# A symbolic execution semantics for TopHat

Appendices


Nico Naus
Computer Science
Open University of the Netherlands
Heerlen, The Netherlands
nico.naus@ou.nl

Tim Steenvoorden
Software Science
Radboud University
Nijmegen, The Netherlands
tim@cs.ru.nl

Markus Klinik
Software Science
Radboud University
Nijmegen, The Netherlands
m.klinik@cs.ru.nl



**ACM Reference Format:**
Nico Naus, Tim Steenvoorden, and Markus Klinik. 2020. A symbolic execution semantics for TopHat: Appendices. In *Proceedings of International Symposium on Implementation and Application of Functional Languages (IFL'19).* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn


## 1 COMPLETE SYMBOLIC SEMANTICS

### 1.1 Symbolic evaluation rules

$$\boxed{\tilde{e}, \tilde{\sigma} \wr \overline{\tilde{v}, \tilde{\sigma}', \varphi}}$$

SE-Value
$$\frac{}{\tilde{v}, \tilde{\sigma} \wr \tilde{v}, \tilde{\sigma}, \mathsf{True}}$$

SE-Pair
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{v}_1, \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2, \tilde{\sigma}' \wr \overline{\tilde{v}_2, \tilde{\sigma}'', \varphi_2}}{\langle \tilde{e}_1, \tilde{e}_2 \rangle, \tilde{\sigma} \wr \overline{\langle \tilde{v}_1, \tilde{v}_2 \rangle, \tilde{\sigma}'', \varphi_1 \land \varphi_2}}$$

SE-First
$$\frac{\tilde{e}, \tilde{\sigma} \wr \overline{\langle \tilde{v}_1, \tilde{v}_2 \rangle, \tilde{\sigma}', \varphi}}{\mathsf{fst}\, \tilde{e}, \tilde{\sigma} \wr \overline{\tilde{v}_1, \tilde{\sigma}', \varphi}}$$

SE-Second
$$\frac{\tilde{e}, \tilde{\sigma} \wr \overline{\langle \tilde{v}_1, \tilde{v}_2 \rangle, \tilde{\sigma}', \varphi}}{\mathsf{snd}\, \tilde{e}, \tilde{\sigma} \wr \overline{\tilde{v}_2, \tilde{\sigma}', \varphi}}$$

SE-Cons
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \tilde{v}_1, \tilde{\sigma}', \varphi_1 \quad \tilde{e}_2, \tilde{\sigma}' \wr \tilde{v}_2, \tilde{\sigma}'', \varphi_2}{\tilde{e}_1 :: \tilde{e}_2, \tilde{\sigma} \wr \tilde{v}_1 :: \tilde{v}_2, \tilde{\sigma}'', \varphi_1 \land \varphi_2}$$

SE-Head
$$\frac{\tilde{e}, \tilde{\sigma} \wr \tilde{v}_1 :: \tilde{v}_2, \tilde{\sigma}', \varphi}{\mathsf{head}\, \tilde{e}, \tilde{\sigma} \wr \tilde{v}_1, \tilde{\sigma}', \varphi}$$

SE-Tail
$$\frac{\tilde{e}, \tilde{\sigma} \wr \tilde{v}_1 :: \tilde{v}_2, \tilde{\sigma}', \varphi}{\mathsf{tail}\, \tilde{e}, \tilde{\sigma} \wr \tilde{v}_2, \tilde{\sigma}', \varphi}$$

SE-App
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\lambda x : \tau . \tilde{e}_1', \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2, \tilde{\sigma}' \wr \overline{\tilde{v}_2, \tilde{\sigma}'', \varphi_2} \quad \tilde{e}_1'[x \mapsto \tilde{v}_2], \tilde{\sigma}'' \wr \overline{\tilde{v}_1, \tilde{\sigma}''', \varphi_3}}{\tilde{e}_1 \tilde{e}_2, \tilde{\sigma} \wr \overline{\tilde{v}_1, \tilde{\sigma}''', \varphi_1 \land \varphi_2 \land \varphi_3}}$$

SE-If
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{v}_1, \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2, \tilde{\sigma}' \wr \overline{\tilde{v}_2, \tilde{\sigma}'', \varphi_2} \qquad \tilde{e}_3, \tilde{\sigma}' \wr \overline{\tilde{v}_3, \tilde{\sigma}''', \varphi_3}}{\textbf{if}\, \tilde{e}_1\, \textbf{then}\, \tilde{e}_2\, \textbf{else}\, \tilde{e}_3, \tilde{\sigma} \wr \overline{\tilde{v}_2, \tilde{\sigma}'', \varphi_1 \land \varphi_2 \land \tilde{v}_1} \cup \overline{\tilde{v}_3, \tilde{\sigma}''', \varphi_1 \land \varphi_3 \land \neg \tilde{v}_1}}$$

SE-Ref
$$\frac{\tilde{e}, \tilde{\sigma} \wr \overline{\tilde{v}, \tilde{\sigma}', \varphi} \qquad l \notin Dom(\sigma')}{\textbf{ref}\, \tilde{e}, \tilde{\sigma} \wr \overline{l, \tilde{\sigma}'[l \mapsto \tilde{v}], \varphi}}$$

SE-Deref
$$\frac{\tilde{e}, \tilde{\sigma} \wr \overline{l, \tilde{\sigma}', \varphi}}{!\tilde{e}, \tilde{\sigma} \wr \overline{\tilde{\sigma}'(l), \tilde{\sigma}', \varphi}}$$

SE-Assign
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{l, \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2, \tilde{\sigma}' \wr \overline{\tilde{v}_2, \tilde{\sigma}'', \varphi_2}}{\tilde{e}_1 := \tilde{e}_2, \tilde{\sigma} \wr \overline{\langle\rangle, \tilde{\sigma}''[l \mapsto \tilde{v}_2], \varphi_1 \land \varphi_2}}$$

SE-Edit
$$\frac{\tilde{e}, \tilde{\sigma} \wr \overline{\tilde{v}, \tilde{\sigma}', \varphi}}{\square\, \tilde{e}, \tilde{\sigma} \wr \overline{\square\, \tilde{v}, \tilde{\sigma}', \varphi}}$$

SE-Update
$$\frac{\tilde{e}, \tilde{\sigma} \wr \overline{l, \tilde{\sigma}', \varphi}}{\blacksquare\, \tilde{e}, \tilde{\sigma} \wr \overline{\blacksquare\, l, \tilde{\sigma}', \varphi}}$$

SE-Then
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{t}_1, \tilde{\sigma}', \varphi}}{\tilde{e}_1 \blacktriangleright \tilde{e}_2, \tilde{\sigma} \wr \overline{\tilde{t}_1 \blacktriangleright \tilde{e}_2, \tilde{\sigma}', \varphi}}$$

SE-Next
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{t}_1, \tilde{\sigma}', \varphi}}{\tilde{e}_1 \triangleright \tilde{e}_2, \tilde{\sigma} \wr \overline{\tilde{t}_1 \triangleright \tilde{e}_2, \tilde{\sigma}', \varphi}}$$

SE-And
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{t}_1, \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2, \tilde{\sigma}' \wr \overline{\tilde{t}_2, \tilde{\sigma}'', \varphi_2}}{\tilde{e}_1 \bowtie \tilde{e}_2, \tilde{\sigma} \wr \overline{\tilde{t}_1 \bowtie \tilde{t}_2, \tilde{\sigma}'', \varphi_1 \land \varphi_2}}$$

SE-Or
$$\frac{\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{t}_1, \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2, \tilde{\sigma}' \wr \overline{\tilde{t}_2, \tilde{\sigma}'', \varphi_2}}{\tilde{e}_1 \blacklozenge \tilde{e}_2, \tilde{\sigma} \wr \overline{\tilde{t}_1 \blacklozenge \tilde{t}_2, \tilde{\sigma}'', \varphi_1 \land \varphi_2}}$$

SE-Fail
$$\frac{}{\text{↯}, \tilde{\sigma} \wr \text{↯}, \tilde{\sigma}, \mathsf{True}}$$


*IFL'19, September 2019, Singapore*
2020. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

## 1.2 Symbolic striding rules

$$\boxed{\tilde{t}, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}', \tilde{\sigma}', \varphi}}$$

SS-ThenStay
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi}}{\tilde{t}_1 \blacktriangleright \tilde{e}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1' \blacktriangleright \tilde{e}_2, \tilde{\sigma}', \varphi}}\; \mathcal{V}\,(\tilde{t}_1', \tilde{\sigma}') = \bot$$

SS-ThenFail
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi} \qquad \tilde{e}_2\; \tilde{v}_1, \tilde{\sigma}' \;\wr\; \overline{\tilde{t}_2, \tilde{\sigma}'', \_}}{\tilde{t}_1 \blacktriangleright \tilde{e}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1' \blacktriangleright \tilde{e}_2, \tilde{\sigma}', \varphi}}\; \mathcal{V}\,(\tilde{t}_1', \tilde{\sigma}') = \tilde{v}_1 \wedge \mathcal{F}\,(\tilde{t}_2, \tilde{\sigma}'')$$

SS-ThenCont
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi_1} \qquad \tilde{e}_2\; \tilde{v}_1, \tilde{\sigma}' \;\wr\; \overline{\tilde{t}_2, \tilde{\sigma}'', \varphi_2}}{\tilde{t}_1 \blacktriangleright \tilde{e}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{t_2, \sigma'', \varphi_1 \wedge \varphi_2}}\; \mathcal{V}\,(\tilde{t}_1', \tilde{\sigma}') = \tilde{v}_1 \wedge \neg\mathcal{F}\,(\tilde{t}_2, \tilde{\sigma}'')$$

SS-OrLeft
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi}}{\tilde{t}_1 \blacklozenge \tilde{t}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi}}\; \mathcal{V}\,(\tilde{t}_1', \tilde{\sigma}') = \tilde{v}_1$$

SS-OrRight
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi_1} \qquad \tilde{t}_2, \tilde{\sigma}' \;\rightsquigarrow\; \overline{\tilde{t}_2', \tilde{\sigma}'', \varphi_2}}{\tilde{t}_1 \blacklozenge \tilde{t}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_2', \tilde{\sigma}'', \varphi_1 \wedge \varphi_2}}\; \mathcal{V}\,(\tilde{t}_1', \tilde{\sigma}') = \bot \wedge \mathcal{V}\,(\tilde{t}_2', \tilde{\sigma}'') = \tilde{v}_2$$

