A EQUIVALENCE OF THE Px86$_{\text{man}}$ OPERATIONAL AND DECLARATIVE SEMANTICS

A.1 Intermediate Operational Semantics

Types.

Notation. In what follows we write WU for $W \cup U$.

Annotated persistent memory

$$M \in \textsc{AMem} \triangleq \left\{ f \in \textsc{Loc} \xrightarrow{\text{fin}} W \;\middle|\; \forall x \in dom(f).\ \mathtt{loc}(f(x)) = x \right\}$$

Annotated persistent buffers

$$PB \in \textsc{APBuff} \triangleq \textsc{Seq}\,\langle W \cup U \cup FO \cup FL \rangle$$

Annotated volatile buffers

$$b \in \textsc{ABuff}_\tau \triangleq \textsc{Seq}\left\langle W \cup \left\{ \begin{array}{l|l} \langle \mathsf{fo}, fo\rangle, \langle \mathsf{pfo}, fo\rangle, & fo \in FO \wedge \mathtt{tid}(fo){=}\tau \\ \langle \mathsf{fl}, fl\rangle, \langle \mathsf{pfl}, fl\rangle & \wedge\ fl \in FL \wedge \mathtt{tid}(fl){=}\tau \\ \langle \mathsf{sf}, sf\rangle, \langle \mathsf{psf}, sf\rangle & \wedge\ sf \in SF \wedge \mathtt{tid}(sf){=}\tau \end{array} \right\} \right\rangle$$

$$b \in \textsc{ABuff} \triangleq \bigcup_{\tau \in \textsc{TId}} \textsc{ABuff}_\tau$$

Annotated volatile buffer maps

$$B \in \textsc{ABMap} \triangleq \left\{ B \in \textsc{TId} \xrightarrow{\text{fin}} \textsc{ABuff} \;\middle|\; \forall \tau \in dom(B).\ B(\tau) \in \textsc{ABuff}_\tau \right\}$$

Annotated labels

$$\begin{array}{lll}
\textsc{ALabels} \ni \lambda ::= & \mathsf{R}\langle r, e\rangle & \text{where } r \in R, e \in WU, \mathtt{loc}(r){=}\mathtt{loc}(e), \mathtt{val_r}(r){=}\mathtt{val_w}(e) \\
\mid & \mathsf{U}\langle u, e\rangle & \text{where } u \in U, e \in WU, \mathtt{loc}(u){=}\mathtt{loc}(e), \mathtt{val_r}(u){=}\mathtt{val_w}(e) \\
\mid & \mathsf{W}\langle w\rangle & \text{where } w \in W \\
\mid & \mathsf{MF}\langle mf\rangle & \text{where } mf \in MF \\
\mid & \mathsf{SF}\langle sf\rangle & \text{where } sf \in SF \\
\mid & \mathsf{FO}\langle fo\rangle & \text{where } fo \in FO \\
\mid & \mathsf{FL}\langle fl\rangle & \text{where } fl \in FL \\
\mid & \mathsf{PSF}\langle sf\rangle & \text{where } sf \in SF \\
\mid & \mathsf{PFO}\langle fo\rangle & \text{where } fo \in FO \\
\mid & \mathsf{PFL}\langle fl\rangle & \text{where } fl \in FL \\
\mid & \mathsf{B}\langle e\rangle & \text{where } e \in W \cup SF \cup FO \cup FL \\
\mid & \mathsf{J}\langle e\rangle & \text{where } e \in FO \cup FL \cup SF \\
\mid & \mathsf{D}\langle e\rangle & \text{where } e \in FO \cup FL \cup SF \\
\mid & \mathsf{PB}\langle e\rangle & \text{where } e \in W \cup U \cup FO \cup FL \\
\mid & \mathcal{E}\langle \tau\rangle & \text{where } \tau \in \textsc{TId}
\end{array}$$

$$\pi \in \textsc{Path} \triangleq \textsc{Seq}\left\langle \textsc{ALabels} \setminus \left\{ \mathcal{E}\langle\tau\rangle \;\middle|\; \tau \in \textsc{TId} \right\} \right\rangle \qquad \text{Event paths}$$

$$\pi \in \textsc{PPath} \triangleq \textsc{Seq}\left\langle \textsc{ALabels} \cap \left\{ \mathsf{B}\langle e\rangle, \mathsf{D}\langle e\rangle, \mathsf{PB}\langle e\rangle \;\middle|\; e \in E \right\} \right\rangle \qquad \text{Propagation paths}$$

$$\theta \in \textsc{Trace} \triangleq \textsc{Path} \times \textsc{PPath} \qquad \text{Traces}$$

$$\mathcal{H} \in \textsc{Hist} \triangleq \textsc{Seq}\,\langle \textsc{Trace} \rangle \qquad \text{Histories}$$

Let

$$\begin{aligned} \textsc{AMem} \ni M_0 &\triangleq \lambda x.\mathit{init}_x \text{ with } \mathtt{lab}(\mathit{init}_x) \triangleq (\mathtt{W}, x, 0) \\ \textsc{APBuff} \ni PB_0 &\triangleq \lambda x.\epsilon \\ \textsc{ABuff} \ni b_0 &\triangleq \epsilon \\ \textsc{ABMap} \ni B_0 &\triangleq \lambda \tau.b_0 \end{aligned}$$

$$\mathsf{P}_{\mathbf{skip}} \triangleq \lambda \tau.v \text{ for some } v \in \textsc{Val}$$

Storage Subsystem

$$\frac{B(\tau){=}b \quad \mathtt{loc}(\mathit{fo}){=}x \quad x \in X \quad (SF \cup W_X \cup \left\{\langle \mathsf{fl}, e\rangle \;\middle|\; \mathtt{loc}(e) \in X\right\}) \cap b = \emptyset}{M, PB, B \xrightarrow{\mathsf{PFO}\langle \mathit{fo}\rangle} M, PB.\mathit{fo}, B[\tau \mapsto b.\langle \mathsf{pfo}, \mathit{fo}\rangle]} \text{ (AM-ProFO)}$$

$$\frac{B(\tau){=}b \quad \mathtt{loc}(\mathit{fl}){=}x \quad x \in X \quad (SF \cup W \cup \left\{\langle \mathsf{fo}, e\rangle, \langle \mathsf{fl}, e'\rangle \;\middle|\; \mathtt{loc}(e) \in X\right\}) \cap b = \emptyset}{M, PB, B \xrightarrow{\mathsf{PFL}\langle \mathit{fl}\rangle} M, PB.\mathit{fl}, B[\tau \mapsto b.\langle \mathsf{pfl}, \mathit{fl}\rangle]} \text{ (AM-ProFL)}$$

$$\frac{B(\tau){=}b \quad (W \cup \{\langle \mathsf{sf}, -\rangle, \langle \mathsf{fo}, -\rangle, \langle \mathsf{fl}, -\rangle\}) \cap b = \emptyset}{M, PB, B \xrightarrow{\mathsf{PSF}\langle \mathit{sf}\rangle} M, PB, B[\tau \mapsto b.\langle \mathsf{psf}, \mathit{sf}\rangle]} \text{ (AM-ProSF)}$$

$$\frac{B(\tau){=}b_1.o.b_2 \quad o \in \{\langle \mathsf{psf}, -\rangle, \langle \mathsf{pfo}, -\rangle, \langle \mathsf{pfl}, -\rangle\}}{M, PB, B \xrightarrow{\mathsf{D}\langle e\rangle} M, PB, B[\tau \mapsto b_1.b_2]} \text{ (AM-BDrop)}$$

$$\frac{B(\tau){=}b \quad \mathtt{loc}(w) \in X \quad \left\{\langle \mathsf{psf}, e\rangle, \langle \mathsf{pfl}, e\rangle, \langle \mathsf{pfo}, e'\rangle \;\middle|\; \mathtt{loc}(e') \in X\right\} \cap b = \emptyset}{M, PB, B \xrightarrow{\mathsf{W}\langle w\rangle} M, PB, B[\tau \mapsto b.w]} \text{ (AM-Write)}$$

$$\frac{B(\tau){=}b \quad \mathtt{loc}(r){=}x \quad \mathtt{rd}(M, PB, b, x){=}e}{M, PB, B \xrightarrow{\mathsf{R}\langle r, e\rangle} M, PB, B} \text{ (AM-Read)}$$

$$\frac{B(\tau){=}\epsilon \qquad \mathtt{loc}(u){=}x \qquad \mathtt{rd}(M, PB, \epsilon, x){=}e}{M, PB, B \xrightarrow{\mathsf{U}\langle u, e\rangle} M, PB[x \mapsto PB(x).u], B} \text{ (AM-RMW)}$$

$$\frac{B(\tau){=}\epsilon}{M, PB, B \xrightarrow{\mathsf{MF}\langle \mathit{mf}\rangle} M, PB, B} \text{ (AM-MF)}$$

$$\frac{B(\tau){=}b \quad \forall e.\ \forall o \in \{\mathsf{psf}, \mathsf{pfo}, \mathsf{pfl}\}.\ \langle o, e\rangle \notin b}{M, PB, B \xrightarrow{\mathsf{SF}\langle \mathit{sf}\rangle} M, PB, B[\tau \mapsto b.\mathit{sf}]} \text{ (AM-SF)}$$

$$\frac{B(\tau){=}\langle \mathsf{psf}, \mathit{sf}\rangle.b'}{M, PB, B \xrightarrow{\mathsf{J}\langle \mathit{sf}\rangle} M, PB, B[\tau \mapsto b']} \text{ (AM-SF2)}$$

$$\frac{B(\tau){=}b \quad \mathtt{loc}(fo) \in X \quad \forall e.\ \langle \mathsf{psf}, e\rangle \notin b \quad \forall e.\mathtt{loc}(e) \in X \Rightarrow \langle \mathsf{pfl}, e\rangle \notin b}{M, PB, B \xrightarrow{\mathsf{FO}\langle fo\rangle} M, PB, B[\tau \mapsto b.\langle \mathsf{fo}, fo\rangle]} \text{ (AM-FO)}$$

$$\frac{B(\tau){=}b_1.\langle \mathsf{pfo}, fo\rangle.b_2 \quad \mathtt{loc}(fo) \in X \quad \forall e.\ \langle \mathsf{psf}, e\rangle \notin b_1 \quad \forall e.\ \mathtt{loc}(e) \in X \Rightarrow \langle \mathsf{pfl}, e\rangle \notin b_1}{M, PB, B \xrightarrow{\mathsf{J}\langle fo\rangle} M, PB, B[\tau \mapsto b_1.b_2]} \text{ (AM-FO2)}$$

$$\frac{B(\tau){=}b \quad \mathtt{loc}(fl) \in X \quad \forall e.\ \langle \mathsf{psf}, e\rangle, \langle \mathsf{pfl}, e\rangle \notin b \quad \forall e.\ \mathtt{loc}(e) \in X \Rightarrow \langle \mathsf{pfo}, e\rangle \notin b}{M, PB, B \xrightarrow{\mathsf{FL}\langle fl\rangle} M, PB, B[\tau \mapsto b.\langle \mathsf{fl}, fl\rangle]} \text{ (AM-FL)}$$

$$\frac{B(\tau){=}b_1.\langle \mathsf{pfl}, fl\rangle.b_2 \quad \mathtt{loc}(fo) \in X \quad \forall e.\ \mathtt{loc}(e) \in X \Rightarrow \langle \mathsf{pfo}, e\rangle \notin b_1 \quad \forall e.\ \langle \mathsf{pfl}, e\rangle, \langle \mathsf{psf}, e\rangle \notin b_1}{M, PB, B \xrightarrow{\mathsf{J}\langle fl\rangle} M, PB, B[\tau \mapsto b]} \text{ (AM-FL2)}$$

$$\frac{\begin{array}{c} B(\tau){=}b_1.w.b_2 \\ (W \cup \{\langle \mathsf{sf}, -\rangle, \langle \mathsf{fl}, -\rangle\}) \cap b_1 = \emptyset \end{array}}{M, PB, B \xrightarrow{\mathsf{B}\langle w\rangle} M, PB.w, B[\tau \mapsto b_1.b_2]} \text{ (AM-BPropW)}$$

$$\frac{B(\tau){=}\langle \mathsf{sf}, sf\rangle.b}{M, PB, B \xrightarrow{\mathsf{B}\langle sf\rangle} M, PB, B[\tau \mapsto b]} \text{ (AM-BPropSF)}$$

$$\frac{\begin{array}{c} B(\tau){=}b_1.\langle \mathsf{fo}, fo\rangle.b_2 \quad \mathtt{loc}(fo) \in X \\ (SF \cup W_X \cup \left\{\langle \mathsf{fl}, e\rangle \;\middle|\; \mathtt{loc}(e) \in X\right\}) \cap b_1 = \emptyset \end{array}}{M, PB, B \xrightarrow{\mathsf{B}\langle fo\rangle} M, PB.fo, B[\tau \mapsto b_1.b_2]} \text{ (AM-BPropFO)}$$

$$\frac{\begin{array}{c} B(\tau){=}b_1.\langle \mathsf{fl}, fl\rangle.b_2 \quad \mathtt{loc}(fl) \in X \\ (SF \cup W \cup \left\{\langle \mathsf{fo}, e\rangle, \langle \mathsf{fl}, -\rangle \;\middle|\; \mathtt{loc}(e) \in X\right\}) \cap b_1 = \emptyset \end{array}}{M, PB, B \xrightarrow{\mathsf{B}\langle fl\rangle} M, PB.fl, B[\tau \mapsto b_1.b_2]} \text{ (AM-BPropFL)}$$

$$\frac{\mathsf{PB}{=}\mathsf{PB}_1.w.\mathsf{PB}_2 \quad w \in W \quad \mathtt{loc}(w){=}x \quad \mathsf{PB}_1 \cap (W_x \cup FO \cup FL){=}\emptyset}{M, PB, B \xrightarrow{\mathsf{PB}\langle w\rangle} M[x \mapsto w], PB_1.PB_2, B} \text{ (AM-PropW)}$$

$$\frac{\mathsf{PB}{=}\mathsf{PB}_1.e.\mathsf{PB}_2 \quad e \in FO \cup FL \quad \mathtt{loc}(e) \in X \quad \mathsf{PB}_1 \cap (W_X \cup FO \cup FL){=}\emptyset}{M, PB, B \xrightarrow{\mathsf{PB}\langle e\rangle} M, PB_1.PB_2, B} \text{ (AM-PropP)}$$

where

$$\mathtt{rd}(M, PB, b, x) \triangleq \begin{cases} e & \text{if } \mathtt{rd_S}(b, x) = e \\ e & \text{else if } PB{=}PB_1.e.PB_2 \\ & \text{and } WU_x \cap PB_2{=}\emptyset \\ & \text{and } e \in WU_x \\ M(x) & \text{otherwise} \end{cases} \qquad \mathtt{rd_S}(b, x) \triangleq \begin{cases} w & \text{if } \exists b_1, b_2.\ b{=}b_1.w.b_2 \\ & \text{and } \mathtt{loc}(w){=}x \\ & \text{and } W_x \cap b_2{=}\emptyset \\ \text{undef} & \text{otherwise} \end{cases}$$

Thread Subsystem

Thread-local steps.

$$\frac{\mathrm{C}_1, \xrightarrow{\lambda} \mathrm{C}_1'}{\textbf{let } a{:=}\mathrm{C}_1 \textbf{ in } \mathrm{C}_2 \xrightarrow{\lambda} \textbf{let } a{:=}\mathrm{C}_1' \textbf{ in } \mathrm{C}_2}\ (\text{AT-Let1}) \qquad \frac{}{\textbf{let } a{:=}v \textbf{ in } \mathrm{C} \xrightarrow{\mathcal{E}\langle\tau\rangle} \mathrm{C}[v/a]}\ (\text{AT-Let2})$$

$$\frac{\mathrm{C}, \xrightarrow{\lambda} \mathrm{C}'}{\textbf{if } (\mathrm{C}) \textbf{ then } \mathrm{C}_1 \textbf{ else } \mathrm{C}_2 \xrightarrow{\lambda} \textbf{if } (\mathrm{C}') \textbf{ then } \mathrm{C}_1 \textbf{ else } \mathrm{C}_2}\ (\text{AT-If1})$$

$$\frac{v{\neq}0 \Rightarrow \mathrm{C}{=}\mathrm{C}_1 \quad v{=}0 \Rightarrow \mathrm{C}{=}\mathrm{C}_2}{\textbf{if } (v) \textbf{ then } \mathrm{C}_1 \textbf{ else } \mathrm{C}_2 \xrightarrow{\mathcal{E}\langle\tau\rangle} \mathrm{C}}\ (\text{T-If2})$$

$$\frac{}{\textbf{repeat } \mathrm{C} \xrightarrow{\mathcal{E}\langle\tau\rangle} \textbf{if } (\mathrm{C}) \textbf{ then } (\textbf{repeat } \mathrm{C}) \textbf{ else } 0}\ (\text{T-Repeat})$$

$$\frac{\mathtt{val_w}(w){=}v \quad \mathtt{loc}(w){=}x}{\textbf{store}(x, v) \xrightarrow{\mathrm{W}\langle w\rangle} v}\ (\text{AT-Write}) \qquad \frac{\mathtt{val_r}(r){=}v \quad \mathtt{loc}(r){=}x}{\textbf{load}(x) \xrightarrow{\mathrm{R}\langle r, w\rangle} v}\ (\text{AT-Read})$$

$$\frac{\mathtt{val_r}(u){=}v \quad \mathtt{val_w}(u){=}v{+}v' \quad \mathtt{loc}(u){=}x}{\textbf{FAA}(x, v) \xrightarrow{\mathrm{U}\langle u, w\rangle} v}\ (\text{AT-FAA}) \qquad \frac{}{\textbf{mfence} \xrightarrow{\mathrm{MF}\langle mf\rangle} 1}\ (\text{AT-MFence})$$

$$\frac{\mathtt{val_r}(r) \neq v_1 \quad \mathtt{loc}(r){=}x}{\textbf{CAS}(x, v_1, v_2) \xrightarrow{\mathrm{R}\langle r, w\rangle} 0}\ (\text{AT-CAS0}) \qquad \frac{\mathtt{val_r}(u){=}v_1 \quad \mathtt{val_w}(u){=}v_2 \quad \mathtt{loc}(u){=}x}{\textbf{CAS}(x, v_1, v_2) \xrightarrow{\mathrm{U}\langle u, w\rangle} 1}\ (\text{AT-CAS1})$$

$$\frac{}{\textbf{sfence} \xrightarrow{\mathrm{SF}\langle sf\rangle} 1}\ (\text{AT-SFence}) \qquad \frac{\mathtt{loc}(fo){=}x}{\textbf{flush}_{\text{opt}}\ x \xrightarrow{\mathrm{FO}\langle fo\rangle} 1}\ (\text{AT-FO1}) \qquad \frac{\mathtt{loc}(fo){=}x}{\textbf{flush}_{\text{opt}}\ x \xrightarrow{\mathrm{J}\langle fo\rangle} 1}\ (\text{AT-FO2})$$

$$\frac{\mathtt{loc}(fl){=}x}{\textbf{flush}\ x \xrightarrow{\mathrm{FL}\langle fl\rangle} 1}\ (\text{AT-FL1}) \qquad \frac{\mathtt{loc}(fl){=}x}{\textbf{flush}\ x \xrightarrow{\mathrm{J}\langle fl\rangle} 1}\ (\text{AT-FL2})$$

Program Steps.

$$\frac{\mathsf{P}(\tau) \xrightarrow{\lambda} \mathsf{C} \quad \mathtt{tid}(\lambda) = \tau}{\mathsf{P} \xrightarrow{\lambda} \mathsf{P}[\tau \mapsto \mathsf{C}]} \text{ (AP-Step)}$$

where:

$$\mathtt{tid}(\lambda) \triangleq \begin{cases} \tau & \text{if } \lambda{=}\mathcal{E}\langle\tau\rangle \\ \mathtt{tid}(\mathtt{event}(\lambda)) & \text{otherwise} \end{cases}$$

$$\begin{aligned}
\mathtt{event}(\mathsf{R}\langle r, w\rangle) &\triangleq r \\
\mathtt{event}(\mathsf{U}\langle u, w\rangle) &\triangleq u \\
\mathtt{event}(\mathsf{W}\langle w\rangle) &\triangleq w \\
\mathtt{event}(\mathsf{MF}\langle mf\rangle) &\triangleq mf \\
\mathtt{event}(\mathsf{SF}\langle sf\rangle) &\triangleq sf \\
\mathtt{event}(\mathsf{FO}\langle fo\rangle) &\triangleq fo \\
\mathtt{event}(\mathsf{FL}\langle fl\rangle) &\triangleq fl \\
\mathtt{event}(\mathsf{PSF}\langle sf\rangle) &\triangleq sf \\
\mathtt{event}(\mathsf{PFO}\langle fo\rangle) &\triangleq fo \\
\mathtt{event}(\mathsf{PFL}\langle fl\rangle) &\triangleq fl \\
\mathtt{event}(\mathsf{B}\langle e\rangle) &\triangleq e \\
\mathtt{event}(\mathsf{J}\langle e\rangle) &\triangleq e \\
\mathtt{event}(\mathsf{D}\langle e\rangle) &\triangleq e \\
\mathtt{event}(\mathsf{PB}\langle e\rangle) &\triangleq e \\
\mathtt{event}(\mathcal{E}\langle\tau\rangle) &\text{ undefined}
\end{aligned}$$

