

# Calculation of Communication with Open Terms

in GenSpect Process Algebra  
(Draft)

Muck van Weerdenburg

M.J.v.Weerdenburg@student.tue.nl

We recall the definition of the communication function  $\gamma$  of [1].

**Definition 0.1.** Let  $m \in \mathbb{B}(\mathbf{A})$  and  $a \in \mathcal{N}_A$ . Also, let  $\vec{d}, \vec{e} \in \overrightarrow{D_{\mathcal{M}}}$ . The function  $\chi : \mathbb{B}(\mathbf{A}) \times \overrightarrow{D_{\mathcal{M}}} \rightarrow B$  is true if, and only if, all actions of the multiaction parameter have the given data vector as parameter, i.e.  $\chi$  is defined as follows:

$$\begin{aligned} \chi(\square, \vec{d}) &= t \\ \chi([a(\vec{e})] \oplus m, \vec{d}) &= \chi(m, \vec{d}) \quad \text{if } \vec{d} = \vec{e} \\ \chi([a(\vec{e})] \oplus m, \vec{d}) &= f \quad \text{if } \vec{d} \neq \vec{e} \end{aligned}$$

**Definition 0.2.** Let  $N_{\mathbb{B}} = \{n \mid n \in \mathbb{B}(\mathcal{N}_A) \wedge 1 < |n|\}$ ,  $a(\vec{d}) \in \mathbf{A}$ ,  $b \in N_{\mathbb{B}}$  and  $m, n, o \in \mathbb{B}(\mathbf{A})$ . Also let  $C : N_{\mathbb{B}} \rightarrow (\mathcal{N}_A \cup \{\tau\})$  with  $\forall \langle b, a \rangle, \langle c, a \rangle \in C (\forall n \in b (n \notin c))$ . The *communication* function  $\gamma : \mathbb{B}(\mathbf{A}) \times (N_{\mathbb{B}} \rightarrow (\mathcal{N}_A \cup \{\tau\})) \rightarrow \mathbb{B}(\mathbf{A})$  is defined by the following definition:

$$\begin{aligned} \gamma(m \oplus n, C) &= [a(\vec{d})] \oplus \gamma(n, C) \quad \exists \langle b, a \rangle \in C (b = \mu(m) \wedge \chi(m, \vec{d})) \\ \gamma(m \oplus n, C) &= \gamma(n, C) \quad \exists \langle b, \tau \rangle \in C (b = \mu(m) \wedge \chi(m, \vec{d})) \\ \gamma(m, C) &= m \quad \neg \exists_{n, o} (m = n \oplus o \wedge \exists_{c \in C} ((c = \langle b, a \rangle \vee c = \langle b, \tau \rangle) \wedge b = \mu(n) \wedge \exists_{\vec{d} \in \overrightarrow{D}} (\chi(n, \vec{d})))) \end{aligned}$$

When working with open terms one encounters the problem that we may not be able to calculate the value of  $\chi(m, \vec{d})$ . As we wish to calculate the possible communications of a certain multiaction, given some communication function, the result will have to be a set of tuples containing a multiaction resulting from communication and a condition, with terms  $\chi(m, \vec{d})$ , indicating what must hold for this communication to be possible.

But first we reformulate  $\gamma$  to  $\gamma'$  as follows, because Definition 0.2 is not really suitable from a implementation point of view. Note that we somewhat ignore the possibility of right hand sides that are  $\tau$ , but this is not directly relevant for the algorithms. If one desires, one can consider  $[\tau(\vec{d})]$  to be equal to  $\square$  to make things fit.

**Definition 0.3.** Let  $N_{\mathbb{B}} = \{n \mid n \in \mathbb{B}(\mathcal{N}_A) \wedge 1 < |n|\}$ ,  $a(\vec{d}) \in \mathbf{A}$ ,  $b \in N_{\mathbb{B}}$  and  $m, n, o \in \mathbb{B}(\mathbf{A})$ . Also let  $C : N_{\mathbb{B}} \rightarrow (\mathcal{N}_A \cup \{\tau\})$  with  $\forall \langle b, a \rangle, \langle c, a \rangle \in C (\forall n \in b (n \notin c))$ . The *communication* function  $\gamma : \mathbb{B}(\mathbf{A}) \times (N_{\mathbb{B}} \rightarrow (\mathcal{N}_A \cup \{\tau\})) \rightarrow \mathbb{B}(\mathbf{A})$  is defined by the following definition:

$$\begin{aligned} \gamma'(\square, C) &= \square \\ \gamma'([a(\vec{d})] \oplus m, C) &= [a(\vec{d})] \oplus \gamma'(m, C) \quad \neg \exists_{n, o, \langle b, c \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d})) \\ \gamma'([a(\vec{d})] \oplus m, C) &= [c(\vec{d})] \oplus \gamma'(o, C) \quad \exists_{n, \langle b, c \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d})) \end{aligned}$$