SS-OrNone
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi_1} \qquad \tilde{t}_2, \tilde{\sigma}' \;\rightsquigarrow\; \overline{\tilde{t}_2', \tilde{\sigma}'', \varphi_2}}{\tilde{t}_1 \blacklozenge \tilde{t}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1' \blacklozenge \tilde{t}_2', \tilde{\sigma}'', \varphi_1 \wedge \varphi_2}}\; \mathcal{V}\,(\tilde{t}_1', \tilde{\sigma}') = \bot \wedge \mathcal{V}\,(\tilde{t}_2', \tilde{\sigma}'') = \bot$$

SS-Edit
$$\frac{}{\square\, \tilde{v}, \tilde{\sigma} \;\rightsquigarrow\; \square\, \tilde{v}, \tilde{\sigma}, \mathsf{True}}$$

SS-Fill
$$\frac{}{\boxtimes\, \beta, \tilde{\sigma} \;\rightsquigarrow\; \boxtimes\, \beta, \tilde{\sigma}, \mathsf{True}}$$

SS-Update
$$\frac{}{\blacksquare\, l, \tilde{\sigma} \;\rightsquigarrow\; \blacksquare\, l, \tilde{\sigma}, \mathsf{True}}$$

SS-Fail
$$\frac{}{\lightning, \tilde{\sigma} \;\rightsquigarrow\; \lightning, \tilde{\sigma}, \mathsf{True}}$$

SS-Xor
$$\frac{}{\tilde{e}_1 \lozenge \tilde{e}_2, \tilde{\sigma} \;\rightsquigarrow\; \tilde{e}_1 \lozenge \tilde{e}_2, \tilde{\sigma}, \mathsf{True}}$$

SS-Next
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi}}{\tilde{t}_1 \triangleright \tilde{e}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1' \triangleright \tilde{e}_2, \tilde{\sigma}', \varphi}}$$

SS-And
$$\frac{\tilde{t}_1, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1', \tilde{\sigma}', \varphi_1} \qquad \tilde{t}_2, \tilde{\sigma}' \;\rightsquigarrow\; \overline{\tilde{t}_2', \tilde{\sigma}'', \varphi_2}}{\tilde{t}_1 \bowtie \tilde{t}_2, \tilde{\sigma} \;\rightsquigarrow\; \overline{\tilde{t}_1' \bowtie \tilde{t}_2', \tilde{\sigma}'', \varphi_1 \wedge \varphi_2}}$$

## 1.3 Symbolic normalisation rules

$$\boxed{\tilde{e}, \tilde{\sigma} \;\wr\wr\; \overline{\tilde{t}, \tilde{\sigma}', \varphi}}$$

SN-Done
$$\frac{\tilde{e}, \tilde{\sigma} \;\wr\; \overline{\tilde{t}, \tilde{\sigma}', \varphi_1} \qquad \tilde{t}, \tilde{\sigma}' \;\rightsquigarrow\; \overline{\tilde{t}', \tilde{\sigma}'', \varphi_2}}{\tilde{e}, \tilde{\sigma} \;\wr\wr\; \overline{\tilde{t}, \tilde{\sigma}', \varphi_1 \wedge \varphi_2}}\; \tilde{\sigma}' = \tilde{\sigma}'' \wedge \tilde{t} = \tilde{t}'$$

SN-Repeat
$$\frac{\tilde{e}, \tilde{\sigma} \;\wr\; \overline{\tilde{t}, \tilde{\sigma}', \varphi_1} \qquad \tilde{t}, \tilde{\sigma}' \;\rightsquigarrow\; \overline{\tilde{t}', \tilde{\sigma}'', \varphi_2} \qquad \tilde{t}', \tilde{\sigma}'' \;\wr\wr\; \overline{\tilde{t}'', \tilde{\sigma}''', \varphi_3}}{\tilde{e}, \tilde{\sigma} \;\wr\wr\; \overline{\tilde{t}'', \tilde{\sigma}''', \varphi_1 \wedge \varphi_2 \wedge \varphi_3}}\; \tilde{\sigma}' \neq \tilde{\sigma}'' \vee \tilde{t} \neq \tilde{t}'$$

### 1.4 Symbolic handling rules

$$\boxed{\tilde t, \tilde\sigma \rightsquigarrow \overline{\tilde t', \tilde\sigma', \tilde\imath, \varphi}}$$

$$\textsc{SH-Change}\quad \frac{\text{fresh } s}{\square\,\tilde v, \tilde\sigma \rightsquigarrow \square\, s, \tilde\sigma, s, \mathsf{True}}\ \tilde v, s : \beta \qquad \textsc{SH-Fill}\quad \frac{\text{fresh } s \quad s : \beta}{\boxtimes\,\beta, \tilde\sigma \rightsquigarrow \square\, s, \tilde\sigma, s, \mathsf{True}} \qquad \textsc{SH-Update}\quad \frac{\text{fresh } s \quad \tilde\sigma(l), s : \beta}{\blacksquare\, l, \tilde\sigma \rightsquigarrow \blacksquare\, l, \tilde\sigma[l \mapsto s], s, \mathsf{True}}$$

$$\textsc{SH-PassNext}\quad \frac{\tilde t_1, \tilde\sigma \rightsquigarrow \overline{\tilde t_1', \tilde\sigma', \tilde\imath, \varphi} \quad \mathcal{V}\,(\tilde t_1', \tilde\sigma') = \bot}{\tilde t_1 \triangleright \tilde e_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1' \triangleright \tilde e_2, \tilde\sigma', \tilde\imath, \varphi}}$$

$$\textsc{SH-PassNextFail}\quad \frac{\tilde t_1, \tilde\sigma \rightsquigarrow \overline{\tilde t_1', \tilde\sigma_1, \tilde\imath, \varphi} \quad \mathcal{V}\,(\tilde t_1', \tilde\sigma_1) = \tilde v_1 \qquad \tilde e_2\ \tilde v_1, \tilde\sigma_1 \Downarrow \overline{\tilde t_2, \tilde\sigma_2, \_} \quad \mathcal{F}\,(\tilde t_2, \tilde\sigma_2)}{\tilde t_1 \triangleright \tilde e_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1' \triangleright \tilde e_2, \tilde\sigma_1, \tilde\imath, \varphi}}$$

$$\textsc{SH-Next}\quad \frac{\tilde t_1, \tilde\sigma \rightsquigarrow \overline{\tilde t_1', \tilde\sigma_1, \tilde\imath, \varphi_1} \quad \tilde e_2\ \tilde v_1, \tilde\sigma_1 \Downarrow \overline{\tilde t_2, \tilde\sigma_2, \varphi_2}}{\tilde t_1 \triangleright \tilde e_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1' \triangleright \tilde e_2, \tilde\sigma_1, \tilde\imath, \varphi_1} \cup \overline{\tilde t_2, \tilde\sigma_2, \mathsf{C}, \varphi_2}}\ \mathcal{V}\,(\tilde t_1', \tilde\sigma_1) = \tilde v_1 \wedge \neg\mathcal{F}\,(\tilde t_2, \tilde\sigma_2)$$

$$\textsc{SH-PassThen}\quad \frac{\tilde t_1, \tilde\sigma \rightsquigarrow \overline{\tilde t_1', \tilde\sigma', \tilde\imath, \varphi}}{\tilde t_1 \blacktriangleright \tilde e_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1' \blacktriangleright \tilde e_2, \tilde\sigma', \tilde\imath, \varphi}}$$

$$\textsc{SH-Pick}\quad \frac{\tilde e_1, \tilde\sigma \Downarrow \overline{\tilde t_1, \tilde\sigma_1, \varphi_1} \quad \tilde e_2, \tilde\sigma \Downarrow \overline{\tilde t_2, \tilde\sigma_2, \varphi_2}}{\tilde e_1 \lozenge \tilde e_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1, \tilde\sigma_1, \mathsf{L}, \varphi_1} \cup \overline{\tilde t_2, \tilde\sigma_2, \mathsf{R}, \varphi_2}}\ \neg\mathcal{F}\,(\tilde t_1, \tilde\sigma_1) \wedge \neg\mathcal{F}\,(\tilde t_2, \tilde\sigma_2)$$

$$\textsc{SH-PickLeft}\quad \frac{\tilde e_1, \tilde\sigma \Downarrow \overline{\tilde t_1, \tilde\sigma_1, \varphi_1} \quad \tilde e_2, \tilde\sigma \Downarrow \overline{\tilde t_2, \tilde\sigma_2, \varphi_2}}{\tilde e_1 \lozenge \tilde e_2, \tilde\sigma \rightsquigarrow \tilde t_1, \tilde\sigma_1, \mathsf{L}, \varphi_1}\ \neg\mathcal{F}\,(\tilde t_1, \tilde\sigma_1) \wedge \mathcal{F}\,(\tilde t_2, \tilde\sigma_2)$$

$$\textsc{SH-PickRight}\quad \frac{\tilde e_1, \tilde\sigma \Downarrow \overline{\tilde t_1, \tilde\sigma_1, \varphi_1} \quad \tilde e_2, \tilde\sigma \Downarrow \overline{\tilde t_2, \tilde\sigma_2, \varphi_2}}{\tilde e_1 \lozenge \tilde e_2, \tilde\sigma \rightsquigarrow \tilde t_2, \tilde\sigma_2, \mathsf{R}, \varphi_2}\ \mathcal{F}\,(\tilde t_1, \tilde\sigma_1) \wedge \neg\mathcal{F}\,(\tilde t_2, \tilde\sigma_2)$$

$$\textsc{SH-And}\quad \frac{\tilde t_1, \tilde\sigma \rightsquigarrow \overline{\tilde t_1', \tilde\sigma_1, \tilde\imath_1, \varphi_1} \quad \tilde t_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_2', \tilde\sigma_2, \tilde\imath_2, \varphi_2}}{\tilde t_1 \bowtie \tilde t_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1' \bowtie \tilde t_2, \tilde\sigma_1, \mathsf{F}\,\tilde\imath_1, \varphi_1} \cup \overline{\tilde t_1 \bowtie \tilde t_2', \tilde\sigma_2, \mathsf{S}\,\tilde\imath_2, \varphi_2}}$$

$$\textsc{SH-Or}\quad \frac{\tilde t_1, \tilde\sigma \rightsquigarrow \overline{\tilde t_1', \tilde\sigma_1, \tilde\imath_1, \varphi_1} \quad \tilde t_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_2', \tilde\sigma_2, \tilde\imath_2, \varphi_2}}{\tilde t_1 \blacklozenge \tilde t_2, \tilde\sigma \rightsquigarrow \overline{\tilde t_1' \blacklozenge \tilde t_2, \tilde\sigma_1, \mathsf{F}\,\tilde\imath_1, \varphi_1} \cup \overline{\tilde t_1 \blacklozenge \tilde t_2', \tilde\sigma_2, \mathsf{S}\,\tilde\imath_2, \varphi_2}}$$