Event-Annotated Operational Semantics

$$\frac{\mathsf{P} \xrightarrow{\mathcal{E}\langle\tau\rangle} \mathsf{P}'}{\Delta \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}', M, PB, B, \mathcal{H}, \pi} \text{ (A-SilentP)}$$

$$\frac{M, PB, B \xrightarrow{\lambda} M', PB', B' \qquad \lambda \in \left\{\mathsf{B}\langle e\rangle, \mathsf{PB}\langle e\rangle, \mathsf{D}\langle e\rangle, \mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle, \mathsf{PSF}\langle e\rangle\right\} \quad \mathtt{fresh}(\lambda, \pi) \quad \mathtt{fresh}(\lambda, \mathcal{H})}{\Delta \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi.\lambda} \text{ (A-PropM)}$$

$$\frac{\mathsf{P} \xrightarrow{\lambda} \mathsf{P}' \quad M, PB, B \xrightarrow{\lambda} M', PB', B' \quad \mathtt{fresh}(\lambda, \pi) \quad \mathtt{fresh}(\lambda, \mathcal{H})}{\Delta \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}', M', PB', B', \mathcal{H}, \pi.\lambda} \text{ (A-Step)}$$

$$\frac{\Delta{=}(\mathsf{P}_0, \mathbf{rec}) \quad M, PB, B \xrightarrow{\pi'}_{\mathrm{p}} M', PB_0, B_0}{\Delta \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathbf{rec}(\mathsf{P}_0, M), M, PB_0, B_0, \mathcal{H}.(\pi, \pi'), \epsilon} \text{ (A-Crash)}$$

with

$$\frac{}{(M, PB_0, B_0) \xrightarrow{\epsilon}_{\mathrm{p}} (M, PB_0, B_0)}$$

$$\frac{(M, PB, B) \xrightarrow{\lambda} (M'', PB'', B'') \quad \exists e.\ \lambda \in \{\mathsf{B}\langle e\rangle, \mathsf{D}\langle e\rangle, \mathsf{PB}\langle e\rangle\} \quad (M'', PB'', B'') \xrightarrow{\pi}_{\mathrm{p}} (M', PB', B')}{(M, PB, B) \xrightarrow{\lambda.\pi}_{\mathrm{p}} (M', PB', B')}$$

and

$$\begin{aligned}\mathsf{fresh}(\lambda,\pi) \triangleq\ & \lambda \notin \pi \wedge \forall e, w.\ \forall w' \neq w.\\ & (\lambda{=}\mathsf{R}\langle e,w\rangle \Rightarrow \mathsf{R}\langle e,w'\rangle \notin \pi) \wedge (\lambda{=}\mathsf{U}\langle e,w\rangle \Rightarrow \mathsf{U}\langle e,w'\rangle \notin \pi)\\ & \wedge (\lambda{=}\mathsf{J}\langle e\rangle \Rightarrow \mathsf{D}\langle e\rangle \notin \pi) \wedge (\lambda{=}\mathsf{D}\langle e\rangle \Rightarrow \mathsf{J}\langle e\rangle \notin \pi)\\ & \wedge (\lambda{=}\mathsf{FO}\langle e\rangle \Rightarrow \mathsf{PFO}\langle e\rangle \notin \pi) \wedge (\lambda{=}\mathsf{PFO}\langle e\rangle \Rightarrow \mathsf{FO}\langle e\rangle \notin \pi)\\ & \wedge (\lambda{=}\mathsf{FL}\langle e\rangle \Rightarrow \mathsf{PFL}\langle e\rangle \notin \pi) \wedge (\lambda{=}\mathsf{PFL}\langle e\rangle \Rightarrow \mathsf{FL}\langle e\rangle \notin \pi)\\ & \wedge (\lambda{=}\mathsf{SF}\langle e\rangle \Rightarrow \mathsf{PSF}\langle e\rangle \notin \pi) \wedge (\lambda{=}\mathsf{PSF}\langle e\rangle \Rightarrow \mathsf{SF}\langle e\rangle \notin \pi)\end{aligned}$$

$$\mathsf{fresh}(\lambda,\mathcal{H}) \triangleq \forall(\pi,\pi') \in \mathcal{H}.\ \mathsf{fresh}(\lambda,\pi.\pi')$$

Definition 5.

$$\begin{aligned}\mathsf{complete}(\pi) \triangleq\ & \forall e.\ \mathsf{W}\langle e\rangle \in \pi \Rightarrow \mathsf{B}\langle e\rangle, \mathsf{PB}\langle e\rangle \in \pi\\ & \mathsf{U}\langle e,-\rangle \in \pi \Rightarrow \mathsf{PB}\langle e\rangle \in \pi\\ & \mathsf{SF}\langle e\rangle \in \pi \Rightarrow \mathsf{B}\langle e\rangle \in \pi\\ & \mathsf{FO}\langle e\rangle \in \pi \Rightarrow \mathsf{B}\langle e\rangle, \mathsf{PB}\langle e\rangle \in \pi\\ & \mathsf{FL}\langle e\rangle \in \pi \Rightarrow \mathsf{B}\langle e\rangle, \mathsf{PB}\langle e\rangle \in \pi\\ & \mathsf{PFO}\langle e\rangle \in \pi \Rightarrow (\mathsf{J}\langle e\rangle \in \pi \wedge \mathsf{PB}\langle e\rangle \in \pi) \vee \mathsf{D}\langle e\rangle \in \pi\\ & \mathsf{PFL}\langle e\rangle \in \pi \Rightarrow (\mathsf{J}\langle e\rangle \in \pi \wedge \mathsf{PB}\langle e\rangle \in \pi) \vee \mathsf{D}\langle e\rangle \in \pi\\ & \mathsf{PSF}\langle e\rangle \in \pi \Rightarrow \mathsf{J}\langle e\rangle \in \pi \vee \mathsf{D}\langle e\rangle \in \pi\end{aligned}$$

$$\begin{aligned}\mathsf{wfp}(\pi,\mathcal{H}) \triangleq & \forall \lambda, \pi_1, \pi_2, e, r, e_1, e_2, \lambda_1, \lambda_2, X.\\ & \mathsf{nodups}(\pi.\pi'.\pi'')\\ & \pi{=}\pi_1.\mathsf{R}\langle r,e\rangle.\pi_2 \vee \pi{=}\pi_1.\mathsf{U}\langle r,e\rangle.\pi_2 \Rightarrow \mathsf{wfrd}(r,e,\pi_1,\pi')\\ & \mathsf{B}\langle e\rangle \in \pi \Rightarrow\\ & \quad \mathsf{W}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \vee \mathsf{SF}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \vee \mathsf{FO}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \vee \mathsf{FL}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle\\ & \mathsf{PB}\langle e\rangle \in \pi \Rightarrow\\ & \quad \mathsf{B}\langle e\rangle \prec_\pi \mathsf{PB}\langle e\rangle \vee \mathsf{U}\langle e,-\rangle \prec_\pi \mathsf{PB}\langle e\rangle \vee \mathsf{J}\langle e\rangle \prec_\pi \mathsf{PB}\langle e\rangle\\ & \mathsf{J}\langle e\rangle \in \pi \Rightarrow \mathsf{PFO}\langle e\rangle \prec_\pi \mathsf{J}\langle e\rangle \vee \mathsf{PFL}\langle e\rangle \prec_\pi \mathsf{J}\langle e\rangle \vee \mathsf{PSF}\langle e\rangle \prec_\pi \mathsf{J}\langle e\rangle\\ & \mathsf{D}\langle e\rangle \in \pi \Rightarrow \mathsf{PFO}\langle e\rangle \prec_\pi \mathsf{D}\langle e\rangle \vee \mathsf{PFL}\langle e\rangle \prec_\pi \mathsf{D}\langle e\rangle \vee \mathsf{PSF}\langle e\rangle \prec_\pi \mathsf{D}\langle e\rangle\\ & \mathsf{J}\langle e\rangle \notin \pi \vee \mathsf{D}\langle e\rangle \notin \pi\\ & \mathsf{FO}\langle e\rangle \notin \pi \vee \mathsf{PFO}\langle e\rangle \notin \pi\\ & \mathsf{FL}\langle e\rangle \notin \pi \vee \mathsf{PFL}\langle e\rangle \notin \pi\\ & \mathsf{SF}\langle e\rangle \notin \pi \vee \mathsf{PSF}\langle e\rangle \notin \pi\\ & \mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle\\ & \mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle\\ & \mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle\end{aligned}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle$$
$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle$$
$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$$
$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$$
$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$$
$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$$
$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$$
$$e_1 \in \mathit{FO} \cup \mathit{FL} \cup \mathit{SF} \wedge e_2 \in \mathit{SF} \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\quad \mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \vee \mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \vee \mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle$$
$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle$$
$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle$$
$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle$$
$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle$$
$$e_1 \in \mathit{SF} \wedge e_2 \in \mathit{FO} \cup \mathit{FL} \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\quad \mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \vee \mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle$$
$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1) = \mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1) = \mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle$$
$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle$$
$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow$$
$$\quad \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle$$
$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle$$
$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow$$
$$\quad \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle$$
$$e_1 \in \mathit{FO} \wedge e_2 \in \mathit{FL} \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\quad \mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle$$
$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$$
$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle$$
$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow$$

$$
\begin{aligned}
&\quad \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \\
&\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \\
&\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \\
&\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \\
&\quad \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \\
&e_1 \in \mathit{FL} \wedge e_2 \in \mathit{FO} \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow \\
&\quad \mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \\
&\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \\
&\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \\
&\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \\
&\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \\
&\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \\
&\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \\
&\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \\
&\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \\
&\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \\
&e_1, e_2 \in \mathit{FL} \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow \\
&\quad \mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \\
&e_1{,}e_2 \in \mathit{WU} \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle{,}\mathsf{U}\langle e_1, -\rangle\} \wedge \lambda_2 \in \{\mathsf{B}\langle e_2\rangle{,}\mathsf{U}\langle e_2, -\rangle\} \wedge \lambda_1 \prec_\pi \lambda_2 \wedge \mathtt{loc}(e_1){=}\mathtt{loc}(e_2) \\
&\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \\
&e_1 \in \mathit{WU} \wedge e_2 \in \mathit{FO} \cup \mathit{FL} \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) {\in} X \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\} \wedge \lambda_1 {\prec_\pi} \mathsf{B}\langle e_2\rangle \\
&\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \\
&e_1 \in \mathit{WU} \wedge e_2 \in \mathit{FO} \cup \mathit{FL} \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) {\in} X \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\} \\
&\wedge \lambda_2 {\in} \{\mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\} \wedge \lambda_1 {\prec_\pi} \lambda_2 \\
&\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \vee \mathsf{D}\langle e_2\rangle \in \pi \\
&e_1 \in \mathit{FO} \cup \mathit{FL} \wedge e_2 \in \mathit{D} \wedge \lambda_1 {\in} \{\mathsf{B}\langle e_1\rangle, \mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle\} \\
&\wedge \lambda_2 {\in} \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, e\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\} \wedge \lambda_1 {\prec_\pi} \lambda_2 \\
&\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \in \pi \vee \mathsf{D}\langle e_2\rangle \in \pi
\end{aligned}
$$

where $\pi' = \pi_1.\cdots.\pi_n$ and $\pi'' = \pi'_1.\cdots.\pi'_n$, when $\mathcal{H} = (\pi_1, \pi'_1).\cdots.(\pi_n, \pi'_n)$; and

$$\mathsf{nodups}(\pi) \triangleq \forall \pi_1, \pi_2, \lambda.\ \pi = \pi_1.\lambda.\pi_2 \Rightarrow \mathsf{fresh}(\lambda, \pi_1.\pi_2)$$

$$\mathsf{wfrd}(r,e,\pi,\pi') \triangleq \left(\begin{array}{l}\exists \pi_1,\pi_2,\lambda.\ \pi=\pi_2.\lambda.\pi_1 \\ \quad \wedge (\lambda{=}\mathsf{B}\langle e\rangle \vee \lambda{=}\mathsf{U}\langle e,-\rangle \vee (\lambda{=}\mathsf{W}\langle e\rangle \wedge \mathtt{tid}(e)=\mathtt{tid}(r))) \\ \quad \wedge \left((\lambda{=}\mathsf{B}\langle e\rangle \vee \lambda{=}\mathsf{U}\langle e,-\rangle) \Rightarrow \left\{\mathsf{B}\langle e'\rangle, \mathsf{U}\langle e',-\rangle \in \pi_1 \;\middle|\; \mathtt{loc}(e')=\mathtt{loc}(r)\right\} = \emptyset\right) \\ \quad \wedge \left\{e' \;\middle|\; \begin{array}{l}\mathsf{W}\langle e'\rangle \in \pi \wedge \mathsf{B}\langle e'\rangle \notin \pi \\ \wedge\, \mathtt{loc}(e')=\mathtt{loc}(r) \wedge \mathtt{tid}(e')=\mathtt{tid}(r)\end{array}\right\} = \emptyset \\ \quad \wedge \left(\lambda{=}\mathsf{W}\langle e\rangle \Rightarrow \mathsf{B}\langle e\rangle \notin \pi_1 \wedge \left\{\mathsf{W}\langle e'\rangle \in \pi_1 \;\middle|\; \begin{array}{l}\mathtt{loc}(e')=\mathtt{loc}(r)\wedge \\ \mathtt{tid}(e')=\mathtt{tid}(r)\end{array}\right\} = \emptyset\right)\end{array}\right) \\ \vee \left(\begin{array}{l}\exists \pi_1,\pi_2.\ \pi'=\pi_2.\mathsf{PB}\langle e\rangle.\pi_1 \\ \quad \wedge \left\{\begin{array}{l|l}\mathsf{B}\langle e'\rangle, \mathsf{U}\langle e',-\rangle \in \pi, & \mathtt{loc}(e')=\mathtt{loc}(r)\wedge \\ \mathsf{W}\langle e''\rangle \in \pi, & \mathtt{loc}(e'')=\mathtt{loc}(r)\wedge \\ \mathsf{PB}\langle e'\rangle \in \pi_1 & \mathtt{tid}(e'')=\mathtt{tid}(r)\end{array}\right\} = \emptyset\end{array}\right) \\ \vee \left(e = \mathit{init}_{\mathtt{loc}(e)} \wedge \left\{\begin{array}{l|l}\mathsf{B}\langle e'\rangle, \mathsf{U}\langle e',-\rangle \in \pi, & \mathtt{loc}(e')=\mathtt{loc}(r)\wedge \\ \mathsf{W}\langle e''\rangle \in \pi, & \mathtt{loc}(e'')=\mathtt{loc}(r)\wedge \\ \mathsf{PB}\langle e'\rangle \in \pi' & \mathtt{tid}(e'')=\mathtt{tid}(r)\end{array}\right\} = \emptyset\right)$$

Definition 6.

$$\mathsf{wf}(M,PB,B,\mathcal{H},\pi) \overset{\text{def}}{\Leftrightarrow} \mathsf{mem}(\mathcal{H},\pi)=M \wedge \mathsf{pbuff}(PB_0,\pi)=PB \wedge \mathsf{bmap}(B_0,\pi)=B \wedge \mathsf{wfp}(\pi,\mathcal{H}) \wedge \mathsf{wfh}(\mathcal{H})$$

where

$$\mathsf{mem}(\mathcal{H},\pi)=M \overset{\text{def}}{\Leftrightarrow} \forall x \in \mathrm{Loc}.\ M(x)=\mathtt{read}(\mathcal{H},\pi,x)$$

$$\mathtt{read}(\mathcal{H},\pi.\lambda,x) \triangleq \begin{cases} e & \exists e \in WU.\ \lambda=\mathsf{PB}\langle e\rangle \wedge \mathtt{loc}(e)=x \\ \mathtt{read}(\mathcal{H},\pi,x) & \text{otherwise}\end{cases}$$

$$\mathtt{read}(\mathcal{H}.(\pi,-),\epsilon,x) \triangleq \mathtt{read}(\mathcal{H},\pi,x)$$

$$\mathtt{read}(\epsilon,\epsilon,x) \triangleq \mathit{init}_x$$

$$\mathsf{pbuff}(PB,\epsilon) \triangleq PB$$

$$\mathsf{pbuff}(PB,\lambda.\pi) \triangleq \begin{cases} \mathsf{pbuff}(PB.e,\pi) & \text{if } \exists e.\ \lambda \in \{\mathsf{B}\langle e\rangle, \mathsf{U}\langle e,-\rangle, \mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle\} \wedge \mathsf{PB}\langle e\rangle \notin \pi \\ \mathsf{pbuff}(PB,\pi) & \text{otherwise}\end{cases}$$

$$\mathsf{bmap}(B,\epsilon) \triangleq B$$

$$\mathsf{bmap}(B,\lambda.\pi) \triangleq \begin{cases} \mathsf{bmap}(B[\tau \mapsto B(\tau).e],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{W}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{B}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B[\tau \mapsto B(\tau).\langle \mathsf{fo},e\rangle],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{FO}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{B}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B[\tau \mapsto B(\tau).\langle \mathsf{fl},e\rangle],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{FL}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{B}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B[\tau \mapsto B(\tau).\langle \mathsf{sf},e\rangle],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{SF}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{B}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B[\tau \mapsto B(\tau).\langle \mathsf{pfo},e\rangle],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{PFO}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{J}\langle e\rangle, \mathsf{D}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B[\tau \mapsto B(\tau).\langle \mathsf{pfl},e\rangle],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{PFL}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{J}\langle e\rangle, \mathsf{D}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B[\tau \mapsto B(\tau).\langle \mathsf{psf},e\rangle],\pi) & \text{if } \exists e,\tau.\ \lambda{=}\mathsf{PSF}\langle e\rangle \wedge \mathtt{tid}(e){=}\tau \wedge \mathsf{J}\langle e\rangle, \mathsf{D}\langle e\rangle \notin \pi \\ \mathsf{bmap}(B,\pi) & \text{otherwise}\end{cases}$$

$$\mathsf{wfh}(\epsilon) \overset{\text{def}}{\Leftrightarrow} \mathit{true}$$

$$\mathsf{wfh}(\mathcal{H}.(\pi,\pi')) \overset{\text{def}}{\Leftrightarrow} \mathsf{wfp}(\pi.\pi',\mathcal{H}) \wedge \mathsf{complete}(\pi.\pi') \wedge \mathsf{wfh}(\mathcal{H})$$

Lemma 1. *For all* **rec**, P, P′, *PB*, *PB*′, *B*, *B*′, $\mathcal{H}$, $\mathcal{H}'$, π, π'*:*

- $\mathsf{wf}(M_0, PB_0, B_0, \epsilon, \epsilon)$

- *if* $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}', M', PB', B', \mathcal{H}', \pi'$ *and* $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$, *then* $\mathsf{wf}(M', PB', B', \mathcal{H}', \pi')$
- *if* $\mathbf{rec} \vdash \mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, M, PB, B, \mathcal{H}, \pi$, *then* $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$

Proof. The proof of the first part follows trivially from the definitions of M_0, PB_0, and B_0. The second part follows straightforwardly by induction on the structure of $\Rightarrow$. The last part follows from the previous two parts and induction on the length of $\Rightarrow^*$. □

