

Determinism and termination in the semantics of the Céu programming language

February 3, 2017

1 The original formulation

The semantics discussed in this section follows as much as possible the original semantics of Céu presented in [?]. Any deviations from original definitions are duly noted in the text.

1.1 Abstract syntax

The *abstract syntax* of Céu programs is given by the following grammar:

$p \in P ::=$	skip	do nothing
	$v := a$	assignment
	<code>await</code> (e)	await event
	<code>emit</code> (e)	emit event
	<code>break</code>	break innermost loop
	<code>if</code> b <code>then</code> p_1 <code>else</code> p_2	conditional
	$p_1; p_2$	sequence
	<code>loop</code> p_1	repetition
	p_1 <code>and</code> p_2	par/and
	p_1 <code>or</code> p_2	par/or
	<code>fin</code> p_1	finalization
	<code>@awaiting</code> (e, n)	awaiting e since reaction n
	<code>@emitting</code> (e, n)	emitting e on stack level n
	p_1 <code>@loop</code> p_2	unwinded loop

(1) `skip` precisa aparecer na gramática já que aparece nos programas em P .
(2) Atribuição agora aparece explicitamente na gramática. Expressões aritméticas e booleanas também estão na gramática mas a sua estrutura interna é omitida.

where $n \in N$ is an integer, $v \in V$ is a memory location (variable) identifier, $e \in E$ is an event identifier, $a \in A$ is an arithmetic expression, $b \in B$ is a boolean expression, and $p, p_1, p_2 \in P$ are programs. We assume the usual structure for arithmetic and boolean expressions, and omit their definition.

1.2 The reaction inner-step relation

The *state* of a Céu program within a reaction is represented by a stack of events $\alpha = e_1 e_2 \dots e_n \in E^*$ together with a memory map $m: v \rightarrow N \in \mathcal{M}$. A *configuration*

is a 4-tuple $\langle p, \alpha, m, n \rangle \in \Delta$ that represents the situation of program p waiting to be evaluated in state $\langle \alpha, m \rangle$ and reaction n . Given an initial configuration, each small-step within a program reaction is determined by the reaction-inner-step relation $\rightarrow \in \Delta \times \Delta$ such that $\langle p, \alpha, m, n \rangle \rightarrow \langle p', \alpha', m', n \rangle$ iff a reaction inner-step of program p in state $\langle \alpha, m \rangle$ and reaction number n evaluates to a modified program p' and a modified state $\langle \alpha', m' \rangle$ in the same reaction (n). Since relation \rightarrow can only relate configurations with the same n , we shall write $\langle p, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha', m' \rangle$ for $\langle p, \alpha, m, n \rangle \rightarrow \langle p', \alpha', m', n \rangle$.

Relation \rightarrow is defined inductively with the help of the auxiliary functions *eval*, *blocked*, and *clear*. The *eval* function evaluates arithmetic or boolean expressions on a given memory; we omit its definition and assume that such evaluation is deterministic and always terminates. The *blocked* function is a predicate that determines if all trails of a program p are blocked on a given event stack and reaction number. And the *clear* function extracts the body of *fin* blocks from a given program.

Definition 1.1. Function *blocked*: $P \times E^* \times N \rightarrow \{0, 1\}$ is defined inductively as follows.

$$\begin{aligned}
& \text{blocked}(\text{skip}, e\alpha, n) = 0 \\
& \text{blocked}(v := a, e\alpha, n) = 0 \\
& \text{blocked}(\text{await}(e'), e\alpha, n) = 0 \\
& \text{blocked}(\text{emit}(e'), e\alpha, n) = 0 \\
& \text{blocked}(\text{break}, e\alpha, n) = 0 \\
& \text{blocked}(\text{if } b \text{ then } p_1 \text{ else } p_2, e\alpha, n) = 0 \\
& \text{blocked}(p_1; p_2, e\alpha, n) = \text{blocked}(p_1, e\alpha, n) \\
& \text{blocked}(\text{loop } p, e\alpha, n) = 0 \\
& \text{blocked}(p_1 \text{ and } p_2, e\alpha, n) = \text{blocked}(p_1, e\alpha, n) \cdot \text{blocked}(p_2, e\alpha, n) \\
& \text{blocked}(p_1 \text{ or } p_2, e\alpha, n) = \text{blocked}(p_1, e\alpha, n) \cdot \text{blocked}(p_2, e\alpha, n) \\
& \text{blocked}(\text{fin } p_1, e\alpha, n) = 0 \\
& \text{blocked}(@\text{awaiting}(e', n'), e\alpha, n) = \begin{cases} 1 & \text{if } e \neq e' \text{ or } n = n' \\ 0 & \text{otherwise} \end{cases} \\
& \text{blocked}(@\text{emitting}(n'), e\alpha, n) = \begin{cases} 1 & \text{if } |e\alpha| \neq n' \\ 0 & \text{otherwise} \end{cases} \\
& \text{blocked}(p_1 @\text{loop } p_2, e\alpha, n) = \text{blocked}(p_1, e\alpha, n)
\end{aligned}$$

Definition 1.2. Function *clear*: $P \rightarrow P'$, where $P' = \{v := a\}$, is defined inductively as follows.

- (1) Além de atribuições P' não deveria conter instruções if-else?.
- (2) No artigo original, $\text{clear}(p_1; p_2)$ retorna apenas $\text{clear}(p_1)$. Acharmos que isso está errado, já que apenas o primeiro *fin* seria considerado. (Confirmar com o Chicão).
- (3) Adicionei $\text{clear}(\text{skip})$.

$$\begin{aligned}
& \text{clear}(\text{skip}) = \text{skip} \\
& \text{clear}(v := a) = v := a \\
& \text{clear}(\text{await}(e')) = \text{skip} \\
& \text{clear}(\text{emit}(e')) = \text{skip} \\
& \text{clear}(\text{break}) = \text{skip} \\
& \text{clear}(\text{if } b \text{ then } p_1 \text{ else } p_2) = \text{skip} \\
& \text{clear}(p_1; p_2) = \text{clear}(p_1) \\
& \text{clear}(\text{loop } p) = \text{clear}(p) \\
& \text{clear}(p_1 \text{ and } p_2) = \text{clear}(p_1); \text{clear}(p_2) \\
& \text{clear}(p_1 \text{ or } p_2) = \text{clear}(p_1); \text{clear}(p_2) \\
& \text{clear}(\text{fin } p) = p \\
& \text{clear}(\text{@awaiting}(e', n')) = \text{skip} \\
& \text{clear}(\text{@emitting}(n')) = \text{skip} \\
& \text{clear}(p_1 \text{ @loop } p_2) = \text{clear}(p_1)
\end{aligned}$$

Definition 1.3 (Reaction inner-step). Relation $\rightarrow \subseteq \Delta \times \Delta$ is defined inductively as follows.