### 1.5 Symbolic driving rules

$$\boxed{\tilde t, \tilde\sigma \Rrightarrow \overline{\tilde t', \tilde\sigma', \tilde\imath, \varphi}}$$

$$\textsc{SI-Handle}\quad \frac{\tilde t, \tilde\sigma \rightsquigarrow \overline{\tilde t', \tilde\sigma', \tilde\imath, \varphi_1} \quad \tilde t', \tilde\sigma' \Downarrow \overline{\tilde t'', \tilde\sigma'', \varphi_2}}{\tilde t, \tilde\sigma \Rrightarrow \overline{\tilde t'', \tilde\sigma'', \tilde\imath, \varphi_1 \wedge \varphi_2}}$$

## 2 TOPHAT SEMANTICS

### 2.1 Typing rules

$$\boxed{\Gamma, \Sigma \vdash e : \tau}$$

$$\text{T-ConstBool}\ \frac{c \in B}{\Gamma, \Sigma \vdash c : \textsc{Bool}} \qquad \text{T-ConstInt}\ \frac{c \in I}{\Gamma, \Sigma \vdash c : \textsc{Int}} \qquad \text{T-ConstString}\ \frac{c \in S}{\Gamma, \Sigma \vdash c : \textsc{String}} \qquad \text{T-Unit}\ \frac{}{\Gamma, \Sigma \vdash \langle\rangle : \textsc{Unit}} \qquad \text{T-Var}\ \frac{x : \tau \in \Gamma}{\Gamma, \Sigma \vdash x : \tau} \qquad \text{T-Loc}\ \frac{\Sigma(l) = \beta}{\Gamma, \Sigma \vdash l : \textsc{Ref}\,\beta}$$

$$\text{T-Pair}\ \frac{\Gamma, \Sigma \vdash e_1 : \tau_1 \quad \Gamma, \Sigma \vdash e_2 : \tau_2}{\Gamma, \Sigma \vdash \langle e_1, e_2\rangle : \tau_1 \times \tau_2} \qquad \text{T-First}\ \frac{\Gamma, \Sigma \vdash e_1 : \tau}{\Gamma, \Sigma \vdash \mathsf{fst}\langle e_1, e_2\rangle : \tau} \qquad \text{T-Second}\ \frac{\Gamma, \Sigma \vdash e_2 : \tau}{\Gamma, \Sigma \vdash \mathsf{snd}\langle e_1, e_2\rangle : \tau} \qquad \text{T-ListEmpty}\ \frac{}{\Gamma, \Sigma \vdash [\,]_\beta : \textsc{List}\,\beta} \qquad \text{T-ListCons}\ \frac{\Gamma, \Sigma \vdash e_1 : \beta \quad \Gamma, \Sigma \vdash e_2 : \textsc{List}\,\beta}{\Gamma, \Sigma \vdash e_1 :: e_2 : \textsc{List}\,\beta}$$

$$\text{T-ListHead}\ \frac{\Gamma, \Sigma \vdash e : \textsc{List}\,\beta}{\Gamma, \Sigma \vdash \mathsf{head}\, e : \beta} \qquad \text{T-ListTail}\ \frac{\Gamma, \Sigma \vdash e : \textsc{List}\,\beta}{\Gamma, \Sigma \vdash \mathsf{tail}\, e : \textsc{List}\,\beta} \qquad \text{T-Abs}\ \frac{\Gamma[x : \tau_1], \Sigma \vdash e : \tau_2}{\Gamma, \Sigma \vdash \lambda x : \tau_1.e : \tau_1 \to \tau_2} \qquad \text{T-App}\ \frac{\Gamma, \Sigma \vdash e_1 : \tau_1 \to \tau_2 \quad \Gamma, \Sigma \vdash e_2 : \tau_1}{\Gamma, \Sigma \vdash e_1 e_2 : \tau_2} \qquad \text{T-Ref}\ \frac{\Gamma, \Sigma \vdash e : \beta}{\Gamma, \Sigma \vdash \mathbf{ref}\, e : \textsc{Ref}\,\beta}$$

$$\text{T-If}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Bool} \quad \Gamma, \Sigma \vdash e_2 : \tau \quad \Gamma, \Sigma \vdash e_3 : \tau}{\Gamma, \Sigma \vdash \mathbf{if}\, e_1\, \mathbf{then}\, e_2\, \mathbf{else}\, e_3 : \tau} \qquad \text{T-Deref}\ \frac{\Gamma, \Sigma \vdash e : \textsc{Ref}\,\beta}{\Gamma, \Sigma \vdash\, !e : \beta} \qquad \text{T-Assign}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Ref}\,\beta \quad \Gamma, \Sigma \vdash e_2 : \beta}{\Gamma, \Sigma \vdash e_1 := e_2 : \textsc{Unit}} \qquad \text{T-Xor}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Task}\,\tau \quad \Gamma, \Sigma \vdash e_2 : \textsc{Task}\,\tau}{\Gamma, \Sigma \vdash e_1 \lozenge e_2 : \textsc{Task}\,\tau}$$

$$\text{T-Edit}\ \frac{\Gamma, \Sigma \vdash e : \beta}{\Gamma, \Sigma \vdash \square\, e : \textsc{Task}\,\beta} \qquad \text{T-Enter}\ \frac{}{\Gamma, \Sigma \vdash \boxtimes\, \beta : \textsc{Task}\,\beta} \qquad \text{T-Update}\ \frac{\Gamma, \Sigma \vdash e : \textsc{Ref}\,\beta}{\Gamma, \Sigma \vdash \blacksquare\, e : \textsc{Task}\,\beta} \qquad \text{T-Fail}\ \frac{}{\Gamma, \Sigma \vdash \lightning : \textsc{Task}\,\tau} \qquad \text{T-Then}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Task}\,\tau_1 \quad \Gamma, \Sigma \vdash e_2 : \tau_1 \to \textsc{Task}\,\tau_2}{\Gamma, \Sigma \vdash e_1 \blacktriangleright e_2 : \textsc{Task}\,\tau_2}$$

$$\text{T-Next}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Task}\,\tau_1 \quad \Gamma, \Sigma \vdash e_2 : \tau_1 \to \textsc{Task}\,\tau_2}{\Gamma, \Sigma \vdash e_1 \rhd e_2 : \textsc{Task}\,\tau_2} \qquad \text{T-And}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Task}\,\tau_1 \quad \Gamma, \Sigma \vdash e_2 : \textsc{Task}\,\tau_2}{\Gamma, \Sigma \vdash e_1 \bowtie e_2 : \textsc{Task}\,(\tau_1 \times \tau_2)} \qquad \text{T-Or}\ \frac{\Gamma, \Sigma \vdash e_1 : \textsc{Task}\,\tau \quad \Gamma, \Sigma \vdash e_2 : \textsc{Task}\,\tau}{\Gamma, \Sigma \vdash e_1 \blacklozenge e_2 : \textsc{Task}\,\tau}$$

### 2.2 Evaluation rules

$$\boxed{e, \sigma \downarrow v, \sigma'}$$

$$\text{E-App}\ \frac{e_1, \sigma \downarrow \lambda x : \tau.e_1', \sigma' \quad e_2, \sigma' \downarrow v_2, \sigma'' \quad e_1'[x \mapsto v_2], \sigma'' \downarrow v_1, \sigma'''}{e_1 e_2, \sigma \downarrow v_1, \sigma'''} \qquad \text{E-IfTrue}\ \frac{e_1, \sigma \downarrow \mathsf{True}, \sigma' \quad e_2, \sigma' \downarrow v_2, \sigma''}{\mathbf{if}\, e_1\, \mathbf{then}\, e_2\, \mathbf{else}\, e_3, \sigma \downarrow v_2, \sigma''} \qquad \text{E-Ref}\ \frac{e, \sigma \downarrow v, \sigma' \quad l \notin Dom(\sigma')}{\mathbf{ref}\, e, \sigma \downarrow l, \sigma'[l \mapsto v]}$$

$$\text{E-IfFalse}\ \frac{e_1, \sigma \downarrow \mathsf{False}, \sigma' \quad e_3, \sigma' \downarrow v_3, \sigma''}{\mathbf{if}\, e_1\, \mathbf{then}\, e_2\, \mathbf{else}\, e_3, \sigma \downarrow v_3, \sigma''} \qquad \text{E-Deref}\ \frac{e, \sigma \downarrow l, \sigma'}{!e, \sigma \downarrow \sigma'(l), \sigma'} \qquad \text{E-Value}\ \frac{}{v, \sigma \downarrow v, \sigma} \qquad \text{E-Assign}\ \frac{e_1, \sigma \downarrow l, \sigma' \quad e_2, \sigma' \downarrow v_2, \sigma''}{e_1 := e_2, \sigma \downarrow \langle\rangle, \sigma''[l \mapsto v_2]} \qquad \text{E-Pair}\ \frac{e_1, \sigma \downarrow v_1, \sigma' \quad e_2, \sigma' \downarrow v_2, \sigma''}{\langle e_1, e_2\rangle, \sigma \downarrow \langle v_1, v_2\rangle, \sigma''}$$

$$\text{E-First}\ \frac{e, \sigma \downarrow \langle v_1, v_2\rangle, \sigma'}{\mathsf{fst}\, e, \sigma \downarrow v_1, \sigma'} \qquad \text{E-Second}\ \frac{e, \sigma \downarrow \langle v_1, v_2\rangle, \sigma'}{\mathsf{snd}\, e, \sigma \downarrow v_2, \sigma'} \qquad \text{E-Cons}\ \frac{e_1, \sigma \downarrow v_1, \sigma' \quad e_2, \sigma' \downarrow v_2, \sigma''}{e_1 :: e_2, \sigma \downarrow v_1 :: v_2, \sigma''} \qquad \text{E-Head}\ \frac{e, \sigma \downarrow v_1 :: v_2, \sigma'}{\mathsf{head}\, e, \sigma \downarrow v_1, \sigma'} \qquad \text{E-Tail}\ \frac{e, \sigma \downarrow v_1 :: v_2, \sigma'}{\mathsf{tail}\, e, \sigma \downarrow v_2, \sigma'}$$

