Determinism and termination in the semantics of the Céu programming language

February 5, 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:

(1) `skip` precisa aparecer na gramática já que aprece 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.

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

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 $\alpha = e_1 e_2 \ldots e_n \in E^*$ together with a memory map $m \colon v \to N \in \mathcal{M}$. 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 $\mathit{blocked}\colon P \times E^* \times N \rightarrow \{0, 1\}$ is defined inductively as follows.

Intuitivamente, um programa p está bloqueado se toda trilha de p está (i) aguardando (`@awaiting`) algum evento que não está no topo da pilha ou que foi emitido na reação atual, ou (ii) acabou de emitir um evento (`@emitting`) e esse evento ainda não foi consumido (i.e., está na pilha). É isso?

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

Definition 1.2. Function $\mathit{clear}\colon P \rightarrow P'$ is defined inductively as follows.

Intuitivamente, dado um programa p, a função *clear* concatena em sequência o corpo das *primeiras* ocorrências de `fin` em cada trilha de p, e retorna a sequência resultante. A sintaxe do corpo de instruções `fin` está restrita a programas em $P' \subseteq P$, viz., que não contém instruções `await`, `@awaiting`, `emit`, `@emitting`, `and`, `or` e `fin`. É isso? Se sim, por que a chamada recursiva para *loops* mas não para *if-else*?

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

Ainda na definição da *clear*, será que não é preciso chamar a função recursivamente também para a segunda metade da sequência? Do jeito que está

$$clear((v := a; \texttt{fin}\,p)\;\texttt{and}\;\texttt{skip}) = \texttt{skip}; \texttt{skip}.$$

Não deveria ser

$$clear((v := a; \texttt{fin}\,p)\;\texttt{and}\;\texttt{skip}) = p; \texttt{skip}?$$

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

(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. (Não fizemos isso para deixar explícitas as condições de cada regra.)

Await and emit

(1.1) $\langle \texttt{await}(e), \alpha, m\rangle \xrightarrow{n} \langle \texttt{@awaiting}(e, n'), \alpha, m\rangle$ with $n' = n + 1$

(1.2) $\langle \texttt{@awaiting}(e, n'), e\alpha, m\rangle \xrightarrow{n} \langle \texttt{skip}, e\alpha, m\rangle$ if $n' \le n$

(1.3) $\langle \texttt{emit}(e), \alpha, m\rangle \xrightarrow{n} \langle \texttt{@emitting}(n'), e\alpha, m\rangle$ with $n' = |\alpha|$

(1.4) $\langle \texttt{@emitting}(n'), \alpha, m\rangle \xrightarrow{n} \langle \texttt{skip}, \alpha, m\rangle$ if $n' = |\alpha|$

Conditionals

(1.5) $\langle \texttt{if}\,b\,\texttt{then}\,p_1\,\texttt{else}\,p_2, \alpha, m\rangle \xrightarrow{n} \langle p_1, \alpha, m\rangle$ if $eval(b, m) = 1$

(1.6) $\langle \texttt{if}\,b\,\texttt{then}\,p_1\,\texttt{else}\,p_2, \alpha, m\rangle \xrightarrow{n} \langle p_2, \alpha, m\rangle$ if $eval(b, m) = 0$

Sequences

$$\langle \texttt{skip}; p, \alpha, m\rangle \xrightarrow{n} \langle p, \alpha, m\rangle \tag{1.7}$$

$$\langle v := a; p, \alpha, m\rangle \xrightarrow{n} \langle p, \alpha, m'\rangle \qquad \text{with } m' = m[v/\mathit{eval}(a)] \tag{1.8}$$

$$\langle \texttt{break}; p, \alpha, m\rangle \xrightarrow{n} \langle \texttt{break}, \alpha, m\rangle \tag{1.9}$$

$$\frac{\langle p_1, \alpha, m\rangle \xrightarrow{n} \langle p_1', \alpha', m'\rangle}{\langle p_1; p_2, \alpha, m\rangle \xrightarrow{n} \langle p_1'; p_2, \alpha', m'\rangle} \qquad \text{if } p_1 \neq \texttt{skip}, v := a, \texttt{break} \tag{1.10}$$

