_1}, s_1) \in \mathcal{M}(\text{delay } e)\). Then \(s_1 = s_0\) and \(|c_{f_1}| = \max(0, \epsilon(e)(s_0))\).

By Lemma D.2.2, \(|c_{f_1}| = \max(0, \nu(e)(\gamma, s_0, c_{f_0}))\).

Using \(s_1 = s_0\) this leads to \([q \land C][c_{f_0} + |c_{f_1}|/time]p\gamma(s_1, c_{f_0})\).

Hence, by Lemma D.2.5 we obtain \([q \land C]q(s_1, c_{f_0} \land c_{f_1})\).

Output

We prove the soundness of the Input Rule. The soundness of the Output Rule is proved similarly. Suppose \(p[t_0/time]q \land \text{ wait to } c?\) at \(t_0\) and comm value \(v \rightarrow (q[v/x] \land C)\).

Assume \([p]q(s_0, c_{f_0})\). Define \(\gamma = (\gamma : t_0 \rightarrow |c_{f_0}|)\). Then, by Lemma D.2.4, \([p[t_0/time]]\gamma(s_0, c_{f_0} \land c_{f_1})\). Consider \((s_0, c_{f_1}, s_1) \in \mathcal{M}(c?x)\). Then

- either \(|c_{f_1}| = \infty\), and for all \(r_1 \in \text{TIME}\): \(c_{f_1}(r_1) = \{c?\}\). Since \(\gamma(t_0) = |c_{f_0}|\), this implies \([\text{wait to } c? \text{ during } [t_0, \infty) \land \text{time } = \infty]q(s_0, c_{f_0} \land c_{f_1})\).

Observe that, by definition, \([\text{comm via } c \text{ during } [\infty, \infty) \text{ value } v]q(s_0, c_{f_0} \land c_{f_1})\).

Hence \([\text{wait to } c? \text{ at } t_0 \text{ and comm value } v]q(s_0, c_{f_0} \land c_{f_1})\). This leads to

\([C]q(s_0, c_{f_0} \land c_{f_1})\). Since \(\text{var}(C) = \emptyset\) and \(t_0\) is fresh, we obtain \([C]q(s_1, c_{f_0} \land c_{f_1})\).

- or \(|c_{f_1}| < \infty\), there exists a \(r \in \text{TIME}\) and a value \(\theta\) such that \(|c_{f_1}| = r + K_c\), for all \(r_1 < r: c_{f_1}(r_1) = \{c?\}\), for all \(r_2, r \leq r_2 < r + K_c: c_{f_1}(r_2) = \{(c, \theta)\}\), and \(s_1 = (s_0 : x \mapsto \theta)\). Hence,

\([\text{wait to } c? \text{ during } [t_0, t) \land \text{comm via } c \text{ during } [t, t + K_c) \text{ value } v \land \text{time } = t + K_c]q(s_0, c_{f_0} \land c_{f_1})\).

Thus \(\exists t \geq t_0: \text{wait to } c? \text{ during } [t_0, t) \land \text{comm via } c \text{ during } [t, t + K_c) \text{ value } v \land \text{time } = t + K_c\).

This leads to \([q[v/x] \land C]q(v \mapsto \theta)(s_0, c_{f_0} \land c_{f_1})\).

Since \(v\) and \(t_0\) do not occur free in \(C\) and \(\text{var}(C) = \emptyset\) we obtain \([C]q(s_1, c_{f_0} \land c_{f_1})\).

By Lemma D.2.3, using that \(\nu(v)((\gamma : v \mapsto \theta), s_0, c_{f_0} \land c_{f_1}) = \theta\),

\([q]q(v \mapsto \theta)((s_0 : x \mapsto \theta), c_{f_0} \land c_{f_1})\). Thus \([q]q(v \mapsto \theta)(s_1, c_{f_0} \land c_{f_1})\).

Since \(v\) and \(t_0\) do not occur free in \(q\) this leads to \([q]q(s_1, c_{f_0} \land c_{f_1})\).

Sequential Composition

Suppose \(\models C_1 : \{p\} S_1 \{r\} \text{ and } \models C_2 : \{r\} S_2 \{q\}\).

Assume \([p]q(s_0, c_{f_0})\). Consider \((s_0, c_{f_1}, s_1) \in \mathcal{M}(S_1; S_2) = \text{SEQV}(\mathcal{M}(S_1), \mathcal{M}(S_2))\).

Thus
• either \((s_0, c_f_1, s_1) \in \mathcal{M}(S_1)\) and \(|c_f_1| = \infty\).

Then, by \(\models C_1 : \{p\} S_1 \{r\}\), we obtain \([C_1] \gamma(s_1, c_f_0 \wedge c_f_1)\).

Further, \([\text{time} = \infty] \gamma(s_1, c_f_0 \wedge c_f_1)\), and thus \([C_1 \wedge \text{time} = \infty] \gamma(s_1, c_f_0 \wedge c_f_1)\).

• or there exist \(\delta, c_f_{11}, c_f_{12}\) such that \(c_f_1 = c_f_{11} \wedge c_f_{12}\), \((s_0, c_f_{11}, \delta) \in \mathcal{M}(S_1)\), \(|c_f_{11}| < \infty\), and \((\delta, c_f_{12}, s_1) \in \mathcal{M}(S_2)\). Then, using \(\models C_1 : \{p\} S_1 \{r\}\), we obtain \([r] \gamma(\delta, c_f_0 \wedge c_f_{11})\). By \(\models C_2 : \{r\} S_2 \{q\}\) this leads to \([C_2] \gamma(s_1, c_f_0 \wedge c_f_{11} \wedge c_f_{12})\) and if \(|c_f_{12}| < \infty\) then \([q] \gamma(s_1, c_f_0 \wedge c_f_{11} \wedge c_f_{12})\). Hence \([C_2] \gamma(s_1, c_f_0 \wedge c_f_1)\) and if \(|c_f_1| < \infty\) then \([q] \gamma(s_1, c_f_0 \wedge c_f_1)\).

**Guarded Command Termination**

Suppose \(\models C : \{p \wedge \neg b_G\} \text{ delay } K_g \{q\}\). Assume \([p \wedge \neg b_G] \gamma(s_0, c_f_0)\). Consider \((s_0, c_f_1, s_1) \in \mathcal{M}(G)\). Since \([-b_G] \gamma(s_0, c_f_0)\), Lemma D.2.1 leads to \(B(-b_G)(s_0)\). Hence \((s_0, c_f_1, s_1) \in \mathcal{M}(\text{delay } K_g)\), and thus \([C \wedge q] \gamma(s_1, c_f_0 \wedge c_f_1)\).

**Guarded Command with Purely Boolean Guards**

Suppose \(\models C_i : \{p \wedge b_i\} \text{ delay } K_g \{S_i \{q_i\}\\}, \forall i \in \{1, \ldots, n\}\).

Assume \([p \wedge b_G] \gamma(s_0, c_f_0)\). Consider \((s_0, c_f_1, s_1) \in \mathcal{M}(\{i : b_i \rightarrow S_i\})\).

Since \([b_G] \gamma(s_0, c_f_0)\), Lemma D.2.1 leads to \(B(b_G)(s_0)\). Hence there exists a \(k \in \{1, \ldots, n\}\) such that \(B(b_k)(s_0)\), and \((s_0, c_f_1, s_1) \in \mathcal{M}(\text{delay } K_g; S_k)\). By Lemma D.2.1, \([b_k] \gamma(s_0, c_f_0)\), thus \([p \wedge b_k] \gamma(s_0, c_f_0)\). Then \(\models C_k : \{p \wedge b_k\} \text{ delay } K_k; S_k \{q_k\}\) leads to \([C_k] \gamma(s_1, c_f_0 \wedge c_f_1)\) and if \(|c_f_1| < \infty\) then \([q_k] \gamma(s_1, c_f_0 \wedge c_f_1)\).

Hence \([V_{i=1}^n C_i] \gamma(s_1, c_f_0 \wedge c_f_1)\) and if \(|c_f_1| < \infty\) then \([V_{i=1}^n q_i] \gamma(s_1, c_f_0 \wedge c_f_1)\).