Graph Operational Semantics

Let

$$\Gamma \in \textsc{GHist} \triangleq \textsc{Seq}\,\langle \textsc{Exec} \times \textsc{Trace}\rangle \qquad \text{Graph histories}$$
$$\mathtt{hist}(.) : \textsc{GHist} \to \textsc{Hist}$$
$$\mathtt{hist}(\epsilon) = \epsilon \qquad \mathtt{hist}((G,\theta).\Gamma) = \theta.\mathtt{hist}(\Gamma)$$

$$\frac{\mathsf{P} \xrightarrow{\mathcal{E}\langle\tau\rangle} \mathsf{P}'}{\Delta \vdash \mathsf{P}, \Gamma, \pi \Rightarrow \mathsf{P}', \Gamma, \pi} \text{ (G-SilentP)}$$

$$\frac{\lambda \in \{\mathsf{B}\langle e\rangle, \mathsf{D}\langle e\rangle, \mathsf{PB}\langle e\rangle, \mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle, \mathsf{PSF}\langle e\rangle\} \quad \mathsf{fresh}(\lambda,\pi) \quad \mathsf{fresh}(\lambda,\Gamma)}{\Delta \vdash \mathsf{P}, \Gamma, \pi \Rightarrow \mathsf{P}, \Gamma, \pi.\lambda} \text{ (G-Prop)}$$

$$\frac{\mathsf{P} \xrightarrow{\lambda} \mathsf{P}' \quad \lambda \neq \mathcal{E}\langle -\rangle \quad \mathsf{fresh}(\lambda,\pi) \quad \mathsf{fresh}(\lambda,\Gamma)}{\Delta \vdash \mathsf{P}, \Gamma, \pi \Rightarrow \mathsf{P}', \Gamma, \pi.\lambda} \text{ (G-Step)}$$

$$\frac{\mathsf{comp}(\pi,\pi') \quad G \text{ is } \mathrm{Px86}_{\mathrm{man}}\text{-consistent} \quad G \prec \mathtt{getG}(\Gamma,\pi,\pi') \quad \Delta{=}(\mathsf{P}_0, \mathtt{rec})}{\Delta \vdash \mathsf{P}, \Gamma, \pi \Rightarrow \mathtt{rec}(\mathsf{P}_0, G), \Gamma.(G,(\pi,\pi')), \epsilon} \text{ (G-Crash)}$$

where

$$\mathsf{fresh}(\lambda,\Gamma) \overset{\text{def}}{\Leftrightarrow} \forall (-,(\pi,\pi')) \in \Gamma.\ \mathsf{fresh}(\lambda,\pi.\pi')$$
$$\mathsf{comp}(.,.) : \textsc{Path} \times \textsc{PPath} \to \{true, false\}$$
$$\mathsf{comp}(\pi,\pi') \overset{\text{def}}{\Leftrightarrow} \forall e.\ \begin{pmatrix}\mathsf{W}\langle e\rangle \in \pi \vee \mathsf{SF}\langle e\rangle \in \pi \\ \vee\ \mathsf{FO}\langle e\rangle \in \pi \vee \mathsf{FL}\langle e\rangle \in \pi\end{pmatrix} \wedge \mathsf{B}\langle e\rangle \notin \pi \Leftrightarrow \mathsf{B}\langle e\rangle \in \pi'$$
$$\wedge \begin{pmatrix}\mathsf{PFO}\langle e\rangle \in \pi \\ \vee\ \mathsf{PFL}\langle e\rangle \in \pi \\ \vee\ \mathsf{PSF}\langle e\rangle \in \pi\end{pmatrix} \wedge \mathsf{J}\langle e\rangle \notin \pi \wedge \mathsf{D}\langle e\rangle \notin \pi \Leftrightarrow \mathsf{D}\langle e\rangle \in \pi'$$
$$\wedge \begin{pmatrix}\mathsf{W}\langle e\rangle \in \pi \vee \mathsf{U}\langle e,-\rangle \in \pi \\ \vee\ \mathsf{FO}\langle e\rangle \in \pi \vee \mathsf{FL}\langle e\rangle \in \pi \\ \vee\ (\mathsf{PFO}\langle e\rangle \in \pi \wedge \mathsf{J}\langle e\rangle \in \pi.\pi') \\ \vee\ (\mathsf{PFL}\langle e\rangle \in \pi \wedge \mathsf{J}\langle e\rangle \in \pi.\pi')\end{pmatrix} \wedge \mathsf{PB}\langle e\rangle \notin \pi \Leftrightarrow \mathsf{PB}\langle e\rangle \in \pi'$$

$$\mathtt{getG}(\Gamma,\pi_1,\pi_2) \triangleq \begin{cases}(E, I, P, \mathrm{po}, \mathrm{rf}, \mathrm{tso}, \mathrm{nvo}) & \text{if } \mathsf{wfp}(\pi_1.\pi_2, \mathtt{hist}(\Gamma)) \wedge \mathsf{complete}(\pi_1.\pi_2) \\ \text{undefined} & \text{otherwise}\end{cases}$$

with $(\pi,\pi'){=}\mathsf{prune}(\pi_1,\pi_2)$ and

$$I \triangleq \begin{cases}\left\{ init_x \;\middle|\; x \in \mathrm{Loc} \right\} & \text{if } \Gamma = \epsilon \\ \left\{ w_x \;\middle|\; \begin{array}{l} x \in \mathrm{Loc} \wedge \exists e.\ e{=}\max\left(G.\mathrm{nvo}|_{G.P \cap WU_x}\right) \\ \wedge\ \mathsf{val_w}(w_x){=}\mathsf{val_w}(e) \wedge \mathsf{tid}(w_x){=}\tau_0 \end{array} \right\} & \text{if } \Gamma = \Gamma'.(G,-)\end{cases}$$
$$E \triangleq I \cup \left\{ e \;\middle|\; \exists \lambda \in \pi.\pi'.\ \mathtt{getE}(\lambda) = e \right\}$$
$$P \triangleq I \cup \left\{ e \in E \;\middle|\; \exists \lambda \in \pi.\ \mathtt{getPE}(\lambda) = e \right\}$$

$$\begin{aligned}
&\mathsf{rf} \triangleq \left\{(w,e) \;\middle|\; \mathsf{R}\langle e,w\rangle \in \pi \vee \mathsf{U}\langle e,w\rangle \in \pi\right\}\\
&\mathsf{po} \triangleq I \times (E \setminus I) \cup \bigcup_{\tau \in \mathrm{TID}} \left\{(e_1,e_2) \;\middle|\; \begin{array}{l}\exists \lambda_1,\lambda_2.\ e_1{=}\mathsf{getE}(\lambda_1) \wedge e_2{=}\mathsf{getE}(\lambda_2) \wedge \lambda_1 \prec_{\pi.\pi'} \lambda_2\\ \wedge\ \mathsf{tid}(e_1){=}\mathsf{tid}(e_2){=}\tau\end{array}\right\}\\
&\mathsf{tso} \triangleq I \times (E \setminus I)\\
&\qquad \cup \left\{(e_1,e_2) \in E \times E \;\middle|\; \exists \lambda_1,\lambda_2.\ e_1{=}\mathsf{getBE}(\lambda_1) \wedge e_2{=}\mathsf{getBE}(\lambda_2) \wedge \lambda_1 \prec_{\pi.\pi'} \lambda_2\right\}\\
&\mathsf{nvo} \triangleq I \times (D \setminus I)\\
&\qquad \cup \left\{(e_1,e_2) \in E \times E \;\middle|\; \exists \lambda_1,\lambda_2.\ e_1{=}\mathsf{getPE}(\lambda_1) \wedge e_2{=}\mathsf{getPE}(\lambda_2) \wedge \lambda_1 \prec_{\pi.\pi'} \lambda_2\right\}
\end{aligned}$$

and

$$\begin{aligned}
&\mathsf{getE}(.) : \textsc{ALabels} \rightharpoonup E\\
&\mathsf{getE}(\lambda) \triangleq \begin{cases} e & \text{if } \exists e.\ \lambda \in \{\mathsf{R}\langle e,-\rangle, \mathsf{U}\langle e,-\rangle, \mathsf{W}\langle e\rangle, \mathsf{MF}\langle e\rangle, \mathsf{SF}\langle e\rangle, \mathsf{FO}\langle e\rangle, \mathsf{FL}\langle e\rangle, \mathsf{J}\langle e\rangle\}\\ \text{undefined} & \text{otherwise}\end{cases}
\end{aligned}$$

$$\begin{aligned}
&\mathsf{getBE}(.) : \textsc{ALabels} \rightharpoonup E\\
&\mathsf{getBE}(\lambda) \triangleq \begin{cases} e & \text{if } \exists e.\ \lambda \in \{\mathsf{R}\langle e,-\rangle, \mathsf{U}\langle e,-\rangle, \mathsf{MF}\langle e\rangle, \mathsf{B}\langle e\rangle, \mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle, \mathsf{PSF}\langle e\rangle\}\\ \text{undefined} & \text{otherwise}\end{cases}
\end{aligned}$$

$$\begin{aligned}
&\mathsf{getPE}(.) : \textsc{ALabels} \rightharpoonup E\\
&\mathsf{getPE}(\lambda) \triangleq \begin{cases} e & \text{if } \exists e.\ \lambda{=}\mathsf{PB}\langle e\rangle\\ \text{undefined} & \text{otherwise}\end{cases}
\end{aligned}$$

and

$$\begin{aligned}
&\mathsf{prune}(\epsilon,\pi_2) \triangleq (\epsilon,\pi_2)\\
&\mathsf{prune}(\lambda.\pi_1,\pi_2) \triangleq \begin{cases} \mathsf{prune}(\pi_1 \setminus \lambda_d, \pi_2 \setminus \lambda_d) & \exists e,\lambda_d.\ \lambda \in \{\mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle, \mathsf{PSF}\langle e\rangle\}\\ & \wedge\ \lambda_d{=}\mathsf{D}\langle e\rangle \wedge \lambda_d \in \pi_1 \cup \pi_2\\ (\lambda.\pi_3,\pi_4) & \text{otherwise}\\ & \text{where } (\pi_3,\pi_4){=}\mathsf{prune}(\pi_1,\pi_2)\end{cases}
\end{aligned}$$

and

$$\begin{aligned}
G_1 \prec G_2 \overset{\text{def}}{\Leftrightarrow}\ & G_1.E{=}G_2.E \wedge G_1.I{=}G_2.I \wedge G_1.P{=}G_2.P\\
& \wedge\ G_1.\mathrm{po}{=}G_2.\mathrm{po} \wedge G_1.\mathsf{rf}{=}G_2.\mathsf{rf} \wedge G_1.\mathsf{nvo}{=}G_2.\mathsf{nvo}\\
& \wedge\ G_1.\mathsf{tso} \subseteq G_2.\mathsf{tso}
\end{aligned}$$

Definition 7.

$$\mathsf{sim}_{\mathsf{rec}}(\mathbf{rec},\mathsf{rec}) \overset{\text{def}}{\Leftrightarrow} \forall G, M, P.\ \mathsf{sim}_{\mathsf{GM}}(G,M) \Rightarrow \mathbf{rec}(P,M){=}\mathsf{rec}(\mathsf{P},G)$$

where

$$\mathsf{sim}_{\mathsf{GM}}(G,M) \overset{\text{def}}{\Leftrightarrow} \forall x,e.\ M(x){=}e \Rightarrow \exists e'.\ \max\left(\mathsf{nvo}|_{P \cap WU_x}\right){=}e' \wedge \mathsf{val}_{\mathsf{w}}(e){=}\mathsf{val}_{\mathsf{w}}(e')$$

A.2 Soundness of the Intermediate Semantics against Px86$_{\mathbf{man}}$ Declarative Semantics

Lemma 2. *For all $\Gamma, \mathcal{H}, \pi, \pi'$ and G, if $G = \mathsf{getG}(\Gamma,\pi,\pi')$ and $\mathcal{H}{=}\mathsf{hist}(\Gamma)$, then G is Px86$_{man}$-consistent.*

Proof. Pick arbitrary $G{=}\langle E, I, P, \mathrm{po}, \mathsf{rf}, \mathsf{tso}, \mathsf{nvo}\rangle$, Γ,$\mathcal{H}$, π and π' such that $G{=}\mathsf{getG}(\Gamma,\pi,\pi')$ and $\mathcal{H}{=}\mathsf{hist}(\Gamma)$. As $G{=}\mathsf{getG}(\Gamma,\pi,\pi')$, we know $\mathsf{wfp}(\pi.\pi',\mathcal{H})$ and $\mathsf{complete}(\pi.\pi')$ hold. It then suffices

to show:

$$I \subseteq P \tag{1}$$
$$P \subseteq D \tag{2}$$
$$I \times (E \setminus I) \subseteq \mathrm{po} \tag{3}$$
$$I \times (E \setminus I) \subseteq \mathrm{tso} \tag{4}$$
$$I \times (D \setminus I) \subseteq \mathrm{nvo} \tag{5}$$
$$dom(\mathrm{nvo}; [P]) \subseteq P \text{ and } P_n = D_n \tag{6}$$
$$I_1 = \left\{ init_x \;\middle|\; x \in \mathrm{Loc} \right\} \text{ and } I_{i+1} = \left\{ \max\left(\mathrm{nvo}|_{P \cap WU_x}\right) \;\middle|\; x \in \mathrm{Loc} \right\} \tag{7}$$
$$\mathrm{po} \text{ is a strict total order on } E \tag{8}$$
$$\mathrm{rf} \subseteq (W \cup U) \times (R \cup U) \text{ and is total and functional on } R \cup U \tag{9}$$
$$\mathrm{tso} \subseteq E \times E \text{ and is total on } E \setminus R \tag{10}$$
$$([W \cup U \cup R]; \mathrm{po}; [W \cup U \cup R]) \setminus (W \times R) \subseteq \mathrm{tso} \tag{11}$$
$$([E]; \mathrm{po}; [MF]) \cup ([MF]; \mathrm{po}; [E]) \subseteq \mathrm{tso} \tag{12}$$
$$\mathrm{rf} \subseteq \mathrm{tso} \cup \mathrm{po} \tag{13}$$
$$\forall (w, r) \in \mathrm{rf}.\ \forall w' \in W. \quad (w', r) \in \mathrm{tso} \cup \mathrm{po} \wedge \mathtt{loc}(w') = \mathtt{loc}(r) \Rightarrow (w, w') \notin \mathrm{tso} \tag{14}$$
$$[E \setminus R]; \mathrm{po}; [SF] \cup [SF]; \mathrm{po}; [E \setminus R] \subseteq \mathrm{tso} \tag{15}$$
$$\forall X \in \mathrm{CL}.\ ([W_X]; \mathrm{po}; [FO_X]) \subseteq \mathrm{tso} \tag{16}$$
$$([U]; \mathrm{po}; [FO]) \cup ([FO]; \mathrm{po}; [U]) \subseteq \mathrm{tso} \tag{17}$$
$$\forall X \in \mathrm{CL}.\ ([FL_X]; \mathrm{po}; [FO_X]) \cup ([FO_X]; \mathrm{po}; [FL_X]) \subseteq \mathrm{tso} \tag{18}$$
$$([W \cup U \cup FL]; \mathrm{po}; [FL]) \cup ([FL]; \mathrm{po}; [W \cup U \cup FL]) \subseteq \mathrm{tso} \tag{19}$$
$$\mathrm{nvo} \text{ is a strict total order on } D \tag{20}$$
$$dom(\mathrm{nvo}; [P]) \subseteq P \tag{21}$$
$$\forall x \in \mathrm{Loc}.\ \mathrm{tso}|_{D_x} \subseteq \mathrm{nvo} \tag{22}$$
$$[FO \cup FL]; \mathrm{tso}; [D] \subseteq \mathrm{nvo} \tag{23}$$
$$\forall X.\ [W_X \cup U_X]; \mathrm{tso}; [FO_X \cup FL_X] \subseteq \mathrm{nvo} \tag{24}$$

The proofs of parts (1), (3), (4), (5), (7), and (8) follow immediately from the construction of G.

RTS. (2)
Pick an arbitrary $e \in P$. We then know there exist $\lambda \in \pi, e$ such that $e{=}\mathtt{getPE}(\lambda)$ and $\lambda{=}\mathsf{PB}\langle e\rangle$, and thus $e \in W \cup U \cup FO \cup FL$. From $\mathrm{wfp}(\pi.\pi', \mathcal{H})$ we then know there exists $\lambda' \in \{\mathsf{B}\langle e\rangle, \mathsf{U}\langle e, -\rangle, \mathsf{J}\langle e\rangle\}$ such that $\lambda' \prec_{\pi.\pi'} \lambda$; and consequently from $\mathrm{wfp}(\pi.\pi', \mathcal{H})$ we know there exists λ'' such that $\lambda'' \in \{\mathsf{W}\langle e\rangle, \mathsf{FO}\langle e\rangle, \mathsf{FL}\langle e\rangle, \mathsf{U}\langle e, , \rangle \mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle\}$ such that $\lambda'' \prec_{\pi.\pi'} \lambda$. That is, $\mathtt{getE}(\lambda'')$=e. As such, from the definitions of E and D we have $e \in D$, as required.

RTS. (6)
Pick an arbitrary e_1, e_2 such that $(e_1, e_2) \in \mathrm{nvo}$ and $e_2 \in P$. From the definition of nvo we then know there exist $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1 = \mathtt{getPE}(\lambda_1)$, $e_2 = \mathtt{getPE}(\lambda_2)$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. On the other hand, from the definition of P and since $e_2 \in P$ we know that $\lambda_2 \in \pi$. As such, since $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and labels in $\pi.\pi'$ are fresh ($\mathrm{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds), we also know that $\lambda_1 \in \pi$. Consequently, since $e_1 = \mathtt{getPE}(\lambda_1)$ and $\lambda_1 \in \pi$, from the definition of P we have $e_1 \in P$, as required.

To demonstrate that $P_n = D_n$, it suffices to show that $D_n \subseteq P_n$, as in part (2) we established that $P_n \subseteq D_n$. Pick arbitrary $e \in D_n$. From the definition of D_n we then know there exists $\lambda \in \pi_n$ such that $\mathtt{getE}(\lambda) = e$ and $e \in WU_n \cup FO_n \cup FL_n$. There are then two cases to consider: 1) $\lambda \in \{\mathsf{W}\langle e\rangle, \mathsf{U}\langle e, -\rangle, \mathsf{FO}\langle e\rangle, \mathsf{FL}\langle e\rangle\}$; or 2) λ=$\mathsf{J}\langle e\rangle$; or In case (1), from $\mathrm{complete}(\pi_n.\pi_n')$ we know that there exists λ' such that $\lambda' = \mathsf{PB}\langle e\rangle$ and $\lambda' \in \pi_n.\pi_n'$. As $\pi_n' = \epsilon$ we know that $\lambda' \in \pi_n$. As such, from the definition of $\mathtt{getPE}(.)$ we know that $\mathtt{getPE}(\lambda') = e$ and thus $e \in P_n$, as required. In case (2), from $\mathrm{wfp}(\pi_n.\pi_n', \mathtt{hist}(\Gamma))$ we know that there exists λ' such that $\lambda' \in \{\mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle\}$ and $\lambda' \in \pi_n.\pi_n'$. As such, from $\mathrm{complete}(\pi_n.\pi_n')$ we know that there exists λ'' such that $\lambda'' = \mathsf{PB}\langle e\rangle$ and $\lambda'' \in \pi_n.\pi_n'$. As $\pi_n'' = \epsilon$ we know that $\lambda'' \in \pi_n$. As such, from the definition of $\mathtt{getPE}(.)$ we know that $\mathtt{getPE}(\lambda'') = e$ and thus $e \in P_n$, as required.

RTS. (9)
To demonstrate that $\mathsf{rf} \subseteq (W \cup U) \times (R \cup U)$, pick an arbitrary $(e_w, e_r) \in \mathsf{rf}$. From the definition of rf we then know there exists $\lambda \in \pi$ such that $\lambda = \mathsf{R}\langle e_r, e_w\rangle$ or $\lambda = \mathsf{U}\langle e_r, e_w\rangle$. As such from the type of annotated labels we know $e_r \in R \cup U$ and $e_w \in W \cup U$.

To demonstrate that rf is total on $R \cup U$, pick an arbitrary $r \in R \cup U$. Form the definition of E we then know there exist $\lambda \in \pi$ and e such that $\lambda = \mathsf{R}\langle r, e\rangle$ or $\lambda = \mathsf{U}\langle r, e\rangle$. As such we know $(e, r) \in \mathsf{rf}$ and thus rf is *total* on $R \cup U$.

To show rf is functional on R, pick an arbitrary $r \in R \cup U$. Form the definition of E we know there exists $\lambda \in \pi$ and e such that either $\lambda = \mathsf{R}\langle r, e\rangle$ or $\lambda = \mathsf{U}\langle r, e\rangle$. From the definition of rf we then know $(e, r) \in \mathsf{rf}$. Moreover, since π contains unique labels ($\mathrm{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds), we know $\forall e' \neq e.\ \mathsf{R}\langle r, e'\rangle \notin \pi$ and thus $\forall e' \neq e.\ (e', r) \notin \mathsf{rf}$. That is, rf is *functional* on R.

RTS. (10)
To demonstrate that $\mathsf{tso} \subseteq E \times E$, pick an arbitrary $(e_1, e_2) \in \mathsf{tso}$. We then know that either: 1) $(e_1, e_2) \in I \times (E \setminus I)$; or 2) there exist λ_1, λ_2 such that e_1=$\mathtt{getBE}(\lambda_1)$, e_2=$\mathtt{getBE}(\lambda_2)$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. In cases (1) we simply have $e_1, e_2 \in E$, as required. In case (2), from $\mathrm{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know there exist λ_1', λ_2' such that e_1=$\mathtt{getE}(\lambda_1')$, e_2=$\mathtt{getE}(\lambda_2')$. As such, from the definition of E we have $e_1, e_2 \in E$, as required.

Transitivity and strictness of tso follow from the definition of tso, transitivity and strictness of $\prec_{\pi.\pi'}$ and the freshness of events in $\pi.\pi'$ ($\mathrm{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds).

To demonstrate that tso is total on $E \setminus R$, pick arbitrary $e_1, e_2 \in E \setminus R$ such that $e_1 \neq e_2$. From the definitions of E we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $e_j = \mathtt{getE}(\lambda_j)$ for $j \in \{1, 2\}$. Moreover from $\mathrm{complete}(\pi.\pi')$ and given the definition of $\mathtt{getBE}(.)$ we know there exist $\lambda_1', \lambda_2' \in \pi.\pi'$ such that $e_j = \mathtt{getBE}(\lambda_j')$ for $j \in \{1, 2\}$. As $e_1 \neq e_2$ and $\pi_j'.\pi_j$ contains fresh labels ($\mathrm{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds), we know that $\lambda_1' \neq \lambda_2'$ and thus either $\lambda_1' \prec_{\pi.\pi'} \lambda_2'$ or $\lambda_2' \prec_{\pi.\pi'} \lambda_1'$. As such, from the definition of tso we have either $(e_1, e_2) \in \mathsf{tso}$ or $(e_2, e_1) \in \mathsf{tso}$, as required.