Loops

$$\langle \texttt{loop}\; p, \alpha, m\rangle \xrightarrow{n} \langle p \,\texttt{@loop}\; p, \alpha, m\rangle \tag{1.11}$$

$$\langle \texttt{skip}\,\texttt{@loop}\; p, \alpha, m\rangle \xrightarrow{n} \langle \texttt{loop}\; p, \alpha, m\rangle \tag{1.12}$$

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

$$\langle \texttt{break}\,\texttt{@loop}\; p, \alpha, m\rangle \xrightarrow{n} \langle \texttt{skip}, \alpha, m\rangle \tag{1.14}$$

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

Par/and

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

$$\langle \texttt{skip and}\; p, \alpha, m\rangle \xrightarrow{n} \langle p, \alpha, m\rangle \tag{1.16}$$

$$\langle p \;\texttt{and skip}, \alpha, m\rangle \xrightarrow{n} \langle p, \alpha, m\rangle \qquad \text{if } \mathit{blocked}(p, \alpha, n) = 1 \tag{1.17}$$

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

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

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

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

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

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

Par/or

$$\langle \texttt{skip or } p, \alpha, m\rangle \xrightarrow{n} \langle p', \alpha, m\rangle \quad \text{with } p' = \mathit{clear}(p) \tag{1.24}$$

$$\langle p \texttt{ or skip}, \alpha, m\rangle \xrightarrow{n} \langle p', \alpha, m\rangle \quad \begin{array}{l}\text{if } \mathit{blocked}(p, \alpha, n) = 1,\\ \text{with } p' = \mathit{clear}(p)\end{array} \tag{1.25}$$

$$\langle v := a \texttt{ or } p, \alpha, m\rangle \xrightarrow{n} \langle p', \alpha, m'\rangle \quad \begin{array}{l}\text{with } m' = m[v/\mathit{eval}(a)]\\ \text{and } p' = \mathit{clear}(p)\end{array} \tag{1.26}$$

$$\langle p \texttt{ or } v := a, \alpha, m\rangle \xrightarrow{n} \langle p', \alpha, m'\rangle \quad \begin{array}{l}\text{if } \mathit{blocked}(p, \alpha, n) = 1,\\ \text{with } m' = m[v/\mathit{eval}(a)]\\ \text{and } p' = \mathit{clear}(p)\end{array} \tag{1.27}$$

$$\langle \texttt{break or } p, \alpha, m\rangle \xrightarrow{n} \langle p'; \texttt{break}, \alpha, m\rangle \quad \text{with } p' = \mathit{clear}(p) \tag{1.28}$$

$$\langle p \texttt{ or break}, \alpha, m\rangle \xrightarrow{n} \langle p'; \texttt{break}, \alpha, m\rangle \quad \begin{array}{l}\text{if } \mathit{blocked}(p, \alpha, n) = 1,\\ \text{with } p' = \mathit{clear}(p)\end{array} \tag{1.29}$$

$$\frac{\langle p_1, \alpha, m\rangle \xrightarrow{n} \langle p_1', \alpha', m'\rangle}{\langle p_1 \texttt{ or } p_2, \alpha, m\rangle \xrightarrow{n} \langle p_1' \texttt{ or } p_2, \alpha', m'\rangle} \quad \begin{array}{l}\text{if } \mathit{blocked}(p_1, \alpha, n) = 0\\ \text{and } p_1 \neq \texttt{skip},\\ v := a, \texttt{break}\end{array} \tag{1.30}$$

$$\frac{\langle p_2, \alpha, m\rangle \xrightarrow{n} \langle p_2', \alpha', m'\rangle}{\langle p_1 \texttt{ or } p_2, \alpha, m\rangle \xrightarrow{n} \langle p_1 \texttt{ or } p_2', \alpha', m'\rangle} \quad \begin{array}{l}\text{if } \mathit{blocked}(p_1, \alpha, n) = 1\\ \text{and } p_2 \neq \texttt{skip},\\ v := a, \texttt{break}\end{array} \tag{1.31}$$

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, $p_2 \in P$, α, α_1, $\alpha_2 \in E^*$, m, m_1, $m_2 \in \mathcal{M}$, and $n \in N$,*

$$\textit{if} \quad \langle p, \alpha, m\rangle \xrightarrow{n} \langle p_1, \alpha_1, m_1\rangle \quad \textit{and} \quad \langle p, \alpha, m\rangle \xrightarrow{n} \langle p_2, \alpha_2, m_2\rangle,$$
$$\textit{then} \quad \langle p_1, \alpha_1, m_1\rangle = \langle p_2, \alpha_2, m_2\rangle.$$

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 the implication is trivially true for $p = \texttt{skip}, v := a, \texttt{break}$, as there are no rules to evaluate such programs.)