$$\text{E-Edit}\ \frac{e, \sigma \downarrow v, \sigma'}{\square\, e, \sigma \downarrow \square\, v, \sigma'} \qquad \text{E-Update}\ \frac{e, \sigma \downarrow l, \sigma'}{\blacksquare\, e, \sigma \downarrow \blacksquare\, l, \sigma'} \qquad \text{E-Then}\ \frac{e_1, \sigma \downarrow t_1, \sigma'}{e_1 \blacktriangleright e_2, \sigma \downarrow t_1 \blacktriangleright e_2, \sigma'} \qquad \text{E-Next}\ \frac{e_1, \sigma \downarrow t_1, \sigma'}{e_1 \rhd e_2, \sigma \downarrow t_1 \rhd e_2, \sigma'} \qquad \text{E-And}\ \frac{e_1, \sigma \downarrow t_1, \sigma' \quad e_2, \sigma' \downarrow t_2, \sigma''}{e_1 \bowtie e_2, \sigma \downarrow t_1 \bowtie t_2, \sigma''}$$

$$\text{E-Or}\ \frac{e_1, \sigma \downarrow t_1, \sigma' \quad e_2, \sigma' \downarrow t_2, \sigma''}{e_1 \blacklozenge e_2, \sigma \downarrow t_1 \blacklozenge t_2, \sigma''}$$

### 2.3 Striding rules

$$\boxed{t, \sigma \;\mapsto\; t', \sigma'}$$

S-ThenStay
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma'}{t_1 \blacktriangleright e_2, \sigma \mapsto {t_1}' \blacktriangleright e_2, \sigma'} \; \mathcal{V}\,({t_1}', \sigma') = \bot$$

S-ThenFail
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma' \qquad e_2\ v_1, \sigma' \downarrow t_2, \sigma''}{t_1 \blacktriangleright e_2, \sigma \mapsto {t_1}' \blacktriangleright e_2, \sigma'} \; \mathcal{V}\,({t_1}', \sigma') = v_1 \wedge \mathcal{F}\,(t_2, \sigma'')$$

S-ThenCont
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma' \qquad e_2\ v_1, \sigma' \downarrow t_2, \sigma''}{t_1 \blacktriangleright e_2, \sigma \mapsto t_2, \sigma''} \; \mathcal{V}\,({t_1}', \sigma') = v_1 \wedge \neg\mathcal{F}\,(t_2, \sigma'')$$

S-OrLeft
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma'}{t_1 \blacklozenge t_2, \sigma \mapsto {t_1}', \sigma'} \; \mathcal{V}\,({t_1}', \sigma') = v_1$$

S-OrRight
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma' \qquad t_2, \sigma' \mapsto {t_2}', \sigma''}{t_1 \blacklozenge t_2, \sigma \mapsto {t_2}', \sigma''} \; \mathcal{V}\,({t_1}', \sigma') = \bot \wedge \mathcal{V}\,({t_2}', \sigma'') = v_2$$

S-OrNone
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma' \qquad t_2, \sigma' \mapsto {t_2}', \sigma''}{t_1 \blacklozenge t_2, \sigma \mapsto {t_1}' \blacklozenge {t_2}', \sigma''} \; \mathcal{V}\,({t_1}', \sigma') = \bot \wedge \mathcal{V}\,({t_2}', \sigma'') = \bot$$

S-Edit
$$\frac{}{\square\, v, \sigma \mapsto \square\, v, \sigma}$$

S-Fill
$$\frac{}{\boxtimes\, \beta, \sigma \mapsto \boxtimes\, \beta, \sigma}$$

S-Update
$$\frac{}{\blacksquare\, l, \sigma \mapsto \blacksquare\, l, \sigma}$$

S-Fail
$$\frac{}{\lightning, \sigma \mapsto \lightning, \sigma}$$

S-Xor
$$\frac{}{e_1 \Diamond e_2, \sigma \mapsto e_1 \Diamond e_2, \sigma}$$

S-Next
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma'}{t_1 \triangleright e_2, \sigma \mapsto {t_1}' \triangleright e_2, \sigma'}$$

S-And
$$\frac{t_1, \sigma \mapsto {t_1}', \sigma' \qquad t_2, \sigma' \mapsto {t_2}', \sigma''}{t_1 \bowtie t_2, \sigma \mapsto {t_1}' \bowtie {t_2}', \sigma''}$$

### 2.4 Normalisation rules

$$\boxed{e, \sigma \;\Downarrow\; t, \sigma'}$$

N-Done
$$\frac{e, \sigma \downarrow t, \sigma' \qquad t, \sigma' \;\mapsto\; t', \sigma'' \qquad \sigma' = \sigma'' \wedge t = t'}{e, \sigma \Downarrow t, \sigma'}$$

N-Repeat
$$\frac{e, \sigma \downarrow t, \sigma' \qquad t, \sigma' \;\mapsto\; t', \sigma'' \qquad t', \sigma'' \;\Downarrow\; t'', \sigma'''}{e, \sigma \Downarrow t'', \sigma'''} \; \sigma' \neq \sigma'' \vee t \neq t'$$

### 2.5 Handling rules

$$\boxed{t, \sigma \xrightarrow{i} t', \sigma'}$$

H-Change
$$\frac{v, v' : \beta}{\square\, v, \sigma \xrightarrow{v'} \square\, v', \sigma}$$

H-Fill
$$\frac{v : \beta}{\boxtimes\, \beta, \sigma \xrightarrow{v} \square\, v, \sigma}$$

H-Update
$$\frac{\sigma(l), v : \beta}{\blacksquare\, l, \sigma \xrightarrow{v} \blacksquare\, l, \sigma[l \mapsto v]}$$

H-Next
$$\frac{e_2\ v_1, \sigma \Downarrow t_2, \sigma'}{t_1 \triangleright e_2, \sigma \xrightarrow{\mathsf{C}} t_2, \sigma'} \; \mathcal{V}\,(t_1, \sigma) = v_1 \wedge \neg\mathcal{F}\,(t_2, \sigma')$$

H-PickLeft
$$\frac{e_1, \sigma \;\Downarrow\; t_1, \sigma'}{e_1 \Diamond e_2, \sigma \xrightarrow{\mathsf{L}} t_1, \sigma'} \; \neg\mathcal{F}\,(t_1, \sigma')$$

H-PickRight
$$\frac{e_2, \sigma \Downarrow t_2, \sigma'}{e_1 \Diamond e_2, \sigma \xrightarrow{\mathsf{R}} t_2, \sigma'} \; \neg\mathcal{F}\,(t_2, \sigma')$$

H-PassThen
$$\frac{t_1, \sigma \xrightarrow{i} t_1', \sigma'}{t_1 \blacktriangleright e_2, \sigma \xrightarrow{i} t_1' \blacktriangleright e_2, \sigma'}$$

H-PassNext
$$\frac{t_1, \sigma \xrightarrow{i} t_1', \sigma'}{t_1 \triangleright e_2, \sigma \xrightarrow{i} t_1' \triangleright e_2, \sigma'}$$

H-FirstAnd
$$\frac{t_1, \sigma \xrightarrow{i} {t_1}', \sigma'}{t_1 \bowtie t_2, \sigma \xrightarrow{\mathsf{F}\, i} {t_1}' \bowtie t_2, \sigma'}$$

H-SecondAnd
$$\frac{t_2, \sigma \xrightarrow{i} {t_2}', \sigma'}{t_1 \bowtie t_2, \sigma \xrightarrow{\mathsf{S}\, i} t_1 \bowtie {t_2}', \sigma'}$$

H-FirstOr
$$\frac{t_1, \sigma \xrightarrow{i} {t_1}', \sigma'}{t_1 \blacklozenge t_2, \sigma \xrightarrow{\mathsf{F}\, i} {t_1}' \blacklozenge t_2, \sigma'}$$

H-SecondOr
$$\frac{t_2, \sigma \xrightarrow{i} {t_2}', \sigma'}{t_1 \blacklozenge t_2, \sigma \xrightarrow{\mathsf{S}\, i} t_1 \blacklozenge {t_2}', \sigma'}$$

### 2.6 Driving rules

$$\boxed{t, \sigma \;\Rightarrow i\; t', \sigma'}$$

I-Handle
$$\frac{t, \sigma \xrightarrow{i} t', \sigma' \qquad t', \sigma' \Downarrow t'', \sigma''}{t, \sigma \;\Rightarrow i\; t'', \sigma''}$$

## 3 SOUNDNESS PROOFS

### 3.1 Proof of soundness of symbolic evaluation semantics

Proof. We prove Lemma 6.5 by induction over the derivation of the symbolic evaluation $e, \sigma \;\wr\; \overline{\tilde{e}, \tilde{\sigma}, \varphi}$.

**Case** SE-Value

Since this case does not generate constraints, any $M$ will do. Since neither the state, nor the expression is altered by the evaluation rule E-Value, this case holds trivially.

**Case** SE-Fail

Since this case does not generate constraints, any $M$ will do. Since neither the state, nor the expression $\lightning$ is altered by the evaluation rule E-Fail, this case holds trivially.

**Case** SE-Pair

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$,we need to demonstrate that $\langle e_1, e_2\rangle, \sigma \downarrow \langle v_1, v_2\rangle, \sigma''$ with $M\langle\tilde{v}_1, \tilde{v}_2\rangle \equiv \langle v_1, v_2\rangle$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.\tilde{e}_1, \tilde{\sigma} \wr \overline{\tilde{v}_1, \tilde{\sigma}', \varphi_1} \wedge M_1\varphi_1 \supset e_1, \sigma \downarrow v_1, \sigma' \wedge M_1\tilde{v}_1 \equiv v_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \downarrow v_2, \sigma'' \wedge M_2\tilde{v}_2 \equiv v_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Note that we have omitted from the second application of the induction hypothesis, the requirement that the symbolic step exists. The fact that this step exists is obtained from SE-pair and omitted to increase readability of this and any following proofs.