RTS. (11)
Pick an arbitrary $(e_1, e_2) \in ([W \cup U \cup R]; \mathrm{po}; [W \cup U \cup R]) \setminus (W \times R)$. From the definition of po we then know there exist τ and $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1 = \mathtt{getE}(\lambda_1)$, $e_2 = \mathtt{getE}(\lambda_2)$, $\mathtt{tid}(e_1) = \mathtt{tid}(e_2) = \tau$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. There are then four cases to consider: 1) $e_1, e_2 \in U \cup R$; or 2) $e_1 \in U \cup R$ and $e_2 \in W$; or 3) $e_1, e_2 \in W$; or 4) $e_1 \in W$ and $e_2 \in U$.

In case (1) we have $\lambda_1 \in \{\mathsf{R}\langle e_1, -\rangle, \mathsf{U}\langle e_1, -\rangle\}$, $\lambda_2 \in \{\mathsf{R}\langle e_2, -\rangle, \mathsf{U}\langle e_2, -\rangle\}$ and thus $\mathtt{getBE}(\lambda_1)$=$e_1$, $\mathtt{getBE}(\lambda_2)$=$e_2$. As such, since $\lambda_1 \prec_{\pi.\pi'} \lambda_2$, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) we have $\lambda_1 \in \{\mathsf{R}\langle e_1, -\rangle, \mathsf{U}\langle e_1, -\rangle\}$, $\lambda_2{=}\mathsf{W}\langle e_2\rangle$ and thus $\mathtt{getBE}(\lambda_1){=}e_1$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ and $\mathsf{complete}(\pi.\pi')$ we know there exists $\lambda_2'{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda_2 \prec_{\pi.\pi'} \lambda_2'$. That is, $\mathtt{getBE}(\lambda_2'){=}e_2$. As such, since $\lambda_1 \prec_{\pi.\pi'} \lambda_2 \prec_{\pi.\pi'} \lambda_2'$, i.e. $\lambda_1 \prec_{\pi.\pi'} \lambda_2'$, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (3) we have $\lambda_1{=}\mathsf{W}\langle e_1\rangle, \lambda_2{=}\mathsf{W}\langle e_2\rangle$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ and $\mathsf{complete}(\pi.\pi')$ we know there exists $\lambda_1'{=}\mathsf{B}\langle e_1\rangle, \lambda_2'{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda_1' \prec_{\pi.\pi'} \lambda_2'$. As such, since $\mathtt{getBE}(\lambda_1'){=}e_1$ and $\mathtt{getBE}(\lambda_2'){=}e_2$, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (4) we have $\lambda_1{=}\mathsf{W}\langle e_1\rangle, \lambda_2{=}\mathsf{U}\langle e_2, -\rangle$ and thus $\mathtt{getBE}(\lambda_2){=}e_2$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ and $\mathsf{complete}(\pi.\pi')$ we know there exists $\lambda_1'{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda_1' \prec_{\pi.\pi'} \lambda_2$. As such, since $\mathtt{getBE}(\lambda_1'){=}e_1$, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

RTS. (12)

To show that $[E]; \mathsf{po}; [MF] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [E]; \mathsf{po}; [MF]$. From the definition of po we then know there exist τ and $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1{=}\mathtt{getE}(\lambda_1)$, $\lambda_2{=}\mathsf{MF}\langle e_2\rangle$, $\mathtt{tid}(e_1){=}\mathtt{tid}(e_2) = \tau$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. There are then three cases to consider: 1) $\lambda_1 \in \{\mathsf{R}\langle e_1, -\rangle, \mathsf{U}\langle e_1, -\rangle, \mathsf{MF}\langle e_1\rangle\}$; or 2) $\lambda_1 \in \{\mathsf{W}\langle e_1\rangle, \mathsf{SF}\langle e_1\rangle, \mathsf{FO}\langle e_1\rangle, \mathsf{FL}\langle e_1\rangle\}$; or 3) $\lambda_1{=}\mathsf{J}\langle e_1\rangle$.

In case (1) we have $\mathtt{getBE}(\lambda_1){=}e_1$ and $\mathtt{getBE}(\lambda_2) = e_2$. As such, since $\lambda_1 \prec_{\pi.\pi'} \lambda_2$, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'{=}\mathsf{B}\langle e_1\rangle$ such that $\lambda' \prec_{\pi.\pi'} \lambda_2$. That is, $\mathtt{getBE}(\lambda'){=}e_1$ and $\mathtt{getBE}(\lambda_2) = e_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (3), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda' \in \{\mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle\}$ such that $\lambda' \prec_{\pi.\pi'} \lambda_1$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\lambda' \prec_{\pi.\pi'} \lambda_2$. On the other hand, we have $\mathtt{getBE}(\lambda'){=}e_1$ and $\mathtt{getBE}(\lambda_2) = e_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

To show $[MF]; \mathsf{po}; [E] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [MF]; \mathsf{po}; [E]$. From the definition of po we then know there exist τ and $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_2{=}\mathtt{getE}(\lambda_2)$, $\lambda_1{=}\mathsf{MF}\langle e_1\rangle$, $\mathtt{tid}(e_1){=}\mathtt{tid}(e_2) = \tau$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. There are then three cases to consider: 1) $\lambda_2 \in \{\mathsf{R}\langle e_2, -\rangle, \mathsf{U}\langle e_2, -\rangle, \mathsf{MF}\langle e_2\rangle\}$; or 2) $\lambda_2 \in \{\mathsf{W}\langle e_2\rangle, \mathsf{SF}\langle e_2\rangle, \mathsf{FO}\langle e_2\rangle, \mathsf{FL}\langle e_2\rangle\}$; or 3) $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1) we have $\mathtt{getBE}(\lambda_1){=}e_1$ and $\mathtt{getBE}(\lambda_2) = e_2$. As such, since $\lambda_1 \prec_{\pi.\pi'} \lambda_2$, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda_2 \prec_{\pi.\pi'} \lambda'$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\lambda_1 \prec_{\pi.\pi'} \lambda'$. Moreover, we have $\mathtt{getBE}(\lambda'){=}e_2$ and $\mathtt{getBE}(\lambda_1) = e_1$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (3), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda' \in \{\mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$ such that $\lambda' \prec_{\pi.\pi'} \lambda_2$. There are now two cases to consider: a) $\lambda_1 \prec_{\pi.\pi'} \lambda'$; or b) $\lambda' \prec_{\pi.\pi'} \lambda_1$. In case (3.a), we then have $\mathtt{getBE}(\lambda'){=}e_2$ and $\mathtt{getBE}(\lambda_1) = e_1$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (3.b), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda''{=}\mathsf{J}\langle e_2\rangle$ such that $\lambda'' \prec_{\pi.\pi'} \lambda_1$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know that $\pi.\pi'$ contains unique labels and thus $\lambda''{=}\lambda_2$. As such, we have $\lambda_2 \prec_{\pi.\pi'} \lambda_1$. This however leads to a contradiction as we also have $\lambda_1 \prec_{\pi.\pi'} \lambda_2$.

RTS. (13)

Pick arbitrary $(w, r) \in \mathsf{rf}$. From the definition of rf we then know there exists $\lambda \in \pi$ such that $\lambda = \mathsf{R}\langle r, w\rangle$. On the other hand, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{wfrd}(r, w, \pi, \pi')$ holds and thus either 1) there exists λ' such that $\lambda'{=}\mathsf{W}\langle w\rangle$ and $\lambda' \prec_\pi \lambda$ and $\mathtt{tid}(w){=}\mathtt{tid}(r)$; or 2) there exists λ' such that $\lambda' \in \{\mathsf{B}\langle w\rangle, \mathsf{U}\langle w, -\rangle\}$ and $\lambda' \prec_\pi \lambda$; or 3) $w \in I$. In case (1) from the definition of po we

then have $(w, r) \in \mathsf{po}$, as required. In case (2) from the definition of tso we then have $(w, r) \in \mathsf{tso}$, as required. In case (3) from the definition of po we then have $(w, r) \in \mathsf{po}$, as required.

RTS. (14)
Pick arbitrary $(w, r) \in \mathsf{rf}$ and $w' \in W$ such that $(w', r) \in \mathsf{tso} \cup \mathsf{po}$ and $\mathtt{loc}(w') = \mathtt{loc}(r)$. If $w' = w$, from the strictness of tso we immediately know that $(w, w') \notin \mathsf{tso}$, as required.

Now let us consider the case where $w' \neq w$. From the construction of rf we then know there exist $\lambda_r \in \pi$ such that either $\lambda_r = \mathsf{R}\langle r, w\rangle$ or $\lambda_r = \mathsf{U}\langle r, w\rangle$. From $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we then know that either 1) there exists $\lambda = \mathsf{B}\langle w\rangle \prec_\pi \lambda_r$; or 2) there exists $\lambda = \mathsf{U}\langle w, -\rangle \prec_\pi \lambda_r$; or 3) there exists $\lambda = \mathsf{W}\langle w\rangle \prec_\pi \lambda_r$ and $\mathtt{tid}(w) = \mathtt{tid}(r)$; or 4) $w \in I$.

On the other hand, from the construction of tso, po and since $(w', r) \in \mathsf{tso} \cup \mathsf{po}$ we know that either: a) there exists $\lambda' = \mathsf{B}\langle w'\rangle \prec_\pi r$; or b) there exists $\lambda' = \mathsf{U}\langle w', -\rangle \prec_\pi r$; or c) there exists $\lambda' = \mathsf{W}\langle w'\rangle \prec_\pi \lambda_r$ and $\mathtt{tid}(w') = \mathtt{tid}(r)$; or d) $w' \in I$.

However, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$, the definition of $\mathsf{wfrd}(.,.,.,.)$ and since $\lambda_\in\{\mathsf{R}\langle r, w\rangle, \mathsf{U}\langle r, w\rangle\}$, in cases (1.a), (1.b), (1.c), (2.a), (2.b), (2.c), (3.a), (3.b), (3.c) we have $\lambda' \prec_\pi \lambda$. Consequently, in cases (1.a), (1.b), (2.a), (2.b) from the definition of tso we have $(w', w) \in \mathsf{tso}$, i.e. $(w, w') \notin \mathsf{tso}$, as required.

In cases (3.a) and (3.b) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ and $\mathsf{complete}(\pi.\pi')$ we additionally know there exist $\lambda'' = \mathsf{B}\langle w\rangle$ such that $\lambda \prec_{\pi.\pi'} \lambda''$ and thus from the transitivity of $\prec$ we have $\lambda' \prec_{\pi.\pi'} \lambda''$. Consequently, from the definition of tso we have $(w', w) \in \mathsf{tso}$, i.e. $(w, w') \notin \mathsf{tso}$, as required.

In cases (1.c) and (2.c) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$, $\mathsf{complete}(\pi.\pi')$ and the definition of $\mathsf{wfrd}(.,.,.,.)$ we additionally know there exist $\lambda''=\mathsf{B}\langle w'\rangle$ such that $\lambda'' \prec_{\pi.\pi'} \lambda$. Consequently, from the definition of tso we have $(w', w) \in \mathsf{tso}$, i.e. $(w, w') \notin \mathsf{tso}$, as required. In case (3.c) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ and $\mathsf{complete}(\pi.\pi')$ we additionally know there exist $\lambda_2=\mathsf{B}\langle w'\rangle$ and $\lambda_1=\mathsf{B}\langle w\rangle$ such that $\lambda_2 \prec_{\pi.\pi'} \lambda_1$. Consequently, from the definition of tso we have $(w', w) \in \mathsf{tso}$, i.e. $(w, w') \notin \mathsf{tso}$, as required.

In cases (2.d), (3.d) from the definition of tso we have $(w', w) \in \mathsf{tso}$, i.e. $(w, w') \notin \mathsf{tso}$, as required. Similarly, in case (1.d) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{W}\langle w\rangle \in \pi$ and thus from the definition of tso we have $(w', w) \in \mathsf{tso}$, i.e. $(w, w') \notin \mathsf{tso}$, as required.

Cases (4.a), (4.b) and (4.c) cannot arise as from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ and the definition of $\mathsf{wfrd}(.,.,.,.)$ we arrive at a contradiction. Case (4.d) cannot arise as $w \neq w'$ and from the definition of I we cannot have two distinct events of the same location in I.

RTS. (15)
To show that $[E \setminus R]; \mathsf{po}; [SF] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [E \setminus R]; \mathsf{po}; [SF]$. From the definition of po we then know there exist τ and $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1=\mathtt{getE}(\lambda_1)$, $e_2 \in SF$, $\lambda_2 \in \{\mathsf{SF}\langle e_2\rangle, \mathsf{J}\langle e_2\rangle\}$, $\mathtt{tid}(e_1)=\mathtt{tid}(e_2) = \tau$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. There are then six cases to consider: 1.1) $\lambda_1 \in \{\mathsf{U}\langle e_1, -\rangle, \mathsf{MF}\langle e_1\rangle\}$ and $\lambda_2=\mathsf{SF}\langle e_2\rangle$; or 1.2) $\lambda_1 \in \{\mathsf{U}\langle e_1, -\rangle, \mathsf{MF}\langle e_1\rangle\}$ and $\lambda_2=\mathsf{J}\langle e_2\rangle$; or 2.1) $\lambda_1 \in \{\mathsf{W}\langle e_1\rangle, \mathsf{SF}\langle e_1\rangle, \mathsf{FO}\langle e_1\rangle, \mathsf{FL}\langle e_1\rangle\}$ and $\lambda_2=\mathsf{SF}\langle e_2\rangle$; or 2.2) $\lambda_1 \in \{\mathsf{W}\langle e_1\rangle, \mathsf{SF}\langle e_1\rangle, \mathsf{FO}\langle e_1\rangle, \mathsf{FL}\langle e_1\rangle\}$ and $\lambda_2=\mathsf{J}\langle e_2\rangle$; or 3.1) $\lambda_1=\mathsf{J}\langle e_1\rangle$ and $\lambda_2=\mathsf{SF}\langle e_2\rangle$; or 3.2) $\lambda_1=\mathsf{J}\langle e_1\rangle$ and $\lambda_2=\mathsf{J}\langle e_2\rangle$.

In case (1.1) we have $\mathtt{getBE}(\lambda_1)=e_1$. We also know that there exists $\lambda'=\mathsf{B}\langle e_2\rangle$ such that $\lambda_2 \prec_{\pi.\pi'} \lambda'$ and thus $\mathtt{getBE}(\lambda') = e_2$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\lambda_1 \prec_{\pi.\pi'} \lambda'$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (1.2) we have $\mathtt{getBE}(\lambda_1)=e_1$. We also know that there exists $\lambda'=\mathsf{PSF}\langle e_2\rangle$ such that $\lambda' \prec_{\pi.\pi'} \lambda_2$ and thus $\mathtt{getBE}(\lambda') = e_2$. There are now two cases to consider: a) $\lambda_1 \prec_{\pi.\pi'} \lambda'$; or b) $\lambda' \prec_{\pi.\pi'} \lambda_1$. In case (a) from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (b) since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda''=\mathsf{J}\langle e_2\rangle$ such that $\lambda'' \prec_{\pi.\pi'} \lambda_1$. As such, since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know $\lambda_2=\lambda''$. That is, $\lambda_2 \prec_{\pi.\pi'} \lambda_1$. This however leads to a contradiction as we also have $\lambda_1 \prec_{\pi.\pi'} \lambda_2$.

In case (2.1), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_1{=}\mathsf{B}\langle e_1\rangle$ and $\lambda'_2{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$. That is, $\mathtt{getBE}(\lambda'_1){=}e_1$ and $\mathtt{getBE}(\lambda'_2) = e_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in$ tso, as required.

Similarly, In case (2.1), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_2{=}\mathsf{PSF}\langle e_2\rangle$ such that $\lambda'_2 \prec_{\pi.\pi'} \lambda_2$. That is, $\mathtt{getBE}(\lambda'_2){=}e_2$. There are now two cases to consider: a) $\lambda_1 \prec_{\pi.\pi'} \lambda'_2$; or b) $\lambda'_2 \prec_{\pi.\pi'} \lambda_1$. In case (a), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_1{=}\mathsf{B}\langle e_1\rangle$ such that $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$. That is, $\mathtt{getBE}(\lambda'_1){=}e_1$. Consequently, from the definition of tso we have $(e_1, e_2) \in$ tso, as required. In case (b), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda''{=}\mathsf{J}\langle e_2\rangle$ such that $\lambda'' \prec_{\pi.\pi'} \lambda_1$. As such, since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know $\lambda_2{=}\lambda''$. That is, $\lambda_2 \prec_{\pi.\pi'} \lambda_1$. This however leads to a contradiction as we also have $\lambda_1 \prec_{\pi.\pi'} \lambda_2$.

In case (3.1), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_1 \in \{\mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle, \mathsf{PSF}\langle e_1\rangle\}$ such that $\lambda'_1 \prec_{\pi.\pi'} \lambda_1$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we also know there exists $\lambda'_2{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda_2 \prec_{\pi.\pi'} \lambda'_2$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$. On the other hand, we have $\mathtt{getBE}(\lambda'_1){=}e_1$ and $\mathtt{getBE}(\lambda'_2) = e_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in$ tso, as required.

In case (3.2), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_1 \in \{\mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle, \mathsf{PSF}\langle e_1\rangle\}$ and $\lambda'_2{=}\mathsf{PSF}\langle e_2\rangle$ such that $\lambda'_1 \prec_{\pi.\pi'} \lambda_1$ and $\lambda'_2 \prec_{\pi.\pi'} \lambda_2$. That is, $\mathtt{getBE}(\lambda'_1){=}e_1$ and $\mathtt{getBE}(\lambda'_2) = e_2$. There are now two cases to consider: a) $\lambda_1 \prec_{\pi.\pi'} \lambda'_2$; or b) $\lambda'_2 \prec_{\pi.\pi'} \lambda_1$. In case (a) from the transitivity of $\prec_{\pi.\pi'}$ we have $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$. As such, from the definition of tso we have $(e_1, e_2) \in$ tso, as required. In case (b), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda''{=}\mathsf{J}\langle e_2\rangle$ such that $\lambda'' \prec_{\pi.\pi'} \lambda_1$. As such, since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know $\lambda_2{=}\lambda''$. That is, $\lambda_2 \prec_{\pi.\pi'} \lambda_1$. This however leads to a contradiction as we also have $\lambda_1 \prec_{\pi.\pi'} \lambda_2$.

To show $[SF]$; po; $[E \backslash R] \subseteq$ tso, pick an arbitrary $(e_1, e_2) \in [SF]$; po; $[E \backslash R]$. From the definition of po we then know there exist τ and $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_2{=}\mathtt{getE}(\lambda_2)$, $\lambda_1{=}\mathsf{SF}\langle e_1\rangle$, $\mathtt{tid}(e_1){=}\mathtt{tid}(e_2) = \tau$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. There are then three cases to consider: 1) $\lambda_2 \in \{\mathsf{U}\langle e_2, -\rangle, \mathsf{MF}\langle e_2\rangle\}$; or 2) $\lambda_2 \in \{\mathsf{W}\langle e_2\rangle, \mathsf{SF}\langle e_2\rangle, \mathsf{FO}\langle e_2\rangle, \mathsf{FL}\langle e_2\rangle\}$; or 3) $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1) we know $\mathtt{getBE}(\lambda_2) = e_2$. We also know that there exists $\lambda'{=}\mathsf{B}\langle e_1\rangle$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda'$ and thus $\mathtt{getBE}(\lambda') = e_1$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\lambda' \prec_{\pi.\pi'} \lambda_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in$ tso, as required.

In case (2), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_1{=}\mathsf{B}\langle e_1\rangle$ and $\lambda'_2{=}\mathsf{B}\langle e_2\rangle$ such that $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in$ tso, as required.

In case (3), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_2 \in \{\mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$ such that $\lambda'_2 \prec_{\pi.\pi'} \lambda_2$. There are now two cases to consider: a) $\lambda_1 \prec_{\pi.\pi'} \lambda'_2$; or b) $\lambda'_2 \prec_{\pi.\pi'} \lambda_1$.

In case (3.a), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda'_1{=}\mathsf{B}\langle e_1\rangle$ such that $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$. On the other hand, we have $\mathtt{getBE}(\lambda'_1){=}e_1$ and $\mathtt{getBE}(\lambda'_2) = e_2$. Consequently, from the definition of tso we have $(e_1, e_2) \in$ tso, as required.

In case (3.b), since $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds, we know there exists $\lambda''{=}\mathsf{J}\langle e_2\rangle$ such that $\lambda'' \prec_{\pi.\pi'} \lambda_1$. Moreover, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know that $\pi.\pi'$ contains unique labels and thus $\lambda''{=}\lambda_2$. As such, we have $\lambda_2 \prec_{\pi.\pi'} \lambda_1$. This however leads to a contradiction as we also have $\lambda_1 \prec_{\pi.\pi'} \lambda_2$.