Await and emit

$$\begin{aligned}
(1.1) \quad & \langle \text{await}(e), \alpha, m \rangle \xrightarrow{n} \langle \text{@awaiting}(e, n'), \alpha, m \rangle \quad \text{with } n' = n + 1 \\
(1.2) \quad & \langle \text{@awaiting}(e, n'), e\alpha, m \rangle \xrightarrow{n} \langle \text{skip}, e\alpha, m \rangle \quad \text{if } n' \leq n \\
(1.3) \quad & \langle \text{emit}(e), \alpha, m \rangle \xrightarrow{n} \langle \text{@emitting}(n'), e\alpha, m \rangle \quad \text{with } n' = |\alpha| \\
(1.4) \quad & \langle \text{@emitting}(n'), \alpha, m \rangle \xrightarrow{n} \langle \text{skip}, \alpha, m \rangle \quad \text{if } n' = |\alpha|
\end{aligned}$$

Conditionals

$$\begin{aligned}
(1.5) \quad & \langle \text{if } b \text{ then } p_1 \text{ else } p_2, \alpha, m \rangle \xrightarrow{n} \langle p_1, \alpha, m \rangle \quad \text{if } \text{eval}(b, m) = 1 \\
(1.6) \quad & \langle \text{if } b \text{ then } p_1 \text{ else } p_2, \alpha, m \rangle \xrightarrow{n} \langle p_2, \alpha, m \rangle \quad \text{if } \text{eval}(b, m) = 0
\end{aligned}$$

Sequences

$$\begin{aligned}
(1.7) \quad & \langle \text{skip}; p, \alpha, m \rangle \xrightarrow{n} \langle p, \alpha, m \rangle \\
(1.8) \quad & \langle v := a; p, \alpha, m \rangle \xrightarrow{n} \langle p, \alpha, m' \rangle \quad \text{with } m' = m[v/\text{eval}(a)] \\
(1.9) \quad & \langle \text{break}; p, \alpha, m \rangle \xrightarrow{n} \langle \text{break}, \alpha, m \rangle \\
(1.10) \quad & \frac{\langle p_1, \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle}{\langle p_1; p_2, \alpha, m \rangle \xrightarrow{n} \langle p'_1; p_2, \alpha', m' \rangle} \quad \text{if } p_1 \neq \text{skip}, v := a, \text{break}
\end{aligned}$$

(1) Adicionamos o mapa de memória (m) à configuração e regras explícitas para atribuição. A avaliação de expressões aritméticas e booleanas está encapsulada na função *eval*.
(2) Adicionamos regras para consumir instruções *skip*.
(3) Adicionamos condições que garantem que a cada passo apenas uma regra é aplicável—não há escolha. Outra forma menos verbosa de fazer isso é dizer que elas devem ser avaliadas na ordem em que foram declaradas. Nesse caso, a primeira que for satisfeita deve ser aplicada.

Loops

$$(1.11) \quad \langle \text{loop } p, \alpha, m \rangle \xrightarrow{n} \langle p @ \text{loop } p, \alpha, m \rangle$$

$$(1.12) \quad \langle \text{skip} @ \text{loop } p, \alpha, m \rangle \xrightarrow{n} \langle \text{loop } p, \alpha, m \rangle$$

$$(1.13) \quad \langle v := a @ \text{loop } p, \alpha, m \rangle \xrightarrow{n} \langle \text{loop } p, \alpha, m' \rangle \quad \text{with } m' = m[v/eval(a)]$$

$$(1.14) \quad \langle \text{break} @ \text{loop } p, \alpha, m \rangle \xrightarrow{n} \langle \text{skip}, \alpha, m \rangle$$

$$(1.15) \quad \frac{\langle p_1, \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle}{\langle p_1 @ \text{loop } p_2, \alpha, m \rangle \xrightarrow{n} \langle p'_1 @ \text{loop } p_2, \alpha', m' \rangle} \quad \begin{array}{l} \text{if } p_1 \neq \text{skip}, \\ v := a, \text{break} \end{array}$$

Par/and

$$(1.16) \quad \langle \text{skip and } p, \alpha, m \rangle \xrightarrow{n} \langle p, \alpha, m \rangle$$

$$(1.17) \quad \langle p \text{ and skip}, \alpha, m \rangle \xrightarrow{n} \langle p, \alpha, m \rangle$$

$$(1.18) \quad \langle v := a \text{ and } p, \alpha, m \rangle \xrightarrow{n} \langle p, \alpha, m' \rangle \quad \text{with } m' = m[v/eval(a)]$$

$$(1.19) \quad \langle p \text{ and } v := a, \alpha, m \rangle \xrightarrow{n} \langle p, \alpha, m' \rangle \quad \begin{array}{l} \text{if } blocked(p, \alpha, n) = 1, \\ \text{with } m' = m[v/eval(a)] \end{array}$$

$$(1.20) \quad \langle \text{break and } p, \alpha, m \rangle \xrightarrow{n} \langle p'; \text{break}, \alpha, m \rangle \quad \text{with } p' = clear(p)$$

$$(1.21) \quad \langle p \text{ and break}, \alpha, m \rangle \xrightarrow{n} \langle p'; \text{break}, \alpha, m \rangle \quad \begin{array}{l} \text{if } blocked(p, \alpha, n) = 1, \\ \text{with } p' = clear(p) \end{array}$$

$$(1.22) \quad \frac{\langle p_1, \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle}{\langle p_1 \text{ and } p_2, \alpha, m \rangle \xrightarrow{n} \langle p'_1 \text{ and } p_2, \alpha', m' \rangle} \quad \begin{array}{l} \text{if } blocked(p_1, \alpha, n) = 0 \\ \text{and } p_1 \neq \text{skip}, \\ v := a, \text{break} \end{array}$$

$$(1.23) \quad \frac{\langle p_2, \alpha, m \rangle \xrightarrow{n} \langle p'_2, \alpha', m' \rangle}{\langle p_1 \text{ and } p_2, \alpha, m \rangle \xrightarrow{n} \langle p_1 \text{ and } p'_2, \alpha', m' \rangle} \quad \begin{array}{l} \text{if } blocked(p_1, \alpha, n) = 1 \\ \text{and } p_2 \neq \text{skip}, \\ v := a, \text{break} \end{array}$$

(1)Adicionamos a condição de o lado esquerdo estar bloqueado nas regras 1.19 e 1.17.