**Guarded Command without Delay**

Consider \(G \equiv \left[ \bigwedge_{i=1}^n b_i; c_i ? x_i \rightarrow S_i \right] b; \text{ delay } e \rightarrow S\). Suppose

\(\models p[t_0/\text{time}] \wedge \neg \text{no (dch}(G)) \text{ during } [t_0, t_0 + K_g] \wedge \)

\(\text{ wait in } G \text{ during } [t_0 + K_g, \infty] \wedge \text{ time } = \infty \rightarrow C_{\text{nonterm}}\)

\(\models p[t_0/\text{time}] \wedge b_i \wedge (\exists t, t_0 + K_g \leq t < \infty : \neg \text{no (dch}(G)) \text{ during } [t_0, t_0 + K_g] \wedge \)

\(\text{ wait in } G \text{ during } [t_0 + K_g, t] \wedge \text{ comm } c_i \text{ in } G \text{ from } t \text{ value } v)\)

\(\rightarrow p_i[v/x_i]\), for \(i = 1, \ldots, n\)

\(\models C_i : \{p_i\} S_i \{q_i\}\), for \(i = 1, \ldots, n\)

Assume \([p \wedge b_G \wedge \neg \gamma] \gamma(s_0, c_f_0)\). Consider \((s_0, c_f_1, s_1) \in \mathcal{M}(G)\). Let \(\gamma = (t_0 \rightarrow |c_f_0|)\).

Then, by Lemma D.2.4, \([p[t_0/\text{time}]] \gamma(s_0, c_f_0 \wedge c_f_1)\). From \([b_G] \gamma(s_0, c_f_0)\) we obtain, by Lemma D.2.1, \(B(b_G)(s_0)\). Similarly, \([-b] \gamma(s_0, c_f_0)\) leads to \(B(\neg b_G)(s_0)\).

Then \((s_0, c_f_1, s_1) \in \text{SEQV}(\mathcal{M}(\text{delay } K_g), \text{NotLimWait}(G), \text{Comm}(G))\), that is,
either \((s_0, cf_1, s_1) \in \text{SEQV}(\mathcal{M}(\text{delay } K_g), \text{NotLimWait}(G))\) with \(|cf_1| = \infty\).

Then for all \(\tau_1, 0 \leq \tau_1 < K_g\): \(cf_1(\tau_1) = \emptyset\) and for all \(\tau_2, K_g \leq \tau_2 < \infty\):
\[
 cf_1(\tau_2) = \{c_i? | B(b_i)(s_0), 1 \leq i \leq n\}.
\]
Since \(\gamma(t_0) = |cf_0|\), this leads to
\[
[\text{no (dch}(G)) \text{ during } [t_0, t_0 + K_g) \wedge \text{wait in } G \text{ during } [t_0 + K_g, \infty]]\gamma(s_0, cf_0^\uparrow cf_1).
\]
By Lemma D.2.4 we obtain \([p[t_0/time]]\gamma(s_0, cf_0^\uparrow cf_1)\). Then the first assumption of the rule this leads to \([\text{Cnonterm}]\gamma(s_0, cf_0^\uparrow cf_1)\). Since \(t_0\) does not occur in \(\text{Cnonterm}\) and \(\text{var}(\text{Cnonterm}) = \emptyset\), we obtain \([\text{Cnonterm}]\gamma(s_1, cf_0^\uparrow cf_1)\).

or there exist \(\tau \in \text{TIME}, \tau \geq K_g\), \(\dot{s}, cf_{11}\) and \(cf_{12}\) such that \(cf_1 = cf_{11}^\uparrow cf_{12}\), for all \(\tau_1, 0 \leq \tau_1 < K_g\): \(cf_{11}(\tau_1) = \emptyset\), for all \(\tau_2, K_g \leq \tau_2 < \tau\):
\[
 cf_{11}(\tau_2) = \{c_i? | B(b_i)(s_0), 1 \leq i \leq n\},
\]
there exists a \(k \in \{1, \ldots, n\}\) and a value \(\theta\) such that \(B(b_k)(s_0)\), for all \(\tau_3, \tau \leq \tau_3 < \tau + K_g\):
\[
 cf_{11}(\tau_3) = \{(c_k, \theta)\}, \quad |cf_{11}| = \tau + K_c,
\]
\(\dot{s} = (s_0 : x_k \leftarrow \theta)\), and \((s_0, cf_{11}, s_1) \in \mathcal{M}(S_k)\). Thus
\[
[\exists t, t_0 + K_g \leq t < \infty : \text{no (dch}(G)) \text{ during } [t_0, t_0 + K_g) \wedge
 wait in G during [t_0 + K_g, t) \wedge \text{comm } c_i \text{ in } G \text{ from } t \text{ value } v]]\gamma(s_0, cf_0^\uparrow cf_{11}).
\]
By Lemma D.2.4, using that \(v\) is fresh, \([p[t_0/time]](\gamma : v \leftarrow \theta)(s_0, cf_0^\uparrow cf_{11})\).

Further, \(B(b_k)(s_0)\) leads by Lemma D.2.1 to \([b_k]\gamma(s_0, cf_0^\uparrow cf_{11}).\)
Then from the second assumption of the rule we obtain
\[
[p[k[v/x_k]]](\gamma : v \leftarrow \theta)(s_0, cf_0^\uparrow cf_{11}).\]
Thus, by Lemma D.2.3,
\[
[p[k]\gamma(s_1, cf_0^\uparrow cf_{11}).\]
By the third assumption of the rule this leads to
\[
[V^k_{i=1}](\gamma : v \leftarrow \theta)(s_1, cf_0^\uparrow cf_{11}^\uparrow cf_{12})\]
and if \(|cf_{12}| < \infty\) then
\[
[V^k_{i=1}](\gamma : v \leftarrow \theta)(s_1, cf_0^\uparrow cf_{11}^\uparrow cf_{12}).\]
Since \(t_0\) and \(v\) are fresh, we obtain
\[
[V^k_{i=1}](\gamma : v \leftarrow \theta)(s_1, cf_0^\uparrow cf_{11}^\uparrow cf_{12}).\]
Thus, by Lemma D.2.3,
\[
[V^k_{i=1}](\gamma : v \leftarrow \theta)(s_1, cf_0^\uparrow cf_{12}).\]
Assume \([p \wedge b]\gamma(s_0, cf_0).\) Consider \((s_0, cf_1, s_1) \in \mathcal{M}(G)\).

Guarded Command with Delay

Consider \(G \equiv [\prod_{i=1}^n b_i, c_i? x_i \rightarrow S_i \prod b; \text{delay } e \rightarrow S]\). Suppose
\[
\models p[t_0/time] \wedge b_i \wedge (\exists t, t_0 + K_g \leq t < \max(0, e) : \text{no (dch}(G)) \text{ during } [t_0, t_0 + K_g) \wedge
 wait in G during [t_0 + K_g, t) \wedge \text{comm } c_i \text{ in } G \text{ from } t \text{ value } v)
\]
\[
\rightarrow p_i[v/x_i], \text{for } i = 1, \ldots, n
\]
\[
\models C_i : \{p_i\} S_i \{q_i\}, \text{for } i = 1, \ldots, n
\]
\[
\models p[t_0/time] \wedge \text{no (dch}(G)) \text{ during } [t_0, t_0 + K_g) \wedge
 wait in G during [t_0 + K_g, t_0 + K_g + \max(0, e)) \wedge
 time = t_0 + K_g + \max(0, e) \rightarrow \dot{p}
\]
\[
\models C : \{\dot{p}\} S \{q\}
\]
Assume \([p \wedge b]\gamma(s_0, cf_0).\) Consider \((s_0, cf_1, s_1) \in \mathcal{M}(G)\).
Define \( \gamma = (\gamma : t_0 \mapsto |c_{f0}|) \). Then, by Lemma D.2.4, \([p[t_0/time]]\gamma(s_0, c_{f0}^c f_1)\).

Since \([b]\gamma(s_0, c_{f0})\), Lemma D.2.1, leads to \(B(b)(s_0)\) and thus \(B(b_0)(s_0)\). Then

- either \((s_0, c_{f1}, s_1) \in SEQV(\mathcal{M}(\text{delay } K_g), \text{LimitedWait}(G), \text{Comm}(G))\).

Then similar to the proof for the previous rule we obtain \(\forall i \in \mathbb{N}, i > 1, \forall i \in \{1, \ldots, k\} : \sigma_i \in \mathcal{M}(G), \sigma_i \neq \sigma_i.\text{init} = \sigma_i.\text{final}, |\sigma_i| < \infty, B(b_G)(\sigma_i.\text{init})\), and either \(|\sigma_k| = \infty\) or \(B(\neg b_G)(\sigma_k.\text{init})\).

Now we consider the cases \(k = 1\) and \(k > 1\):

- If \(k = 1\) then \(B(\neg b_G)(\sigma_1.\text{init})\) and \(s_0 = \sigma_1.\text{init}\), thus \([p \wedge \neg b_G]\gamma(s_0, c_{f0})\).

Since \((s_0, c_{f1}, s_1) = \sigma_1\) and \(\sigma_1 \in \mathcal{M}(G)\), we obtain from (D.2):

\([C_{\text{term}}]\gamma(s_1, c_{f0}^c f_1)\), and if \(|c_{f1}| < \infty\) then \([q]\gamma(s_1, c_{f0}^c f_1)\).