RTS. (16)
Pick an arbitrary X. To show that $[W_X]$; po; $[FO_X] \subseteq$ tso, pick an arbitrary $(e_1, e_2) \in [W_X]$; po; $[FO_X]$, i.e. $(e_1, e_2) \in$ po, $e_1 \in W$, $e_2 \in FO$ and $\mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_1{=}\mathsf{W}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{FO}\langle e_2\rangle$; or 2) $\lambda_1{=}\mathsf{W}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1), from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know that $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$. There are now two cases to consider: i) $\mathsf{W}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFO}\langle e_2\rangle$; or ii) $\mathsf{PFO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{W}\langle e_1\rangle$. In case (2.i) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFO}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (2.ii) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know there exists $\lambda{=}\mathsf{J}\langle e_2\rangle$ such that $\mathsf{J}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{W}\langle e_1\rangle$. As the labels in $\pi.\pi'$ are unique, this however leads to contradiction as we also have $\mathsf{W}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$.

RTS. (17)

To show $[U]; \mathsf{po}; [FO] \subseteq \mathsf{tso}$ pick an arbitrary $(e_1, e_2) \in [U]; \mathsf{po}; [FO]$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_1{=}\mathsf{U}\langle e_1, -\rangle$ and $\lambda_2{=}\mathsf{FO}\langle e_2\rangle$; or 2) $\lambda_1{=}\mathsf{U}\langle e_1, -\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{FO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$ and thus from transitivity of $\prec_{\pi.\pi'}$ we have $\mathsf{U}\langle e_1, -\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$. There are now two cases to consider: i) $\mathsf{U}\langle e_1, -\rangle \prec_{\pi.\pi'} \mathsf{PFO}\langle e_2\rangle$; or ii) $\mathsf{PFO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{U}\langle e_1, -\rangle$. In case (3.i) from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (3.ii) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know there exists $\lambda{=}\mathsf{J}\langle e_2\rangle$ such that $\mathsf{J}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{U}\langle e_1, -\rangle$. As the labels in $\pi.\pi'$ are unique, this however leads to contradiction as we also have $\mathsf{U}\langle e_1, -\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$.

To show $[FO]; \mathsf{po}; [U] \subseteq \mathsf{tso}$ pick an arbitrary $(e_1, e_2) \in [FO]; \mathsf{po}; [U]$. That is, $(e_1, e_2) \in \mathsf{po}$, $e_2 \in U$ and $e_1 \in FO$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_2{=}\mathsf{U}\langle e_2, -\rangle$ and $\lambda_1{=}\mathsf{FO}\langle e_1\rangle$; or 2) $\lambda_2{=}\mathsf{U}\langle e_2, -\rangle$ and $\lambda_1{=}\mathsf{J}\langle e_1\rangle$.

In case (1) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{U}\langle e_2, -\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFO}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_1\rangle$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we know $\mathsf{PFO}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{U}\langle e_2, -\rangle$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

RTS. (18)

Pick an arbitrary X. To show $[FL_X]; \mathsf{po}; [FO_X] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [FL_X]; \mathsf{po}; [FO_X]$; i.e. $(e_1, e_2) \in \mathsf{po}$, $e_1 \in FL$, $e_2 \in FO$ and $\mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_1{=}\mathsf{FL}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{FO}\langle e_2\rangle$; or 2) $\lambda_1{=}\mathsf{FL}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$; or 3) $\lambda_1{=}\mathsf{J}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{FO}\langle e_2\rangle$; or 4) $\lambda_1{=}\mathsf{J}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$. There are now two cases to consider: i) $\mathsf{FL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFO}\langle e_2\rangle$; or ii) $\mathsf{PFO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{FL}\langle e_1\rangle$. In case (2.i) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFO}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (2.ii), from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{J}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{FL}\langle e_1\rangle$. This however leads to a contradiction as we also have $\mathsf{FL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$.

In case (3) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_1\rangle$ and $\mathsf{FO}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$, and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (4) since $e_1 \in FL$ and $e_2 \in FO$, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFO}\langle e_2\rangle$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

The proof of $[FO_X]; \mathsf{po}; [FL_X] \subseteq \mathsf{tso}$ is analogous and is omitted here.

RTS. (19)
To show $[W]; \mathsf{po}; [FL] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [W]; \mathsf{po}; [FL]$, i.e. $(e_1, e_2) \in \mathsf{po}$, $e_1 \in W$ and $e_2 \in FL$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_1{=}\mathsf{W}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{FL}\langle e_2\rangle$; or 2) $\lambda_1{=}\mathsf{W}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1), from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know that $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFL}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$. There are now two cases to consider: i) $\mathsf{W}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFL}\langle e_2\rangle$; or ii) $\mathsf{PFL}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{W}\langle e_1\rangle$. In case (2.i) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFL}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (2.ii) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know there exists $\lambda{=}\mathsf{J}\langle e_2\rangle$ such that $\mathsf{J}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{W}\langle e_1\rangle$. As the labels in $\pi.\pi'$ are unique, this however leads to contradiction as we also have $\mathsf{W}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$.

To show $[FL]; \mathsf{po}; [W] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [FL]; \mathsf{po}; [W]$; i.e. $(e_1, e_2) \in \mathsf{po}$, $e_2 \in W$ and $e_1 \in FL$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_2{=}\mathsf{W}\langle e_2\rangle$ and $\lambda_1{=}\mathsf{FL}\langle e_1\rangle$; or 2) $\lambda_2{=}\mathsf{W}\langle e_2\rangle$ and $\lambda_1{=}\mathsf{J}\langle e_1\rangle$.

In case (1), from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know that $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_1\rangle$ and $\mathsf{W}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we know $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

To show $[FL]; \mathsf{po}; [FL] \subseteq \mathsf{tso}$, pick an arbitrary $(e_1, e_2) \in [FL]; \mathsf{po}; [FL]$; i.e. $(e_1, e_2) \in \mathsf{po}$ and $e_1, e_2 \in FL$. From the definition of po we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $\lambda_1 \prec_{\pi.\pi'} \lambda_2$ and either 1) $\lambda_1{=}\mathsf{FL}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{FL}\langle e_2\rangle$; or 2) $\lambda_1{=}\mathsf{FL}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$; or 3) $\lambda_1{=}\mathsf{J}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{FL}\langle e_2\rangle$; or 4) $\lambda_1{=}\mathsf{J}\langle e_1\rangle$ and $\lambda_2{=}\mathsf{J}\langle e_2\rangle$.

In case (1) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (2) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we know $\mathsf{PFL}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$. There are now two cases to consider: i) $\mathsf{FL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFL}\langle e_2\rangle$; or ii) $\mathsf{PFL}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{FL}\langle e_1\rangle$. In case (2.i) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{B}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFL}\langle e_2\rangle$ and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required. In case (2.ii), from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{J}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{FL}\langle e_1\rangle$. This however leads to a contradiction as we also have $\mathsf{FL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_2\rangle$.

In case (3) from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{J}\langle e_1\rangle$ and $\mathsf{FL}\langle e_2\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$. As such, from the transitivity of $\prec_{\pi.\pi'}$ we have $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{B}\langle e_2\rangle$, and thus from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

In case (4) since $e_1, e_2 \in FL$, from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{PFL}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PFL}\langle e_2\rangle$. Consequently, from the definition of tso we have $(e_1, e_2) \in \mathsf{tso}$, as required.

The proof of $([U]; \mathsf{po}; [FL]) \cup ([FL]; \mathsf{po}; [U]) \subseteq \mathsf{tso}$ is analogous to that of part (17) and is omitted here.

RTS. (20)
Transitivity and strictness of nvo follow from the definition of nvo, transitivity and strictness of $\prec_{\pi.\pi'}$ and the freshness of events in $\pi.\pi'$ ($\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds).

To demonstrate that nvo is total on D, pick arbitrary $e_1, e_2 \in D$ such that $e_1 \neq e_2$. From the definitions of E we know there exist $\lambda_1, \lambda_2 \in \pi$ such that $e_j = \mathtt{getE}(\lambda_j)$ for $j \in \{1, 2\}$. Moreover

from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$, $\mathsf{complete}(\pi.\pi')$ and given the definition of $\mathtt{getPE}(.)$ we know there exist $\lambda'_1, \lambda'_2 \in \pi.\pi'$ such that $e_j = \mathtt{getPE}(\lambda'_j)$ for $j \in \{1, 2\}$. As $e_1 \neq e_2$ and $\pi'_j.\pi_j$ contains fresh labels ($\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ holds), we know that $\lambda'_1 \neq \lambda'_2$ and thus either $\lambda'_1 \prec_{\pi.\pi'} \lambda'_2$ or $\lambda'_2 \prec_{\pi.\pi'} \lambda'_1$. As such, from the definition of nvo we have either $(e_1, e_2) \in \mathsf{nvo}$ or $(e_2, e_1) \in \mathsf{nvo}$, as required.

RTS. (21)
Pick an arbitrary $e \in \mathit{dom}(\mathsf{nvo}; [P])$, i.e. there exists $e' \in P$ such that $(e, e') \in \mathsf{nvo}$. From the definition of nvo we then know there exists λ, λ' such that $\mathtt{getPE}(\lambda)$=e, $\mathtt{getPE}(\lambda')$=e' and $\lambda \prec_{\pi.\pi'} \lambda'$. Moreover, since $e' \in P$, from the definition of P we know $\lambda' \in \pi$ and thus $\lambda \prec_{\pi} \lambda'$. As such, we know $\lambda \in \pi$. Consequently, since $\mathtt{getPE}(\lambda)$=e, from the definition of P we have $e \in P$, as required.

RTS. (22)
Pick an arbitrary x and $(e_1, e_2) \in \mathsf{tso}|_{D_x}$; that is, $e_1, e_2 \in D$ and $\mathtt{loc}(e_1) = \mathtt{loc}(e_2) = x$. From the definition of tso we then know there exist $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1 = \mathtt{getBE}(\lambda_1)$, $e_2 = \mathtt{getBE}(\lambda_2)$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. There are now three cases to consider:
1) $e_1, e_2 \in W \cup U$, i.e. $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\}$, $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, -\rangle\}$;
2) $e_1 \in W \cup U$, $e_2 \in \mathit{FO} \cup \mathit{FL}$, i.e. $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\}$, $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$;
3) $e_1 \in \mathit{FO} \cup \mathit{FL}$, $e_2 \in D$, i.e. $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle\}$ and $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, -\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$.
In all three cases from $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we have $\mathsf{PB}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PB}\langle e_2\rangle$ and thus from the definition of nvo we have $(e_1, e_2) \in \mathsf{nvo}$, as required.

RTS. (23)
Pick an arbitrary $(e_1, e_2) \in [\mathit{FO} \cup \mathit{FL}]; \mathsf{tso}; [D]$; that is, $e_1 \in \mathit{FO} \cup \mathit{FL}$ and $e_2 \in D$. From the definition of tso we then know there exist $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1 = \mathtt{getBE}(\lambda_1)$, $e_2 = \mathtt{getBE}(\lambda_2)$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. That is, $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle\}$ and $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, -\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$. From $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we then have $\mathsf{PB}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PB}\langle e_2\rangle$ and thus from the definition of nvo we have $(e_1, e_2) \in \mathsf{nvo}$, as required.

RTS. (24)
Pick an arbitrary X and $(e_1, e_2) \in [W_X \cup U_X]; \mathsf{tso}; [\mathit{FO}_X \cup \mathit{FL}_X]$; that is, $e_1 \in W \cup U$, $e_2 \in \mathit{FO} \cup \mathit{FL}$ and $\mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X$. From the definition of tso we then know there exist $\lambda_1, \lambda_2 \in \pi.\pi'$ such that $e_1 = \mathtt{getBE}(\lambda_1)$, $e_2 = \mathtt{getBE}(\lambda_2)$ and $\lambda_1 \prec_{\pi.\pi'} \lambda_2$. That is, $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\}$ and $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$. From $\mathsf{wfp}(\pi.\pi', \mathtt{hist}(\Gamma))$ we then have $\mathsf{PB}\langle e_1\rangle \prec_{\pi.\pi'} \mathsf{PB}\langle e_2\rangle$ and thus from the definition of nvo we have $(e_1, e_2) \in \mathsf{nvo}$, as required. □

Theorem 3 (soundness). *For all* **rec**, $\mathsf{P}, M, \mathcal{H} = (\pi_1, \pi'_1). \cdots .(\pi_{n-1}, \pi'_{n-1}), \pi_n$ *and* $\pi'_n = \epsilon$*:*

$$\mathbf{rec} \vdash \mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, M, PB_0, B_0, \mathcal{H}, \pi_n$$

then

(1) $\mathsf{P}, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, \Gamma, \pi_n$ *where*

$$\begin{array}{ll} \Gamma = \Gamma_n & \\ \Gamma_1{=}\epsilon \qquad \Gamma_{j+1}{=}(G_1, (\pi_1, \pi'_1)). \cdots .(G_j, (\pi_j, \pi'_j)) & \textit{for } j \in \{1 \cdots n{-}1\} \\ G_i = \mathtt{getG}(\Gamma_i, \pi_i, \pi'_i) \qquad & \textit{for } i \in \{1 \cdots n\} \end{array}$$

(2) The chain $C = G_1, \cdots, G_n$ *is Px86$_{man}$-valid.*

Proof. Pick arbitrary $\mathsf{P}, M, \mathcal{H} = (\pi_1, \pi'_1). \cdots .(\pi_{n-1}, \pi'_{n-1}), \pi_n$ such that

$$\mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, M, PB_0, B_0, \mathcal{H}, \pi_n$$

and let $\pi'_n = \epsilon$. The proof of the first part follows from Lemma 1, Lemma 2 and by induction on the length of the event-annotated transition $\Rightarrow^*$.
For the second part, for each $i \in \{1 \cdots n\}$ and $G_i = \mathtt{getG}(\Gamma_i, \pi_i, \pi'_i) \in C$, from Lemma 2 we know G_i is Px86_{man}-consistent. As such, from the definition of validity we have C is Px86_{man}-valid. □

A.3 Completeness of the Intermediate Semantics against Px86_{man} Declarative Semantics

Definition 8. Given a Px86_{man}-consistent execution G, the set of traces induced by G, written $\mathtt{traces}(G)$, includes those non-empty histories that satisfy the following condition:

$$\mathcal{H}.(\pi, \pi') \in \mathtt{traces}(G) \overset{\text{def}}{\Leftrightarrow} \mathsf{norm}(\pi.\pi') \wedge \exists G'.\ \mathtt{getG}(\mathcal{H}, \pi, \pi') {=} G' \wedge G \prec G'$$

where

$$\mathsf{norm}(\pi) \overset{\text{def}}{\Leftrightarrow} \forall e.\ \mathsf{D}\langle e\rangle \notin \pi$$

Given a Px86_{man}-valid chain $C{=}G_1, \cdots, G_n$, the set of traces induced by C, written $\mathtt{traces}(C)$, includes those non-empty histories $\mathcal{H} = (\pi_1, \pi'_1), \cdots, (\pi_n, \pi'_n)$ that satisfy the following conditions:

$$\mathcal{H} \in \mathtt{traces}(C) \overset{\text{def}}{\Leftrightarrow} \forall \lambda \in \pi'_n.\ \exists e \in SF.\ \lambda{=}\mathsf{B}\langle e\rangle \wedge \bigwedge_{i=1}^{n} \mathcal{H}_i.(\pi_i, \pi'_i) \in \mathtt{traces}(G_i)$$

where $\mathcal{H}_1 = \epsilon$ and $\mathcal{H}_j = \mathcal{H}_{j-1}.(\pi_j, \pi'_j)$ for $j \in \{2 \cdots n\}$.

Lemma 3. *For all chains $C = G_1, \cdots, G_n$, if C is Px86$_{man}$-valid, then $\mathtt{traces}(C) \neq \emptyset$.*

Proof. Pick an arbitrary Px86_{man}-valid chain $C = G_1, \cdots, G_n$. We then show how to construct $(\pi_1, \pi'_1), \cdots, (\pi_n, \pi'_n)$ such that $\mathcal{H}_i.(\pi_i, \pi'_i) \in \mathtt{traces}(G_i)$ for all $i \in \{1 \cdots n\}$, where $\pi'_n = \epsilon$ and $\mathcal{H}_i$ is as defined above.

For each $i \in \{1 \cdots n\}$, given $\mathcal{H}_i$ as defined above and $G_i = (I, P, E, \mathsf{po}, \mathsf{rf}, \mathsf{tso}, \mathsf{nvo})$, we construct (π_i, π'_i) as follows. Let $\{r_1 \cdots r_q\}$ denote an enumeration of $G_i.R$ and $\{w_1, \cdots, w_s\}$ denote an enumeration of $G_k.WU$. For each $j \in \{1 \cdots q\}$ and $l \in \{0 \cdots s{-}1\}$ where $(w, r_j) \in \mathsf{rf}$, we then define

$$\mathsf{tso}_j^{l+1} \triangleq \begin{cases} \left(\mathsf{tso}_j^l \cup \left\{(r_j, w_{l+1})\right\}\right)^+ & \text{if } (r_j, w_{l+1}) \notin \mathsf{tso}_j^l \cup (\mathsf{tso}_j^l)^{-1} \\ & \text{and } (w, w_{l+1}) \in \mathsf{tso} \\ \mathsf{tso}_j^l & \text{otherwise} \end{cases}$$

where $\mathsf{tso}_1^0 = \mathsf{tso}$ and $\mathsf{tso}_{j+1}^0 = \mathsf{tso}_j^s$ for $j \in \{1 \cdots q{-}1\}$. Note that each tso_j^l is 1) total on writes and includes tso; and 2) is a strict partial order on E. We next show that:

$$\begin{aligned} &\forall j \in \{1 \cdots q\}.\ \forall l \in \{0 \cdots s\}.\ \forall w, r.\ \forall w' \in W \cup U. \\ &\quad (w, r) \in \mathsf{rf} \wedge (w', r) \in \mathsf{tso}_j^l \cup \mathsf{po} \wedge \mathtt{loc}(w) = \mathtt{loc}(w') \Rightarrow (w, w') \notin \mathsf{tso}_j^l \end{aligned} \tag{RFJ}$$

We proceed by double induction on j and l.

Base case $j = 1$ and $l = 0$
As G_i is Px86_{man}-valid, we know that the desired property holds of tso and thus of $\mathsf{tso}_1^0 = \mathsf{tso}$ by definition.

Inductive case $j = 1$ and $l = a{+}1$ with $0 \leq a < s$

$$\begin{aligned} &\forall l' \in \{1 \cdots a\}.\ \forall w, r.\ \forall w' \in W \cup U. \\ &\quad (w, r) \in \mathsf{rf} \wedge (w', r) \in \mathsf{tso}_1^{l'} \cup \mathsf{po} \wedge \mathtt{loc}(w) = \mathtt{loc}(w') \Rightarrow (w, w') \notin \mathsf{tso}_1^{l'} \end{aligned} \tag{I.H.}$$

From the definition of tso_1^l, we know either i) $\mathsf{tso}_1^l{=}\mathsf{tso}_1^a$; or ii) $\mathsf{tso}_1^l{=}\left(\mathsf{tso}_1^a \cup \{(r_1, w_l)\}\right)^+$ where $(w, r_1) \in \mathsf{rf}$, $(r_1, w_l) \notin \mathsf{tso}_1^a \cup (\mathsf{tso}_1^a)^{-1}$ and $(w, w_l) \in \mathsf{tso}$. In case (i) the result follows from (I.H.).

In case (ii) we proceed by contradiction. Let us assume there exists w_c, w_c', r_c such that $(w_c, r_c) \in \mathsf{rf}$, $(w_c', r_c) \in \mathsf{tso}_1^l \cup \mathsf{po} \wedge \mathtt{loc}(w_c) = \mathtt{loc}(w_c')$ and $(w_c, w_c') \in \mathsf{tso}_1^l$. As $(w_c, w_c') \in \mathsf{tso}_1^l$ and tso_1^l is a strict partial order, we know that $w_c \neq w_c'$. On the other hand, from the definition of as tso_1^l and since $(w_c, w_c') \in \mathsf{tso}_1^l$, we know $(w_c, w_c') \in \mathsf{tso}_1^a$. Consequently, from (I.H.) we know that $(w_c', r_c) \notin \mathsf{tso}_1^a \cup \mathsf{po}$. As such, form the definition of tso_1^l we know that $w_c' \xrightarrow{\mathsf{tso}_1^a} r_1 \xrightarrow{\mathsf{tso}_1^l} w_l \xrightarrow{\mathsf{tso}_1^a} r_c$. However, as tso_1^a is strict and is total on writes, we know that either a) $(w_l, w_c') \in \mathsf{tso}_1^a$; or b) $(w_c', w_l) \in \mathsf{tso}_1^a$. In case (ii.a) we then have $w_l \xrightarrow{\mathsf{tso}_1^a} w_c' \xrightarrow{\mathsf{tso}_1^a} r_1$, contradicting the assumption that $(r_1, w_l) \notin \mathsf{tso}_1^a \cup (\mathsf{tso}_1^a)^{-1}$. In case (ii.b) we have $w_c' \xrightarrow{\mathsf{tso}_1^a} w_l \xrightarrow{\mathsf{tso}_1^a} r_c$, i.e. $(w_c', r_c) \in \mathsf{tso}_1^a$. This however contradicts the result above that $(w_c', r_c) \notin \mathsf{tso}_1^a \cup \mathsf{po}$.