Par/or

$$\begin{aligned}
(1.24) \quad & \langle \text{skip or } p, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha, m \rangle && \text{with } p' = \text{clear}(p) \\
(1.25) \quad & \langle p \text{ or skip}, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha, m \rangle && \text{if } \text{blocked}(p, \alpha, n) = 1, \\
& && \text{with } p' = \text{clear}(p) \\
(1.26) \quad & \langle v := a \text{ or } p, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha, m' \rangle && \text{with } m' = m[v/\text{eval}(a)] \\
& && \text{and } p' = \text{clear}(p) \\
(1.27) \quad & \langle p \text{ or } v := a, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha, m' \rangle && \text{if } \text{blocked}(p, \alpha, n) = 1, \\
& && \text{with } m' = m[v/\text{eval}(a)] \\
& && \text{and } p' = \text{clear}(p) \\
(1.28) \quad & \langle \text{break or } p, \alpha, m \rangle \xrightarrow{n} \langle p'; \text{break}, \alpha, m \rangle && \text{with } p' = \text{clear}(p) \\
(1.29) \quad & \langle p \text{ or break}, \alpha, m \rangle \xrightarrow{n} \langle p'; \text{break}, \alpha, m \rangle && \text{if } \text{blocked}(p, \alpha, n) = 1, \\
& && \text{with } p' = \text{clear}(p) \\
(1.30) \quad & \frac{\langle p_1, \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle}{\langle p_1 \text{ or } p_2, \alpha, m \rangle \xrightarrow{n} \langle p'_1 \text{ or } p_2, \alpha', m' \rangle} && \text{if } \text{blocked}(p_1, \alpha, n) = 0 \\
& && \text{and } p_1 \neq \text{skip}, \\
& && v := a, \text{break} \\
(1.31) \quad & \frac{\langle p_2, \alpha, m \rangle \xrightarrow{n} \langle p'_2, \alpha', m' \rangle}{\langle p_1 \text{ or } p_2, \alpha, m \rangle \xrightarrow{n} \langle p_1 \text{ or } p'_2, \alpha', m' \rangle} && \text{if } \text{blocked}(p_1, \alpha, n) = 1 \\
& && \text{and } p_2 \neq \text{skip}, \\
& && v := a, \text{break}
\end{aligned}$$

The next theorem establishes that the reaction inner-step relation is deterministic, i.e., that it is in fact a *partial* function.

Theorem 1.4 (Determinism of the inner-step relation). *For all $p, p_1, p_2 \in P$, $\alpha, \alpha_1, \alpha_2 \in E^*$, $m, m_1, m_2 \in \mathcal{M}$, and $n \in N$,*

$$\begin{aligned}
& \text{if } \langle p, \alpha, m \rangle \xrightarrow{n} \langle p_1, \alpha_1, m_1 \rangle \text{ and } \langle p, \alpha, m \rangle \xrightarrow{n} \langle p_2, \alpha_2, m_2 \rangle, \\
& \text{then } \langle p_1, \alpha_1, m_1 \rangle = \langle p_2, \alpha_2, m_2 \rangle.
\end{aligned}$$

Proof. By induction on the structure of derivations. Suppose

$$d_1 \Vdash \langle p, \alpha, m \rangle \xrightarrow{n} \langle p_1, \alpha_1, m_1 \rangle \quad \text{and} \quad d_2 \Vdash \langle p, \alpha, m \rangle \xrightarrow{n} \langle p_2, \alpha_2, m_2 \rangle,$$

for some derivations d_1 and d_2 . Then there are ten possibilities depending on the structure of p . (Note that p cannot be equal to **skip**, $v := a$, or **break**, as there are no rules to evaluate such programs.)

[Case 1] $p = \text{await}(e)$, for some $e \in E$. Then d_1 and d_2 are instances of axiom (1.1), and as such, $p_1 = p_2 = @awaiting(e, n')$ with $n' = n + 1$, and $\alpha_1 = \alpha_2 = \alpha$ and $m_1 = m_2 = m$.

[Case 2] $p = @awaiting(e, n')$, for some $e \in E$ and $n' \in N$. Then d_1 and d_2 are instances of axiom (1.2), with $n' \leq n$. Thus $p_1 = p_2 = \text{skip}$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 3] $p = \text{emit}(e)$, for some $e \in E$. Then d_1 and d_2 are instances of axiom (1.3), and as such, $p_1 = p_2 = @\text{emitting}(n')$ with $n' = |\alpha|$, and $\alpha_1 = \alpha_2 = e\alpha$ and $m_1 = m_2 = m$.

[Case 4] $p = @\text{emitting}(e, n')$, for some $e \in E$ and $n' \in N$. Then d_1 and d_2 are instances of axiom (1.4) with $n' = |\alpha|$. Thus $p_1 = p_2 = \text{skip}$, $\alpha_1 = \alpha_2 = \alpha$ and $m_1 = m_2 = m$.

[Case 5] $p = \text{if } b \text{ then } p' \text{ else } p''$, for some $b \in B$ and $p', p'' \in P$.

[Case 5.1] $\text{eval}(b, m) = 1$. Then d_1 and d_2 are instances of axiom (1.5), and as such, $p_1 = p_2 = p'$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 5.2] $\text{eval}(b, m) = 0$. Then d_1 and d_2 are instances of axiom (1.6), and as such, $p_1 = p_2 = p''$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 6] $p = p'; p''$, for some $p', p'' \in P$.

[Case 6.1] $p' = \text{skip}$. Then d_1 and d_2 are instances of axiom (1.7), and as such, $p_1 = p_2 = p''$, $\alpha_1 = \alpha_2 = \alpha$ and $m_1 = m_2 = m$.

[Case 6.2] $p' = v := a$, for some $v \in V$ and $a \in A$. Then d_1 and d_2 are instances of axiom (1.8), and as such, $p_1 = p_2 = p''$, $\alpha_1 = \alpha_2 = \alpha$ and, as eval is a total function, $m_1 = m_2 = m[v/\text{eval}(a)]$.