**Lemma 0.4.** Definition 0.2 and 0.3 define equivalent functions. That is,  $\gamma(m, C) = \gamma'(m, C)$ , for all  $m$  and  $C$ .

**Proof 0.4.** The defining equations of  $\gamma'$  are complete, so we only need to show that  $\gamma'$  is sound (with respect to  $\gamma$ ). We do this by induction on  $m$ .

Case  $\square$ :

$$\begin{aligned} & \gamma'(\square, C) \\ = & \square \\ = & \gamma(\square, C) \end{aligned}$$

Case  $[a(\vec{d})] \oplus m$ . We do case distinction on the possibility of  $a(\vec{d})$  to participate in a communication. Case  $\exists_{n, \langle b, a \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d}))$ :

$$\begin{aligned} & \gamma'([a(\vec{d})] \oplus m, C) \\ = & [a(\vec{d})] \oplus \gamma'(o, C) \\ = & [c(\vec{d})] \oplus \gamma(o, C) \\ = & \gamma([a(\vec{d})] \oplus n) \oplus o, C) \\ = & \gamma([a(\vec{d})] \oplus (n \oplus o), C) \\ = & \gamma([a(\vec{d})] \oplus m, C) \end{aligned}$$

Case  $\neg \exists_{n, \langle b, a \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d}))$ , with  $m'$  and  $m''$  such that  $\gamma(m, C) = m'' \oplus \gamma(m', C)$  and  $\gamma(m', C) = m'$ :

$$\begin{aligned} & \gamma'([a(\vec{d})] \oplus m, C) \\ = & [a(\vec{d})] \oplus \gamma'(m, C) \\ = & [a(\vec{d})] \oplus \gamma(m, C) \\ = & [a(\vec{d})] \oplus m'' \oplus \gamma(m', C) \\ = & [a(\vec{d})] \oplus m'' \oplus m' \\ = & m'' \oplus [a(\vec{d})] \oplus m' \\ = & m'' \oplus \gamma([a(\vec{d})] \oplus m', C) \\ = & \gamma([a(\vec{d})] \oplus m, C) \end{aligned}$$

□

Taking as basis the new definition, we now define the function we are really interested in. That is, the communication function on open terms. We use the set  $T_{\mathbb{B}}$  of (open) boolean terms and assume that expression depending on action arguments  $\vec{d}$  are such terms.

**Definition 0.5.** Let  $\mathbb{B}(\mathbf{A}')$  be the set of bags of actions with open data parameters. The extension of the communication operator over open data terms  $\bar{\gamma}(m, C) : \mathbb{B}(\mathbf{A}') \times (N_{\mathbb{B}} \rightarrow (\mathcal{N}_{\mathcal{A}} \cup \{\tau\})) \rightarrow \mathcal{P}(\mathbb{B}(\mathbf{A}') \times T_{\mathbb{B}})$  is defined as follows.

$$\begin{aligned} \bar{\gamma}(\square, C) &= \{\{\square, true\}\} \\ \bar{\gamma}([a(\vec{d})] \oplus m, C) &= \{(r, e) \mid \exists_{n, o, \langle b, c \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}(o, c)} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \\ & \quad (e = \chi(n, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r')\} \cup \\ & \quad \{([a(\vec{d})] \oplus r, e \wedge \neg \exists_{n, o, \langle b, c \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \\ & \quad \chi(n, \vec{d}))) \mid \langle r, e \rangle \in \bar{\gamma}(m, C)\} \end{aligned}$$

**Theorem 0.6.** Let  $m \in \mathbb{B}(\mathbf{A}')$  and  $\sigma$  an assignment of variables to closed terms. Then the following holds:

$$\forall_{\langle r, e \rangle \in \bar{\gamma}(m, C)} (e\sigma \Rightarrow r\sigma = \gamma(m, C))$$