Inductive case $j = b{+}1$ **and** $l = 0$ **with** $1 \leq b < q{-}1$

$$\begin{array}{c}\forall j' \in \{1 \cdots b\}.\ \forall l' \in \{1 \cdots s\}.\ \forall w, r.\ \forall w' \in W \cup U.\\ (w, r) \in \mathsf{rf} \wedge (w', r) \in \mathsf{tso}_{j'}^{l'} \Rightarrow (w, w') \notin \mathsf{tso}_{j'}^{l'}\end{array} \tag{I.H.}$$

As $\mathsf{tso}_j^0 \triangleq \mathsf{tso}_b^s$, the desired result holds immediately from (I.H.).

Inductive case $j = b{+}1$ **and** $l = a{+}1$ **with** $1 \leq b < q{-}1$ **and** $0 \leq a < s$

$$\begin{array}{c}\forall l' \in \{1 \cdots a\}.\ \forall w, r.\ \forall w' \in W \cup U.\\ (w, r) \in \mathsf{rf} \wedge (w', r) \in \mathsf{tso}_j^{l'} \Rightarrow (w, w') \notin \mathsf{tso}_j^{l'}\end{array} \tag{I.H.}$$

From the definition of tso_j^l, we know either i) $\mathsf{tso}_j^l = \mathsf{tso}_j^a$; or ii) $\mathsf{tso}_j^l = \left(\mathsf{tso}_j^a \cup \{(r_j, w_l)\}\right)^+$ when $(w, r_j) \in \mathsf{rf}$, $(r_j, w_l) \notin \mathsf{tso}_j^a \cup (\mathsf{tso}_j^a)^{-1}$ and $(w, w_l) \in \mathsf{tso}$. In case (i) the result follows from (I.H.).

In case (ii), we proceed by contradiction. Let us assume there exists w_c, w_c', r_c such that $(w_c, r_c) \in \mathsf{rf}$, $(w_c', r_c) \in \mathsf{tso}_j^l \cup \mathsf{po} \wedge \mathtt{loc}(w_c) = \mathtt{loc}(w_c')$ and $(w_c, w_c') \in \mathsf{tso}_j^l$. As $(w_c, w_c') \in \mathsf{tso}_j^l$ and tso_j^l is a strict partial order, we know that $w_c \neq w_c'$. On the other hand, from the definition of as tso_1^l and since $(w_c, w_c') \in \mathsf{tso}_j^l$, we know $(w_c, w_c') \in \mathsf{tso}_1^a$. Consequently, from (I.H.) we know that $(w_c', r_c) \notin \mathsf{tso}_j^a \cup \mathsf{po}$. As such, form the definition of tso_j^l we know that $w_c' \xrightarrow{\mathsf{tso}_j^a} r_j \xrightarrow{\mathsf{tso}_j^l} w_l \xrightarrow{\mathsf{tso}_j^a} r_c$. However, as tso_j^a is strict and is total on writes, we know that either a) $(w_l, w_c') \in \mathsf{tso}_j^a$; or b) $(w_c', w_l) \in \mathsf{tso}_j^a$. In case (ii.a) we then have $w_l \xrightarrow{\mathsf{tso}_j^a} w_c' \xrightarrow{\mathsf{tso}_j^a} r_j$, contradicting the assumption that $(r_j, w_l) \notin \mathsf{tso}_j^a \cup (\mathsf{tso}_j^a)^{-1}$. In case (ii.b) we have $w_c' \xrightarrow{\mathsf{tso}_j^a} w_l \xrightarrow{\mathsf{tso}_j^a} r_c$, i.e. $(w_c', r_c) \in \mathsf{tso}_j^a$. This however contradicts the result above that $(w_c', r_c) \notin \mathsf{tso}_j^a \cup \mathsf{po}$. □

Let tso_t denote an extension of tso_q^s to a strict total order on E. Once again, we demonstrate that:

$$\forall w, r.\ \forall w' \in W \cup U.\ (w, r) \in \mathsf{rf} \wedge (w', r) \in \mathsf{tso}_t \wedge \mathtt{loc}(w) = \mathtt{loc}(w') \Rightarrow (w, w') \notin \mathsf{tso}_t \tag{RF}$$

Pick arbitrary w, w', r such that $(w, r) \in \mathsf{rf} \wedge \mathtt{loc}(w) = \mathtt{loc}(w')$ and $(w', r) \in \mathsf{tso}_t$. There are two cases to consider: 1) $(w', r) \in \mathsf{tso}_q^s$; or 2) $(w', r) \in \mathsf{tso}_t \setminus \mathsf{tso}_q^s$. In case (1) the result holds from (RFJ) established above. In case (2), as $(w', r) \in \mathsf{tso}_t \setminus \mathsf{tso}_q^s$ and tso_t is a strict total extension of tso_q^s, we know that $(r, w'), (w', r) \notin \mathsf{tso}_q^s$. As such, from the definition of tso_q^s we know that $(w, w') \notin \mathsf{tso}_q^s$. As tso_q^s is total on writes, we then know that $(w', w) \in \mathsf{tso}_q^s \subseteq \mathsf{tso}_t$. As tso_t is a strict total order,

we have $(w, w') \notin \mathsf{tso}_t$, as required. □

Let $\mathcal{D} \triangleq \left\{e \in FO \cup FL \cup SF \;\middle|\; \nexists r \in R.(r, e) \in G.\mathrm{po} \wedge (e, r) \in \mathsf{tso}_t\right\}$ and $\mathcal{P} \triangleq (FO \cup FL \cup SF) \setminus \mathcal{D}$. Let $e_1, \cdots, e_n$ be an enumeration of $G_i.E \backslash I$ according to tso_t and $\pi^0 = \lambda_1. \cdots .\lambda_n$, where $\lambda_k = \mathsf{genBL}(e_k, G_i)$ for $k \in \{1, \cdots, n\}$ and:

$$\mathsf{genBL}(e, G) \triangleq \begin{cases} \mathsf{B}\langle e\rangle & \text{if } e \in \mathcal{D} \cup W \\ \mathsf{PFO}\langle e\rangle & \text{if } e \in FO \cap \mathcal{P} \\ \mathsf{PFL}\langle e\rangle & \text{if } e \in FL \cap \mathcal{P} \\ \mathsf{PSF}\langle e\rangle & \text{if } e \in SF \cap \mathcal{P} \\ \mathsf{genL}(e, G) & \text{otherwise} \end{cases}$$

$$\mathsf{genL}(e, G) \triangleq \begin{cases} \mathsf{R}\langle e, w\rangle & \text{if } e \in R \wedge (w, e) \in \mathsf{rf} \\ \mathsf{U}\langle e, w\rangle & \text{if } e \in U \wedge (w, e) \in \mathsf{rf} \\ \mathsf{MF}\langle e\rangle & \text{if } e \in MF \\ \mathsf{W}\langle e\rangle & \text{if } e \in W \\ \mathsf{FO}\langle e\rangle & \text{if } e \in FO \cap \mathcal{D} \\ \mathsf{FL}\langle e\rangle & \text{if } e \in FL \cap \mathcal{D} \\ \mathsf{SF}\langle e\rangle & \text{if } e \in SF \cap \mathcal{D} \\ \mathsf{J}\langle e\rangle & \text{otherwise} \end{cases}$$

Let $d_1, \cdots d_m$ denote an enumeration of $\mathcal{D} \cup W$ that respects po^{-1}. For each $j \in \{1 \cdots m\}$, let $A_j \triangleq \left\{e \;\middle|\; (d_j, e) \in \mathrm{po}\right\}$ and $\pi^j = \mathsf{addD}(\pi^{j-1}, d_j, A_j)$, where:

$$\mathsf{addD}(\pi, d, A) \triangleq \begin{cases} \mathsf{genL}(d, G_i).\pi & \text{if } \exists e, \pi'.\ e \in A \wedge \pi = \mathsf{genL}(e, G_i).\pi' \\ \mathsf{genL}(d, G_i).\pi & \text{else if } \exists \pi'.\ \pi = \mathsf{B}\langle d\rangle.\pi' \\ \lambda.\mathsf{addD}(\pi', d, A) & \text{else if } \exists \lambda, \pi'.\ \pi = \lambda.\pi' \\ \text{undefined} & \text{otherwise} \end{cases}$$

Note that for each $j \in \{1 \cdots m\}$, π^j is always defined as $\mathsf{B}\langle d_j\rangle \in \pi^0$ and thus $\mathsf{B}\langle d_j\rangle \in \pi^j$.

Let $c_{m+1}, \cdots c_k$ denote an enumeration of $\mathcal{P}$ that respects po. For each $j \in \{m{+}1 \cdots k\}$, let $B_j \triangleq \left\{e \;\middle|\; (e, c_j) \in \mathrm{po}\right\}$ and $\pi^j = \mathsf{addC}(\pi^{j-1}, c_j, B_j)$, where:

$$\mathsf{addC}(\pi, c, B) \triangleq \begin{cases} \pi.\mathsf{J}\langle c\rangle & \text{if } \exists e, \pi'.\ e \in B \wedge \pi = \pi'.\mathsf{genL}(e, G_i) \\ \pi.\mathsf{J}\langle c\rangle & \text{else if } \exists \pi'.\ \pi = \pi'.\mathsf{genBL}(c, G_i) \\ \mathsf{addC}(\pi', c, B).\lambda & \text{else if } \exists \lambda, \pi'.\ \pi = \pi'.\lambda \\ \text{undefined} & \text{otherwise} \end{cases}$$

Note that for each $j \in \{m{+}1 \cdots k\}$, π^j is always defined as $\mathsf{genBL}(c_j, G_i) \in \pi^0$ and thus $\mathsf{B}\langle c_j\rangle \in \pi^j$.

Let $a_{k+1}, \cdots, a_o$ denote an enumeration of $G_i.D$ according to nvo. Note that as G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent and thus $dom(G_i.\mathsf{nvo}; [G_i.P]) \subseteq G_i.P$, we know there exists p such that $a_{k+1}, \cdots, a_p \in G_i.P$ and $a_{p+1}, \cdots, a_o \in G_i.(D \setminus P)$.

We define $\pi_i \triangleq \pi^k.\lambda_1. \cdots .\lambda_p$ and $\pi'_i \triangleq \lambda_{p+1}. \cdots .\lambda_o$, where $\lambda_j \triangleq \mathsf{PB}\langle a_j\rangle$ for $j \in \{k{+}1, \cdots, o\}$.

Note that it is straightforward to show that for all e, e':

$$\begin{aligned} (e, e') \in G_i.\mathrm{po} &\Leftrightarrow \mathsf{genL}(e, G_i) \prec_{\pi_i.\pi'_i} \mathsf{genL}(e', G_i) \wedge \mathtt{tid}(e){=}\mathtt{tid}(e') \\ (e, e') \in \mathsf{tso}_t &\Leftrightarrow \mathsf{genBL}(e, G_i) \prec_{\pi_i.\pi'_i} \mathsf{genBL}(e', G_i) \\ (e, e') \in G_i.\mathsf{nvo} &\Leftrightarrow \mathsf{PB}\langle e\rangle \prec_{\pi_i.\pi'_i} \mathsf{PB}\langle e\rangle \end{aligned} \tag{25}$$

Moreover, from the definitions of π_i, π_i we know $\mathsf{norm}(\pi_i.\pi_i')$ holds. Let G_i'=$(I, P, E, \mathsf{po}, \mathsf{rf}, \mathsf{tso}_t, \mathsf{nvo})$. Note that $G_i \prec G_i'$; and since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent, from the definition of G_i' and (RF) above, we also know G_i' is $\mathrm{Px86}_{\mathrm{man}}$-consistent. We next show that $\mathsf{wfp}(\pi_i.\pi_i', \mathcal{H}_i)$ and $\mathsf{complete}(\pi_i.\pi_i')$ hold. As such, from the definition of $\mathtt{getG}(.,.)$ and G_i' we have $\mathtt{getG}(\mathcal{H}_i, \pi_i, \pi_i')$=$G_i'$, as required.

Goal: $\mathsf{wfp}(\pi_i.\pi_i', \mathcal{H}_i)$
Let $\pi = \pi_i.\pi_i'$. We are then required to show that for all $\lambda, \pi_1, \pi_2, e, r, e_1, e_2$:

$$\mathsf{nodups}(\pi.\pi''.\pi''') \tag{26}$$

$$\pi{=}\pi_2.\mathsf{R}\langle r, e\rangle.\pi_1 \vee \pi{=}\pi_2.\mathsf{U}\langle r, e\rangle.\pi_1 \Rightarrow \mathsf{wfrd}(r, e, \pi_1, \pi'') \tag{27}$$

$$\mathsf{B}\langle e\rangle \in \pi \Rightarrow$$
$$\quad \mathsf{W}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \vee \mathsf{SF}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \vee \mathsf{FO}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \vee \mathsf{FL}\langle e\rangle \prec_\pi \mathsf{B}\langle e\rangle \tag{28}$$

$$\mathsf{PB}\langle e\rangle \in \pi \Rightarrow$$
$$\quad \mathsf{B}\langle e\rangle \prec_\pi \mathsf{PB}\langle e\rangle \vee \mathsf{U}\langle e, -\rangle \prec_\pi \mathsf{PB}\langle e\rangle \vee \mathsf{J}\langle e\rangle \prec_\pi \mathsf{PB}\langle e\rangle \tag{29}$$

$$\mathsf{J}\langle e\rangle \in \pi \Rightarrow \mathsf{PFO}\langle e\rangle \prec_\pi \mathsf{J}\langle e\rangle \vee \mathsf{PFL}\langle e\rangle \prec_\pi \mathsf{J}\langle e\rangle \vee \mathsf{PSF}\langle e\rangle \prec_\pi \mathsf{J}\langle e\rangle \tag{30}$$

$$\mathsf{D}\langle e\rangle \in \pi \Rightarrow \mathsf{PFO}\langle e\rangle \prec_\pi \mathsf{D}\langle e\rangle \vee \mathsf{PFL}\langle e\rangle \prec_\pi \mathsf{D}\langle e\rangle \vee \mathsf{PSF}\langle e\rangle \prec_\pi \mathsf{D}\langle e\rangle \tag{31}$$

$$\mathsf{J}\langle e\rangle \notin \pi \vee \mathsf{D}\langle e\rangle \notin \pi \tag{32}$$

$$\mathsf{FO}\langle e\rangle \notin \pi \vee \mathsf{PFO}\langle e\rangle \notin \pi \tag{33}$$

$$\mathsf{FL}\langle e\rangle \notin \pi \vee \mathsf{PFL}\langle e\rangle \notin \pi \tag{34}$$

$$\mathsf{SF}\langle e\rangle \notin \pi \vee \mathsf{PSF}\langle e\rangle \notin \pi \tag{35}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \tag{36}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \tag{37}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \tag{38}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \tag{39}$$

$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle{\prec_\pi}\mathsf{MF}\langle e_2\rangle \tag{40}$$

$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle{\prec_\pi}\mathsf{MF}\langle e_2\rangle \tag{41}$$

$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle{\prec_\pi}\mathsf{MF}\langle e_2\rangle \tag{42}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{43}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{44}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{45}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{46}$$

$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle{\prec_\pi}\mathsf{SF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle{\prec_\pi}\mathsf{SF}\langle e_2\rangle \tag{47}$$

$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle{\prec_\pi}\mathsf{SF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle{\prec_\pi}\mathsf{SF}\langle e_2\rangle \tag{48}$$

$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle{\prec_\pi}\mathsf{SF}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle{\prec_\pi}\mathsf{SF}\langle e_2\rangle \tag{49}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{50}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{51}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{52}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{53}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \tag{54}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \tag{55}$$

$$\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \tag{56}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \tag{57}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \tag{58}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \tag{59}$$

$$e_1 \in \mathit{FO} \cup \mathit{FL} \cup \mathit{SF} \wedge e_2 \in \mathit{SF} \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \vee \mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \vee \mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \tag{60}$$

$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \tag{61}$$

$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{62}$$

$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \tag{63}$$

$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \tag{64}$$

$$e_1 \in \mathit{SF} \wedge e_2 \in \mathit{FO} \cup \mathit{FL} \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \vee \mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \tag{65}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1) = \mathtt{tid}(e_2) \wedge \mathsf{B}\langle e_2\rangle \in \pi \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{66}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1) = \mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{67}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{68}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \tag{69}$$

$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow$$
$$\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \tag{70}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{71}$$

$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow$$
$$\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{72}$$

$$e_1 \in \mathit{FO} \wedge e_2 \in \mathit{FL} \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \tag{73}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{74}$$

$$\mathsf{FO}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \tag{75}$$

$$\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow$$
$$\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \tag{76}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{77}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \tag{78}$$

$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \Rightarrow$$
$$\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FO}\langle e_2\rangle \tag{79}$$

$$e_1 \in \mathit{FL} \wedge e_2 \in \mathit{FO} \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow$$
$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \tag{80}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{81}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{82}$$

$$\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \tag{83}$$

$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{W}\langle e_2\rangle \tag{84}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{85}$$

$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{U}\langle e_2, e\rangle \tag{86}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle \tag{87}$$

$$\mathsf{FL}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \tag{88}$$

$$\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \Rightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \prec_\pi \mathsf{FL}\langle e_2\rangle \tag{89}$$

$$\begin{aligned} &e_1, e_2 \in FL \wedge \mathtt{tid}(e_1){=}\mathtt{tid}(e_2) \wedge \mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi \Rightarrow \\ &\quad \mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PFL}\langle e_2\rangle \Leftrightarrow \mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle \end{aligned} \tag{90}$$

$$\begin{aligned} &e_1{,}e_2 \in WU \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle{,}\mathsf{U}\langle e_1, -\rangle\} \wedge \lambda_2 \in \{\mathsf{B}\langle e_2\rangle{,}\mathsf{U}\langle e_2, -\rangle\} \wedge \lambda_1 \prec_\pi \lambda_2 \wedge \mathtt{loc}(e_1){=}\mathtt{loc}(e_2) \\ &\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \end{aligned} \tag{91}$$

$$\begin{aligned} &e_1 \in WU \wedge e_2 \in FO \cup FL \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\} \wedge \lambda_1 \prec_\pi \mathsf{B}\langle e_2\rangle \\ &\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \end{aligned} \tag{92}$$

$$\begin{aligned} &e_1 \in WU \wedge e_2 \in FO \cup FL \wedge \mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\} \\ &\wedge \ \lambda_2 \in \{\mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\} \wedge \lambda_1 \prec_\pi \lambda_2 \\ &\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \vee \mathsf{D}\langle e_2\rangle \in \pi \end{aligned} \tag{93}$$

$$\begin{aligned} &e_1 \in FO \cup FL \wedge e_2 \in D \wedge \lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle\} \\ &\wedge \lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, e\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\} \wedge \lambda_1 \prec_\pi \lambda_2 \\ &\quad \Rightarrow \mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle \vee \mathsf{D}\langle e_1\rangle \in \pi \vee \mathsf{D}\langle e_2\rangle \in \pi \end{aligned} \tag{94}$$

where $\pi'' = \pi_1.\cdots.\pi_{k-1}$ and $\pi''' = \pi'_1.\cdots.\pi'_{k-1}$.

The proof of parts (26) and (28)-(35) follow immediately from the construction of $\pi_i.\pi'_i$.

For part (27), pick arbitrary π_1, π_2, r, e such that $\pi{=}\pi_1.\mathsf{R}\langle r, e\rangle.\pi_2$ or $\pi{=}\pi_1.\mathsf{U}\langle r, e\rangle.\pi_2$. From the construction of π we then know $(e, r) \in \mathsf{rf}$. There are two cases to consider: 1) $e \in E \setminus I$; 2) $e \in I$.

In case (1), as G_i is $\mathrm{Px86}_{\mathrm{man}}$-valid, we know that $(e, r) \in \mathsf{rf} \subseteq \mathsf{tso} \cup \mathsf{po} \subseteq \mathsf{tso}_t \cup \mathsf{po}$. As such, from the construction of π we know there exists π_3 such that $\pi_1 = \pi_3.\lambda.-$ and $\lambda{=}\mathsf{B}\langle e\rangle \vee \lambda{=}\mathsf{U}\langle e, -\rangle \vee (\lambda{=}\mathsf{W}\langle e\rangle \wedge \mathtt{tid}(e) = \mathtt{tid}(r))$. There are two more cases to consider: i) $\lambda{=}\mathsf{B}\langle e\rangle \vee \lambda{=}\mathsf{U}\langle e, -\rangle$; or ii) $\lambda{=}\mathsf{W}\langle e\rangle$.

In case (i) let us assume there exists e' such that $\mathtt{loc}(e'){=}\mathtt{loc}(r)$ and $\mathsf{B}\langle e'\rangle \in \pi_3$ or $\mathsf{U}\langle e', -\rangle \in \pi_3$. From the construction of π we then have $e' \in W$, $(e', r) \in \mathsf{tso}_t$ and $(e, e') \in \mathsf{tso}_t$. This however contradicts our result in (RF) and thus we have $\left\{\mathsf{B}\langle e'\rangle, \mathsf{U}\langle e', -\rangle \in \pi_3 \;\middle|\; \mathtt{loc}(e'){=}\mathtt{loc}(r)\right\} = \emptyset$, as required. Similarly, let us assume there exists e' such that $\mathtt{loc}(e'){=}\mathtt{loc}(r)$, $\mathtt{tid}(e') = \mathtt{tid}(r)$, $\mathsf{W}\langle e'\rangle \in \pi_3$ and $\mathsf{B}\langle e'\rangle \notin \pi_3$. From the construction of π we then have $e' \in W$, $(e', r) \in \mathsf{po}$ and $(e, e') \in \mathsf{po} \cap W \times W \subseteq \mathsf{tso}_t$. This however contradicts our result in (RF) and thus we have $\left\{e' \;\middle|\; \begin{array}{l}\mathsf{W}\langle e'\rangle \in \pi_3 \wedge \mathsf{B}\langle e'\rangle \notin \pi_3 \\ \mathtt{loc}(e'){=}\mathtt{loc}(r) \wedge \mathtt{tid}(e') = \mathtt{tid}(r)\end{array}\right\} = \emptyset$, as required.