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

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

[Case 3] $p = \texttt{emit}(e)$. Then d_1 and d_2 are instances of axiom (1.3), and as such, $p_1 = p2 = \texttt{@emitting}(n')$ with $n' = |\alpha|$, and $\alpha_1 = \alpha_2 = e\alpha$ and $m_1 = m_2 = m$.

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

[Case 5] $p = \texttt{if}\, b\, \texttt{then}\, p'\, \texttt{else}\, p''$.

[Case 5.1] $eval(b, m) = 1$. 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] $eval(b, m) = 0$. 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] $p = p'; p''$.

[Case 6.1] $p' = \texttt{skip}$. 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] $p' = v := a$. 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, $m_1 = m_2 = m[v/eval(a)]$.

[Case 6.3] $p' = \texttt{break}$. 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] $p' \neq \texttt{skip}, v := a, \texttt{break}$. 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 $d_1' \prec d_1$ and $d_2' \prec d_2$, 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] $p = \texttt{loop}\, p'$. Then d_1 and d_2 are instances of axiom (1.11), and as such, $p_1 = p_2 = p'\, \texttt{@loop}\, p'$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 8] $p = p'\, \texttt{@loop}\, p''$.

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

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

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

[Case 8.4] $p = p'\, \texttt{@loop}\, p''$ with $p' \neq \texttt{skip}, v := a, \texttt{break}$. 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 $d_1' \prec d_1$ and $d_2' \prec d_2$, by induction hypothesis, $\alpha_1 = \alpha$, $m_1 = m_2$, and $p_1' = p_2'$, which implies

$$p_1 = p_1'\, \texttt{@loop}\, p'' = p_2'\, \texttt{@loop}\, p'' = p_2.$$

[Case 9] $p = p'$ and p''.

[Case 9.1] $p' = \texttt{skip}$. 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] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = \texttt{skip}$. If $blocked(p', \alpha, m) = 0$, this case becomes Case 9.7. Otherwise, if $blocked(p', \alpha, m) = 1$, d_1 and d_2 are instances of axiom (1.17), and as such, $p_1 = p_2 = p'$, $\alpha_1 = \alpha_2 = \alpha$, and $m_1 = m_2 = m$.

[Case 9.3] $p' = v := a$. 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, $m_1 = m_2 = m[v/eval(a)]$.

[Case 9.4] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = v := a$. If $blocked(p', \alpha, m) = 0$, this case becomes Case 9.7. Otherwise, if $blocked(p', \alpha, m) = 1$, 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, $m_1 = m_2 = m[v/eval(a)]$.

[Case 9.5] $p' = \texttt{break}$. 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, $p_1 = p_2 = clear(p''); \texttt{break}$.

[Case 9.6] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = \texttt{break}$. If $blocked(p', \alpha, m) = 0$, this case becomes Case 9.7. Otherwise, if $blocked(p', \alpha, m) = 1$, 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, $p_1 = p_2 = clear(p'); \texttt{break}$.

[Case 9.7] $p', p'' \neq \texttt{skip}, v := a, \texttt{break}$. If $blocked(p', \alpha, m) = 0$, 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 $d_1' \prec d_1$ and $d_2' \prec d_2$, 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, $blocked(p', \alpha, m) = 1$, 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 $d_1'' \prec d_1$ and $d_2'' \prec d_2$, 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] $p = p'$ or p''.

[Case 10.1] $p' = \texttt{skip}$. 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] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = \texttt{skip}$. If $\mathit{blocked}(p', \alpha, m) = 0$, this case becomes Case 10.7. Otherwise, if $\mathit{blocked}(p', \alpha, m) = 1$, 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, $p_1 = p_2 = \mathit{clear}(p')$.

[Case 10.3] $p' = v := a$. 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, $m_1 = m_2 = m[v/\mathit{eval}(a)]$ and $p_1 = p_2 = \mathit{clear}(p'')$.