Note that we can rewrite  $\neg \exists_{n, o, \langle b, c \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d}))$  as follows.

$$\begin{aligned} & \neg \exists_{n, o, \langle b, c \rangle \in C} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d})) \\ \equiv & \forall_{n, o, \langle b, c \rangle \in C} (\neg (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \chi(n, \vec{d}))) \\ \equiv & \forall_{n, o, \langle b, c \rangle \in C} (\neg (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n)) \vee \neg \chi(n, \vec{d})) \\ \equiv & \forall_{n, o, \langle b, c \rangle \in C} ((m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n)) \Rightarrow \neg \chi(n, \vec{d})) \end{aligned}$$

**Definition 0.7.**

$$\begin{aligned} \bar{\gamma}'(\llbracket, C \rrbracket) &= \{ \langle \llbracket, true \rangle \} \\ \bar{\gamma}'([a(\vec{d})] \oplus m, C) &= \{ \langle r, e \rangle \mid \exists_{n, o, \langle b, c \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o, C)} (m = n \oplus o \wedge b = \mu([a(\vec{d})] \oplus n) \wedge \\ & \quad (e = \chi(n, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \end{aligned}$$

**Lemma 0.8.**

$$\bar{\gamma}([a(\vec{d})] \oplus m, C) = \bar{\gamma}'([a(\vec{d})] \oplus m, C) \cup \{ \langle [a(\vec{d})] \oplus r, e \rangle \mid \forall_{\langle r', e' \rangle \in \bar{\gamma}'([a(\vec{d})] \oplus m, C)} (\neg e') \mid \langle r, e \rangle \in \bar{\gamma}(m, C) \}$$

We now concentrate on  $\bar{\gamma}'$ .

**Definition 0.9.**

$$\phi(m, \vec{d}, w, n, C) = \{ \langle r, e \rangle \mid \exists_{o, o', \langle b, c \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o' \oplus w, C)} (n = o \oplus o' \wedge b = \mu(m \oplus o) \wedge \\ (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \}$$

**Lemma 0.10.**  $\bar{\gamma}'([a(\vec{d})] \oplus m, C) = \phi([a(\vec{d})], \vec{d}, \llbracket, m, C)$

And finally with  $\phi$ :

$$\begin{aligned} & \phi(m, \vec{d}, w, \llbracket, C) \\ = & \{ \langle r, e \rangle \mid \exists_{o, o', \langle b, c \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o' \oplus w, C)} (\llbracket = o \oplus o' \wedge b = \mu(m \oplus o) \wedge \\ & \quad (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle r, e \rangle \mid \exists_{\langle b, c \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(\llbracket \oplus w, C)} (b = \mu(m \oplus \llbracket) \wedge \\ & \quad (e = \chi(\llbracket, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle r, e \rangle \mid \exists_{\langle b, c \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(w, C)} (b = \mu(m) \wedge e = e' \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle [c(\vec{d})] \oplus r', e' \rangle \mid \exists_{\langle b, c \rangle \in C} (b = \mu(m)) \wedge \langle r', e' \rangle \in \bar{\gamma}'(w, C) \} \\ & \phi(m, \vec{d}, w, [a(\vec{f})] \oplus n, C) \\ = & \{ \langle r, e \rangle \mid \exists_{o, o', \langle b, a \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o', C)} ([a(\vec{f})] \oplus n = o \oplus o' \wedge b = \mu(m \oplus o) \wedge \\ & \quad (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \end{aligned}$$

Here  $a(\vec{d})$  can be in  $o$  or in  $o'$ . Assume it is in  $o$ .