[Case 6.3] $p' = \text{break}$. Then d_1 and d_2 are instances of axiom (1.9), and as such, $p_1 = p_2 = p'$, $\alpha_1 = \alpha_2 = \alpha$ and $m_1 = m_2 = m$.

[Case 6.4] $p' \neq \text{skip}, v := a, \text{break}$. Then d_1 and d_2 are instances of rule (1.10). Thus there are derivations d'_1 and d'_2 such that

$$d'_1 \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha_1, m_1 \rangle \quad \text{and} \quad d'_2 \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_2, \alpha_2, m_2 \rangle,$$

for some $p'_1, p'_2 \in P$. Since $d'_1 < d_1$ and $d'_2 < d_2$, by induction hypothesis, $\alpha_1 = \alpha_2$, $m_1 = m_2$, and $p'_1 = p'_2$, which implies

$$p_1 = p'_1; p'' = p'_2; p'' = p_2.$$

[Case 7] $p = \text{loop } p'$, for some $p' \in P$. Then d_1 and d_2 are instances of axiom (1.11), and as such, $p_1 = p_2 = p' @ \text{loop } p'$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 8] $p = p' @ \text{loop } p''$, for some $p', p'' \in P$.

[Case 8.1] $p = \text{skip} @ \text{loop } p'$, for some $p' \in P$. Then d_1 and d_2 are instances of axiom (1.12), and as such, $p_1 = p_2 = \text{loop } p'$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 8.2] $p = v := a @ \text{loop } p'$, for some $a \in A$, $v \in V$, and $p' \in P$. Then d_1 and d_2 are instances of axiom (1.13), and as such, $p_1 = p_2 = \text{loop } p'$, $\alpha_1 = \alpha_2 = \alpha$, and as eval is a total function, $m_1 = m_2 = m[v/\text{eval}(a)]$.

[Case 8.3] $p = \text{break} @ \text{loop } p'$, for some $p' \in P$. Then d_1 and d_2 are instances of axiom (1.14), and as such, $p_1 = p_2 = \text{skip}$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 8.4] $p = p' @ \text{loop } p''$, for some $p', p'' \in P$ such that $p' \neq \text{skip}, v := a, \text{break}$. Then d_1 and d_2 are instances of rule (1.15). Thus there are derivations d'_1 and d'_2 such that

$$d'_1 \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha_1, m_1 \rangle \quad \text{and} \quad d'_2 \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_2, \alpha_2, m_2 \rangle,$$

for some $p'_1, p'_2 \in P$. Since $d'_1 < d_1$ and $d'_2 < d_2$, by induction hypothesis, $\alpha_1 = \alpha, m_1 = m_2$, and $p'_1 = p'_2$, which implies

$$p_1 = p'_1 @ \text{loop } p'' = p'_2 @ \text{loop } p'' = p_2.$$

[Case 9] $p = p'$ and p'' , for some $p', p'' \in P$.

[Case 9.1] $p = \text{skip}$ and p' , for some $p' \in P$ and $p' \neq \text{break}$. Then d_1 and d_2 are instances of axiom (1.16), and as such, $p_1 = p_2 = p', \alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 9.2] $p = p'$ and skip , for some $p' \in P$ and $p' \neq \text{break}$. Similar to Case 9.1.

[Case 9.3] $p = v := a$ and p' , for some $v \in V, a \in A$ and $p' \in P$. Then d_1 and d_2 are instances of axiom (1.18), and as such, $p_1 = p_2 = p', \alpha_1 = \alpha_2 = \alpha$, and as eval is a total function, $m_1 = m_2 = m[v/\text{eval}(a)]$.

[Case 9.4] $p = p'$ and $v := a$, for some $v \in V, a \in A$ and $p' \in P$. Then either $\text{blocked}(p') = 0$ or $\text{blocked}(p') = 1$. If $\text{blocked}(p') = 0$ then this case becomes Case 9.7. Otherwise, if $\text{blocked}(p') = 1$, then d_1 and d_2 are instances of axiom (1.19), and as such, $p_1 = p_2 = p', \alpha_1 = \alpha_2 = \alpha$, and as eval is a total function, $m_1 = m_2 = m[v/\text{eval}(a)]$.

[Case 9.5] $p = \text{break}$ and p' , for some $p' \in P$. Then d_1 and d_2 are instances of axiom (1.20), and as such, $\alpha_1 = \alpha_2 = \alpha, m_1 = m_2 = m$, and as clear is a total function, $p_1 = p_2 = \text{clear}(p'); \text{break}$.

[Case 9.6] $p = p'$ and break , for some $p' \in P$. Then either $\text{blocked}(p') = 0$ or $\text{blocked}(p') = 1$. If $\text{blocked}(p') = 0$ then this case becomes Case 9.7. Otherwise, if $\text{blocked}(p') = 1$, then d_1 and d_2 are instances of axiom (1.21), and as such, $\alpha_1 = \alpha_2 = \alpha, m_1 = m_2 = m$, and as clear is a total function, $p_1 = p_2 = \text{clear}(p'); \text{break}$.

[Case 9.7] $p = p'$ and p'' , for some p' and $p'' \in P$. Then there are two possibilities. If $\text{blocked}(p') = 0$ then d_1 and d_2 are instances of (1.22). Thus there are derivations d'_1 and d'_2 such that

$$d'_1 \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha_1, m_1 \rangle \quad \text{and} \quad d'_2 \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_2, \alpha_2, m_2 \rangle,$$

for some $p'_1, p'_2 \in P$. Since $d'_1 < d_1$ and $d'_2 < d_2$, by induction hypothesis, $\alpha_1 = \alpha_2, m_1 = m_2$, and $p'_1 = p'_2$, which implies

$$p_1 = (p'_1 \text{ and } p'') = (p'_2 \text{ and } p'') = p_2.$$

If, however, $blocked(p') = 1$ and $p'' \neq \text{skip}, v := a, \text{break}$, then d_1 and d_2 are instances of (1.23). Thus there are derivations d_1'' and d_2'' such that

$$d_1'' \Vdash \langle p'', \alpha, m \rangle \xrightarrow{n} \langle p_1'', \alpha_1, m_1 \rangle \quad \text{and} \quad d_2'' \Vdash \langle p'', \alpha, m \rangle \xrightarrow{n} \langle p_2'', \alpha_2, m_2 \rangle,$$

for some $p_1'', p_2'' \in P$. Since $d_1'' < d_1$ and $d_2'' < d_2$, by induction hypothesis, $\alpha_1 = \alpha_2$, $m_1 = m_2$, and $p_1'' = p_2''$, which implies

$$p_1 = (p' \text{ and } p_1'') = (p' \text{ and } p_2'') = p_2.$$

[Case 10] $p = p' \text{ or } p''$, for some $p', p'' \in P$.