[Case 10.4] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = v := a$. If $\mathit{blocked}(p', \alpha, m) = 0$, this case becomes Case 10.7. Otherwise, if $\mathit{blocked}(p', \alpha, m) = 1$, 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, $m_1 = m_2 = m[v/\mathit{eval}(a)]$ and $p_1 = p_2 = \mathit{clear}(p')$.

[Case 10.5] $p' = \texttt{break}$. 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, $p_1 = p_2 = \mathit{clear}(p''); \texttt{break}$.

[Case 10.6] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = \texttt{break}$. If $\mathit{blocked}(p', \alpha, m) = 0$, this case becomes Case 10.7. Otherwise, if $\mathit{blocked}(p', \alpha, m) = 1$, 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, $p_1 = p_2 = \mathit{clear}(p'); \texttt{break}$.

[Case 10.7] $p', p'' \neq \texttt{skip}, v := a, \texttt{break}$. If $\mathit{blocked}(p', \alpha, m) = 0$, 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 $d_1' \prec d_1$ and $d_2' \prec d_2$, by induction hypothesis, $\alpha_1 = \alpha_2$, $m_1 = m_2$, and $p_1' = p_2'$, which implies

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

If, however, $\mathit{blocked}(p', \alpha, m) = 1$, 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 $d_1'' \prec d_1$ and $d_2'' \prec d_2$, by induction hypothesis, $\alpha_1 = \alpha_2$, $m_1 = m_2$, and $p_1'' = p_2''$, which implies

$$p_1 = (p' \texttt{ or } p_1'') = (p' \texttt{ or } p_2'') = p_2. \qquad \square$$

The next lemma establishes that given a nontrivial program p either it is possible to advance p by an inner-step or all its trails are blocked, but not both. [FIXME: Faltou incluir $\texttt{fin}\, p$ na lista dos programas triviais, i.e., que não avançam.]

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

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

but not both.

Proof. By induction on the structure of programs.

[Case 1] $p = \texttt{await}(e)$. Then by axiom (1.1),

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

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

[Case 2] $p = \texttt{@awaiting}(e, n')$.

[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),

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

and by Definition 1.1, $\mathit{blocked}(\texttt{@awaiting}(e, n'), \alpha, n) = 0$.
If, however, $e \neq \alpha_{[1]}$, then there is no such δ, as no rule is applicable. And by Definition 1.1, $\mathit{blocked}(\texttt{@awaiting}(e, n'), \alpha, n) = 1$.

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

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

E $\mathit{blocked}(\texttt{@awaiting}(e, n'), \alpha, n) = 1$. 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

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

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, $\mathit{blocked}(\texttt{@awaiting}(e, n'), \alpha, n) = 1$.

[Case 3] $p = \texttt{emit}(e)$. Then by axiom (1.3),

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

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

[Case 4] $p = \texttt{@emitting}(e, n')$.

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

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

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

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

[Case 5] $p = \texttt{if}\, b\, \texttt{then}\, p'\, \texttt{else}\, p''$. By axioms (1.5) and (1.6), if $eval(b, m) = 1$, $\delta = \langle p', \alpha, m\rangle$, otherwise $\delta = \langle p'', \alpha, m\rangle$. And by Definition 1.1,

$$blocked(\texttt{if}\, b\, \texttt{then}\, p'\, \texttt{else}\, p'', e\alpha, n) = 0.$$

[Case 6] $p = p'; p''$.

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

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

[Case 6.2] $p' = v := a$. By axiom (1.8), $\delta = \langle p'', \alpha, m[v/eval(a)]\rangle$, and by Definition 1.1,

$$blocked(v := a; p'', \alpha, n) = blocked(v := a, \alpha, n) = 0.$$

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

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

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

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

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

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

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

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

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

[Case 8] $p = p'\, \texttt{@loop}\, p''$.

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

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

[Case 8.2] $p' = v := a$. By axiom (1.13), $\delta = \langle \texttt{loop}\, p'', \alpha, m[v/eval(a)] \rangle$, and by Definition 1.1,

$$blocked(v := a\, \texttt{@loop}\, p'', \alpha, n) = blocked(v := a, \alpha, n) = 0.$$

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

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

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

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

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

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

and by Definition 1.1,

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

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

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

[Case 9] $p = p'$ and p''.