$$\begin{aligned} & \{ \langle r, e \rangle \mid \exists_{o, o', \langle b, a \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o' \oplus w, C)} ([a(\vec{f})] \oplus n = o \oplus o' \wedge b = \mu(m \oplus o) \wedge \\ & \quad (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle r, e \rangle \mid \exists_{o, o', \langle b, a \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o' \oplus w, C)} (n = (o \oplus [a(\vec{f})]) \oplus o' \wedge b = \mu(m \oplus [a(\vec{f})] \oplus (o \oplus [a(\vec{f})]))) \wedge \\ & \quad (e = \chi([a(\vec{f})] \oplus (o \oplus [a(\vec{f})]), \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle r, e \rangle \mid \exists_{o'', o', \langle b, a \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o' \oplus w, C)} (n = o'' \oplus o' \wedge b = \mu(m \oplus [a(\vec{f})] \oplus o'') \wedge \\ & \quad (e = (\vec{f} = \vec{d}) \wedge \chi(o'', \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle r, e \wedge (\vec{f} = \vec{d}) \rangle \mid \exists_{o'', o', \langle b, a \rangle \in C, \langle r', e' \rangle \in \bar{\gamma}'(o' \oplus w, C)} (n = o'' \oplus o' \wedge b = \mu(m \oplus [a(\vec{f})] \oplus o'') \wedge \\ & \quad (e = \chi(o'', \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r') \} \\ = & \{ \langle r, e \wedge (\vec{f} = \vec{d}) \rangle \mid \langle r, e \rangle \in \phi(m \oplus [a(\vec{f})], \vec{d}, w, n, C) \} \end{aligned}$$

Now assume it is in  $o'$ .

$$\begin{aligned}
& \{\langle r, e \rangle \mid \exists_{o, o', (b, a) \in C, \langle r', e' \rangle \in \bar{\gamma}(o' \oplus w, C)} ([a(\vec{f})] \oplus n = o \oplus o' \wedge b = \mu(m \oplus o) \wedge \\
& \quad (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r')\} \\
= & \{\langle r, e \rangle \mid \exists_{o, o', (b, a) \in C, \langle r', e' \rangle \in \bar{\gamma}([a(\vec{f})] \oplus (o' \oplus [a(\vec{f})])) \oplus w, C)} (n = o \oplus (o' \oplus [a(\vec{f})]) \wedge b = \mu(m \oplus o) \wedge \\
& \quad (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r')\} \\
= & \{\langle r, e \rangle \mid \exists_{o, o', (b, a) \in C, \langle r', e' \rangle \in \bar{\gamma}([a(\vec{f})] \oplus o' \oplus w, C)} (n = o \oplus o') \wedge b = \mu(m \oplus o) \wedge \\
& \quad (e = \chi(o, \vec{d}) \wedge e') \wedge r = [c(\vec{d})] \oplus r')\} \\
= & \phi(m, \vec{d}, w \oplus [a(\vec{f})], n, C)
\end{aligned}$$

To conclude, we write an algorithm that uses what we have proven.

$$\begin{aligned}
\bar{\gamma}(m, C) = & \llbracket \mathbf{var} S, T : \mathcal{P}(\mathbb{B}(\mathbf{A}') \times T_{\mathbb{B}}); \mathbf{var} b : T_{\mathbb{B}}; \\
& \mid \\
& \mathbf{if} \ m = \square \quad \rightarrow S := \{\langle \square, true \rangle\} \\
& \mid \ m = [a(\vec{d})] \oplus n \rightarrow S, T := \phi([a(\vec{d})], \vec{d}, \square, n, C), \bar{\gamma}(n, C) \\
& \quad ; b := \forall_{\langle r, e \rangle \in S} (\neg e) \\
& \quad ; S := S \cup \{\langle [a(\vec{d})] \oplus r, e \wedge b \rangle \mid \langle r, e \rangle \in T\} \\
& \mathbf{fi} \\
& ; \mathbf{return} S \\
& \rrbracket
\end{aligned}$$

$$\begin{aligned}
\phi(m, \vec{d}, w, n, C) = & \llbracket \mathbf{var} S, T : \mathcal{P}(\mathbb{B}(\mathbf{A}') \times T_{\mathbb{B}}); \\
& \mid \\
& \mathbf{if} \ n = \square \quad \rightarrow \mathbf{if} \ \exists_{(b, c) \in C} (b = \mu(m)) \quad \rightarrow T := \bar{\gamma}(w, C) \\
& \quad ; S := \{\langle [c(\vec{d})] \oplus r, e \rangle \mid \langle r, e \rangle \in T\} \\
& \quad \mid \neg \exists_{(b, c) \in C} (b = \mu(m)) \rightarrow S := \emptyset \\
& \quad \mathbf{fi} \\
& \mid \ n = [a(\vec{f})] \oplus o \rightarrow T := \phi(m \oplus [a(\vec{f})], \vec{d}, w, o, C) \\
& \quad ; T := \{\langle r, e \wedge (\vec{f} = \vec{d}) \rangle \mid \langle r, e \rangle \in T\} \\
& \quad ; S := T \cup \phi(m, \vec{d}, w \oplus [a(\vec{f})], o, C) \\
& \mathbf{fi} \\
& ; \mathbf{return} S \\
& \rrbracket
\end{aligned}$$

If we analyse this algorithm focussing on the length of input  $m$ , we see that it is  $O(2^{|m|})$ . It basically takes the first action in  $m$  and computes the result given that this action participates in a communication and the result given that it does not.