[Case 10.1] $p = \text{skip or } p'$, for some $p' \in P$. Then d_1 and d_2 are instances of axiom (1.24), and as such, $\alpha_1 = \alpha_2 = \alpha$, $m_1 = m_2 = m$, and as $clear$ is a total, $p_1 = p_2 = clear(p')$.

[Case 10.2] $p = p' \text{ or skip}$, for some $p' \in P$. Then either $blocked(p') = 0$ or $blocked(p') = 1$. If $blocked(p') = 0$ then this case becomes Case 10.7. Otherwise, if $blocked(p') = 1$, then d_1 and d_2 are instances of axiom (1.25), and as such, $\alpha_1 = \alpha_2 = \alpha$, $m_1 = m_2 = m$, and as $clear$ is a total function, $p_1 = p_2 = clear(p')$.

[Case 10.3] $p = v := a \text{ or } p'$, for some $v \in V$, $a \in A$ and $p' \in P$. Then d_1 and d_2 are instances of axiom (1.26), and as such, $\alpha_1 = \alpha_2 = \alpha$, and as $eval$ and $clear$ are total functions, $m_1 = m_2 = m[v/eval(a)]$ and $p_1 = p_2 = clear(p')$.

[Case 10.4] $p = p' \text{ or } v := a$, for some $v \in V$, $a \in A$ and $p' \in P$. Then either $blocked(p') = 0$ or $blocked(p') = 1$. If $blocked(p') = 0$ then this case becomes Case 10.7. Otherwise, if $blocked(p') = 1$, then d_1 and d_2 are instances of axiom (1.27), and as such, $\alpha_1 = \alpha_2 = \alpha$, and as $eval$ and $clear$ are total functions, $m_1 = m_2 = m[v/eval(a)]$ and $p_1 = p_2 = clear(p')$.

[Case 10.5] $p = \text{break or } p'$, for some $p' \in P$. Then d_1 and d_2 are instances of axiom (1.28), and as such, $\alpha_1 = \alpha_2 = \alpha$, $m_1 = m_2 = m$, and as $clear$ is a total function, $p_1 = p_2 = clear(p'); \text{break}$.

[Case 10.6] $p = p' \text{ or break}$, for some $p' \in P$. Then either $blocked(p') = 0$ or $blocked(p') = 1$. If $blocked(p') = 0$ then this case becomes Case 10.7. Otherwise, if $blocked(p') = 1$, then d_1 and d_2 are instances of axiom (1.29), and as such, $\alpha_1 = \alpha_2 = \alpha$, $m_1 = m_2 = m$, and as $clear$ is a total function, $p_1 = p_2 = clear(p'); \text{break}$.

[Case 10.7] $p = p' \text{ or } p''$, for some p' and $p'' \in P$. Then there are two possibilities. If $blocked(p') = 0$ then d_1 and d_2 are instances of (1.30). Thus there are derivations d_1' and d_2' such that

$$d_1' \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p_1', \alpha_1, m_1 \rangle \quad \text{and} \quad d_2' \Vdash \langle p', \alpha, m \rangle \xrightarrow{n} \langle p_2', \alpha_2, m_2 \rangle,$$

for some $p_1', p_2' \in P$. Since $d_1' < d_1$ and $d_2' < d_2$, by induction hypothesis, $\alpha_1 = \alpha_2$, $m_1 = m_2$, and $p_1' = p_2'$, which implies

$$p_1 = (p_1' \text{ or } p'') = (p_2' \text{ or } p'') = p_2.$$

If, however, $blocked(p') = 1$ and $p'' \neq \text{skip}, v := a, \text{break}$, then d_1 and d_2 are instances of (1.31). Thus there are derivations d_1'' and d_2'' such that

$$d_1'' \Vdash \langle p'', \alpha, m \rangle \xrightarrow{n} \langle p_1'', \alpha_1, m_1 \rangle \quad \text{and} \quad d_2'' \Vdash \langle p'', \alpha, m \rangle \xrightarrow{n} \langle p_2'', \alpha_2, m_2 \rangle,$$

for some $p_1'', p_2'' \in P$. Since $d_1'' < d_1$ and $d_2'' < d_2$, by induction hypothesis, $\alpha_1 = \alpha_2$, $m_1 = m_2$, and $p_1'' = p_2''$, which implies

$$p_1 = (p' \text{ or } p_1'') = (p' \text{ or } p_2'') = p_2. \quad \square$$

The next lemma establishes that given a program either it is possible to advance it by an inner-step or all its trails are blocked, but not both.

Lemma 1.5. *For all $p \in P$, $\alpha \in E^*$, $m \in \mathcal{M}$, and $n \in N$, if $p \neq \text{skip}, v := a, \text{break}$ then either*

$$\exists \delta \in \Delta(\langle p, \alpha, m \rangle \xrightarrow{n} \delta) \quad \text{or} \quad blocked(p, \alpha, n) = 1,$$

but not both.

Proof. By induction on the structure of programs.

[Case 1] $p = \text{await}(e)$, for some $e \in E$. Then by axiom (1.1),

$$\langle \text{await}(e), \alpha, m \rangle \xrightarrow{n} \langle @\text{awaiting}(e, n'), \alpha, m \rangle = \delta,$$

where $n' = n + 1$. And by Definition 1.1, $blocked(\text{await}(e), \alpha, n) = 0$.

[Case 2] $p = @\text{awaiting}(e, n')$, for some $e \in E$ and $n' \in N$.

[Case 2.1] $n' < n$. If e is the top-of-stack event in α , in symbols $e = \alpha_{[1]}$, then by axiom (1.2),

$$\langle @\text{awaiting}(e, n'), \alpha, m \rangle \xrightarrow{n} \langle \text{skip}, \alpha, m \rangle = \delta.$$

And by Definition 1.1, $blocked(@\text{awaiting}(e, n'), \alpha, n) = 0$.