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from E-Pair and the induction steps above that $\langle e_1, e_2\rangle, \sigma \downarrow \langle v_1, v_2\rangle, \sigma''$, $M\langle\tilde{v}_1, \tilde{v}_2\rangle \equiv \langle v_1, v_2\rangle$ and $M\tilde{\sigma}'' \equiv \sigma''$.

**Case** SE-First

For all mappings $M$ such that $M\varphi$, we need to show that $\mathsf{fst}\, e, \sigma \downarrow v_1, \sigma'$ with $M\tilde{v}_1 \equiv v_1$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi \supset e, \sigma \downarrow \langle v_1, v_2\rangle, \sigma' \wedge M_1\langle\tilde{v}_1, \tilde{v}_2\rangle \equiv \langle v_1, v_2\rangle \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi$, we obtain from E-First and the induction step above that $\mathsf{fst}\, e, \sigma \downarrow v_1, \sigma'$ with $M\tilde{v}_1 \equiv v_1 s$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Second

For all mappings $M$ such that $M\varphi$, we need to show that $\mathsf{snd}\, e, \sigma \downarrow v_2, \sigma'$ with $M\tilde{v}_2 \equiv v_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi \supset e, \sigma \downarrow \langle v_1, v_2\rangle, \sigma' \wedge M_1\langle\tilde{v}_1, \tilde{v}_2\rangle \equiv \langle v_1, v_2\rangle \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi$, we obtain from E-Second and the induction step above that $\mathsf{snd}\, e, \sigma \downarrow v_2, \sigma'$ with $M\tilde{v}_2 \equiv v_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Cons

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $e_1 :: e_2, \sigma \downarrow v_1 :: v_2, \sigma''$ with $M\tilde{v}_1 :: \tilde{v}_2 \equiv v_1 :: v_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \downarrow v_1, \sigma' \wedge M_1\tilde{v}_1 \equiv v_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \downarrow v_2, \sigma'' \wedge M_2\tilde{v}_2 \equiv v_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from E-Cons and the induction steps above that $e_1 :: e_2, \sigma \downarrow v_1 :: v_2, \sigma''$ with $M(\tilde{v}_1 :: \tilde{v}_2) \equiv v_1 :: v_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

**Case** SE-Head

For all mappings $M$ such that $M\varphi$, we need to show that $\mathsf{head}\, e, \sigma \downarrow v_1, \sigma'$ with $M\tilde{v}_1 \equiv v_1$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi \supset e, \sigma \downarrow v_1 :: v_2, \sigma' \wedge M_1(\tilde{v}_1 :: \tilde{v}_2) \equiv v_1 :: v_2 \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi$, we obtain from E-Head and the induction step above that $\mathsf{head}\, e, \sigma \downarrow v_1, \sigma'$ with $M\tilde{v}_1 \equiv v_1$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Tail

For all mappings $M$ such that $M\varphi$, we need to show that $\mathsf{tail}\, e, \sigma \downarrow v_2, \sigma'$ with $M\tilde{v}_2 \equiv v_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi \supset e, \sigma \downarrow v_1 :: v_2, \sigma' \wedge M_1(\tilde{v}_1 :: \tilde{v}_2) \equiv v_1 :: v_2 \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi$, we obtain from E-Tail and the induction step above that $\mathsf{tail}\, e, \sigma \downarrow v_2, \sigma'$ with $M\tilde{v}_2 \equiv v_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-App

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2 \wedge \varphi_3)$, we need to demonstrate that $e_1 e_2, \sigma \downarrow v_1, \sigma'''$ with $M\tilde{v}_1 \equiv v_1$ and $M\tilde{\sigma}''' \equiv \sigma'''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \downarrow \lambda x : \tau.e_1', \sigma' \wedge M_1\lambda x : \tau.\tilde{e}_1' \equiv \lambda x : \tau.e_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \downarrow v_2, \sigma'' \wedge M_2\tilde{v}_2 \equiv v_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$ and $\forall M_3.M_3\varphi_3 \supset e_1'[x \mapsto v_2], \sigma'' \downarrow v_1, \sigma''' \wedge M_3\tilde{v}_1 \equiv v_1 \wedge M_3\tilde{\sigma}''' \equiv \sigma'''$.

Since $M$ satisfies $\varphi_1$, $\varphi_2$ and $\varphi_3$, we obtain from E-App and the induction steps above that $e_1 e_2, \sigma \downarrow v_1, \sigma'''$ with $M\tilde{v}_1 \equiv v_1$ and $M\tilde{\sigma}''' \equiv \sigma'''$.

**Case** SE-If

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2 \wedge \tilde{v}_1)$, we need to demonstrate that $\textbf{if}\, e_1\, \textbf{then}\, e_2\, \textbf{else}\, e_3, \sigma \ \downarrow\ v_2, \sigma''$ with $M\tilde{v}_2 = v_2$ and $M\tilde{\sigma}'' = \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \ \downarrow\ v_1, \sigma' \wedge M_1\tilde{v}_1 \equiv v_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \ \downarrow\ v_2, \sigma'' \wedge M_2\tilde{v}_2 \equiv v_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies $\varphi_1$, $\varphi_2$ and $\tilde{v}_1$, we know that $v_1 = \mathsf{True}$.

From E-IfTrue and the induction steps above, we obtain that $\textbf{if}\, e_1\, \textbf{then}\, e_2\, \textbf{else}\, e_3, \sigma \ \downarrow\ v_2, \sigma''$ with $M\tilde{v}_2 = v_2$ and $M\tilde{\sigma}'' = \sigma''$.

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_3 \wedge \neg\tilde{v}_1)$, we need to demonstrate that $\textbf{if}\, e_1\, \textbf{then}\, e_2\, \textbf{else}\, e_3, \sigma \ \downarrow\ v_3, \sigma''$ with $M\tilde{v}_3 = v_3$ and $M\tilde{\sigma}'' = \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \ \downarrow\ v_1, \sigma' \wedge M_1\tilde{v}_1 \equiv v_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_3.M_3\varphi_3 \supset e_3, \sigma' \ \downarrow\ v_3, \sigma'' \wedge M_3\tilde{v}_3 \equiv v_3 \wedge M_3\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies $\varphi_1$, $\varphi_3$ and $\neg\tilde{v}_1$, we know that $v_1 = \mathsf{False}$.

From E-IfFalse and the induction steps above, we obtain that $\textbf{if}\, e_1\, \textbf{then}\, e_2\, \textbf{else}\, e_3, \sigma \ \downarrow\ v_3, \sigma''$ with $M\tilde{v}_3 = v_3$ and $M\tilde{\sigma}'' = \sigma''$.

**Case** SE-Ref

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $\textbf{ref}\, e, \sigma \ \downarrow\ l, \sigma'[l \mapsto v]$ with $Ml \equiv l$ and $M\tilde{\sigma}'[l \mapsto \tilde{v}] \equiv \sigma'[l \mapsto v]$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi \supset e, \sigma \downarrow v, \sigma' \wedge M_1\tilde{v} \equiv v \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from E-Ref and the induction steps above that $\textbf{ref}\, e, \sigma \ \downarrow\ l, \sigma'[l \mapsto v]$.

We assume that the assignment of location references happens in a deterministic manner, and that we can therefore conclude that exactly the same $l$ is used in both cases. Since $l$ cannot contain any symbols, $Ml \equiv l$ holds trivially.

This, together with $M\tilde{\sigma}' \equiv \sigma'$ obtained from the induction hypothesis, we can conclude that $M\tilde{\sigma}'[l \mapsto \tilde{v}] \equiv \sigma'[l \mapsto v]$.

**Case** SE-Deref

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $!e, \sigma \ \downarrow\ \sigma'(l), \sigma'$ with $M\tilde{\sigma}'(l) \equiv \sigma'(l)$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi \supset e, \sigma \downarrow l, \sigma' \wedge M_1 l \equiv l \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from E-Deref and the induction step above that $!e, \sigma \downarrow \sigma'(l), \sigma'$ with $M\tilde{\sigma}'(l) \equiv \sigma'(l)$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Assign

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$, we need to demonstrate that $e_1 := e_2, \sigma \ \downarrow\ \langle\rangle, \sigma''[l \mapsto v_2]$ with $M\langle\rangle \equiv \langle\rangle$, which holds true trivially, and $M\tilde{\sigma}''[l \mapsto \tilde{v}_2] \equiv \sigma''[l \mapsto v_2]$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \downarrow l, \sigma' \wedge M_1 l \equiv l \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \downarrow v_2, \sigma'' \wedge M_2\tilde{v}_2 \equiv v_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from E-Assign and the induction steps above that $e_1 := e_2, \sigma \ \downarrow\ \langle\rangle, \sigma''[l \mapsto v_2]$ with $M\tilde{\sigma}''[l \mapsto \tilde{v}_2] \equiv \sigma''[l \mapsto v_2]$.

**Case** SE-Edit

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $\square\, e, \sigma \ \downarrow\ \square\, v, \sigma'$ with $M \square\, \tilde{v} \equiv \square\, v$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset e, \sigma \downarrow v, \sigma' \wedge M_1\tilde{v} \equiv v \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from E-Edit and the induction step above that $\square\, e, \sigma \ \downarrow\ \square\, v, \sigma'$ with $M \square\, \tilde{v} \equiv \square\, v$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Update

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $\blacksquare\, e, \sigma \ \downarrow\ \blacksquare\, l, \sigma'$ with $M \blacksquare\, l \equiv \blacksquare\, l$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset e, \sigma \downarrow l, \sigma' \wedge M_1 l \equiv l \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from E-Update and the induction step above that $\blacksquare\, e, \sigma \ \downarrow\ \blacksquare\, l, \sigma'$ with $M \blacksquare\, l \equiv \blacksquare\, l$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Then

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $e_1 \blacktriangleright e_2, \sigma \ \downarrow\ t_1 \blacktriangleright e_2, \sigma'$ with $M\tilde{t_1} \blacktriangleright \tilde{e}_2 \equiv t_1 \blacktriangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset e, \sigma \downarrow t_1, \sigma' \wedge M_1\tilde{t}_1 \equiv t_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from E-Then and the induction step above that $e_1 \blacktriangleright e_2, \sigma \ \downarrow\ t_1 \blacktriangleright e_2, \sigma'$ with $M\tilde{t_1} \blacktriangleright \tilde{e}_2 \equiv t_1 \blacktriangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Next