However, looking at  $\phi$ , we can see that the algorithm needlessly tries to find a part in  $n$ , such that  $m$  with this part can communicate, if  $m$  is not even a subbag of a left hand side of a communication in  $C$ . So, we propose to add an extra check to  $\phi$  to prevent this behaviour and making the algorithm more (or precisely) in the order of  $O(2^{|m_1|} + |m_2|)$ , with  $m = m_1 \oplus m_2$  and  $m_1$  contains actions that occur in a left hand side of a communication in  $C$  and  $m_2$  actions that do not.

$$\begin{aligned}
\phi(m, \vec{d}, w, n, C) = & \llbracket \text{var } S, T : \mathcal{P}(\mathbb{B}(\mathbf{A}') \times T_{\mathbb{B}}); \\
& \text{var } b : \text{bool}; \\
& | \\
& b := \exists_{o,c}(\langle \mu(m) \oplus o, c \rangle \in C) \\
& ; \text{if } \neg b \quad \rightarrow S := \emptyset \\
& \quad \parallel b \wedge n = \square \quad \rightarrow \text{if } \exists_{(b,c) \in C}(b = \mu(m)) \quad \rightarrow T := \bar{\gamma}(w, C) \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; S := \{\langle [c(\vec{d})] \oplus r, e \rangle \mid \langle r, e \rangle \in T\} \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \parallel \neg \exists_{(b,c) \in C}(b = \mu(m)) \quad \rightarrow S := \emptyset \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{fi} \\
& \quad \parallel b \wedge n = [a(\vec{f})] \oplus o \quad \rightarrow T := \phi(m \oplus [a(\vec{f})], \vec{d}, w, o, C) \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; T := \{\langle r, e \wedge (\vec{f} = \vec{d}) \rangle \mid \langle r, e \rangle \in T\} \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; S := T \cup \phi(m, \vec{d}, w \oplus [a(\vec{f})], o, C) \\
& \text{fi} \\
& ; \text{return } S \\
& \rrbracket
\end{aligned}$$

Another problem with the above code is that it can generate a lot of negative conditions to indicate that certain communication do not happen. This appears to be at least exponential.

We solve this by removing the problematic  $\forall$  in  $\bar{\gamma}$ . Instead we add an extra parameter to  $\bar{\gamma}$  and  $\phi$  indicating which actions will not communicate. Then, in the final case of  $\bar{\gamma}$ , where  $m = \square$ , we use a new function  $\psi$  to calculate a more reasonable condition indicating that the remaining actions do not communicate.

Note that the following algorithm deviates in a significant way of the previous version, which means that its validity is not guaranteed and additional proofs will be needed.

$$\begin{aligned}
\bar{\gamma}(m, C, r) = & \llbracket \text{var } S, T : \mathcal{P}(\mathbb{B}(\mathbf{A}') \times T_{\mathbb{B}}); \\
& | \\
& \text{if } m = \square \quad \rightarrow S := \{\langle r, \psi(r, C) \rangle\} \\
& \quad \parallel m = [a(\vec{d})] \oplus n \quad \rightarrow S, T := \phi([a(\vec{d})], \vec{d}, \square, n, C, r), \bar{\gamma}(n, C, [a(\vec{d})] \oplus r) \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; S := S \cup T \\
& \text{fi} \\
& ; \text{return } S \\
& \rrbracket
\end{aligned}$$