If, however, $e \neq \alpha_{[1]}$, then there is no such δ , as no rule is applicable. And by Definition 1.1, $blocked(@\text{awaiting}(e, n'), \alpha, n) = 1$.

[Case 2.2] $n' = n$. [FIXME: Pelo axioma (1.2),

$$\langle @\text{awaiting}(e, n'), \alpha, m \rangle \xrightarrow{n} \langle \text{skip}, \alpha, m \rangle.$$

E $blocked(@\text{awaiting}(e, n'), \alpha, n) = 1$. Ou seja, ambos os lados do “ou” deram verdadeiro, o que invalida o lema.]

[Case 2.3] $n' > n$. If $e = \alpha_{[1]}$ then [FIXME: Não existe tal δ e

$$blocked(@\text{awaiting}(e, n'), \alpha, n) = 0.$$

O que, novamente, invalida o lema.]

If, however, $e \neq \alpha_{[1]}$, then there is no such δ (no rule is applicable) and, by Definition 1.1, $blocked(@\text{awaiting}(e, n'), \alpha, n) = 1$.

[Case 3] $p = \text{emit}(e)$, for some $e \in E$. Then by axiom (1.3),

$$\langle \text{emit}(e), \alpha, m \rangle \xrightarrow{n} \langle @\text{emitting}(n'), e\alpha, m \rangle = \delta,$$

where $n' = |\alpha|$. And by Definition 1.1, $\text{blocked}(\text{emit}(e'), e\alpha, n) = 0$.

[Case 4] $p = @\text{emitting}(e, n')$, for some $e \in E$ and $n' \in N$.

[Case 4.1] $n' = |\alpha|$. By axiom (1.4),

$$\langle @\text{emitting}(e, n'), \alpha, m \rangle \xrightarrow{n} \langle \text{skip}, \alpha, m \rangle = \delta.$$

And by Definition 1.1, $\text{blocked}(@\text{emitting}(e, n'), \alpha, n) = 0$.

[Case 4.2] $n' \neq |\alpha|$. Then there is no such δ (no rule is applicable) and, by Definition 1.1, $\text{blocked}(@\text{emitting}(e, n'), \alpha, n) = 1$.

[Case 5] $p = \text{if } b \text{ then } p' \text{ else } p''$, for some $b \in B$ and $p', p'' \in P$. By axioms (1.5) and (1.6), if $\text{eval}(b, m) = 1$, $\delta = \langle p', \alpha, m \rangle$, otherwise $\delta = \langle p'', \alpha, m \rangle$. And by Definition 1.1, $\text{blocked}(\text{if } b \text{ then } p' \text{ else } p'', e\alpha, n) = 0$

[Case 6] $p = p'; p''$, for some $p', p'' \in P$. By Definition 1.1 $\text{blocked}(p'; p'', \alpha, n) = \text{blocked}(p', \alpha, n)$

[Case 6.1] $p' = \text{skip}$. By axiom (1.7), $\delta = \langle p'', \alpha, m \rangle$, and by Definition 1.1, $\text{blocked}(\text{skip}, \alpha, n) = 0$.

[Case 6.2] $p' = v := a$, for some $v \in V$ and $a \in A$. By axiom (1.8), $\delta = \langle p'', \alpha, m[v/\text{eval}(a)] \rangle$, and by Definition 1.1, $\text{blocked}(v := a, \alpha, n) = 0$.

[Case 6.3] $p' = \text{break}$. By axiom (1.9), $\delta = \langle \text{Break}, \alpha, m \rangle$, and by Definition 1.1, $\text{blocked}(\text{break}, \alpha, n) = 0$.

[Case 6.4] $p' \neq \text{skip}, v := a, \text{break}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

Suppose $\langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle$, for some $p'_1 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.10),

$$\langle p'; p'', \alpha, m \rangle \xrightarrow{n} \langle p'_1; p'', \alpha', m' \rangle.$$

And by Definition 1.1,

$$\text{blocked}(p'; p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 0.$$

If, however, $\text{blocked}(p', \alpha, n) = 1$, then there is no such d (no rule is applicable) and by Definition 1.1,

$$\text{blocked}(p'; p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 1.$$

[Case 7] $p = \text{loop } p'$, for some $p' \in P$. By axiom (1.11), $\delta = \langle p' @ \text{loop } p', \alpha, m \rangle$. And by Definition 1.1, $\text{blocked}(\text{loop } p', \alpha, m) = 0$.

[Case 8] $p = p' @ \text{loop } p''$, for some $p', p'' \in P$. By [Definition 1.1](#) $\text{blocked}(p' @ \text{loop } p'', \alpha, n) = \text{blocked}(p', \alpha, n)$

[Case 8.1] $p' = \text{skip}$. By axiom (1.12), $\delta = \langle \text{loop } p'', \alpha, m \rangle$, and by [Definition 1.1](#), $\text{blocked}(\text{skip}, \alpha, n) = 0$.

[Case 8.2] $p' = v := a$, for some $v \in V$ and $a \in A$. By axiom (1.13), $\delta = \langle \text{loop } p'', \alpha, m[v/\text{eval}(a)] \rangle$, and by [Definition 1.1](#), $\text{blocked}(v := a, \alpha, n) = 0$.

[Case 8.3] $p' = \text{break}$. By axiom (1.14), $\delta = \langle \text{skip}, \alpha, m \rangle$, and by [Definition 1.1](#), $\text{blocked}(\text{break}, \alpha, n) = 0$.

[Case 8.4] $p' \neq \text{skip}, v := a, \text{break}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

Suppose $\langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle$, for some $p'_1 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.15),

$$\langle p' @ \text{loop } p'', \alpha, m \rangle \xrightarrow{n} \langle p'_1 @ \text{loop } p'', \alpha', m' \rangle.$$

And by [Definition 1.1](#),

$$\text{blocked}(p' @ \text{loop } p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 0.$$

If, however, $\text{blocked}(p', \alpha, n) = 1$, then there is no such d (no rule is applicable) and by [Definition 1.1](#),

$$\text{blocked}(p' @ \text{loop } p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 1.$$

[Case 9] $p = p' \text{ and } p''$, for some $p' \in P$, By [Definition 1.1](#)

$$\text{blocked}(p' \text{ and } p'', \alpha, n) = \text{blocked}(p', \alpha, n) \cdot \text{blocked}(p'', \alpha, n)$$

[Case 9.1] $p' = \text{skip}$ and $p'' \neq \text{break}$. By axiom (1.16), $\delta = \langle p'', \alpha, m \rangle$, and by [Definition 1.1](#), $\text{blocked}(\text{skip}, \alpha, n) = 0$.