[Case 9.1] $p' = \texttt{skip}$. By axiom (1.16), $\delta = \langle p'', \alpha, m \rangle$, and by Definition 1.1,

$$\begin{aligned} blocked(\texttt{skip and}\ p'', \alpha, n) &= blocked(\texttt{skip}, \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 0. \end{aligned}$$

[Case 9.2] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = \texttt{skip}$. By induction hypothesis, exactly one of the following hold:

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

If $blocked(p', \alpha, n) = 0$, this case becomes Case 9.7.
If, however, $blocked(p', \alpha, n) = 1$, then by axiom (1.17), $\delta = \langle p', \alpha, m \rangle$, and by Definition 1.1,

$$\begin{aligned} blocked(p'\ \texttt{and skip}, \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(\texttt{skip}, \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

[Case 9.3] $p' = v \coloneqq a$. By axiom (1.18), $\delta = \langle p'', \alpha, m[v/eval(a)]\rangle$, and by Definition 1.1

$$\begin{aligned} blocked(v \coloneqq a \texttt{ and } p'', \alpha, n) &= blocked(v \coloneqq a, \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 0. \end{aligned}$$

[Case 9.4] $p' \neq \texttt{skip}, v \coloneqq a, \texttt{break}$ and $p'' = v \coloneqq a$. By induction hypothesis, exactly one of the following hold:

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

If $blocked(p', \alpha, n) = 0$, this case becomes Case 9.7.
If, however, $blocked(p', \alpha, n) = 1$, then by axiom (1.19), $\delta = \langle p', \alpha, m[v/eval(a)]\rangle$, and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ and } v \coloneqq a, \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(v \coloneqq a, \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

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

$$\begin{aligned} blocked(\texttt{break and } p'') &= blocked(\texttt{break}) \cdot blocked(p'', \alpha, n) \\ &= 0. \end{aligned}$$

[Case 9.6] $p' \neq \texttt{skip}, v \coloneqq a, \texttt{break}$ and $p'' = \texttt{break}$. By induction hypothesis, exactly one of the following hold:

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

If $blocked(p', \alpha, n) = 0$, this case becomes Case 9.7.
If, however, $blocked(p', \alpha, n) = 1$, then by axiom (1.21), $\delta = \langle clear(p'), \alpha, m\rangle$, and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ and } \texttt{break}) &= blocked(p') \cdot blocked(\texttt{break}, \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

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

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

If $\langle p', \alpha, m\rangle \xrightarrow{n} \langle p_1', \alpha', m'\rangle$, for some $p_1' \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$, then by rule (1.22),

$$\langle p' \texttt{ and } p'', \alpha, m\rangle \xrightarrow{n} \langle p_1' \texttt{ and } p'', \alpha', m'\rangle.$$

And by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ and } p'', \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 0 \cdot blocked(p'', \alpha, n) = 0. \end{aligned}$$

If, however, $blocked(p', \alpha, n) = 1$ then, again, by induction hypothesis,

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

but not both. Suppose $\langle p'', \alpha, m\rangle \xrightarrow{n} \langle p''_2, \alpha', m'\rangle$, for some $p''_2 \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.23),

$$\langle p' \texttt{ and } p'', \alpha, m\rangle \xrightarrow{n} \langle p' \texttt{ and } p''_2, \alpha', m'\rangle,$$

and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ and } p'', \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

Suppose $blocked(p'', \alpha, n) = 1$. Then there is no such δ (no rule is applicable) and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ and } p'', \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 1 \cdot 1 = 1. \end{aligned}$$

[Case 10] $p = p' \texttt{ or } p''$.

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

$$\begin{aligned} blocked(\texttt{skip or } p'', \alpha, n) &= blocked(\texttt{skip}, \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 0. \end{aligned}$$

[Case 10.2] $p' \neq \texttt{skip}, v := a, \texttt{break}$ and $p'' = \texttt{skip}$. By induction hypothesis, exactly one of the following hold:

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

If $blocked(p', \alpha, n) = 0$, this case becomes Case 10.7.
If, however, $blocked(p', \alpha, n) = 1$, then by axiom (1.25), $\delta = \langle clear(p'), \alpha, m\rangle$, and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ or skip}, \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(\texttt{skip}, \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

[Case 10.3] $p' = v := a$. By axiom (1.26), $\delta = \langle clear(p''), \alpha, m[v/eval(a)]\rangle$, and by Definition 1.1

$$\begin{aligned} blocked(v := a \texttt{ or } p'', \alpha, n) &= blocked(v := a, \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 0. \end{aligned}$$