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $e_1 \triangleright e_2, \sigma \ \downarrow\ t_1 \triangleright e_2, \sigma'$ with $M\tilde{t_1} \triangleright e_2 \equiv t_1 \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset e, \sigma \downarrow t_1, \sigma' \wedge M_1\tilde{t}_1 \equiv t_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from E-Next and the induction step above that $e_1 \rhd e_2, \sigma \ \downarrow\ t_1 \rhd e_2, \sigma'$ with $M\tilde{t}_1 \rhd e_2 \ \equiv\ t_1 \rhd e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SE-Or

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$, we need to demonstrate that $e_1 \blacklozenge e_2, \sigma \ \downarrow\ t_1 \blacklozenge t_2, \sigma''$ with $M\tilde{t}_1 \blacklozenge \tilde{t}_2 \equiv t_1 \blacklozenge t_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \downarrow t_1, \sigma' \wedge M_1\tilde{t}_1 \equiv t_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \downarrow t_2, \sigma'' \wedge M_2\tilde{t}_2 \equiv t_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from E-Or and the induction steps above that $e_1 \blacklozenge e_2, \sigma \ \downarrow\ t_1 \blacklozenge t_2, \sigma''$ with $M\tilde{t}_1 \blacklozenge \tilde{t}_2 \equiv t_1 \blacklozenge t_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

**Case** SE-And

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$, we need to demonstrate that $e_1 \bowtie e_2, \sigma \ \downarrow\ t_1 \bowtie t_2, \sigma''$ with $M\tilde{t}_1 \bowtie \tilde{t}_2 \equiv t_1 \bowtie t_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset e_1, \sigma \downarrow t_1, \sigma' \wedge M_1\tilde{t}_1 \equiv t_1 \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset e_2, \sigma' \downarrow t_2, \sigma'' \wedge M_2\tilde{t}_2 \equiv t_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from E-And and the induction steps above that $e_1 \bowtie e_2, \sigma \ \downarrow\ t_1 \bowtie t_2, \sigma''$ with $M\tilde{t}_1 \bowtie \tilde{t}_2 \equiv t_1 \bowtie t_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

□

## 3.2 Proof of soundness of symbolic striding semantics

Proof. We prove Lemma 6.4 by induction over the derivation $t, \sigma \ \rightsquigarrow\ \overline{\tilde{t}, \tilde{\sigma}, \varphi}$.

**Case** SS-ThenStay,SS-ThenFail

For all mappings $M$ such that $M\varphi$ we need to demonstrate that $t_1 \blacktriangleright e_2, \sigma \ \mapsto\ t_1' \blacktriangleright e_2, \sigma'$ with $M\tilde{t}_1' \blacktriangleright e_2 \equiv t_1' \blacktriangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset t_1, \sigma \mapsto t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from S-ThenStay and S-ThenFail respectively, and the induction step above that $t_1 \blacktriangleright e_2, \sigma \ \mapsto\ t_1' \blacktriangleright e_2, \sigma'$ with $M\tilde{t}_1' \blacktriangleright e_2 \equiv t_1' \blacktriangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SS-ThenCont

For all mappings $M$ such that $M\varphi_1 \wedge M\varphi_2$ we need to demonstrate that $t_1 \blacktriangleright e_2, \sigma \ \mapsto\ t_2, \sigma''$ with $M\tilde{t}_2 \equiv t_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi_1 \supset t_1, \sigma \mapsto t_1', \sigma' \supset M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

From Lemma 6.5 we know that $\forall M_2.M_2\varphi_2 \supset e_2 v_1, \sigma' \downarrow t_2, \sigma'' \quad M_2\tilde{t}_2 \equiv t_2 \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from S-ThenCont, the induction step and application of Lemma 6.5 above that $t_1 \blacktriangleright e_2, \sigma \ \mapsto\ t_2, \sigma''$ with $M\tilde{t}_2 \equiv t_2$ and $M\tilde{\sigma}'' \equiv \sigma''$.

**Case** SS-OrLeft

For all mappings $M$ such that $M\varphi$ we have to demonstrate that $t_1 \blacklozenge t_2, \sigma \ \mapsto\ t_1', \sigma'$ with $M\tilde{t}_1' \equiv t_1'$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset t_1, \sigma \mapsto t_1', \sigma' \quad M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from S-OrLeft and the induction step above that $t_1 \blacklozenge t_2, \sigma \ \mapsto\ t_1', \sigma'$ with $M\tilde{t}_1' \equiv t_1'$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SS-OrRight

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$ we need to demonstrate that $t_1 \blacklozenge t_2, \sigma \ \mapsto\ t_2', \sigma''$ with $M\tilde{t}_2' \equiv t_2'$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset t_1, \sigma \mapsto t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset t_2, \sigma' \mapsto t_2', \sigma'' \wedge M_2\tilde{t}_2' \equiv t_2' \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, and from the premise we have that $\mathcal{V}\ (\tilde{t}', \tilde{\sigma}') = \bot$, we obtain from S-OrRight and the induction steps above that $t_1 \blacklozenge t_2, \sigma \ \mapsto\ t_2', \sigma''$ with $M\tilde{t}_2' \equiv t_2'$ and $M\tilde{\sigma}'' \equiv \sigma''$.

**Case** SS-OrNone

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$ we need to demonstrate that $t_1 \blacklozenge t_2, \sigma \ \mapsto\ t_1' \blacklozenge t_2', \sigma''$ with $M\tilde{t}_1' \blacklozenge \tilde{t}_2' \equiv t_1' \blacklozenge t_2'$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset t_1, \sigma \mapsto t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset t_2, \sigma' \mapsto t_2', \sigma'' \wedge M_2\tilde{t}_2' \equiv t_2' \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from S-OrNone and the induction steps above that $t_1 \blacklozenge t_2, \sigma \ \mapsto\ t_1' \blacklozenge t_2', \sigma''$ with $M\tilde{t}_1' \blacklozenge \tilde{t}_2' \equiv t_1' \blacklozenge t_2'$ and $M\tilde{\sigma}'' \equiv \sigma''$.

**Case** SS-Edit

For all mappings $M$, we need to demonstrate that $\square\, v, \sigma \ \mapsto\ \square\, v, \sigma$ with $M \square\, v \equiv \square\, v$ and $M\sigma \equiv \sigma$. This follows trivially from S-Edit.

**Case** SS-FILL

For all mappings $M$, we need to demonstrate that $\boxtimes\beta, \sigma \;\mapsto\; \boxtimes\beta, \sigma$ with $M \boxtimes\beta \equiv \boxtimes\beta$ and $M\sigma \equiv \sigma$. This follows trivially from S-FILL.

**Case** SS-UPDATE

For all mappings $M$, we need to demonstrate that $\blacksquare\, l, \sigma \;\mapsto\; \blacksquare\, l, \sigma$ with $M \blacksquare\, l \equiv \blacksquare\, l$ and $M\sigma \equiv \sigma$. This follows trivially from S-UPDATE.

**Case** SS-FAIL

For all mappings $M$, we need to demonstrate that $\lightning, \sigma \;\mapsto\; \lightning, \sigma$ with $M \lightning \equiv \lightning$ and $M\sigma \equiv \sigma$. This follows trivially from S-FAIL.

**Case** SS-XOR

For all mappings $M$, we need to demonstrate that $e_1 \lozenge e_2, \sigma \;\mapsto\; e_1 \lozenge e_2, \sigma$ with $Me_1 \lozenge e_2 \equiv e_1 \lozenge e_2$ and $M\tilde{\sigma} \equiv \sigma$. This follows trivially from S-XOR.

**Case** SS-NEXT

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $t_1 \rhd e_2, \sigma \;\mapsto\; t_1' \rhd e_2, \sigma'$ with $M\tilde{t}_1' \rhd e_2 \equiv t_1' \rhd e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

From the induction hypothesis, we obtain the following. $\forall M_1.M_1\varphi \supset t_1, \sigma \mapsto t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi$, we obtain from S-NEXT and the induction step above that $t_1 \rhd e_2, \sigma \;\mapsto\; t_1' \rhd e_2, \sigma'$ with $M\tilde{t}_1' \rhd e_2 \equiv t_1' \rhd e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SS-AND

For all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2)$ we need to demonstrate that $t_1 \bowtie t_2, \sigma \;\mapsto\; t_1' \bowtie t_2', \sigma''$ with $M\tilde{t}_1' \bowtie \tilde{t}_2' \equiv t_1' \bowtie t_2'$ and $M\tilde{\sigma}'' \equiv \sigma''$.

From the induction hypothesis, we obtain the following.

$\forall M_1.M_1\varphi_1 \supset t_1, \sigma \mapsto t_1', \sigma' \quad M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset t_2, \sigma' \mapsto t_2', \sigma'' \quad M_2\tilde{t}_2' \equiv t_2' \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain from S-AND and the induction steps above that $t_1 \bowtie t_2, \sigma \;\mapsto\; t_1' \bowtie t_2', \sigma''$ with $M\tilde{t}_1' \bowtie \tilde{t}_2' \equiv t_1' \bowtie t_2'$ and $M\tilde{\sigma}'' \equiv \sigma''$. □

## 3.3 Proof of soundness of symbolic normalisation semantics

PROOF. We prove Lemma 6.3 by induction over the derivation $e, \sigma \;\Downarrow\; \overline{\tilde{t}, \tilde{\sigma}, \varphi}$.

The base case is when the SN-Done rule applies. Provided that $M(\varphi_1 \wedge \varphi_2)$, we need to demonstrate that $e, \sigma \;\Downarrow\; t, \sigma'$ with $M\tilde{t} \equiv t$ and $M\tilde{\sigma}' \equiv \sigma'$.

By Lemma 6.5 and 6.4, we know that

$\forall M_1.M_1\varphi_1 \supset e, \sigma \downarrow t, \sigma' \wedge M_1\tilde{t} \equiv t \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset t, \sigma' \mapsto t', \sigma'' \wedge M_2\tilde{t}' \equiv t' \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we have $e, \sigma \;\Downarrow\; t, \sigma'$ with $M\tilde{\sigma}' \equiv \sigma'$.

The induction step is when SN-REPEAT applies. In this case, for all mappings $M$ such that $M(\varphi_1 \wedge \varphi_2 \wedge \varphi_3)$, we need to demonstrate that $e, \sigma \;\Downarrow\; t'', \sigma'''$ with $M\tilde{t}'' \equiv t''$ and $M\tilde{\sigma}''' \equiv \sigma'''$.