[Case 9.2] $p'' = \text{skip}$ and $p' \neq \text{break}$. By axiom (1.17), $\delta = \langle p', \alpha, m \rangle$, and by [Definition 1.1](#), $\text{blocked}(\text{skip}, \alpha, n) = 0$.

[Case 9.3] $p' = v := a$, for some $v \in V$ and $a \in A$. By axiom (1.18), $\delta = \langle p'', \alpha, m[v/\text{eval}(a)] \rangle$, and by [Definition 1.1](#), $\text{blocked}(v := a, \alpha, n) = 0$.

[Case 9.4] $p'' = v := a$, for some $v \in V$ and $a \in A$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

If $\text{blocked}(p', \alpha, m) = 0$, then the derivation of δ is similar to [Case 9.7](#).

If, however, $\text{blocked}(p', \alpha, n) = 1$, then by axiom (1.19), $\delta = \langle p', \alpha, m[v/\text{eval}(a)] \rangle$.

Either way, $\text{blocked}(v := a, \alpha, n) = 0$.

[Case 9.5] $p' = \text{break}$. By axiom (1.20), $\delta = \langle \text{clear}(p''), \alpha, m \rangle$, and by Definition 1.1, $\text{blocked}(\text{break}, \alpha, n) = 0$.

[Case 9.6] $p'' = \text{break}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

If $\text{blocked}(p', \alpha, n) = 0$, then the derivation of δ is similar to Case 9.7.

If, however, $\text{blocked}(p', \alpha, n) = 1$, then by axiom (1.21), $\delta = \langle \text{clear}(p'), \alpha, m \rangle$. Either way, $\text{blocked}(\text{break}, \alpha, n) = 0$.

[Case 9.7] $p' \neq \text{skip}, v := a, \text{break}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

Suppose $\langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle$, for some $p'_1 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.22),

$$\langle p' \text{ and } p'', \alpha, m \rangle \xrightarrow{n} \langle p'_1 \text{ and } p'', \alpha', m' \rangle.$$

And by Definition 1.1,

$$\text{blocked}(p' \text{ and } p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 0.$$

Now suppose $\langle p'', \alpha, m \rangle \xrightarrow{n} \langle p''_1, \alpha', m' \rangle$, for some $p''_1 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.23),

$$\langle p' \text{ and } p'', \alpha, m \rangle \xrightarrow{n} \langle p' \text{ and } p''_1, \alpha', m' \rangle.$$

And by Definition 1.1,

$$\text{blocked}(p' \text{ and } p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 0.$$

[Case 10] $p = p' \text{ or } p''$, for some $p' \in P$, By Definition 1.1

$$\text{blocked}(p' \text{ or } p'', \alpha, n) = \text{blocked}(p', \alpha, n) \cdot \text{blocked}(p'', \alpha, n)$$

[Case 10.1] $p' = \text{skip}$. By axiom (1.24), $\delta = \langle \text{clear}(p''), \alpha, m \rangle$, and by Definition 1.1, $\text{blocked}(\text{skip}, \alpha, n) = 0$.

[Case 10.2] $p'' = \text{skip}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

If $\text{blocked}(p', \alpha, n) = 0$, then the derivation of δ is similar to Case 10.7.

If, however, $\text{blocked}(p', \alpha, n) = 1$, then by axiom (1.25), $\delta = \langle \text{clear}(p'), \alpha, m \rangle$. Either way, $\text{blocked}(\text{skip}, \alpha, n) = 0$.

[Case 10.3] $p' = v := a$, for some $v \in V$ and $a \in A$. By axiom (1.26), $\delta = \langle \text{clear}(p''), \alpha, m[v/\text{eval}(a)] \rangle$, and by Definition 1.1, $\text{blocked}(v := a, \alpha, n) = 0$.

[Case 10.4] $p'' = v := a$, for some $v \in V$ and $a \in A$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

If $\text{blocked}(p', \alpha, m) = 0$, then the derivation of δ is similar to [Case 10.7](#).

If, however, $\text{blocked}(p', \alpha, n) = 1$, then by axiom (1.27), $\delta = \langle \text{clear}(p'), \alpha, m[v/\text{eval}(a)] \rangle$.

Either way, $\text{blocked}(v := a, \alpha, n) = 0$.

[Case 10.5] $p' = \text{break}$. By axiom (1.28), $\delta = \langle \text{clear}(p''), \alpha, m \rangle$, and by [Definition 1.1](#), $\text{blocked}(\text{break}, \alpha, n) = 0$.

[Case 10.6] $p'' = \text{break}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

If $\text{blocked}(p', \alpha, n) = 0$, then the derivation of δ is similar to [Case 10.7](#).

If, however, $\text{blocked}(p', \alpha, n) = 1$, then by axiom (1.29), $\delta = \langle \text{clear}(p'), \alpha, m \rangle$.

Either way, $\text{blocked}(\text{break}, \alpha, n) = 0$.

[Case 10.7] $p' \neq \text{skip}, v := a, \text{break}$. By induction hypothesis, exactly one of the following hold:

$$\exists d' \in \Delta(\langle p', \alpha, m \rangle \xrightarrow{n} d') \quad \text{or} \quad \text{blocked}(p', \alpha, n) = 1.$$

Suppose $\langle p', \alpha, m \rangle \xrightarrow{n} \langle p'_1, \alpha', m' \rangle$, for some $p'_1 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.30),

$$\langle p' \text{ or } p'', \alpha, m \rangle \xrightarrow{n} \langle p'_1 \text{ or } p'', \alpha', m' \rangle.$$

And by [Definition 1.1](#),

$$\text{blocked}(p' \text{ or } p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 0.$$

Now suppose $\langle p'', \alpha, m \rangle \xrightarrow{n} \langle p''_1, \alpha', m' \rangle$, for some $p''_1 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.31),

$$\langle p' \text{ or } p'', \alpha, m \rangle \xrightarrow{n} \langle p' \text{ and } p''_1, \alpha', m' \rangle.$$