[Case 10.4] $p' \neq \texttt{skip}, v \coloneqq a, \texttt{break}$ and $p'' = v \coloneqq a$. By induction hypothesis, exactly one of the following hold:

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

If $\mathit{blocked}(p', \alpha, n) = 0$, this case becomes Case 10.7.
If, however, $\mathit{blocked}(p', \alpha, n) = 1$, then by axiom (1.27),

$$\delta = \langle \mathit{clear}(p'), \alpha, m[v/\mathit{eval}(a)]\rangle,$$

and by Definition 1.1,

$$\begin{aligned} \mathit{blocked}(p' \texttt{ or } v \coloneqq a, \alpha, n) &= \mathit{blocked}(p', \alpha, n) \cdot \mathit{blocked}(v \coloneqq a, \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

[Case 10.5] $p' = \texttt{break}$. By axiom (1.28), $\delta = \langle \mathit{clear}(p''); \texttt{break}, \alpha, m\rangle$, and by Definition 1.1,

$$\begin{aligned} \mathit{blocked}(\texttt{break or } p'') &= \mathit{blocked}(\texttt{break}) \cdot \mathit{blocked}(p'', \alpha, n) \\ &= 0. \end{aligned}$$

[Case 10.6] $p' \neq \texttt{skip}, v \coloneqq a, \texttt{break}$ and $p'' = \texttt{break}$. By induction hypothesis, exactly one of the following hold:

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

If $\mathit{blocked}(p', \alpha, n) = 0$, this case becomes Case 10.7.
If, however, $\mathit{blocked}(p', \alpha, n) = 1$, then by axiom (1.29),

$$\delta = \langle \mathit{clear}(p'); \texttt{break}, \alpha, m\rangle,$$

and by Definition 1.1,

$$\begin{aligned} \mathit{blocked}(p' \texttt{ or } \texttt{break}) &= \mathit{blocked}(p') \cdot \mathit{blocked}(\texttt{break}, \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

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

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

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

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

And by Definition 1.1,

$$\begin{aligned} \mathit{blocked}(p' \texttt{ or } p'', \alpha, n) &= \mathit{blocked}(p', \alpha, n) \cdot \mathit{blocked}(p'', \alpha, n) \\ &= 0 \cdot \mathit{blocked}(p'', \alpha, n) = 0. \end{aligned}$$

If, however, $blocked(p', \alpha, n) = 1$ then, again, by induction hypothesis,

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

but not both. Suppose $\langle p'', \alpha, m\rangle \xrightarrow{n} \langle p_2'', \alpha', m'\rangle$, for some $p_2'' \in P$, $\alpha' \in E^*$, and $m' \in \mathcal{M}$. Then by rule (1.31),

$$\langle p' \texttt{ or } p'', \alpha, m\rangle \xrightarrow{n} \langle p' \texttt{ or } p_2'', \alpha', m'\rangle,$$

and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ or } p'', \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 1 \cdot 0 = 0. \end{aligned}$$

Suppose $blocked(p'', \alpha, n) = 1$. Then there is no such δ (no rule is applicable) and by Definition 1.1,

$$\begin{aligned} blocked(p' \texttt{ or } p'', \alpha, n) &= blocked(p', \alpha, n) \cdot blocked(p'', \alpha, n) \\ &= 1 \cdot 1 = 1. \end{aligned}$$

□

1.3 The reaction outer-step relation

From the previous inner-step relation we define the outer-step relation ($\Rightarrow$) which when necessary pops the event stack and advances blocked programs. The auxiliary function *pop* is used to pop the event stack.

Definition 1.6. Function $pop\colon E^* \to E^*$ is defined as follows.

$$\begin{aligned} pop(\varepsilon) &= \varepsilon \\ pop(e\alpha) &= \alpha \end{aligned}$$

Definition 1.7 (Reaction outer-step)**.**

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

$$\langle p, e\alpha, m\rangle \overset{n}{\Rightarrow} \langle p, \alpha', m\rangle \quad \begin{array}{l} \text{if } blocked(p, \alpha, n) = 1, \\ \text{with } \alpha' = pop(\alpha) \end{array} \tag{1.33}$$

The next two theorem establish that the outer-steps are deterministic and always terminate for nontrivial programs, i.e., that it is in fact the reaction outer-step relation is in fact a *total* function (for nontrivial programs).