- If \(k > 1\) then prove by induction on \(i\):

\([p]\gamma(\sigma_i.\text{final}, c_{f0}^c \sigma_i.\text{comm} \cdots \sigma_i.\text{comm})\), for \(i = 1, \ldots, k - 1\).
Basis For \( i = 1 \) we have \( B(b_G)(\sigma_1.init) \) and \( s_0 = \sigma_1.init \), thus
\[
[p \land b_G] \gamma(s_0, cf_0). \quad \text{Since } \sigma_1 = (\sigma_1.init, \sigma_1.comm, \sigma_1.final) \in \mathcal{M}(G) \text{ and } |\sigma_1.comm| = |\sigma_1| < \infty, \text{ we obtain from (D.1)}:
\[
[p] \gamma(\sigma_1.final, cf_0^\bot \sigma_1.comm).
\]

Induction Consider \( i \) with \( 1 < i \leq k - 1 \). By the induction hypothesis,
\[
[p] \gamma(\sigma_{i-1}.final, cf_0^\bot \sigma_1.comm \cdots \sigma_{i-1}.comm).
\]
Since \( B(b_G)(\sigma_i.init) \) and \( \sigma_i.init = \sigma_{i-1}.final \), we obtain
\[
[p \land b_G] \gamma(\sigma_{i-1}.final, cf_0^\bot \sigma_1.comm \cdots \sigma_{i-1}.comm).
\]
Together with \( \sigma_i = (\sigma_i.init, \sigma_i.comm, \sigma_i.final) \in \mathcal{M}(G) \) and \( |\sigma_i.comm| = |\sigma_i| < \infty \),
this leads by (D.1) to
\[
[p] \gamma(\sigma_i.final, cf_0^\bot \sigma_1.comm \cdots \sigma_{i-1}.comm).
\]
With \( i = k - 1 \) we obtain
\[
[p] \gamma(\sigma_{k-1}.final, cf_0^\bot \sigma_1.comm \cdots \sigma_{k-1}.comm).
\]
Now there are two possibilities:

(a) Either \( |\sigma_k| = \infty \).

Since \( B(-b_G)(\sigma_k.init) \) and \( \sigma_k \in \mathcal{M}(G) \) imply \( |\sigma_k| < \infty \),
we must have \( B(b_G)(\sigma_k.init) \). Together with \( \sigma_k.init = \sigma_{k-1}.final \) this leads to
\[
[p \land b_G] \gamma(\sigma_{k-1}.final, cf_0^\bot \sigma_1.comm \cdots \sigma_{k-1}.comm).
\]
With \( \sigma_k \in \mathcal{M}(G) \), this leads by (D.1) to
\[
[C] \gamma(\sigma_k.final, cf_0^\bot \sigma_1.comm \cdots \sigma_k.comm),
\]
and hence
\[
C[\infty/time] \gamma(s_1, cf_0^\bot cf_1),
\]
and hence
\[
\forall t_1 < \infty \exists t_2 > t_1 : C[t_2/time] \gamma(s_1, cf_0^\bot cf_1).
\]
Thus, by (D.3),
\[
C_{nonterm} \gamma(s_1, cf_0^\bot cf_1).
\]

(b) Or \( B(-b_G)(\sigma_k.init) \), and thus \( |\sigma_k| < \infty \).

Since \( \sigma_k.init = \sigma_{k-1}.final \), this leads to
\[
[p \land -b_G] \gamma(\sigma_{k-1}.final, cf_0^\bot \sigma_1.comm \cdots \sigma_{k-1}.comm).
\]
With \( \sigma_k \in \mathcal{M}(G) \) we obtain from (D.2):
\[
[C_{term}] \gamma(\sigma_k.final, cf_0^\bot \sigma_1.comm \cdots \sigma_k.comm) \text{ and, since } |\sigma_k.comm| < \infty,
\]
\[
[q] \gamma(\sigma_k.final, cf_0^\bot \sigma_1.comm \cdots \sigma_k.comm).
\]
By \( s_1 = \sigma_k.final \) and \( cf_1 = \sigma_1.comm \cdots \sigma_k.comm \), this leads to
\[
[C_{term}] \gamma(s_1, cf_0^\bot cf_1) \text{ and } q \gamma(s_1, cf_0^\bot cf_1).
\]

2. There exists an infinite sequence of models \( \sigma_1, \sigma_2, \ldots \) such that
\( (s_0, cf_1, s_1) = \sigma_1 \sigma_2 \cdots, \) with for all \( i \geq 1: \sigma_i \in \mathcal{M}(G), \sigma_{i+1.init} = \sigma_i.final, \)
\( |\sigma_i| < \infty, \) and \( B(b_G)(\sigma_i.init) \). We prove, by induction on \( i \) that, for all \( i \geq 1:
\]
\[
[p \land C] \gamma(\sigma_i.final, cf_0^\bot \sigma_1.comm \cdots \sigma_i.comm).
\]
Basis For \( i = 1 \) we have \( B(b_G)(\sigma_1.init) \) and \( s_0 = \sigma_1.init \), thus \( [p \land b_G] \gamma(s_0,c_f_0) \).

Since \( \sigma_1 = (\sigma_1.init,\sigma_1.comm,\sigma_1.final) \in \mathcal{M}(G) \) and \( |\sigma_1.comm| = |\sigma_1| < \infty \), we obtain from (D.1): \( [p \land C] \gamma(\sigma_1.final,c_f_0 \land \sigma_1.comm) \).

Induction Consider \( i \) with \( 1 < i \leq k - 1 \). By the induction hypothesis:

\[
[p] \gamma(\sigma_{i-1}.final,c_f_0 \land \sigma_{i-1}.comm) \cdot \sigma_{i-1}.comm).
\]

Since \( B(b_G)(\sigma_1.init) \) and \( \sigma_i.init = \sigma_{i-1}.final \), we obtain

\[
[p \land b_G] \gamma(\sigma_{i-1}.final,c_f_0 \land \sigma_{i-1}.comm) \cdot \sigma_{i-1}.comm).
\]

Together with \( \sigma_i = (\sigma_i.init,\sigma_i.comm,\sigma_i.final) \in \mathcal{M}(G) \) and \( |\sigma_i.comm| = |\sigma_i| < \infty \), this leads by (D.1) to \( [p \land C] \gamma(\sigma_i.final,c_f_0 \land \sigma_i.comm \cdots \sigma_i.comm) \).

By Lemma D.2.4, for \( i \geq 1 \), \( [p \land C] \gamma(\sigma_i.final,c_f_0 \land \sigma_i.comm \cdots \sigma_i.comm) \), implies

\[
[C][c_f_0 \land \sigma_i.comm \cdots \sigma_i.comm]/[time] \gamma
\]

Thus \( [C][c_f_0 \land \sigma_i.comm \cdots \sigma_i.comm]/[time] \gamma(\sigma_i.final,c_f_0 \land c_f_i) \).

Since there are no program variables in \( C \), this leads to

\[
[C][c_f_0 \land \sigma_i.comm \cdots \sigma_i.comm]/[time] \gamma(s_1,c_f_0 \land c_f_i), \text{ for } i \geq 1.
\]

Observe that for all \( \tau_1 \in \text{TIME} \) there exists a \( i \) such that \( |c_f_0 \land \sigma_i.comm \cdots \sigma_i.comm| > \tau_1 \).

Hence, for all \( \tau_1 \in \text{TIME} \) there exists a \( \tau_2 > \tau_1 \) such that \( [C][\tau_2]/[time] \gamma(s_1,c_f_0 \land c_f_i) \).

This leads to \( \forall \tau_1 < \infty \exists \tau_2 > \tau_1 : \gamma(s_1,c_f_0 \land c_f_i) \), and thus, by (D.3),

\[
[C_{\text{nonterm}}] \gamma(s_1,c_f_0 \land c_f_i).
\]

Since \( |c_f_i| = \infty \) we also have \( [\text{time} = \infty] \gamma(s_1,c_f_0 \land c_f_i) \).

