$$
\begin{array}{rcl}
M(\texttt{Call c}, \sigma, \sigma_g) & = & \text{let} \quad ((\texttt{p}_1, \ldots \texttt{p}_2), \texttt{b}) = \mathit{funInfo}(\texttt{c.n}) \\
& & \qquad\qquad\qquad\qquad\qquad\quad \sigma_p = \bigcup_{1 \le i \le n} \langle \texttt{p}_i, M(\texttt{c.a}_i, \sigma, \sigma_g) \rangle \\
& & \qquad\qquad\qquad\qquad\quad (\sigma', \sigma'_g, v) = M_p(\texttt{b}, \sigma_p, \sigma_g) \\
& & \text{in} \qquad\qquad\qquad\quad (v, \sigma'_g)
\end{array}
$$

$$
M_p(s, (\sigma, \sigma_g, v)) = \begin{cases}
(\sigma, \sigma_g, M(\texttt{r.e}, \sigma \cup \sigma_g)) & \text{if } s = \texttt{Return r} \\
& \quad \wedge\ v = \textsc{NoVal} \\
\text{let} \quad \sigma' \cup \sigma'_g = M(s, \sigma \cup \sigma_g) & \\
\text{in} \quad (\sigma', \sigma'_g, v) & \text{if } v = \textsc{NoVal} \\
(\sigma, \sigma_g, v) & \text{otherwise}
\end{cases}
$$

$$
\begin{aligned}
allocate(d_1, \ldots, d_k, \mu, \gamma) = \ \text{let} \quad \gamma' &= \gamma \bigcup_{1 \leq i \leq k} \{\langle v_i, a + i - 1 \rangle\} \\
\mu' &= \mu \overline{\cup} \bigcup_{0 \leq i < k} \{\langle a + i - 1 \rangle\} \\
a' &= a + k \\
\text{in} \quad & (\gamma', \mu', a')
\end{aligned}
$$

$$
\begin{array}{rcl}
M(\texttt{Call c}, \overbrace{\gamma, \mu, a, fp}^{\sigma}) & = & \text{let} \quad \sigma' = activate(\texttt{c.n}, (M(\texttt{c.a}_1, \sigma), \\
& & \qquad \ldots M(\texttt{c.a}_k, \sigma), \sigma) \\
& & \quad (\sigma'', v) = M(bodyMap(\texttt{c.n}), \sigma') \\
& & \quad \sigma''' = deactivate(\texttt{c.n}, \sigma'') \\
& & \text{in} \quad (v, \sigma''')
\end{array}
$$

$$
\begin{array}{rcl}
activate(\texttt{id}, (v_1, \ldots v_k), \sigma) & = & \text{let} \quad \gamma', \mu', a' = allocate(dlink, paramMap(\texttt{id}), \sigma) \\
& & \qquad \text{(doesn't change } fp\text{)} \\
& & \quad \mu'' = \mu' \overline{\cup} \{\underbrace{\langle a, fp \rangle}_{dlink}, \langle a+1, v_1 \rangle, \ldots \langle a+k, v_k \rangle\} \\
& & \quad fp' = a \\
& & \text{in} \quad (\gamma', \mu'', a', fp;)
\end{array}
$$

$$
\begin{array}{rcl}
deactivate(\texttt{id}, \sigma) & = & \text{let} \quad fp' = \mu(fp) \\
& & \quad (\gamma', \mu', a') = deallocate(dlink, paramMap(\texttt{id}), \sigma) \\
& & \text{in} \quad (\gamma', \mu', a', fp')
\end{array}
$$