Theorem 1.8 (Determinism of the outer-step relation)**.** *For all p, p_1, $p_2 \in P$, α, α_1, $\alpha_2 \in E^*$, m, m_1, $m_2 \in \mathcal{M}$, and $n \in N$,*

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

Proof. The implication is vacuously true for the trivial programs `skip`, $v := a$, and `break`. Suppose p is not one of these trivial programs, and 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. By Lemma 1.5, exactly one of the following hold:

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

[Case 1] $\langle p, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha', m' \rangle = \delta$ and $\mathit{blocked}(p, \alpha, n) = 0$. Then d_1 and d_2 are instances of rule (1.32). Thus $p_1 = p_2 = p'$, $\alpha_1 = \alpha_2 = \alpha'$, and $m_1 = m_2 = m'$.

[Case 2] There is no such δ and $\mathit{blocked}(p, \alpha, n) = 1$. Then d_1 and d_2 are instances of rule (1.33). Thus $p_1 = p_2 = p$, $m_1 = m_2 = m$, and $\alpha_1 = \alpha_2 = \mathit{pop}(\alpha)$. □

Theorem 1.9 (Termination of the outer-step relation). *For all $p \in P$, $\alpha \in E$, and $m \in \mathcal{M}$, if $p \neq$ `skip`$, v := a,$ `break` then*

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

Proof. Let $p \neq$ `skip`$, v := a,$ `break`. By Lemma 1.5, exactly one of the following hold:

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

[Case 1] $\langle p, \alpha, m \rangle \xrightarrow{n} \langle p', \alpha', m' \rangle = \delta$ and $\mathit{blocked}(p, \alpha, n) = 0$. Then, by rule (1.32),

$$\langle p, \alpha, m \rangle \overset{n}{\Rightarrow} \langle p', \alpha', m' \rangle.$$

[Case 2] There is no such δ and $\mathit{blocked}(p, \alpha, n) = 1$. Then, by rule (1.33),

$$\langle p, \alpha, m \rangle \overset{n}{\Rightarrow} \langle p, \mathit{pop}(\alpha), m \rangle.$$ □

1.4 The reaction relation

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

Definition 1.10 (Reaction). Let $p, p' \in P$, $\alpha \in E^*$, $m, m' \in \mathcal{M}$. Then

$$\langle p, \alpha, m \rangle \overset{n}{\vDash} \langle p', m' \rangle \quad \text{iff} \quad \langle p, \alpha, m, n \rangle \overset{*}{\Rightarrow} \langle p', \varepsilon, m', n \rangle.$$

The next two theorems establish, respectively, that reactions are deterministic and always terminate (for nontrivial programs). The proofs of these theorems depend on the assumption that every execution path within the body of a loop instruction (`loop` p) contains an occurrence of `break` or `await`(e). This assumption is checked statically by the Céu compiler.

Na nova semântica basta dizer que e é um evento externo.

[FIXME: Antes do determinismo e terminação precisa de um lema para grantir a correção da definição anterior, i.e., que um número finito de passos eventualmente leva a uma configuração com pilha vazia. Nessa prova vai ser preciso usar a hipótese do loop. A prova deve sair por indução no número de passos.]

Theorem 1.11 (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 $n \in N$,*

$$\textit{if} \quad \langle p, \alpha, m \rangle \overset{n}{\vDash} \langle p_1, m_1 \rangle \quad \textit{and} \quad \langle p, \alpha, m \rangle \overset{n}{\vDash} \langle p_2, m_2 \rangle,$$
$$\textit{then} \quad \langle p_1, m_1 \rangle = \langle p_2, m_2 \rangle.$$

Proof. [TODO: ?] □

Theorem 1.12 (Termination of the reaction relation)**.** *For all $p \in P$, $\alpha \in E$, and $m \in \mathcal{M}$, if $p \neq$ `skip`, $v := a$, `break` then*

$$\langle p, \alpha, m \rangle \overset{n}{\vDash} \langle p', m' \rangle,$$

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

$$\langle \varepsilon, \alpha, m, n \rangle \rightsquigarrow \langle \alpha, m, n \rangle \tag{2.1}$$

Assignment

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

Conditionals

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

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

Await

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

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

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

Emit

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

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

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