$$\begin{aligned}
\phi(m, \vec{d}, w, n, C, r) = & \llbracket \text{var } S, T : \mathcal{P}(\mathbb{B}(\mathbf{A}') \times T_{\mathbb{B}}); \\
& \text{var } b : \text{bool}; \\
& | \\
& b := \exists_{o,c}(\langle \mu(m) \oplus o, c \rangle \in C) \\
& ; \text{if } \neg b \quad \rightarrow S := \emptyset \\
& \quad \parallel b \wedge n = \square \quad \rightarrow \text{if } \exists_{(b,c) \in C}(b = \mu(m)) \quad \rightarrow T := \bar{\gamma}(w, C, r) \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; S := \{\langle [c(\vec{d})] \oplus r, e \rangle \mid \langle r, e \rangle \in T\} \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \parallel \neg \exists_{(b,c) \in C}(b = \mu(m)) \quad \rightarrow S := \emptyset \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{fi} \\
& \quad \parallel b \wedge n = [a(\vec{f})] \oplus o \quad \rightarrow T := \phi(m \oplus [a(\vec{f})], \vec{d}, w, o, C, r) \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; T := \{\langle r, e \wedge (\vec{f} = \vec{d}) \rangle \mid \langle r, e \rangle \in T\} \\
& \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad ; S := T \cup \phi(m, \vec{d}, w \oplus [a(\vec{f})], o, C, r) \\
& \text{fi} \\
& ; \text{return } S \\
& \rrbracket
\end{aligned}$$

$$\psi(m, C) = \llbracket \text{var } b : T_{\mathbb{B}}; \\ \mid \\ \text{if } m = [] \rightarrow b := \text{true} \\ \parallel m = [a(\vec{d})] \oplus n \rightarrow b := \psi'(a(\vec{d}), n, C) \wedge \psi(n, C) \\ \text{fi} \\ ; \text{return } b \\ \rrbracket$$

$$\psi'(a(\vec{d}), m, C) = \llbracket \text{var } b : T_{\mathbb{B}}; \\ \text{var } c : \text{bool}; \\ \mid \\ \text{if } m = [] \rightarrow b := \text{true} \\ \parallel m = [b(\vec{e})] \oplus n \rightarrow c := \exists_{o,d}(\langle [a, b] \oplus o, d \rangle \in C) \\ \quad ; \text{if } c \wedge \xi(\langle [a(\vec{d}), b(\vec{e})], n, C \rangle) \rightarrow b := \psi'(a(\vec{d}), n, C) \wedge (\vec{d} \neq \vec{e}) \\ \quad \parallel \neg c \vee \neg \xi(\langle [a(\vec{d}), b(\vec{e})], n, C \rangle) \rightarrow b := \psi'(a(\vec{d}), n, C) \\ \quad \text{fi} \\ \text{fi} \\ ; \text{return } b \\ \rrbracket$$

$$\xi(m, n, C) = \llbracket \text{var } b : \text{bool}; \\ \mid \\ \text{if } n = [] \rightarrow b := \exists_d(\langle m, d \rangle \in C) \\ \parallel n = [a(\vec{d})] \oplus o \rightarrow \text{if } \exists_d(\langle [a] \oplus m, d \rangle \in C) \rightarrow b := \text{true} \\ \quad \parallel \exists_{b,o',d}(\langle [a, b] \oplus m \oplus o', d \rangle \in C) \rightarrow b := \xi([a] \oplus m, o, C) \vee \xi(m, o, C) \\ \quad \parallel \neg \exists_{o',d}(\langle [a] \oplus m \oplus o', d \rangle \in C) \rightarrow b := \xi(m, o, C) \\ \quad \text{fi} \\ \text{fi} \\ ; \text{return } b \\ \rrbracket$$

Naturally, functions  $\psi$  and  $\psi'$  can easily be transformed to the following non-recursive implementation.

$$\psi(m, C) = \llbracket \text{var } b : T_{\mathbb{B}}; \\ \mid \\ b := \text{true} \\ ; \text{do } m = [a(\vec{d})] \oplus n \rightarrow b, m := b \wedge \psi'(a(\vec{d}), n, C), n \\ \text{od} \\ ; \text{return } b \\ \rrbracket$$

$$\begin{aligned}
\psi'(a(\vec{d}), m, C) = & \llbracket \text{var } b : T_{\mathbb{B}}; \\
& \text{var } c : \text{bool}; \\
& | \\
& b := \text{true} \\
& ; \text{ do } m = [b(\vec{e})] \oplus n \rightarrow c := \exists_{o,d}(\langle [a, b] \oplus o, d \rangle \in C) \\
& \quad ; \text{ if } c \wedge \xi(\langle [a(\vec{d}), b(\vec{e})], n, C \rangle \rightarrow b := b \wedge (\vec{d} \neq \vec{e}) \\
& \quad \quad \parallel \neg c \vee \neg \xi(\langle [a(\vec{d}), b(\vec{e})], n, C \rangle \rightarrow \text{skip} \\
& \quad \quad \mathbf{fi} \\
& \quad ; m := n \\
& \text{od} \\
& ; \text{return } b \\
& \rrbracket
\end{aligned}$$