Similarly, in case (ii) we know that either $\mathsf{B}\langle e\rangle \in \pi_3$ or $\mathsf{B}\langle e\rangle \notin \pi_3$. In the former case the desired result follows from the proof of case (i). In the latter case, let us assume there exists e' such that $\mathtt{loc}(e'){=}\mathtt{loc}(r)$, $\mathtt{tid}(e') = \mathtt{tid}(r)$ and $\mathsf{W}\langle e'\rangle \in \pi_3$. From the construction of π we then have $e' \in W \setminus U$, $(e', r) \in \mathsf{po}$ and $(e, e') \in \mathsf{po} \cap W \times W \subseteq \mathsf{tso}_t$. This however contradicts our result in (RF) and thus we have $\left\{\mathsf{W}\langle e'\rangle \in \pi_3 \;\middle|\; \mathtt{loc}(e'){=}\mathtt{loc}(r) \wedge \mathtt{tid}(e') = \mathtt{tid}(r)\right\} = \emptyset$, as required.

In case (2), as G_i is $\mathrm{Px86}_{\mathrm{man}}$-valid, we know either i) $i = 1 \wedge e = \mathit{init}_{\mathtt{loc}(e)}$; or ii) $i > 0 \wedge \exists w.\ w = \max\left(G_{i-1}.\mathsf{nvo}|_{G_{i-1}.P \cap W_{\mathtt{loc}(e)}}\right) \wedge \mathtt{val}_{\mathsf{w}}(w) = \mathtt{val}_{\mathsf{w}}(e)$. Let us now assume there exists e' such that $\mathsf{B}\langle e'\rangle \in \pi_1$ or $\mathsf{U}\langle e', -\rangle \in \pi_1$, and $\mathtt{loc}(e'){=}\mathtt{loc}(r)$. That is, $e' \in W$. From the construction of π we then have $(e', r) \in \mathsf{tso}_t$ and $(e, e') \in \mathsf{tso}_t$. This however contradicts our result in (RF) and thus we have $\left\{\mathsf{B}\langle e'\rangle, \mathsf{U}\langle e', -\rangle \in \pi_1 \;\middle|\; \mathtt{loc}(e'){=}\mathtt{loc}(r)\right\} = \emptyset$. Similarly, let us assume there exists e' such that $\mathtt{loc}(e'){=}\mathtt{loc}(r)$, $\mathtt{tid}(e') = \mathtt{tid}(r)$, $\mathsf{W}\langle e'\rangle \in \pi_1$. That is, $e' \in W \setminus U$. From the construction of π we

then have $(e', r) \in \mathrm{po}$ and $(e, e') \in \mathrm{po} \cap W \times W \subseteq \mathrm{tso}_t$. This however contradicts our result in (RF) and thus we have $\left\{\mathsf{W}\langle e'\rangle \in \pi_1 \;\middle|\; \mathtt{loc}(e')=\mathtt{loc}(r) \wedge \mathtt{tid}(e') = \mathtt{tid}(r)\right\} = \emptyset$. In case (i), as $\mathcal{H}_i = \epsilon$, we know $\pi'' = \epsilon$ and thus we simply have

$$\left\{\mathsf{PB}\langle e'\rangle \in \pi'' \;\middle|\; \mathtt{loc}(e')=\mathtt{loc}(r)\right\} = \emptyset$$

as required.

In case (ii), we then know either:

a) for all $b \in \{1 \cdots i{-}1\}$, $e \in G_b.I$ and $G_b.W_{\mathtt{loc}(e)} \setminus G_b.I = \emptyset$ and thus $e = \mathit{init}_{\mathtt{loc}(e)}$; or

b) there exists $a \in \{1 \cdots i{-}1\}$ such that $e \in G_a.P \setminus I$, $\forall e' \in G_a.W_{\mathtt{loc}(e)}.\ (e', e) \in G_a.\mathrm{nvo}$ and for all $b \in \{a{+}1 \cdots i{-}1\}$, $e \in G_b.I$ and $G_b.W_{\mathtt{loc}(e)} \setminus G_b.I = \emptyset$.

In case (a), let us assume there exists e' such that $\mathsf{PB}\langle e'\rangle \in \pi''$ and $\mathtt{loc}(e') = \mathtt{loc}(r) = \mathtt{loc}(e)$. We then know there exists $b \in \{1 \cdots i{-}1\}$ such that $e \in G_b.W_{\mathtt{loc}(e)} \setminus G_b.I$, leading to a contradiction. As such, we have

$$\left\{\mathsf{PB}\langle e'\rangle \in \pi'' \;\middle|\; \mathtt{loc}(e')=\mathtt{loc}(r)\right\} = \emptyset$$

as required.

In case (b), from the construction of $\pi_1 \cdots \pi_{i-1}$, we know there exists π_3, π_4 such that $\pi_a = \pi_3.\mathsf{PB}\langle e\rangle.\pi_4$, and $\pi'' = \pi_{i-1}.\cdots.\pi_a.\cdots.\pi_1$. Let us assume there exists e' such that $\mathsf{PB}\langle e'\rangle \in \pi_{i-1}.\cdots.\pi_{a+1}$ and $\mathtt{loc}(e') = \mathtt{loc}(r) = \mathtt{loc}(e)$. We then know either there exists $b \in \{i{-}1 \cdots a{+}1\}$ such that $e \in G_b.W_{\mathtt{loc}(e)} \setminus G_b.I$, leading to a contradiction. Similarly, let us assume there exists e' such that $\mathsf{PB}\langle e'\rangle \in \pi_3$ and $\mathtt{loc}(e') = \mathtt{loc}(r) = \mathtt{loc}(e)$. We then know $(e, e') \in G_a.\mathrm{nvo}$, leading to a contradiction. As such, we have $\left\{\mathsf{PB}\langle e'\rangle \in \pi_{i-1}.\cdots.\pi_{a+1}.\pi_3 \;\middle|\; \mathtt{loc}(e')=\mathtt{loc}(r)\right\} = \emptyset$, as required.

For part (36), pick arbitrary e_1, e_2, such that $\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle$ and $\mathtt{tid}(e_1)=\mathtt{tid}(e_2)$. That is, $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in G_i.\mathrm{po}$ and thus since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent, we have $(e_1, e_2) \in G_i.\mathrm{tso} \subseteq G_i.\mathrm{tso}_t$. Consequently, from the construction of π we have $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$, i.e. $\mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle$, as required.

The proofs of parts (37)-(39) are analogous and is thus omitted here.

For part (40), pick arbitrary e_1, e_2, such that $\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle$ and $\mathtt{tid}(e_1)=\mathtt{tid}(e_2)$. That is, $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in G_i.\mathrm{tso}_t$. Since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent and thus $G_i.\mathrm{tso}$ is total on $G_i.E \setminus R$, we also have $(e_1, e_2) \in G_i.\mathrm{tso}$. As $\mathtt{tid}(e_1)=\mathtt{tid}(e_2)$, there are now two cases to consider: 1) $(e_1, e_2) \in G_i.\mathrm{po}$; or 2) $(e_2, e_1) \in G_i.\mathrm{po}$.

In case (1) from (25) we have $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$, i.e. $\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle$, as required. In case (2) since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent, we have $(e_2, e_1) \in G_i.\mathrm{tso}$. Since we also have $(e_1, e_2) \in G_i.\mathrm{tso}$, from the transitivity of $G_i.\mathrm{tso}$ we have $(e_1, e_1) \in G_i.\mathrm{tso}$. This however leads to a contradiction as since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent, we know that $G_i.\mathrm{tso}$ is acyclic.

The proof of parts (41)-(42) are analogous and thus omitted here.

For part (43), pick arbitrary e_1, e_2, such that $\mathsf{W}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle$ and $\mathtt{tid}(e_1)=\mathtt{tid}(e_2)$. That is, $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in G_i.\mathrm{po}$ and thus since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent, we have $(e_1, e_2) \in G_i.\mathrm{tso} \subseteq G_i.\mathrm{tso}_t$. Consequently, from the construction of π we have $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$, i.e. $\mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$, as required.

The proofs of parts (44)-(46), (50)-(53), (66)-(68), (71), (74), (77), (81)-(82), (85) and (87) are analogous and thus omitted here.

For part (47), pick arbitrary e_1, e_2, such that $\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle$ and $\mathtt{tid}(e_1)=\mathtt{tid}(e_2)$. From the construction of π we then know that $\mathsf{SF}\langle e_2\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$. As such, from the transitivity of π we have $\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{B}\langle e_2\rangle$. That is, $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$. As such, from (25) we know

$(e_1, e_2) \in G_i.\mathsf{tso}_t$. As such, since G_i is Px86_{man}-consistent and thus $G_i.\mathsf{tso}$ is total on $G_i.E \setminus R$, we also have $(e_1, e_2) \in G_i.\mathsf{tso}$. As $\mathtt{tid}(e_1)$=$\mathtt{tid}(e_2)$, there are now two cases to consider: 1) $(e_1, e_2) \in G_i.\mathsf{po}$; or 2) $(e_2, e_1) \in G_i.\mathsf{po}$.

In case (1) from (25) we have $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$, i.e. $\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{SF}\langle e_2\rangle$, as required. In case (2) since G_i is Px86_{man}-consistent, we have $(e_2, e_1) \in G_i.\mathsf{tso}$. Since we also have $(e_1, e_2) \in G_i.\mathsf{tso}$, from the transitivity of $G_i.\mathsf{tso}$ we have $(e_1, e_1) \in G_i.\mathsf{tso}$. This however leads to a contradiction as since G_i is Px86_{man}-consistent, we know that $G_i.\mathsf{tso}$ is acyclic.

The proofs of parts (48), (49), (61)-(64), (70), (72), (76), (79), (84), (86) and (89) are analogous and thus omitted here.

For part (54), pick arbitrary e_1, e_2, such that $\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle$ and $\mathtt{tid}(e_1)$=$\mathtt{tid}(e_2)$. From the construction of π we then know that $\mathsf{PFO}\langle e_2\rangle \prec_\pi \mathsf{J}\langle e_2\rangle$. As such, from the transitivity of π we have $\mathsf{SF}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle$. That is, $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in G_i.\mathsf{po}$. As such, since G_i is Px86_{man}-consistent, we have $(e_1, e_2) \in G_i.\mathsf{tso} \subseteq G_i.\mathsf{tso}_t$. Consequently, from the construction of π we have $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$, i.e. $\mathsf{B}\langle e_1\rangle \prec_\pi \mathsf{PFO}\langle e_2\rangle$, as required.

The proofs of parts (55), (56)-(59), (69), (75), (78), (83) and (88) are analogous and thus omitted here.

For part (60), pick arbitrary $e_1, e_2, \lambda_1, \lambda_2$ such that $e_1 \in FO \cup FL \cup SF$, $e_2 \in SF$, $\mathtt{tid}(e_1)$=$\mathtt{tid}(e_2)$ and $\mathsf{J}\langle e_1\rangle, \mathsf{J}\langle e_2\rangle \in \pi$.

For the $\Rightarrow$ direction, let us assume that $\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$ or $\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$ or $\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$. That is, $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in G_i.\mathsf{tso}_t$. Since G_i is Px86_{man}-consistent and thus $G_i.\mathsf{tso}$ is total on $G_i.E \setminus R$, we also have $(e_1, e_2) \in G_i.\mathsf{tso}$. As $\mathtt{tid}(e_1)$=$\mathtt{tid}(e_2)$, there are now two cases to consider: 1) $(e_1, e_2) \in G_i.\mathsf{po}$; or 2) $(e_2, e_1) \in G_i.\mathsf{po}$.

In case (1) from (25) we have $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$, i.e. $\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{MF}\langle e_2\rangle$, as required. In case (2) since G_i is Px86_{man}-consistent, we have $(e_2, e_1) \in G_i.\mathsf{tso}$. Since we also have $(e_1, e_2) \in G_i.\mathsf{tso}$, from the transitivity of $G_i.\mathsf{tso}$ we have $(e_1, e_1) \in G_i.\mathsf{tso}$. This however leads to a contradiction as since G_i is Px86_{man}-consistent, we know that $G_i.\mathsf{tso}$ is acyclic.

For the $\Leftarrow$ direction, let us assume that $\mathsf{J}\langle e_1\rangle \prec_\pi \mathsf{J}\langle e_2\rangle$. That is, $\mathsf{genL}(e_1, G_i) \prec_\pi \mathsf{genL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in G_i.\mathsf{po}$. As $\mathtt{tid}(e_1)$=$\mathtt{tid}(e_2)$ and G_i is Px86_{man}-consistent, we also have $(e_1, e_2) \in G_i.\mathsf{tso} \subseteq \mathsf{tso}_t$. Consequently, from (25) we have $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$, i.e. $\mathsf{PFO}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$ or $\mathsf{PFL}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$ or $\mathsf{PSF}\langle e_1\rangle \prec_\pi \mathsf{PSF}\langle e_2\rangle$, as required.

The proofs of parts (65), (73), (80) and (90) are analogous and thus omitted here.

For part (91), pick arbitrary $e_1, e_2, x, \lambda_1, \lambda_2$ such that $e_1, e_2 \in W \cup U$, $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\}$, $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, -\rangle\}$, $\lambda_1 \prec_\pi \lambda_2$ and $\mathtt{loc}(e_1)$=$\mathtt{loc}(e_2)$=x. That is, $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in \mathsf{tso}_t$. Since G_i is Px86_{man}-consistent and thus $G_i.\mathsf{tso}$ is total on $G_i.E \setminus R$, we also have $(e_1, e_2) \in G_i.\mathsf{tso}$. As G_i is Px86_{man}-consistent and thus $G_i.\mathsf{tso}|_{D_x} \subseteq G_i.\mathsf{nvo}$, we have $(e_1, e_2) \in G_i.\mathsf{nvo}$. As such, from (25) we know $\mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle$, as required.

We prove parts (92) and (93) together. Pick arbitrary $e_1, e_2, X, \lambda_1, \lambda_2$ such that $e_1 \in WU$, $e_2 \in FO \cup FL$, $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{U}\langle e_1, -\rangle\}$, $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$, $\lambda_1 \prec_\pi \lambda_2$ and $\mathtt{loc}(e_1), \mathtt{loc}(e_2) \in X$. That is, $\mathsf{genBL}(e_1, G_i) \prec_\pi \mathsf{genBL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in \mathsf{tso}_t$. Since G_i is Px86_{man}-consistent and thus $G_i.\mathsf{tso}$ is total on $G_i.E \setminus R$, we also have $(e_1, e_2) \in G_i.\mathsf{tso}$. As G_i is Px86_{man}-consistent and thus $G_i.[W_X \cup U_X]; G_i.\mathsf{tso}; G_i.[FO_X \cup FL_X] \subseteq G_i.\mathsf{nvo}$, we have $(e_1, e_2) \in G_i.\mathsf{nvo}$. As such, from (25) we know $\mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle$, as required.

For part (94), pick arbitrary $e_1, e_2, \lambda_1, \lambda_2$ such that $e_1 \in FO \cup FL$, $e_2 \in D$, $\lambda_1 \prec_\pi \lambda_2$, $\lambda_1 \in \{\mathsf{B}\langle e_1\rangle, \mathsf{PFO}\langle e_1\rangle, \mathsf{PFL}\langle e_1\rangle\}$ and $\lambda_2 \in \{\mathsf{B}\langle e_2\rangle, \mathsf{U}\langle e_2, -\rangle, \mathsf{PFO}\langle e_2\rangle, \mathsf{PFL}\langle e_2\rangle\}$. That is, $\mathtt{genBL}(e_1, G_i) \prec_\pi \mathtt{genBL}(e_2, G_i)$. As such, from (25) we know $(e_1, e_2) \in \mathsf{tso}_t$. Since G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent and thus $G_i.\mathsf{tso}$ is total on $G_i.E \setminus R$, we also have $(e_1, e_2) \in G_i.\mathsf{tso}$. As G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent and thus $G_i.[FO \cup FL]; G_i.\mathsf{tso}; G_i.[D] \subseteq G_i.\mathsf{nvo}$, we have $(e_1, e_2) \in G_i.\mathsf{nvo}$. As such, from (25) we know $\mathsf{PB}\langle e_1\rangle \prec_\pi \mathsf{PB}\langle e_2\rangle$, as required.

Goal: $\mathsf{complete}(\pi_i.\pi_i')$
Follows immediately from the construction of $\pi_i.\pi_i'$. □

Definition 9. Given a $\Gamma = (G_1, (\pi_1, \pi_1')).\cdots.(G_n, (\pi_n, \pi_n'))$ and an event path π, let

$$\mathsf{wf}(\Gamma, \pi) \overset{\text{def}}{\Leftrightarrow} \mathsf{wfh}(\mathcal{H}) \wedge \mathsf{wfp}(\pi, \mathcal{H}) \wedge \bigwedge_{i=1}^{n} G_i \prec \mathtt{getG}(\mathcal{H}_i, \pi_i, \pi_i')$$

where $\mathcal{H}_1{=}\epsilon$; $\mathcal{H}_{i+1} = (\pi_1, \pi_1').\cdots.(\pi_i, \pi_i')$ for $i \in \{1 \cdots n\}$; and $\mathcal{H}{=}\mathtt{hist}(\Gamma){=}\mathcal{H}_n$.

Lemma 4. *Let $C = G_1, \cdots, G_n$ denote a $\mathrm{Px86}_{man}$-valid chain. For all $(\pi_1, \pi_1').\cdots.(\pi_n, \pi_n') \in \mathtt{traces}(C)$ and for all $i \in \{1 \cdots n\}$:*

$$\pi_i.\pi_i' = \pi.\pi' \Rightarrow \mathsf{wf}(\Gamma_i, \pi)$$

where $\Gamma_1{=}\epsilon$ and $\Gamma_{j+1}{=}(G_1, (\pi_1, \pi_1')).\cdots.(G_j, (\pi_j, \pi_j'))$ for $j \in \{1 \cdots i{-}1\}$.

Proof. Pick an arbitrary $\mathrm{Px86}_{\mathrm{man}}$-valid chain $C = G_1, \cdots, G_n$ and $(\pi_1, \pi_1').\cdots.(\pi_n, \pi_n') \in \mathtt{traces}(C)$. We proceed by induction on i.

Base case $i = 1$
Pick arbitrary $(\pi_1, \pi_1') \in \mathtt{traces}(G_1)$ and π, π' such that $\pi_1.\pi_1' = \pi.\pi'$. We are then required to show $\mathsf{wf}(\Gamma_1, \pi)$, where $\Gamma_1 = \epsilon$. It thus suffices to show:

$$\mathsf{wfh}(\epsilon) \wedge \mathsf{wfp}(\pi, \epsilon) \wedge G_1 \prec \mathtt{getG}(\epsilon, \pi_1, \pi_1')$$

The first conjunct follows trivially from the definition of $\mathsf{wfh}(\epsilon)$. The third conjunct follows immediately from the fact that $(\pi_1, \pi_1') \in \mathtt{traces}(G_1)$ and the definition of $\mathtt{traces}(.)$. Consequently, from the definition of $\mathtt{getG}(\epsilon, \pi_1, \pi_1')$ we know $\mathsf{wfp}(\pi_1.\pi_1', \epsilon)$ holds implying the result in the second conjunct.

Inductive case $i = j{+}1$

$$\forall k \leq j.\ \forall (\pi_1, \pi_1').\cdots.(\pi_k, \pi_k') \in \mathtt{traces}(G_k).\ \forall \pi^1, \pi^2.\ \pi_k.\pi_k' = \pi^1.\pi^2 \Rightarrow \mathsf{wf}(\Gamma_k', \pi^1) \qquad \text{(I.H.)}$$

where $\Gamma_1'{=}\epsilon$ and $\Gamma_{l+1}'{=}(G_1, (\pi_1, \pi_1')).\cdots.(G_l, (\pi_l, \pi_l'))$ for $l \in \{1 \cdots j{-}1\}$.

Pick arbitrary $(\pi_1, \pi_1').\cdots.(\pi_i, \pi_i') \in \mathtt{traces}(G_i)$ and π, π' such that $\pi_i.\pi_i' = \pi.\pi'$. We are then required to show $\mathsf{wf}(\Gamma_i, \pi)$. It thus suffices to show:

$$\mathsf{wfh}(\mathtt{hist}(\Gamma_i)) \wedge \mathsf{wfp}(\pi, \mathtt{hist}(\Gamma_i)) \wedge \bigwedge_{k=1}^{j} G_k \prec \mathtt{getG}(\Gamma_k, \pi_k, \pi_k')$$

where $\Gamma_1{=}\epsilon$ and $\Gamma_{l+1}{=}(G_1, (\pi_1, \pi_1')).\cdots.(G_l, (\pi_l, \pi_l'))$ for $l \in \{1 \cdots j{-}1\}$.