Parallel Composition

Assume

\[
C_i : \{p_i\} S_i \{q_i\}, \text{ for } i = 1,2
\]

(4.4)

\[
\exists t_1,t_2 : \text{time} = max(t_1,t_2) \land \bigcup_{i=1}^{2} C_i[t_i/time] \land \text{no dch}(S_i) \text{ during } [t_i,\text{time}) \rightarrow C
\]

(4.5)

\[
\exists t_1,t_2 : \text{time} = max(t_1,t_2) \land \bigcup_{i=1}^{2} q_i[t_i/time] \land \text{no dch}(S_i) \text{ during } [t_i,\text{time}) \rightarrow q
\]

(4.6)

\[
dch(C_i,q_i) \subseteq dch(S_i) \text{ and } var(q_i) \subseteq var(S_i), \text{ for } i = 1,2
\]

(4.7)

\[
t_1 \text{ and } t_2 \text{ are fresh logical variables}
\]

(4.8)

Assume \( [p_1 \land p_2] \gamma(s_0,c_f_0) \). Consider any \( (s_0,c_f_1,s_1) \in \mathcal{M}(S_1||S_2) \).

Then \( dch(c_f_1) \subseteq dch(S_1) \cup dch(S_2) \), and for \( i = 1,2 \) there exist \( (s_0,c_f_{i1},s_{i1}) \in \mathcal{M}(S_i) \) such that \( |c_f_i| = max(|c_f_{i1}|,|c_f_{i2}|) \),

\[
|c_f_i|_{dch(S_i)}(\tau) = \begin{cases} c_f_{i1}(\tau) & \text{for all } \tau < |c_f_{i1}| \\ \emptyset & \text{for all } \tau, |c_f_{i1}| \leq \tau < |c_f_i| \end{cases}
\]

Assume \( [p_1 \land p_2] \gamma(s_0,c_f_0) \). Consider any \( (s_0,c_f_1,s_1) \in \mathcal{M}(S_1||S_2) \).

Then \( dch(c_f_1) \subseteq dch(S_1) \cup dch(S_2) \), and for \( i = 1,2 \) there exist \( (s_0,c_f_{i1},s_{i1}) \in \mathcal{M}(S_i) \) such that \( |c_f_i| = max(|c_f_{i1}|,|c_f_{i2}|) \),

\[
|c_f_i|_{dch(S_i)}(\tau) = \begin{cases} c_f_{i1}(\tau) & \text{for all } \tau < |c_f_{i1}| \\ \emptyset & \text{for all } \tau, |c_f_{i1}| \leq \tau < |c_f_i| \end{cases}
\]
if $|c_{f1}| < \infty$ then $s_1(x) = \begin{cases} s_{1i}(x) & \text{if } x \in \text{var}(S_i) \\ s_0(x) & \text{if } x \notin \text{var}(S_i \| S_2) \end{cases}$

By (D.4) we obtain $[C_i]_7(s_{1i}, c_{f0} \wedge c_{f_{i1}})$ and if $|c_{f_{i1}}| < \infty$ then $[q_i]_7(s_{1i}, c_{f0} \wedge c_{f_{i1}})$.

Define $\hat{\gamma} = (\gamma : t_1 \mapsto |c_{f0}| + |c_{f_{i1}}|, t_2 \mapsto |c_{f0}| + |c_{f_{12}}|)$. Then, using (D.8),

$[C_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge c_{f_{i1}})$ and if $|c_{f_{i1}}| < \infty$ then $[q_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge c_{f_{i1}})$.

Using Lemma D.2.4, we obtain $[C_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}]_{\text{dch}(S_i)})$ and

(since $|c_{f1}| < \infty$ implies $|c_{f_{i1}}| < \infty \land |c_{f_{12}}| < \infty$)

if $|c_{f1}| < \infty$ then $[q_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}]_{\text{dch}(S_i)})$.

By (D.7) this leads to $[C_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$ and,

if $|c_{f1}| < \infty$ then $[q_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$.

Since $\text{var}(C_i) = \emptyset$, $\text{var}(q_i) \subseteq \text{var}(S_i)$ and $s_{1i}(x) = s_1(x)$ for $x \in \text{var}(S_i)$, we obtain

$[C_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$ and, if $|c_{f1}| < \infty$ then $[q_i[t_i/time]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$.

Furthermore, $[\text{time} = \max(t_1, t_2)]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$ and

$[\bigwedge_{i=1}^2 \text{no dch}(S_i) \text{ during } [t_i, \text{time}]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$. Thus

$[\exists t_1, t_2 : \text{time} = \max(t_1, t_2) \land \bigwedge_{i=1}^2 \text{C}[i[t_i/time]] \land \text{no dch}(S_i) \text{ during } [t_i, \text{time}]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$

and if $|c_{f0}\sigma| < \infty$ then $[\exists t_1, t_2 : \text{time} = \max(t_1, t_2) \land \bigwedge_{i=1}^2 q_i[t_i/time]$

$\land \text{no dch}(S_i) \text{ during } [t_i, \text{time}]]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$.

Hence, by (D.5) and (D.6), $[C]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$ and if $|c_{f0}\sigma| < \infty$ then $[q]_7(\hat{s}_{1i}, c_{f0} \wedge [c_{f1}])$. 


Appendix E

Soundness of the Proof Systems in Chapter 5

E.1 Soundness of the Proof System in Section 5.5.2

Well-Formedness Axiom

The soundness of the extended Well-Formedness Axiom can be proved as follows.

Consider any environment $\gamma$ and $\sigma \in M(P)$. Then Lemma 5.4.3 implies that

for all $\tau < |\sigma|$, $p_1 \in \sigma(\tau).exec$ and $p_2 \in \sigma(\tau).req$ imply $p_1 \geq p_2$.

Thus $\delta_1 \in \sigma(\tau).req$ and $\delta_2 \in \sigma(\tau).exec$ imply $\delta_1 \leq \delta_2$.

Hence $\delta_1 \in (\sigma \uparrow \tau)(0).req$ and $\delta_2 \in (\sigma \uparrow \tau)(0).exec$ imply $\delta_1 \leq \delta_2$.

Then, $\delta_1 \in S(req)(\gamma \uparrow \tau, \sigma \uparrow \tau)$ and $\delta_2 \in S(exec)(\gamma \uparrow \tau, \sigma \uparrow \tau)$ imply $\delta_1 \leq \delta_2$, for all $\tau < |\sigma|$.

Since, for all $\tau \geq |\sigma|$, $S(req)(\gamma \uparrow \tau, \sigma \uparrow \tau) = S(exec)(\gamma \uparrow \tau, \sigma \uparrow \tau) = \emptyset$, we obtain

for all $\tau$: if $\delta_1 \in S(req)(\gamma \uparrow \tau, \sigma \uparrow \tau)$ and $\delta_2 \in S(exec)(\gamma \uparrow \tau, \sigma \uparrow \tau)$ then $\delta_1 \leq \delta_2$.

Thus, $(\sigma \uparrow \tau, \gamma \uparrow \tau) \models req \leq exec$, and hence $(\sigma, \gamma) \models \Box(req \leq exec)$.

Atomic, Delay, Send, and Receive

First consider $\sigma \in SEQ(Request, Execute(K))$. Then

- either $|\sigma| = \infty$ and for all $\tau \in TIME$: $\sigma(\tau).req = \{0\}$ and $\sigma(\tau).comm = \sigma(\tau).exec = \emptyset$. Thus, for all $\tau \in TIME$: $(\sigma \uparrow \tau)(0).req = \{0\}$ and $(\sigma \uparrow \tau)(0).comm = (\sigma \uparrow \tau)(0).exec = \emptyset$. Hence, for any $\gamma$ and any $cset \subseteq DCHAN$,

$$\langle \sigma, \gamma \rangle \models \Box(req = \{0\} \land exec = \emptyset \land \text{noact}(cset)),$$

and thus

$$\langle \sigma, \gamma \rangle \models Request(cset) \cup Execute(cset, K).$$

- or there exists a $\tau \in TIME$, such that for all $\tau_1 < \tau$: $\sigma(\tau_1).req = \{0\}$ and $\sigma(\tau_1).comm = \sigma(\tau_1).exec = \emptyset$, $\sigma(\tau).exec = \{0\}$, for all $\tau_2, 0 < \tau_2 < K$:
\[
\sigma(\tau + \tau_2).\text{exec} = \{\infty\}, \text{for all } \tau_3 < K: \sigma(\tau + \tau_3).\text{comm} = \sigma(\tau + \tau_3).\text{req} = \emptyset, \text{and } |\sigma| = \tau + K. \text{ Consider any } \gamma \text{ and any } \text{cset } \subseteq \text{DCHAN}.
\]

Then, for all \(\tau_1 < \tau\): \((\sigma \uparrow \tau_1, \gamma \uparrow \tau_1) \models \text{req} = \{0\} \land \text{exec} = \emptyset \land \text{noact(\text{cset})},\) and thus, for all \(\tau_1 < \tau: (\sigma \uparrow \tau_1, \gamma \uparrow \tau_1) \models \text{Request(\text{cset})}.
\]