**Theorem 0.11.**

$$\bar{\gamma}(m, C, r) = \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \bar{\gamma}(m, C)\}$$

**Proof 0.11.**

$$\begin{aligned}
& \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \bar{\gamma}(\llbracket \cdot \rrbracket, C)\} \\
= & \{ \} \\
& \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \{\langle \llbracket \cdot \rrbracket \rangle \text{true}\}\} \\
= & \{ \} \\
& \{\langle r \oplus \llbracket \cdot \rrbracket, \text{true} \wedge \psi(r, C) \rangle\} \\
= & \{ \} \\
& \{\langle r, \psi(r, C) \rangle\} \\
= & \{ \} \\
& \bar{\gamma}(\llbracket \cdot \rrbracket, C, r) \\
& \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \bar{\gamma}([a(\vec{d})] \oplus m, C)\} \\
= & \{ \} \\
& \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \bar{\gamma}'([a(\vec{d})] \oplus m, C) \cup \{([a(\vec{d})] \oplus r, e \wedge \forall_{\langle r', e' \rangle \in \bar{\gamma}'([a(\vec{d})] \oplus m, C)}(\neg e')) \mid \langle r, e \rangle \in \bar{\gamma}(m, C)\}\} \\
= & \{ \} \\
& \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C) \cup \{([a(\vec{d})] \oplus r, e \wedge \forall_{\langle r', e' \rangle \in \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C)}(\neg e')) \mid \langle r, e \rangle \in \bar{\gamma}(m, C)\}\} \\
= & \{ \} \\
& \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C)\} \cup \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \{([a(\vec{d})] \oplus r, e \wedge \forall_{\langle r', e' \rangle \in \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C)}(\neg e')) \mid \langle r, e \rangle \in \bar{\gamma}(m, C)\}\} \\
= & \{ \} \\
& \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C, r) \cup \{\langle r \oplus r', e \wedge \psi(r, C) \rangle : \langle r', e \rangle \in \{([a(\vec{d})] \oplus r, e \wedge \forall_{\langle r', e' \rangle \in \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C)}(\neg e')) \mid \langle r, e \rangle \in \bar{\gamma}(m, C)\}\} \\
= & \{ \mathbf{X} \} \\
& \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C, r) \cup \{\langle [a(\vec{d})] \oplus r \oplus r', e \wedge \psi([a(\vec{d})] \oplus r, C) \rangle : \langle r', e \rangle \in \bar{\gamma}(m, C)\} \\
= & \{ \} \\
& \phi([a(\vec{d})], \vec{d}, \llbracket \cdot \rrbracket, m, C, r) \cup \bar{\gamma}(m, C, [a(\vec{d})] \oplus r) \\
= & \{ \} \\
& \bar{\gamma}([a(\vec{d})] \oplus m, C, r)
\end{aligned}$$

□

**Corollary 0.12.**

$$\bar{\gamma}(m, C) = \bar{\gamma}(m, C, \square)$$

**Proof 0.12.**

$$\begin{aligned}
& \bar{\gamma}(m, C, \square) \\
= & \{ \} \\
& \{ \langle \square \oplus r', e \wedge \psi(\square, C) \rangle : \langle r', e \rangle \in \bar{\gamma}(m, C) \} \\
= & \{ \} \\
& \{ \langle r', e \wedge true \rangle : \langle r', e \rangle \in \bar{\gamma}(m, C) \} \\
= & \{ \} \\
& \{ \langle r', e \rangle : \langle r', e \rangle \in \bar{\gamma}(m, C) \} \\
= & \{ \} \\
& \bar{\gamma}(m, C)
\end{aligned}$$

□

## References

- [1] M.J. van Weerdenburg, *GenSpect Process Algebra*, Master's thesis, Eindhoven University of Technology, 2004