And by [Definition 1.1](#),

$$\text{blocked}(p' \text{ or } p'', \alpha, n) = \text{blocked}(p', \alpha, n) = 0.$$

□

1.3 The reaction outer-step relation

From the previous inner-step relation we define an outer-step relation (\Rightarrow) that when necessary pops the event stack and advances blocked programs. [\[TODO: Improve this description.\]](#)

Definition 1.6 (Reaction outer-step).

$$(1.32) \quad \frac{\langle p, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha', m' \rangle}{\langle p, \alpha, m \rangle \xRightarrow{n} \langle p', \alpha', m' \rangle} \quad \text{if } \text{blocked}(p, \alpha, n) = 0$$

$$(1.33) \quad \langle p, e\alpha, m \rangle \xRightarrow{n} \langle p, \alpha, m \rangle \quad \text{if } \text{blocked}(p, \alpha, n) = 1$$

Theorem 1.7 (Determinism of the outer-step relation). *For all $p, p_1, p_2 \in P$, $\alpha, \alpha_1, \alpha_2 \in E^*$, $m, m_1, m_2 \in \mathcal{M}$, and $n \in N$,*

$$\begin{aligned} &\text{if } \langle p, \alpha, m \rangle \xRightarrow{n} \langle p_1, \alpha_1, m_1 \rangle \quad \text{and} \quad \langle p, \alpha, m \rangle \xRightarrow{n} \langle p_2, \alpha_2, m_2 \rangle, \\ &\text{then } \langle p_1, \alpha_1, m_1 \rangle = \langle p_2, \alpha_2, m_2 \rangle. \end{aligned}$$

Proof. [TODO: By induction on the structure of derivations.] □

Theorem 1.8 (Termination of the outer-step relation). *For all $p \in P$, $\alpha \in E$, and $m \in \mathcal{M}$, if $p \neq \text{skip}, v := a, \text{break}$ then*

$$\exists \delta \in \Delta (\langle p, \alpha, m \rangle \xRightarrow{n} \delta).$$

Proof. [TODO: Directly from Lemma 1.5.] □

1.4 The reaction relation

From the reflexive-transitive closure of the outer-step relation ($\xRightarrow{*}$) we define the reaction relation $\models \subseteq \Delta \times (P \times \mathcal{M} \times N)$, which computes a full program reaction. Given an initial configuration, the reaction relation evaluates it until the event stack becomes empty.

Definition 1.9 (Reaction). Let $p, p' \in P$, $\alpha \in E^*$, $m, m' \in \mathcal{M}$. Then

$$\langle p, \alpha, m \rangle \models \langle p', m' \rangle \quad \text{iff} \quad \langle p, \alpha, m \rangle \xRightarrow{*} \langle p', \varepsilon, m' \rangle.$$

The next two theorems establish, respectively, that reactions are deterministic and always terminate (for the nontrivial programs $p \neq \text{skip}, v := a, \text{break}$).

Theorem 1.10 (Determinism of the reaction relation). *For all $p, p_1, p_2 \in P$, $\alpha \in E^*$, $m, m_1, m_2 \in \mathcal{M}$, and $n \in N$,*

$$\begin{aligned} &\text{if } \langle p, \alpha, m \rangle \models \langle p_1, m_1 \rangle \quad \text{and} \quad \langle p, \alpha, m \rangle \models \langle p_2, m_2 \rangle, \\ &\text{then } \langle p_1, m_1 \rangle = \langle p_2, m_2 \rangle. \end{aligned}$$

Proof. [TODO: ?] □

Theorem 1.11 (Termination of the reaction relation). *For all $p \in P$, $\alpha \in E$, and $m \in \mathcal{M}$, if $p \neq \text{skip}, v := a, \text{break}$ then*

$$\langle p, \alpha, m \rangle \models \langle p', m' \rangle,$$

for some $p' \in P$ and $m' \in \mathcal{M}$.

Proof. [TODO: ?] □

2 Big-step version of the original formulation

[TODO: Minha ideia aqui é fazer uma versão big-step da formulação original. E no final comparar as duas versões, i.e., mostrar que são equivalentes.]

Definition 2.1. [TODO: Parcial e provavelmente incorreta.]

Empty program

$$(2.1) \quad \langle \varepsilon, \alpha, m, n \rangle \rightsquigarrow \langle \alpha, m, n \rangle$$

Assignment

$$(2.2) \quad \langle v := a, \alpha, m, n \rangle \rightsquigarrow \langle \alpha, m[v/eval(a)], n \rangle$$

Conditionals

$$(2.3) \quad \frac{\langle p_1, \alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle}{\langle \text{if } b \text{ then } p_1 \text{ else } p_2, \alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle} \quad \text{if } eval(b, m) = 1$$

$$(2.4) \quad \frac{\langle p_2, \alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle}{\langle \text{if } b \text{ then } p_1 \text{ else } p_2, \alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle} \quad \text{if } eval(b, m) = 0$$

Await

$$(2.5) \quad \frac{\langle @awaiting(e, n+1), e\alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle}{\langle await(e), \alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle}$$

$$(2.6) \quad \langle @awaiting(e', n'), e\alpha, m, n \rangle \rightsquigarrow \langle \alpha, m, n \rangle \quad \text{if } e' = e \text{ and } n' < n$$

$$(2.7) \quad \frac{\langle @awaiting(e', n'), \alpha, m, n \rangle \rightsquigarrow \langle \alpha'', m'', n'' \rangle}{\langle @awaiting(e', n'), e\alpha, m, n \rangle \rightsquigarrow \langle \alpha'', m'', n'' \rangle} \quad \text{if } e' \neq e \text{ or } n' \geq n$$

Emit

$$(2.8) \quad \frac{\langle @emitting(|\alpha|), e\alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle}{\langle emit(e), \alpha, m, n \rangle \rightsquigarrow \langle \alpha', m', n' \rangle}$$

$$(2.9) \quad \langle @emitting(n'), e\alpha, m, n \rangle \rightsquigarrow \langle \alpha, m, n \rangle \quad \text{if } |e\alpha| = n'$$

$$(2.10) \quad \frac{\langle @emitting(n'), \alpha, m, n \rangle \rightsquigarrow \langle \alpha'', m'', n'' \rangle}{\langle @emitting(n'), e\alpha, m, n \rangle \rightsquigarrow \langle \alpha'', m'', n'' \rangle} \quad \text{if } |e\alpha| \neq n'$$