Furthermore, \((\sigma \uparrow \tau, \gamma \uparrow \tau) \models \text{req} = \emptyset \land \text{exec} = \{0\} \land \text{noact(\text{cset})},\) for all \(\tau_2, 0 < \tau_2 < K: ((\sigma \uparrow \tau_1) \uparrow \tau_2, (\gamma \uparrow \tau_1) \uparrow \tau_2) \models \text{req} = \emptyset \land \text{exec} = \{\infty\} \land \text{noact(\text{cset})},\) and \(|\sigma| = \tau = K. \text{ Hence } (\sigma \uparrow \tau, \gamma \uparrow \tau) \models \text{Execute(cset, } K),\) and thus \((\sigma, \gamma) \models \text{Request(\text{cset})} \cup \text{Execute(\text{cset, } K)}.
\]

Observe that if \(\sigma \in \text{Delay(d), } \sigma \in \text{SEQ(WaitSend(c), Comm(c))},\) or \(\sigma \in \text{SEQ(WaitRec(c), Comm(c))}\) then \(\sigma(\tau).\text{req} = \sigma(\tau).\text{exec} = \emptyset,\) for all \(\tau < |\sigma|.
\]

Hence, for any \(\gamma\) and any \(\tau < |\sigma|, (\sigma \uparrow \tau, \gamma \uparrow \tau) \models \text{req} = \text{exec} = \emptyset.\) Since, for all \(\tau \geq |\sigma|, (\sigma \uparrow \tau, \gamma \uparrow \tau) \models \text{req} = \text{exec} = \emptyset,\) this leads to \((\sigma, \gamma) \models \Box (\text{req} = \text{exec} = \emptyset).
\]

Now the soundness of the axioms for atomic(d), delay d, c! and c? follows from the observations above and the soundness of the axioms from Section 3.3 (see the proof in Appendix B.1).

**Guarded Command**

Similar to the proof above, soundness of the Guarded Command Rule is easily proved from the observations above and the soundness of the Rule for Guarded Command with Delay from Section 3.3.

**Priority Assignment**

Suppose \(\models S \text{ sat } \varphi\) and
\[
\models \varphi[r/\text{req}, e/\text{exec}] \land \Box \big((r = \{0\} \rightarrow \text{req} = \{p\}) \land (r \neq \{0\} \rightarrow \text{req} = r)\big) \land \Box \big((e = \{0\} \rightarrow \text{exec} = \{p\}) \land (e \neq \{0\} \rightarrow \text{exec} = e)\big) \rightarrow \varphi_1.
\]

Consider any \(\gamma\) and \(\sigma \in \mathcal{M}(\text{prio } p (S)).\) Then there exists \(\sigma_1 \in \mathcal{M}(S)\) such that \(\sigma = \sigma_1[p/0].\) Thus \(|\sigma| = |\sigma_1|\) and for all \(\tau < |\sigma|,
\[
\sigma(\tau).\text{comm} = \sigma(\tau).\text{comm} \\
\sigma(\tau).\text{req} = \{p | p' \in \sigma_1(\tau).\text{req} \land p' \neq 0\} \cup \{p | 0 \in \sigma_1(\tau).\text{req}\} \\
\sigma(\tau).\text{exec} = \{p | p' \in \sigma_1(\tau).\text{exec} \land p' \neq 0\} \cup \{p | 0 \in \sigma_1(\tau).\text{exec}\}.
\]

By \(\models S \text{ sat } \varphi\) we obtain \((\sigma_1, \gamma) \models \varphi.\) Define \(\hat{\gamma}\) such that
\[
\hat{\gamma}(r)(\tau) = \begin{cases} 
\sigma_1(\tau).\text{req} & \text{for } \tau < |\sigma_1| \\
\emptyset & \text{for } \tau \geq |\sigma_1| 
\end{cases}
\]

and \(\hat{\gamma}(e)(\tau) = \begin{cases} 
\sigma_1(\tau).\text{exec} & \text{for } \tau < |\sigma_1| \\
\emptyset & \text{for } \tau \geq |\sigma_1| 
\end{cases}\)

and \(\hat{\gamma}(u) = \gamma(u),\) for any other \(u \in \text{SPVAR}.\) Then \(\langle \sigma_1, \hat{\gamma} \rangle \models \varphi[r/\text{req}, e/\text{exec}].\)

By \(\sigma_1.\text{comm} = \sigma.\text{comm}\) this leads to \(\langle \sigma, \hat{\gamma} \rangle \models \varphi[r/\text{req}, e/\text{exec}].\) Since the syntactic
restrictions for programs require that $S$ does not contain any parallel composition, we can easily see that $0 \in \sigma_1(\tau).req$ implies $\sigma_1(\tau).req = \{0\}$.

Thus, for all $\tau < |\sigma|$, $\sigma_1(\tau).req = \{0\}$ implies $\sigma(\tau).req = \{p\}$ and $\sigma_1(\tau).req \neq \{0\}$ implies $\sigma(\tau).req = \sigma_1(\tau).req$.

Hence, for all $\tau < |\sigma|$, $\hat{\gamma}(\tau)(\tau) = \{0\}$ implies $\sigma(\tau).req = \{p\}$ and $\hat{\gamma}(\tau)(\tau) \neq \{0\}$ implies $\sigma(\tau).req = \hat{\gamma}(\tau)(\tau)$.

Then, for all $\tau < |\sigma|$, $\langle \sigma \uparrow \tau, \gamma \uparrow \tau \rangle \vdash (r = \{0\} \rightarrow req = \{p\}) \land (r \neq \{0\} \rightarrow req = r)$.

Since, for all $\tau \geq |\sigma|$, $\langle \sigma \uparrow \tau, \gamma \uparrow \tau \rangle \vdash req = \emptyset \land r = \emptyset$, we obtain for all $\tau \in TIME$: $\langle \sigma \uparrow \tau, \gamma \uparrow \tau \rangle \vdash (r = \{0\} \rightarrow req = \{p\}) \land (r \neq \{0\} \rightarrow req = r)$.

Hence $\langle \sigma, \gamma \rangle \vdash \square ((r = \{0\} \rightarrow req = \{p\}) \land (r \neq \{0\} \rightarrow req = r))$.

Similarly, $\langle \sigma, \gamma \rangle \vdash \square ((e = \{0\} \rightarrow exec = \{p\}) \land (e \neq \{0\} \rightarrow exec = e))$.

Together with $\langle \sigma, \hat{\gamma} \rangle \vdash \varphi[r/req, e/exec]$ this leads to $\langle \sigma, \hat{\gamma} \rangle \vdash \varphi_1$.

Since $r$ and $e$ are fresh logical variables we obtain $\langle \sigma, \gamma \rangle \vdash \varphi_1$.

**Parallel Composition**

Suppose $\models P_1 \text{ sat } \varphi_1$, $\models P_2 \text{ sat } \varphi_2$, and

$\models (\varphi_1[r_1/req, e_1/exec] \land (\varphi_2[r_2/req, e_2/exec] \land \square (\text{noact}(dch(P_2)) \land r_2 = e_2 = \emptyset)) \lor$

$(\varphi_2[r_2/req, e_2/exec] \land (\varphi_1[r_1/req, e_1/exec] \land \square (\text{noact}(dch(P_1)) \land r_1 = e_1 = \emptyset))) \land$

$\square (e_1 = \emptyset \lor e_2 = \emptyset) \rightarrow \varphi[r_1 \cup r_2/req, e_1 \cup e_2/exec]$

Let $\gamma$ be any environment. Consider $\sigma \in \mathcal{M}(P_1 \parallel P_2)$. Then

dch(\sigma) \subseteq dch(P_1) \cup dch(P_2), and for $i = 1, 2$ there exist $\sigma_i \in \mathcal{M}(P_i)$ such that

$|\sigma| = \max(|\sigma_1|, |\sigma_2|)$, and for all $\tau < |\sigma|$: $\sigma_i[dch(P_i)](\tau).comm = \sigma_i^+(\tau).comm$, $\sigma_i(\tau).req = \sigma_i^+(\tau).req \cup \sigma_i^+(	au).req$, $\sigma_i(\tau).exec = \sigma_i^+(\tau).exec \cup \sigma_i^+(\tau).exec$, and $\sigma_i^+(\tau).exec = \emptyset \lor \sigma_i^+(\tau).exec = \emptyset$.

Suppose $|\sigma_1| \leq |\sigma_2|$, and thus $|\sigma| = |\sigma_2|$.

Define $\hat{\gamma}$ such that, for $i = 1, 2$, $\hat{\gamma}(\tau)(\tau) = \sigma_i^+(\tau).req$, $\hat{\gamma}(\tau)(\tau) = \sigma_i^+(\tau).exec$, and $\hat{\gamma}(u) = \gamma(u)$, for any other $u \in SPVAR$.