Again by Lemma 6.5 and 6.4, we know that

$\forall M_1.M_1\varphi_1 \supset e, \sigma \downarrow t, \sigma' \wedge M_1\tilde{t} \equiv t \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset t, \sigma' \mapsto t', \sigma'' \wedge M_2\tilde{t}' \equiv t' \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.

Furthermore, we know by applying the induction hypothesis that $\forall M_3.M_3\varphi_3 \supset t', \sigma'' \Downarrow t'', \sigma''' \wedge M_3\tilde{t}'' \equiv t'' \wedge M_3\tilde{\sigma}''' \equiv \sigma'''$.

Since $M$ satisfies $\varphi_1$, $\varphi_2$ and $\varphi_3$, we obtain from N-REPEAT, the application of lemmas and the induction step above that $e, \sigma \;\Downarrow\; t'', \sigma'''$ with $M\tilde{t}'' \equiv t''$ and $M\tilde{\sigma}''' \equiv \sigma'''$. □

## 3.4 Proof of soundness of symbolic handling semantics

PROOF. We prove Lemma 6.2 by induction over the derivation $t, \sigma \;\rightsquigarrow\; \tilde{t}, \tilde{\sigma}, \tilde{\iota}, \varphi$.

**Case** SH-CHANGE

For all mappings $M$, we need to demonstrate that $\square\, v, \sigma \xrightarrow{Ms} \square\, Ms, \sigma$ with $M \square\, s \equiv \square\, Ms$ and $M\sigma \equiv \sigma$.

This follows trivially from H-CHANGE.

**Case** SH-FILL

For all mappings $M$, we need to demonstrate that $\boxtimes\beta, \sigma \xrightarrow{Ms} \square\, Ms, \sigma$ with $M \square\, s \equiv \square\, Ms$ and $M\sigma \equiv \sigma$.

This follows trivially from H-FILL.

**Case** SH-UPDATE

For all mappings $M$, we need to demonstrate that

$\blacksquare\, l, \sigma \xrightarrow{Ms} \blacksquare\, l, \sigma[l \mapsto Ms]$ with $M \blacksquare\, l \equiv \blacksquare\, l$ and $M\sigma[l \mapsto s] \equiv \sigma[l \mapsto Ms]$.

$\blacksquare\, l, \sigma \xrightarrow{Ms} \blacksquare\, l, \sigma[l \mapsto Ms]$ follows trivially from H-Update. $M \blacksquare\, l \equiv \blacksquare\, l$ follows trivially, since locations cannot contain symbols. $M\sigma[l \mapsto s] \equiv \sigma[l \mapsto Ms]$ follows trivially.

**Case** SH-Next

For all mappings $M$ such that $M\varphi_1$, we need to demonstrate that $t_1 \triangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \triangleright e_2, \sigma'$ with $M\tilde{t}_1' \triangleright e_2 \equiv t_1' \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

By the induction hypothesis we obtain the following. $\forall M_1 . M_1\varphi_1 \supset t_1, \sigma \xrightarrow{M_1\tilde{\imath}} t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi_1$, we obtain from H-PassNext and the induction step above that $t_1 \triangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \triangleright e_2, \sigma'$ with $M\tilde{t}_1' \triangleright e_2 \equiv t_1' \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

For all mappings $M$ such that $M\varphi_2$, we need to demonstrate that $t_1 \triangleright e_2, \sigma \xrightarrow{\mathsf{C}} t_2, \sigma'$ with $M\tilde{t}_2 \equiv t_2$ and $M\tilde{\sigma}' \equiv \sigma'$.
From Lemma 6.3 we obtain that $\forall M_1 . M_1\varphi \supset e_2 v_1, \sigma \Downarrow t_2, \sigma' \wedge M\tilde{t}_2 \equiv t_2 \wedge M\tilde{\sigma}' \equiv \sigma'$.
This together with H-Next gives us exactly what we need to prove this case.

**Case** SH-PassNext

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $t_1 \triangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \triangleright e_2, \sigma'$ with $M\tilde{t}_1' \triangleright e_2 \equiv t_1' \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

By the induction hypothesis we obtain the following. $\forall M_1 . M_1\varphi_1 \supset t_1, \sigma \xrightarrow{M_1\tilde{\imath}} t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi$, we obtain from H-PassNext and the induction step above that $t_1 \triangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \triangleright e_2, \sigma'$ with $M\tilde{t}_1' \triangleright e_2 \equiv t_1' \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SH-PassNextFail

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $t_1 \triangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \triangleright e_2, \sigma'$ with $M\tilde{t}_1' \triangleright e_2 \equiv t_1' \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

By the induction hypothesis we obtain the following. $\forall M_1 . M_1\varphi_1 \supset t_1, \sigma \xrightarrow{M_1\tilde{\imath}} t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$.
Since $M$ satisfies $\varphi$ and from the premise of SH-PassNextFail we have $\mathcal{F}\,(\tilde{t}_2, \tilde{\sigma}'')$, we obtain from H-PassNextFail and the induction step above that $t_1 \triangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \triangleright e_2, \sigma'$ with $M\tilde{t}_1' \triangleright e_2 \equiv t_1' \triangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SH-PassThen

For all mappings $M$ such that $M\varphi$, we need to demonstrate that $t_1 \blacktriangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \blacktriangleright e_2, \sigma'$ with $M\tilde{t}_1' \blacktriangleright e_2 \equiv t_1' \blacktriangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

By the induction hypothesis we obtain the following. $\forall M_1 . M_1\varphi_1 \supset t_1, \sigma \xrightarrow{M_1\tilde{\imath}} t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi$, we obtain from H-PassThen and the induction step above that $t_1 \blacktriangleright e_2, \sigma \xrightarrow{M\tilde{\imath}} t_1' \blacktriangleright e_2, \sigma'$ with $M\tilde{t}_1' \blacktriangleright e_2 \equiv t_1' \blacktriangleright e_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SH-Pick

We have that $M\varphi_1$ and/or $M\varphi_2$. In the first case, the proof is identical to the SH-PickLeft rule. In the second case, the proof is identical to the SH-PickRight rule.

**Case** SH-PickLeft

For all mappings $M$ such that $M\varphi_1$, we need to demonstrate that $e_1 \lozenge e_2, \sigma \xrightarrow{\mathsf{L}} t_1, \sigma'$ with $M\tilde{t}_1 \equiv t_1$ and $M\tilde{\sigma}' \equiv \sigma'$.
From Lemma 6.3 we obtain that $\forall M_1 . M_1\varphi \supset e_1, \sigma \Downarrow t_1, \sigma' \wedge M\tilde{t}_1 \equiv t_1 \wedge M\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi_1$, we obtain from H-PickLeft and the application of Lemma 6.3 above that $e_1 \lozenge e_2, \sigma \xrightarrow{\mathsf{L}} t_1, \sigma'$ with $M\tilde{t}_1 \equiv t_1$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SH-PickRight

For all mappings $M$ such that $M\varphi_2$, we need to demonstrate that $e_1 \lozenge e_2, \sigma \xrightarrow{\mathsf{R}} t_2, \sigma'$ with $M\tilde{t}_2 \equiv t_2$ and $M\tilde{\sigma}_2 \equiv \sigma'$.
From Lemma 6.3 we obtain that $\forall M_1 . M_1\varphi \supset e_2, \sigma \Downarrow t_2, \sigma' \wedge M\tilde{t}_2 \equiv t_2 \wedge M\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi_2$, we obtain from H-PickRight and the application of Lemma 6.3 above that $e_1 \lozenge e_2, \sigma \xrightarrow{\mathsf{R}} t_2, \sigma'$ with $M\tilde{t}_2 \equiv t_2$ and $M\tilde{\sigma}_2 \equiv \sigma'$.

**Case** SH-And

For all mappings $M$ such that $M\varphi_1$, we need to demonstrate that $t_1 \bowtie t_2, \sigma \xrightarrow{M\,\mathsf{F}\,\tilde{\imath}} t_1' \bowtie t_2, \sigma'$ with $M\tilde{t}_1' \bowtie t_2 \equiv t_1' \bowtie t_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

By the induction hypothesis we obtain the following. $\forall M_1 . M_1\varphi_1 \supset t_1, \sigma \xrightarrow{M_1\tilde{\imath}} t_1', \sigma' \wedge M_1\tilde{t}_1' \equiv t_1' \wedge M_1\tilde{\sigma}' \equiv \sigma'$.

Since $M$ satisfies $\varphi_1$, we obtain from H-FirstAnd and the induction step above that $t_1 \bowtie t_2, \sigma \xrightarrow{M\,\mathsf{F}\,\tilde{\imath}} t_1' \bowtie t_2, \sigma'$ with $M\tilde{t}_1' \bowtie t_2 \equiv t_1' \bowtie t_2$ and $M\tilde{\sigma}' \equiv \sigma'$.

For all mappings $M$ such that $M\varphi_2$, we need to demonstrate that $t_1 \bowtie t_2, \sigma \xrightarrow{M\,\mathsf{S}\,\tilde{\imath}} t_1 \bowtie t_2', \sigma'$ with $Mt_1 \bowtie \tilde{t}_2' \equiv t_1 \bowtie t_2'$ and $M\tilde{\sigma}' \equiv \sigma'$.

By the induction hypothesis we obtain the following. $\forall M_1.M_1\varphi_1 \supset t_2, \tilde{\sigma} \xrightarrow{M_1\tilde{\imath}} t_2', \sigma' \wedge M_1\tilde{t}_2' \equiv t_2' \wedge M_1\tilde{\sigma}' \equiv \sigma'$

Since $M$ satisfies $\varphi_2$, we obtain from H-SecondAnd and the induction step above that $t_1 \bowtie t_2, \sigma \xrightarrow{M\,\mathsf{S}\,\tilde{\imath}} t_1 \bowtie t_2', \sigma'$ with $Mt_1 \bowtie \tilde{t}_2' \equiv t_1 \bowtie t_2'$ and $M\tilde{\sigma}' \equiv \sigma'$.

**Case** SH-Or

This case is proven in the same way as SH-And.