By $\models P_i \text{ sat } \varphi_i$, we obtain $\langle \sigma_i, \gamma \rangle \vdash \varphi_i$, and thus $\langle \sigma_1, \hat{\gamma} \rangle \models \varphi_1[r_1/req, e_1/exec]$.

Similarly to the soundness proof of the Rule for Parallel Composition from Section 3.3, given in Appendix B.1, we can prove

$\langle \sigma, \hat{\gamma} \rangle \models (\varphi_1[r_1/req, e_1/exec] \land (\varphi_2[r_2/req, e_2/exec] \land \square (\text{noact}(dch(P_2)) \land r_2 = e_2 = \emptyset)) \lor$

$\varphi_2[r_2/req, e_2/exec] \land (\varphi_1[r_1/req, e_1/exec] \land \square (\text{noact}(dch(P_1)) \land r_1 = e_1 = \emptyset)))$.

Further, from $\sigma_i^+(\tau).exec = \emptyset \lor \sigma_i^+(\tau).exec = \emptyset$ we obtain

$\langle \sigma, \hat{\gamma} \rangle \models \square (e_1 = \emptyset \lor e_2 = \emptyset)$.

Hence, the assumption of the rule leads to $\langle \sigma, \hat{\gamma} \rangle \models \varphi[r_1 \cup r_2/req, e_1 \cup e_2/exec]$.

Since $\sigma(\tau).req = \sigma_1^+(\tau).req \cup \sigma_2^+(\tau).req = \hat{\gamma}(r_1)(\tau) \cup \hat{\gamma}(r_2)(\tau)$ and, similarly,
\(\sigma(\tau).\text{exec} = \gamma(e_1)(\tau) \cup \gamma(e_2)(\tau)\), we obtain \(\langle \sigma, \gamma \rangle \models \varphi\).

**Processor Closure**

Suppose \(\models S \models \varphi_1\), and \(\models \varphi_1 \land \Box (\text{exec} = \emptyset \rightarrow \text{req} = \emptyset) \rightarrow \varphi_2\).

Consider \(\sigma \in M(\ll S \gg)\). Then there exists a \(\sigma_1 \in M(S)\) such that \(|\sigma| = |\sigma_1|\), and for all \(\tau < |\sigma|\): \(\sigma(\tau).\text{comm} = \sigma_1(\tau).\text{comm}\), \(\sigma_1(\tau).\text{exec} = \emptyset = \sigma_1(\tau).\text{req} = \emptyset\). Let \(\gamma\) be any arbitrary environment. By \(\models S \models \varphi_1\) we obtain \(\langle \sigma_1, \gamma \rangle \models \varphi_1\). Since, for \(\tau \geq |\sigma_1|\), \(\langle \sigma_1 \uparrow \tau, \gamma \uparrow \tau \rangle \models \text{exec} = \emptyset \land \text{req} = \emptyset\), we have \(\langle \sigma_1, \gamma \rangle \models \Box (\text{exec} = \emptyset \land \text{req} = \emptyset)\).

Thus \(\langle \sigma_1, \gamma \rangle \models \varphi_2\). Since \(\text{req}\) and \(\text{exec}\) do not occur in \(\varphi_2\), we obtain \(\langle \sigma, \gamma \rangle \models \varphi_2\).

**E.2 Soundness of the Proof System in Section 5.6.2**

**Well-Formedness Axiom**

The soundness of the extended Well-Formedness Axiom can be proved as follows.

Consider any environment \(\gamma\) and \(\sigma \in M(P)\). Then Lemma 5.4.3 implies that for all \(\tau < |\sigma|\), \(p_1 \in \sigma(\tau).\text{exec}\) and \(p_2 \in \sigma(\tau).\text{req}\) imply \(p_1 \geq p_2\).

Thus \(\delta_1 \in \sigma(\tau).\text{req}\) and \(\delta_2 \in \sigma(\tau).\text{exec}\) imply \(\delta_1 \leq \delta_2\).

Hence, \(\delta_1 \in \mathcal{P}(\text{req}(\tau))(\gamma, \sigma)\) and \(\delta_2 \in \mathcal{P}(\text{exec}(\tau))(\gamma, \sigma)\) imply \(\delta_1 \leq \delta_2\).

Thus, for all \(\gamma\), for all \(\tau < |\sigma|\), \([\text{req}(\tau) \leq \text{exec}(\tau)]\gamma\sigma\).

Then \(\forall t < \text{time} : \text{req}(t) \leq \text{exec}(t)\gamma\sigma\) and thus \(\forall t < \text{time} : \text{Prio}(t)\gamma\sigma\).

**Atomic**

Suppose \(\models (\exists t \geq t_0 : \text{request during } [t_0, t) \land \text{execute during } [t, t + d) \land \text{time} = t + d) \rightarrow C\).

Assume \([\text{time} = t_0] \gamma \hat{\sigma}\), then \(\gamma(t_0) = |\hat{\sigma}|\).

Consider \(\sigma \in M(\text{atomic}(d)) = \text{SEQ(Request, Execute(d))}\).

Then there exists a \(\tau \in \text{TIME} \cup \{\infty\}\) such that for all \(\tau_1 < \tau\): \(\sigma(\tau_1).\text{req} = \{0\}\) and \(\sigma(\tau_1).\text{exec} = \emptyset\), if \(\tau < \infty\) then \(\sigma(\tau).\text{exec} = \{0\}\), for all \(\tau_2, \tau < \tau_2 < \tau + d\): \(\sigma(\tau_2).\text{exec} = \{\infty\}\), for all \(\tau_3, \tau \leq \tau_3 < \tau + d\): \(\sigma(\tau_3).\text{req} = \emptyset\), and \(|\sigma| = \tau + d\).

Let \(\gamma\) be any environment. Then there exists a \(\tau \in \text{TIME} \cup \{\infty\}\) such that
\([\text{req}[0, \tau) = \{0\} \land \text{exec}[0, \tau) = \emptyset]\gamma\sigma\), \([\{\tau < \infty \rightarrow \text{exec}(\tau) = \{0\}]\gamma\sigma\),
\([\text{exec}(\tau, \tau + d) = \{\infty\} \land \text{req}[\tau, \tau + d) = \emptyset]\gamma\sigma\), and \([\text{time} = \tau + d]\gamma\sigma\).

By the definition of the abbreviations
\([\text{request during } [0, \tau]]\gamma\hat{\sigma}\), \([\text{execute during } [\tau, \tau + d]]\gamma\hat{\sigma}\), and \([\text{time} = \tau + d]\gamma\hat{\sigma}\).

Since \(\gamma(t_0) = |\hat{\sigma}|\), \([\text{request during } [t_0, t_0 + \tau]]\gamma\hat{\sigma}\),
\([\text{execute during } [t_0 + \tau, t_0 + \tau + d]]\gamma\hat{\sigma}\), and \([\text{time} = t_0 + \tau + d]\gamma\hat{\sigma}\).
Hence $\exists t \geq t_0 : \text{request during } [t_0, t) \land \text{execute during } [t, t + d) \land \text{time} = t + d] \gamma \delta \sigma$.

By the assumption of the rule we obtain $[C] \gamma \delta \sigma$, and then $|\sigma| < \infty$ leads to $[C \land \text{time} < \infty] \gamma \delta \sigma$.

**Delay**

Suppose $\models (\exists t \geq t_0 : \text{request during } [t_0, t) \land \text{execute during } [t, t + K_e) \land$

$\neg \text{ReqExec during } [t + K_e, t + K_e + d) \land \text{time} = t + K_e + d) \rightarrow C$.

Assume $[\text{time} = t_0] \gamma \delta$, then $\gamma(t_0) = |\delta|$.

Consider $\sigma \in \mathcal{M}(\text{delay } d) = \text{SEQ}(\text{Request}, \text{Execute}(K_e), \text{Delay}(d))$. Then there exists a $\tau \in \text{TIME} \cup \{\infty\}$ such that

for all $\tau_1 < \tau$: $\sigma(\tau_1).\text{req} = \{0\}$ and $\sigma(\tau_1).\text{exec} = \emptyset$,

if $\tau < \infty$ then $\sigma(\tau).\text{exec} = \{0\}$, for all $\tau_2, \tau < \tau < \tau + K_e: \sigma(\tau_2).\text{exec} = \{\infty\}$,

for all $\tau_3, \tau \leq \tau_3 < \tau + K_e: \sigma(\tau_3).\text{req} = \emptyset$,

for all $\tau_4, \tau + K_e \leq \tau_4 < \tau + K_e + d: \sigma(\tau_4).\text{req} = \sigma(\tau_4).\text{exec} = \emptyset$,

and $|\sigma| = \tau + K_e + d$. Thus

$\exists t \geq t_0 : \text{req}(t_0, t) = \{0\} \land \text{exec}(t_0, t) = \emptyset \land$

$(t < \infty \rightarrow \text{exec}(t) = \{0\}) \land \text{exec}(t, t + K_e) = \{\infty\} \land \text{req}(t, t + K_e) = \emptyset \land$

$\text{req}(t + K_e, t + K_e + d) = \emptyset \land \text{exec}(t + K_e, t + K_e + d) = \emptyset \land \text{time} = t + K_e + d] \gamma \delta \sigma$.

By the definition of the abbreviations

$\exists t \geq t_0 : \text{request during } [t_0, t) \land \text{execute during } [t, t + K_e) \land$

$\neg \text{ReqExec during } [t + K_e, t + K_e + d) \land \text{time} = t + K_e + d] \gamma \delta \sigma$.

Hence, by the assumption of the rule, $[C] \gamma \delta \sigma$, and if $|\sigma| < \infty$ then $[C \land \text{time} < \infty] \gamma \delta \sigma$.

**Send and Receive**

We prove the soundness of the Send Rule. The soundness of the Receive Rule can be proved similarly. Suppose

$\models (\exists t_1 \geq t_0 : \text{ReqExec}(t_0, \{c!, c\}, t_1) \land \exists t \geq t_1 : \text{wait to c! at } t_1 \text{ until comm at } t \land$

$\neg \text{ReqExec during } [t_1, t + K_e) \land \text{time} = t + K_e) \rightarrow C$.

Assume $[\text{time} = t_0] \gamma \delta$, then $\gamma(t_0) = |\delta|$.

Consider $\sigma \in \mathcal{M}(c!) = \text{SEQ}(\text{Request}, \text{Execute}(K_e), \text{WaitSend}(c), \text{Comm}(c))$. Then there exists a $\tau \in \text{TIME} \cup \{\infty\}$ such that for all $\tau_1 < \tau$: $\sigma(\tau_1).\text{req} = \{0\}$ and $\sigma(\tau_1).\text{exec} = \emptyset$,

if $\tau < \infty$ then $\sigma(\tau).\text{exec} = \{0\}$, for all $\tau_2, \tau < \tau < \tau + K_e: \sigma(\tau_2).\text{exec} = \{\infty\}$,

for all $\tau_3, \tau \leq \tau_3 < \tau + K_e: \sigma(\tau_3).\text{comm} = \sigma(\tau_3).\text{req} = \emptyset$,

and there exists a $\hat{\tau} \in \text{TIME} \cup \{\infty\}$ such that $\hat{\tau} \geq \tau + K_e$,

for all $\tau_4, \tau + K_e \leq \tau_4 < \hat{\tau}$: $\sigma(\tau_4).\text{comm} = \{c!\}$, $\sigma(\tau_4).\text{req} = \sigma(\tau_4).\text{exec} = \emptyset$. 


for all $\tau_6, \hat{\tau} \leq \tau_5 < \hat{\tau} + K_c$: $\sigma(\tau_5).comm = \{ c \}, \sigma(\tau_5).req = \sigma(\tau_5).exec = \emptyset,$ and $|\sigma| = \hat{\tau} + K_c$. Then

$$\exists t_1 \geq t_0 \exists t_3 : t_1 = t_3 + K_c \land \text{request during } [t_0, t_3) \land \text{execute during } [t_3, t_1) \land$$

$$\neg \{ c_1, c \} \land \exists t \geq t_1 : \text{wait to } c_1 \text{ at } t_1 \text{ until } \text{comm at } t \land$$

$$\text{req}[t_1, t + K_c) = \emptyset \land \text{exec}[t_1, t + K_c) = \emptyset \land \text{time} = t + K_c \gamma_\delta \sigma.$$  

By definition of the abbreviations

$$\exists t_1 \geq t_0 : \text{ReqExec}(t_0, \{ c_1, c \}, t_1) \land \exists t \geq t_1 : \text{wait to } c_1 \text{ at } t_1 \text{ until } \text{comm at } t \land$$

$$\neg \text{ReqExec during } [t_1, t + K_c) \land \text{time} = t + K_c \gamma_\delta \sigma.$$  

Hence, by the assumption of the rule, $[C] \gamma_\delta \sigma$, and if $|\sigma| < \infty$ then $[C \land \text{time} < \infty] \gamma_\delta \sigma$.

**Guarded Command**

To show the soundness of the rules for guarded commands, first observe that a model from the semantics of a guarded command is the concatenation of a model from the set $\text{SEQ} (\text{Request, Execute}(K_c))$ and a model in which the req- and exec-fields are empty and the comm-field corresponds to the semantics from Chapter 3. Also the rules for guarded commands from Section 5.6 are essentially the same as the rules from Section 3.4, with the addition of a requesting and executing period (represented by the assertion $\text{ReqExec}(t_0, \{ c_1, \ldots, c_n \}, t_1)$) and empty req- and exec-fields during subsequent periods. The correspondence between the set $\text{SEQ} (\text{Request, Execute}(K_c))$ and the assertion $\text{ReqExec}(t_0, \{ c_1, \ldots, c_n \}, t_1)$ can be shown similar to the proofs given above.

**Priority Assignment**

Suppose $\models C : \{ \hat{p} \} S \{ q \},$

$\models C[\hat{p}/\text{req}, e/\text{exec}] \land \text{ReplacePrio}(t_0, p) \rightarrow C_1,$ and

$\models q[\hat{p}/\text{req}, e/\text{exec}] \land \text{ReplacePrio}(t_0, p) \rightarrow q_1.$  

Assume $[\hat{p} \land \text{time} = t_0] \gamma_\delta \sigma$, then $\gamma(t_0) = |\hat{\delta}|$. Consider $\sigma \in M(prior p (S))$. Then there exists $\sigma_1 \in M(S)$ such that $\sigma = \sigma_1[p/0]$. Thus $|\sigma| = |\sigma_1|$ and for all $\tau < |\sigma|,$

$$\sigma(\tau).comm = \sigma(\tau).comm$$

$$\sigma(\tau).req = \{ p \mid p \in \sigma_1(\tau).req \land p \neq 0 \} \cup \{ p \mid 0 \in \sigma_1(\tau).req \}$$

$$\sigma(\tau).exec = \{ p \mid p \in \sigma_1(\tau).exec \land p \neq 0 \} \cup \{ p \mid 0 \in \sigma_1(\tau).exec \}.$$  

Since $[\hat{p}] \gamma_\delta \sigma$ and $\models C : \{ \hat{p} \} S \{ q \}$, we obtain $[C] \gamma_\delta \sigma_1$, and if $|\sigma_1| < \infty$ then $[q] \gamma_\delta \sigma_1$.

Define $\hat{\gamma}$ such that

$$\hat{\gamma}(r)(\tau) = \begin{cases} 
\sigma_1(\tau).\text{req} & \text{for } \tau < |\sigma_1| \\
\emptyset & \text{for } \tau \geq |\sigma_1| 
\end{cases}$$

$$\hat{\gamma}(e)(\tau) = \begin{cases} 
\sigma_1(\tau).\text{exec} & \text{for } \tau < |\sigma_1| \\
\emptyset & \text{for } \tau \geq |\sigma_1| 
\end{cases}$$

and $\hat{\gamma}(u) = \gamma(u)$, for any other $u \in \text{SPVAR}.$
Then \([C[r/req,e/exec]]\hat{\sigma}\), and if \(|\sigma_1| < \infty\) then \([q[r/req,e/exec]]\hat{\sigma}\).

By \(\sigma_{1\.comm} = \sigma\.comm\) this leads to \([C[r/req,e/exec]]\hat{\sigma}\), and if \(|\sigma| = |\sigma_1| < \infty\) then \([q[r/req,e/exec]]\hat{\sigma}\).

Since the syntactic restrictions for programs require that \(S\) does not contain any parallel composition, we can easily see that \(0 \in \sigma_1(\tau).req\) implies \(\sigma_1(\tau).req = \{0\}\).

Thus, for all \(\tau < |\sigma|, \sigma_1(\tau).req = \{0\}\) implies \(\sigma(\tau).req = \{p\}\) and \(\sigma_1(\tau).req \neq \{0\}\) implies \(\sigma(\tau).req = \sigma_1(\tau).req\).

Hence, for all \(\tau < |\sigma|, \hat{\gamma}(\tau)(\tau) = \{0\}\) implies \(\sigma(\tau).req = \{p\}\) and \(\hat{\gamma}(\tau)(\tau) \neq \{0\}\) implies \(\sigma(\tau).req = \hat{\gamma}(\tau)(\tau)\).