□

## 3.5 Proof of soundness of symbolic interacting semantics

Proof. We prove Theorem 6.1 by induction on $\tilde{t}, \tilde{\sigma} \rightsquigarrow \overline{\tilde{t}', \tilde{\sigma}', \tilde{\imath}, \varphi}$. There is only one rule that applies, namely SI-Handle.
Provided that $M(\varphi_1 \wedge \varphi_2)$, we need to demonstrate that $t, \sigma \Rightarrow M\tilde{\imath}\ \ t'', \sigma''$ with $M\tilde{t}'' \equiv t''$ and $M\tilde{\sigma}'' \equiv \sigma''$.
Lemma 6.3 and Lemma 6.2 respectively give us that
$\forall M_1.M_1\varphi_1 \supset t, \sigma \xrightarrow{M_1\tilde{\imath}} t', \sigma' \wedge M_1\tilde{t}' \equiv t' \wedge M_1\tilde{\sigma}' \equiv \sigma'$ and $\forall M_2.M_2\varphi_2 \supset t', \sigma' \Downarrow t'', \sigma'' \wedge M_2\tilde{t}'' \equiv t'' \wedge M_2\tilde{\sigma}'' \equiv \sigma''$.
Since $M$ satisfies both $\varphi_1$ and $\varphi_2$, we obtain exactly what we need to prove, namely $t, \sigma \Rightarrow \tilde{\imath}\ \ t'', \sigma''\ M\tilde{t}'' \equiv t''$ and $M\tilde{\sigma}'' \equiv \sigma''$.

□

# 4 COMPLETENESS PROOFS

## 4.1 Proof of completeness of the symbolic handling semantics

Proof. We prove Lemma 6.8 by induction over the derivation $t, \sigma \xrightarrow{i} t', \sigma'$.

**Case** H-Change

By the SH-Change rule, we have $\square\, v, \sigma \rightsquigarrow \square\, s, \tilde{\sigma}, s, \mathsf{True}$, and $s \sim v'$ holds by definition of input simulation.

**Case** H-Fill

By the SH-Fill rule, we have $\boxtimes\, \beta, \sigma \rightsquigarrow \square\, s, \tilde{\sigma}, s, \mathsf{True}$, and $s \sim v$ holds by definition of input simulation.

**Case** H-Update

By the SH-Update rule, we have $\blacksquare\, l, \sigma \rightsquigarrow \blacksquare\, l, \tilde{\sigma}[l \mapsto s], s, \mathsf{True}$, and $s \sim v$ holds by definition of input simulation.

**Case** H-Next

By the SH-Next rule, we have $t_1 \rhd e_2, \sigma \rightsquigarrow \overline{\tilde{t}_1' \rhd e_2, \tilde{\sigma}_1, \tilde{\imath}, \varphi_1} \cup \overline{t_2, \tilde{\sigma}_2, \mathsf{C}, \varphi_2}$, and $\mathsf{C} \sim \mathsf{C}$ holds by definition of input simulation.

**Case** H-PassNext

By application of the induction hypothesis, we obtain the following.
For all $t_1, \sigma, i$ such that $t_1, \sigma \xrightarrow{i} t_1', \sigma'$ there exists an $\tilde{\imath} \sim i$ such that $t_1, \sigma \rightsquigarrow \overline{\tilde{t}_1, \tilde{\sigma}, \tilde{\imath}, \varphi}$. From this we can conclude that there exists a symbolic execution $t_1 \rhd e_2, \sigma \rightsquigarrow \overline{\tilde{t}_1 \rhd e_2, \tilde{\sigma}, \tilde{\imath}, \varphi}$, and that $\tilde{\imath} \sim i$.

**Case** H-PassThen

By application of the induction hypothesis, we obtain the following.
For all $t_1, \sigma, i$ such that $t_1, \sigma \xrightarrow{i} t_1', \sigma'$ there exists an $\tilde{\imath} \sim i$ such that $t_1, \sigma \rightsquigarrow \overline{\tilde{t}_1, \tilde{\sigma}, \tilde{\imath}, \varphi}$. From this we can conclude that there exists a symbolic execution $t_1 \blacktriangleright e_2, \sigma \rightsquigarrow \overline{\tilde{t}_1 \blacktriangleright e_2, \tilde{\sigma}, \tilde{\imath}, \varphi}$, and $\tilde{\imath} \sim i$.

**Case** H-PickLeft

Lemma 6.9 gives us the following.
There exists a symbolic execution $e_1, \sigma \between \overline{\tilde{t}_1, \tilde{\sigma}, \varphi_1}$. There exists a symbolic execution $e_2, \tilde{\sigma} \between \overline{\tilde{t}_2, \tilde{\sigma}', \varphi_2}$.

We can now conclude that a symbolic execution exists. Either by the SH-PickLeft rule, in case $\mathcal{F}\ (\tilde{t}_2, \tilde{\sigma}')$, or by the SH-Pick rule in case $\neg\mathcal{F}\ (\tilde{t}_2, \tilde{\sigma}')$. We have that $\mathsf{L} \sim \mathsf{L}$ holds by definition.

**Case** H-PickRight

Lemma 6.9 gives us the following.
There exists a symbolic execution $e_1, \sigma \between \overline{t_1, \tilde{\sigma}, \varphi_1}$. There exists a symbolic execution $e_2, \tilde{\sigma} \between \overline{t_2, \tilde{\sigma}', \varphi_2}$.

We can now conclude that a symbolic execution exists. Either by the SH-PickRight rule, in case $\mathcal{F}\ (\tilde{t}_1, \tilde{\sigma})$, or by the SH-Pick rule in case $\neg\mathcal{F}\ (t_1, \tilde{\sigma})$.

We have that $\mathsf{R} \sim \mathsf{R}$ holds by definition.

**Case** H-FirstOr

By application of the induction hypothesis, we obtain the following. For all $t_1, \sigma, i$ such that $t_1, \sigma \xrightarrow{i} t_1', \sigma'$ there exists an $\tilde{\imath} \sim i$ such that $t_1, \sigma \rightsquigarrow \tilde{t}_1, \tilde{\sigma}, \tilde{\imath}, \varphi$.

From SH-Or, and the conclusion of the induction hypothesis, we can conclude that there exists a symbolic input, namely $\mathsf{F}\,\tilde{\imath}$, such that $t_1 \blacklozenge t_2, \sigma \rightsquigarrow \overline{\tilde{t}_1' \blacklozenge t_2, \tilde{\sigma}, \mathsf{F}\,\tilde{\imath}, \varphi}$. From $\tilde{\imath} \sim i$ and by definition of input simulation, we can conclude that $\mathsf{F}\,\tilde{\imath} \sim \mathsf{F}\,i$.

**Case** H-SecondOr

By application of the induction hypothesis, we obtain the following. For all $t_2, \sigma, i$ such that $t_2, \sigma \xrightarrow{i} t_2', \sigma'$ there exists an $\tilde{\imath} \sim i$ such that $t_2, \sigma \rightsquigarrow \tilde{t}_2, \tilde{\sigma}, \tilde{\imath}, \varphi$.

From SH-Or, and the induction step above, we can conclude that there exists a symbolic input such that $t_1 \blacklozenge t_2, \sigma \rightsquigarrow \overline{\tilde{t}_1 \blacklozenge t_2', \tilde{\sigma}', \mathsf{S}\,\tilde{\imath}, \varphi}$, namely $\mathsf{S}\,\tilde{\imath}$. From $\tilde{\imath} \sim i$ and by definition of input simulation, we can conclude that $\mathsf{S}\,\tilde{\imath} \sim \mathsf{S}\,i$.

**Case** H-FirstAnd

By application of the induction hypothesis, we obtain the following. For all $t_1, \sigma, i$ such that $t_1, \sigma \xrightarrow{i} t_1', \sigma'$ there exists an $\tilde{\imath} \sim i$ such that $t_1, \sigma \rightsquigarrow \tilde{t}_1, \tilde{\sigma}, \tilde{\imath}, \varphi$.

From SH-And, and the conclusion of the induction step above, we can conclude that there exists a symbolic input, namely $\mathsf{F}\,\tilde{\imath}$ such that $t_1 \bowtie t_2, \sigma \rightsquigarrow \overline{\tilde{t}_1' \bowtie t_2, \tilde{\sigma}, \mathsf{F}\,\tilde{\imath}, \varphi}$. From $\tilde{\imath} \sim i$ and by definition of input simulation, we can conclude that $\mathsf{F}\,\tilde{\imath} \sim \mathsf{F}\,i$.

**Case** H-SecondAnd

By application of the induction hypothesis, we obtain the following. For all $t_2, \sigma, i$ such that $t_2, \sigma \xrightarrow{i} t_2', \sigma'$ there exists an $\tilde{\imath} \sim i$ such that $t_2, \sigma \rightsquigarrow \tilde{t}_2, \tilde{\sigma}, \tilde{\imath}, \varphi$.

From SH-And, and the conclusion of the induction step above, we can conclude that there exists a symbolic input, namely $\mathsf{S}\,\tilde{\imath}$ such that $t_1 \bowtie t_2, \sigma \rightsquigarrow \overline{t_1 \bowtie \tilde{t}_2, \tilde{\sigma}, \mathsf{S}\,\tilde{\imath}, \varphi}$. From $\tilde{\imath} \sim i$ and by definition of input simulation, we can conclude that $\mathsf{S}\,\tilde{\imath} \sim \mathsf{S}\,i$.

□

## 4.2 Proof of completeness of the symbolic interaction semantics

Proof. The proof of Theorem 6.7 consists of one case, since the interacting semantics consists of one rule, namely I-Handle

$$\frac{t, \sigma \xrightarrow{i} t', \sigma' \qquad t', \sigma' \Downarrow t'', \sigma''}{t, \sigma \Rightarrow i \;\; t'', \sigma''}.$$

By Lemma 6.8 we obtain the following. $t, \sigma \xrightarrow{i} t', \sigma' \supset \exists\tilde{\imath}.t, \sigma \rightsquigarrow \tilde{t}, \tilde{\sigma}, \tilde{\imath}, \varphi \wedge \tilde{\imath} \sim i$

Then by Lemma 6.9 we obtain the following. $t', \sigma' \Downarrow t'', \sigma'' \supset t', \sigma' \rightsquigarrow \tilde{t}', \tilde{\sigma}', \varphi'$

From the above, together with the SI-Handle rule, we can conclude that there exists a symbolic execution $t, \sigma \rightsquigarrow \tilde{t}'', \tilde{\sigma}'', \tilde{\imath}, \varphi \wedge \tilde{\imath} \sim i$.

□