Then, for all \(\tau < |\sigma|, \hat{\gamma}(\tau)(\tau) = \{0\}\) implies \(\sigma(\tau).req = \{p\}\) and \(\hat{\gamma}(\tau)(\tau) \neq \{0\}\) implies \(\sigma(\tau).req = \hat{\gamma}(\tau)(\tau)\).

Since \(\gamma(t_0) = |\hat{\sigma}|\) this leads to
\([Vt,t_0 \leq t < time : (r(t) = \{0\} \rightarrow req(t) = \{p\}) \wedge (r(t) \neq \{0\} \rightarrow req(t) = \{\tau\})]\hat{\sigma}\).

Similarly,
\([Vt,t_0 \leq t < time : (e(t) = \{0\} \rightarrow exec(t) = \{p\}) \wedge (e(t) \neq \{0\} \rightarrow exec(t) = e(t))]\hat{\sigma}\).

Thus \([ReplacePrio(t_0,p)]\hat{\sigma}\).

Hence the assumptions of the rule lead to \([C_1]\hat{\sigma}\), and if \(|\sigma| < \infty\) then \([q_1]\hat{\sigma}\).

Since \(r\) and \(e\) are fresh, we obtain \([C_1]\hat{\sigma}\), and if \(|\sigma| < \infty\) then \([q_1]\hat{\sigma}\).

**Parallel Composition**

Suppose \(\models C_i : \{p_i\} P_i \{q_i\}, i = 1,2,\)
\(\models \exists t_1,t_2 : time = \max(t_1,t_2) \wedge \bigwedge_{i=1}^{2} C_i[t_i/time,r_i/req,e_i/exec] \wedge\)
not active \(dch(P_i)\) during \([t_i,\text{time}]\) \(\forall t,t_0 \leq t < \text{time} :\)
\((e_1(t) = \emptyset \vee e_2(t) = \emptyset) \wedge (r(t) = r_1(t) \cup r_2(t)) \wedge (exec(t) = e_1(t) \cup e_2(t)) \rightarrow C\)
\(\models \exists t_1,t_2 : time = \max(t_1,t_2) \wedge \bigwedge_{i=1}^{2} q_i[t_i/time,r_i/req,e_i/exec] \wedge\)
not active \(dch(P_i)\) during \([t_i,\text{time}]\) \(\forall t,t_0 \leq t < \text{time} :\)
\((e_1(t) = \emptyset \vee e_2(t) = \emptyset) \wedge (r(t) = r_1(t) \cup r_2(t)) \wedge (exec(t) = e_1(t) \cup e_2(t)) \rightarrow q_i\).

Assume \([p_1 \wedge p_2 \wedge time = t_0]\hat{\gamma}\), then \(\gamma(t_0) = |\hat{\sigma}|\). Consider \(\sigma \in \mathcal{M}(P_1 \parallel P_2)\).

Then \(dch(\sigma) \subseteq dch(P_1) \cup dch(P_2)\), and for \(i = 1,2\) there exist \(\sigma_i \in \mathcal{M}(P_i)\) such that
\(\sigma = \max(|\sigma_1|,|\sigma_2|)\), and for all \(\tau < |\sigma|: [\sigma]_{dch(P_i)(\tau)}.\text{comm} = \sigma_i^+(\tau).\text{comm},\)
\(\sigma(\tau).\text{req} = \sigma_i^+(\tau).\text{req} \cup \sigma_i^+(\tau).\text{req}, \sigma(\tau).\text{exec} = \sigma_i^+(\tau).\text{exec} \cup \sigma_i^+(\tau).\text{exec} , \text{and} \)
\(\sigma_i^+(\tau).\text{exec} = \emptyset \vee \sigma_i^+(\tau).\text{exec} = \emptyset\)

By \(\models C_i : \{p_i\} P_i \{q_i\}\) we obtain \([C_i]\hat{\sigma}\), and if \(|\sigma| < \infty\) then \([q_i]\hat{\sigma}\).

Define \(\hat{\gamma}\) such that, for \(i = 1,2,\)
\(\hat{\gamma}(r_i)(\tau) = \sigma_i^+(\tau).\text{req}, \hat{\gamma}(e_i)(\tau) = \sigma_i^+(\tau).\text{exec} , \text{and} \)
\(\hat{\gamma}(u) = \gamma(u)\), for any other \(u \in \text{SPVAR}\). Then \([C_i[t_i/time,r_i/req,e_i/exec]]\hat{\sigma}\), and if \(|\sigma| < \infty\) then \([q_i[t_i/time,r_i/req,e_i/exec]]\hat{\sigma}\).

Since these assertions now only refer to the \(\text{comm}\)-field of \(\hat{\sigma}\), we can use the proof from
Appendix C.1 for the Parallel Composition Rule of Section 3.4 to obtain
\[ C_1[t_i/time, r_i/req, e_i/exec] \Rightarrow \sigma, \text{ and if } |\sigma| < \infty \text{ (and hence } |\sigma| < \infty) \text{ then} \]
\[ q_i[t_i/time, r_i/req, e_i/exec] \Rightarrow \sigma. \]
Also \[ time = max(t_1, t_2) \Rightarrow \sigma \], and
\[ \text{[no dch}(P_i) \text{ during } [t_i, time) \land req[t_i, time) \land exec[t_i, time)] \Rightarrow \sigma, \]
and thus \[ \text{[not active dch}(P_i) \text{ during } [t_i, time)] \Rightarrow \sigma. \]
Furthermore, from \( \sigma^+(\tau).exec = \emptyset \lor \sigma^+(\tau).exec = \emptyset \) we obtain
\[ \forall t, t_0 \leq t < time : (e_1(t) = \emptyset \lor e_2(t) = \emptyset) \Rightarrow \sigma. \]
Since \( \sigma(\tau).req = \sigma^+(\tau).req \lor \sigma^+(\tau).req = \gamma(r_1)(\tau) \lor \gamma(r_2)(\tau) \), we have
\[ \forall t, t_0 \leq t < time : req(t) = r_1(t) \lor r_2(t) \Rightarrow \sigma. \]
Similarly, \[ \forall t, t_0 \leq t < time : exec(t) = e_1(t) \lor e_2(t) \Rightarrow \sigma. \]
Hence, the assumptions of the rule lead to \[ C \Rightarrow \sigma, \text{ and if } |\sigma| < \infty \text{ then } q \Rightarrow \sigma. \]
Since \( t_i, r_i, \) and \( e_i \) are fresh logical variables,
we obtain \[ C \Rightarrow \sigma, \text{ and if } |\sigma| < \infty \text{ then } q \Rightarrow \sigma. \]

**Processor Closure**

Suppose \( \models C \models \{ p \} S \{ q \}, \]
\[ \models C \land (\forall t, t_0 \leq t < time : exec(t) = \emptyset \rightarrow req(t) = \emptyset) \rightarrow C_1, \text{ and} \]
\[ \models q \land (\forall t, t_0 \leq t < time : exec(t) = \emptyset \rightarrow req(t) = \emptyset) \rightarrow q_1. \]
Assume \( p \land time = t_0 \Rightarrow \gamma \sigma, \) then \( \gamma(t_0) = \sigma. \) Consider \( \sigma \in M(S) \).
Then there exists a \( \sigma_1 \in M(S) \) such that \( |\sigma| = |\sigma_1|, \) and for all \( \tau < |\sigma|: \)
\( \sigma(\tau).comm = \sigma_1(\tau).comm, \sigma_1(\tau).exec = \emptyset \rightarrow \sigma_1(\tau).req = \emptyset. \)
Since \( p \Rightarrow \sigma \) and \( \models C \models \{ p \} S \{ q \}, \) we obtain \[ C \Rightarrow \sigma, \text{ and if } |\sigma| < \infty \text{ then } q \Rightarrow \sigma. \]
Further, for all \( \tau < |\sigma_1|, \) \[ exec(\tau) = \emptyset \rightarrow req(\tau) = \emptyset \Rightarrow \sigma_1, \]
and thus
\[ \forall t, t_0 \leq t < time : exec(t) = \emptyset \rightarrow req(t) = \emptyset \Rightarrow \sigma_1. \]
Hence, by the assumptions of the rule, \[ C_1 \Rightarrow \sigma_1, \text{ and if } |\sigma_1| < \infty \text{ then } q_1 \Rightarrow \sigma_1. \]
Since \( |\sigma| = |\sigma_1| \) and \( req \) and \( exec \) do not occur in \( C_1 \) and \( q_1, \) we obtain
\[ C_1 \Rightarrow \sigma, \text{ and if } |\sigma| < \infty \text{ then } q_1 \Rightarrow \sigma. \]
Bibliography