The last conjunct follows from the definition of $\mathtt{traces}(.)$ and the fact that $(\pi_1, \pi_1').\cdots.(\pi_i, \pi_i') \in \mathtt{traces}(G_i)$. Similarly, as $(\pi_1, \pi_1').\cdots.(\pi_i, \pi_i') \in \mathtt{traces}(G_i)$, from the definition of $\mathtt{traces}(.)$ we know $G_i \prec \mathtt{getG}(\Gamma_i, \pi_i, \pi_i')$ and thus $\mathsf{wfp}(\pi_i.\pi_i', \mathtt{hist}(\Gamma_i))$ holds implying the second conjunct.

For the first conjunct, we have $\texttt{hist}(\Gamma_i)$=$\texttt{hist}(\Gamma_j)$.(π_i, π_i'). As $(\pi_1, \pi_1').\cdots.(\pi_i, \pi_i') \in \texttt{traces}(G_i)$, from the definition of traces(.). we know $G_i \prec \texttt{getG}(\Gamma_i, \pi_i, \pi_i')$ and thus $\texttt{wfp}(\pi_i.\pi_i', \texttt{hist}(\Gamma_i))$ and $\texttt{complete}(\pi_i.\pi_i')$ hold. On the other hand, from (I.H.) we have $\texttt{wfh}(\texttt{hist}(\Gamma_j))$. As such, from the definition of wfh(.) we have $\texttt{wfh}(\Gamma_i)$, as required.

□

Lemma 5. *Let $C = G_1, \cdots, G_n$ denote a Px86$_{man}$-valid chain of $\langle \mathsf{P}, \texttt{rec}\rangle$. For each G_i, let $e_i^1, \cdots, e_i^m$ denote an enumeration of $G_i.E \setminus I$ that respects $G_i.\mathrm{po}$. Then there exists $\mathsf{P}_i^1 \cdots \mathsf{P}_i^m$ such that:*

- $\mathsf{P}_i^{j-1} \ (\xrightarrow{\mathcal{E}\langle\tau\rangle})^* \xrightarrow{\texttt{genL}(e_i^j, G_i)} (\xrightarrow{\mathcal{E}\langle\tau\rangle})^* \ \mathsf{P}_i^j$, *for $i \in \{1\cdots n\}$ and $j \in \{1\cdots m\}$*
- $\mathsf{P}_n^m = \mathsf{P}_{\mathbf{skip}}$

where $\mathsf{P}_1^0 = \mathsf{P}$ and $\mathsf{P}_i^0 = \texttt{rec}(\mathsf{P}, G_{i-1})$ for $i \in \{2\cdots n\}$.

Lemma 6. *Let $C = G_1, \cdots, G_n$ denote a Px86$_{man}$-valid chain of program $\langle \mathsf{P}, \texttt{rec}\rangle$. For all $\theta_1.\cdots.\theta_n \in \texttt{traces}(C)$, and for all $i \in \{1\cdots n\}$:*

(1) if $i < n$ then
$(\mathsf{P}, \texttt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_{i+1}^0, \Gamma_{i+1}, \epsilon$
(2) $(\mathsf{P}, \texttt{rec}) \vdash \mathsf{P}_n^0, \Gamma_n, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, \Gamma_n, \pi_n$

where $\mathsf{P}_1^0 = \mathsf{P}$; $\mathsf{P}_{j+1}^0 = \texttt{rec}(\mathsf{P}, G_j)$; $\Gamma_1 = \epsilon$ and Γ_{j+1}=$(G_1, \theta_1).\cdots.(G_j, \theta_j)$, for $j \in \{1\cdots n{-}1\}$.

Proof. Pick an arbitrary program P and a Px86$_{\text{man}}$-valid chain C=$G_1, \cdots, G_n$ of P. Let $\mathsf{P}_1^0 = \mathsf{P}$ and $\mathsf{P}_j^0 = \texttt{rec}(\mathsf{P}, G_{j-1})$ for $j \in \{2\cdots n\}$. Pick an arbitrary $(\pi_1, \pi_1').\cdots.(\pi_n, \pi_n') \in \texttt{traces}(C)$, and $i \in \{1\cdots n\}$. Let $\Gamma_1 = \epsilon$ and $\Gamma_{j+1} = (G_1, (\pi_1, \pi_1')).\cdots.(G_j, (\pi_j, \pi_j'))$ for $j \in \{1\cdots n{-}1\}$. Let $\mathcal{H}_1 = \epsilon$ and $\mathcal{H}_{j+1} = (\pi_1, \pi_1').\cdots.(\pi_j, \pi_j')$ for $j \in \{1\cdots n{-}1\}$.

Part (1). Assume $i < n$. From the definitions of traces(.) and getG(., ., .) we know π_i respects $G_i.\mathrm{po}$. That is, π_i is of the form: $s_0.\texttt{genL}(e_1, G_i).s_1.\cdots.\texttt{genL}(e_m, G_i).s_m$, where:
i) For each $j \in \{i\cdots m\}$, $s_j = \lambda_{(j,1)}.\cdots.\lambda_{(j,k_j)}$ and each $\lambda_{(j,r)}$ is either of the form $\mathsf{B}\langle{-}\rangle$ or $\mathsf{PB}\langle{-}\rangle$ or $\mathsf{PFO}\langle{-}\rangle$ or $\mathsf{PFL}\langle{-}\rangle$ or $\mathsf{PSF}\langle{-}\rangle$, for $r \in \{1\cdots k_j\}$;
ii) $s_0 = \lambda_{(1,1)}.\cdots.\lambda_{(1,k_1)}$ and each $\lambda_{(1,r)}$ is either of the form $\mathsf{PFO}\langle{-}\rangle$ or $\mathsf{PFL}\langle{-}\rangle$ or $\mathsf{PSF}\langle{-}\rangle$, for $r \in \{1\cdots k_1\}$; and
iii) $e_1\cdots e_m$ is an enumeration of $G_i.E$ respecting $G_i.\mathrm{po}$ (if $(e, e') \in G_i.\mathrm{po}$ then $\texttt{genL}(e, G_i) \prec_{\pi_i} \texttt{genL}(e', G_i)$).

Moreover, from the definition of traces(.) we know $G_i \prec \texttt{getG}(\mathcal{H}_i, \pi_i, \pi_i')$. Additionally, from Lemma 4 we know:

$$\forall \lambda, p, q.\ \pi_i.\pi_i' = p.\lambda.q \Rightarrow \texttt{fresh}(\lambda, p.q) \wedge \texttt{fresh}(\lambda, \Gamma_i) \tag{95}$$

From (G-Prop) we thus have $(\mathsf{P}, \texttt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_i^0, \Gamma_i, s_0$. There are now two cases to consider: 1) $m = 0$; or 2) $m > 0$.

In case (1), we then have $\pi_i = s_0$. Since $\pi_i' \in \textsc{PPath}$ (and thus each label in π_i' is of the form $\mathsf{B}\langle{-}\rangle$, $\mathsf{PB}\langle{-}\rangle$ or $\mathsf{D}\langle{-}\rangle$), and from the definition of $\texttt{traces}(C)$ we know that $\texttt{norm}(\pi_i.\pi_i')$ holds (i.e. $\pi_i.\pi_i'$ contains no $\mathsf{D}\langle{-}\rangle$ entries), we know that each label in π_i' is of the form $\mathsf{B}\langle{-}\rangle$ or $\mathsf{PB}\langle{-}\rangle$. As such, since each label in π_i is of the form $\mathsf{PFO}\langle{-}\rangle$ or $\mathsf{PFL}\langle{-}\rangle$ or $\mathsf{PSF}\langle{-}\rangle$, and from the definition of getG(., ., .) in $\texttt{traces}(C)$ we know that $\texttt{wfp}(\mathcal{H}_i, \pi_i.\pi_i')$ and $\texttt{complete}(\pi_i.\pi_i')$holds, we then know $s_0 = \pi_i = \pi_i' = \epsilon$. As such, we have $(\mathsf{P}, \texttt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_i^0, \Gamma_i, \epsilon$. Moreover, since $\pi_i' = \epsilon$ then $\texttt{comp}(\pi_i, \pi_i')$ holds. As such from (G-Crash) we have $(\mathsf{P}, \texttt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_{i+1}^0, \Gamma_{i+1}, \epsilon$, as required.

In case (2) from Lemma 5 we know there exists $\mathsf{P}_i^1 \cdots \mathsf{P}_i^m$ such that for $j \in \{1 \cdots m\}$:

$$(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^{j-1} \ (\xrightarrow{\mathcal{E}\langle\tau\rangle})^* \xrightarrow{\mathsf{genL}(e_i^j, G_i)} (\xrightarrow{\mathcal{E}\langle\tau\rangle})^* \ \mathsf{P}_i^j \tag{96}$$

For $j \in \{1 \cdots m\}$, from (96) we know there exist $\mathsf{P}'_j, \mathsf{P}''_j$ such that $(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^{j-1}(\xrightarrow{\mathcal{E}\langle\tau\rangle})^*\mathsf{P}'_j \xrightarrow{\mathsf{genL}(e_i^j, G_i)} \mathsf{P}''_j(\xrightarrow{\mathcal{E}\langle\tau\rangle})^* \ \mathsf{P}_i^j$. Let $p_0 = s_0$ and $p_j = s_0.\mathsf{genL}(e_1, G_i).s_1. \cdots .s_j.\mathsf{genL}(e_j, G_i).s_j$, for $j \in \{1 \cdots m\}$. As such, from (G-SilentP), (G-Step), (G-Prop), and (95) we then have:

$$\begin{array}{ll} & (\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^{j-1}, \Gamma_i, p_{j-1} \\ \Rightarrow^* & \mathsf{P}'_j, \Gamma_i, p_{j-1} \\ \Rightarrow & \mathsf{P}''_j, \Gamma_i, \mathsf{genL}(e_j, G_i).p_{j-1} \\ \Rightarrow^* & \mathsf{P}_i^j, \Gamma_i, \mathsf{genL}(e_j, G_i).p_{j-1} \\ \Rightarrow & \mathsf{P}_i^j, \Gamma_i, p_j \end{array}$$

Consequently, we have

$$(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_i^0, \Gamma_i, p_0 \Rightarrow^* \mathsf{P}_i^1, \Gamma_i, p_1 \Rightarrow^* \cdots \Rightarrow^* \mathsf{P}_i^m, \Gamma_i, p_m$$

That is, we have

$$(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_i^m, \Gamma_i, \pi_i$$

On the other hand from Lemma 4 and the definition of $\mathtt{getG}(.,.,.)$ we know that $\mathtt{comp}(\pi, \pi')$ holds. As such, since $G_i \prec \mathtt{getG}(\mathcal{H}_i, \pi_i, \pi'_i)$ and G_i is $\mathrm{Px86}_{\mathrm{man}}$-consistent, from (G-Crash) we have

$$(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^m, \Gamma_i, \pi_i \Rightarrow^* \mathsf{P}_{i+1}^m], \Gamma_{i+1}, \epsilon$$

That is, we have $(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_i^0, \Gamma_i, \epsilon \Rightarrow^* \mathsf{P}_{i+1}^m, \Gamma_{i+1}, \epsilon$, as required.

Part (2). From $\mathtt{traces}(G_n)$ we know π_n respects $G_n.\mathrm{po}$. That is, π_n is of form: $s_0.\mathsf{genL}(e_1, G_n).s_1 . \cdots .\mathsf{genL}(e_m, G_n).s_m$, where:
i) For each $j \in \{i \cdots m\}$, $s_j = \lambda_{(j,1)}. \cdots .\lambda_{(j,k_j)}$ and each $\lambda_{(j,r)}$ is either of the form $\mathsf{B}\langle-\rangle$ or $\mathsf{PB}\langle-\rangle$ or $\mathsf{PFO}\langle-\rangle$ or $\mathsf{PFL}\langle-\rangle$ or $\mathsf{PSF}\langle-\rangle$, for $r \in \{1 \cdots k_j\}$;
ii) $s_0 = \lambda_{(1,1)}. \cdots .\lambda_{(1,k_1)}$ and each $\lambda_{(1,r)}$ is either of the form $\mathsf{PFO}\langle-\rangle$ or $\mathsf{PFL}\langle-\rangle$ or $\mathsf{PSF}\langle-\rangle$, for $r \in \{1 \cdots k_1\}$; and
iii) $e_1 \cdots e_m$ is an enumeration of $G_n.E$ respecting $G_n.\mathrm{po}$ (if $(e, e') \in G_n.\mathrm{po}$ then $\mathsf{genL}(e, G_n) \prec_{\pi_n} \mathsf{genL}(e', G_n)$).

Moreover, since $(\pi_n, \pi'_n) \in \mathtt{traces}(G_n)$, from the definition of $\mathtt{traces}(.)$ we know that $G_n \prec \mathtt{getG}(\mathcal{H}_n, \pi_n, \pi'_n)$. Additionally, from Lemma 4 we know:

$$\pi'_n = \epsilon \wedge \forall \lambda, p, q.\ \pi_n.\pi'_n = p.\lambda.q \Rightarrow \mathsf{fresh}(\lambda, p.q) \wedge \mathsf{fresh}(\lambda, \Gamma_n) \tag{97}$$

From (G-Prop) we thus have $(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_n^0, \Gamma_n, \epsilon \Rightarrow^* \mathsf{P}_n^0, \Gamma_n, s_0$. There are now two cases to consider: 1) $m = 0$; or 2) $m > 0$.

In case (1), $\pi_n = s_0$ and from Lemma 5 we also know $\mathsf{P}_n^0 = \mathsf{P}_{\mathbf{skip}}$. In steps similar to those above we can then establish that $s_0 = \pi_n = \pi'_n = \epsilon$. As such, we trivially have $(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_n^0, \Gamma_n, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, \Gamma_n, \epsilon$, as required.

In case (2), in similar steps to that of the proof of part (1) we have: $(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}_n^0, \Gamma_n, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, \Gamma_n, \pi_n$ as required.

□

Corollary 1. *Let $C = G_1, \cdots, G_n$ denote a Px86$_{man}$-valid chain of program* P. *Then, there exists $\theta_1.\cdots.\theta_n \in \mathtt{traces}(C)$, with $\theta_n = (\pi_n, -)$ such that:*

$$(\mathsf{P}, \mathtt{rec}) \vdash \mathsf{P}, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, (G_1, \theta_1).\cdots.(G_{n-1}, \theta_{n-1}), \pi_n$$

Proof. Follows from Lemma 3 and Lemma 6. □

Given an execution path π and a graph history Γ, the set of configurations induced by Γ and π, written $\mathtt{confs}(\Gamma, \pi)$, includes those configurations that satisfy the following condition:

$$\mathtt{confs}(\Gamma, \pi) \triangleq \left\{(M, PB, B) \;\middle|\; \mathsf{wf}(M, PB, B, \mathtt{hist}(\Gamma), \pi)\right\}$$

Definition 10.

$$\begin{aligned} \mathsf{norm}(\Gamma, \pi) &\overset{\text{def}}{\Leftrightarrow} \mathsf{norm}(\mathtt{hist}(\Gamma)) \wedge \mathsf{norm}(\pi) \\ \mathsf{norm}(\epsilon) &\overset{\text{def}}{\Leftrightarrow} \mathsf{true} \\ \mathsf{norm}((\pi_1, \pi_2).\mathcal{H}) &\overset{\text{def}}{\Leftrightarrow} \mathsf{norm}(\pi_1.\pi_2) \wedge \mathsf{norm}(\mathcal{H}) \end{aligned}$$

Lemma 7. *For all* $\mathsf{P}_0, \mathtt{rec}, \mathbf{rec}, \mathsf{P}, \mathsf{P}', \Gamma, \Gamma', \pi, \pi'$*:*
if

$$\begin{aligned} &\mathsf{wf}(\Gamma, \pi) \wedge \mathsf{norm}(\Gamma, \pi) \\ &\wedge \mathsf{wf}(\Gamma', \pi') \wedge \mathsf{norm}(\Gamma', \pi') \\ &\wedge \mathsf{sim}_{\mathsf{rec}}(\mathbf{rec}, \mathtt{rec}) \\ &\wedge (\mathsf{P}_0, \mathtt{rec}) \vdash \mathsf{P}, \Gamma, \pi \Rightarrow \mathsf{P}', \Gamma', \pi' \end{aligned}$$

then for all $(M, PB, B) \in \mathtt{confs}(\Gamma, \pi)$*, there exists* $(M', PB', B) \in \mathtt{confs}(\Gamma', \pi')$ *such that*

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathtt{hist}(\Gamma), \pi \Rightarrow^* \mathsf{P}', M', PB', B', \mathtt{hist}(\Gamma'), \pi'$$

Proof. Pick arbitrary $\mathsf{P}_0, \mathtt{rec}, \mathbf{rec}, \mathsf{P}, \mathsf{P}', \Gamma, \Gamma', \pi, \pi'$ such that $\mathsf{wf}(\Gamma, \pi)$, $\mathsf{norm}(\Gamma, \pi)$, $\mathsf{wf}(\Gamma', \pi')$, $\mathsf{norm}(\Gamma', \pi')$, $\mathsf{sim}_{\mathsf{rec}}(\mathbf{rec}, \mathtt{rec})$, and $(\mathsf{P}_0, \mathtt{rec}) \vdash \mathsf{P}, \Gamma, \pi \Rightarrow \mathsf{P}', \Gamma', \pi'$. Pick an arbitrary $(M, PB, B) \in \mathtt{confs}(\Gamma, \pi)$. Let $\mathcal{H}$=$\mathtt{hist}(\Gamma)$. From the $\mathtt{confs}(.,.)$ definition we know that $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds. We proceed by induction on the structure of $\Rightarrow$.

Case (G-SilentP)
From (G-SilentP) we know $\mathsf{P} \xrightarrow{\mathcal{E}\langle\tau\rangle} \mathsf{P}'$, and Γ'=Γ, π'=π. As such, from (A-SilentP) we have $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}', M, PB, B, \mathcal{H}, \pi$. Moreover, as $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, the required result holds immediately.

Case (G-Prop)
From (G-Prop) and since $\mathsf{norm}(\Gamma', \pi')$ (i.e. $\forall e.\ \mathsf{D}\langle e\rangle \notin \pi$) we know there exists e and $\lambda \in \left\{\mathsf{B}\langle e\rangle, \mathsf{PB}\langle e\rangle, \mathsf{PFO}\langle e\rangle, \mathsf{PFL}\langle e\rangle, \mathsf{PSF}\langle e\rangle\right\}$ such that π'=$\pi.\lambda$, $\mathsf{fresh}(\lambda, \pi)$, $\mathsf{fresh}(\lambda, \Gamma)$, P'=P, and Γ'=Γ. From the $\mathsf{fresh}(.,.)$ definition we know $\mathsf{fresh}(\lambda, \mathcal{H})$ holds. There are six cases to consider: 1) $\lambda = \mathsf{PFO}\langle e\rangle$; or 2) $\lambda = \mathsf{PFL}\langle e\rangle$; or 2) $\lambda = \mathsf{PSF}\langle e\rangle$; or 3) $\lambda = \mathsf{B}\langle e\rangle$ and $e \in W$; or 4) $\lambda = \mathsf{B}\langle e\rangle$ and $e \in SF \cup FO \cup FL$; or 5) $\lambda = \mathsf{PB}\langle e\rangle$ and $e \in W \cup U$; or 6) $\lambda = \mathsf{PB}\langle e\rangle$ and $e \in FO \cup FL$.

For case (1), let $\mathtt{loc}(e)$=x and $\mathsf{B}(\tau) = b$. In what follows we demonstrate $b \cap (W_x \cup SF \cup \left\{\langle \mathsf{fo}, e\rangle, \langle \mathsf{fl}, e\rangle \;\middle|\; \mathtt{loc}(e) \in X\right\}) = \emptyset$. As such, from (AM-BFetchFO), we have: $M, PB, B \xrightarrow{\mathsf{PFO}\langle e\rangle} M, PB, B[\tau \mapsto b.\langle \mathsf{pfo}, e\rangle]$. That is, there exists $M' = M$, $PB' = PB$ and $B' = B[\tau \mapsto b.\langle \mathsf{pfo}, e\rangle]$ such that $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$, as required.

We next demonstrate that $b \cap (W_x \cup SF \cup \{\langle \mathsf{fo}, e'\rangle, \langle \mathsf{fl}, e'\rangle \mid \mathtt{loc}(e') \in X\}) = \emptyset$. We proceed by contradiction. Let us suppose there exists $w \in W_x$ such that $w \in b$. Since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, we then know that $\mathsf{W}\langle w\rangle \in \pi$ and $\mathsf{B}\langle w\rangle \notin \pi$. On the other hand, since $\mathsf{W}\langle w\rangle \prec_{\pi'} \mathsf{PFO}\langle e\rangle$ and $\mathsf{wf}(\Gamma', \pi')$, we have $\mathsf{B}\langle w\rangle \prec_{\pi'} \mathsf{PFO}\langle e\rangle$, i.e. $\mathsf{B}\langle w\rangle \in \pi$, leading to contradiction. Similarly, let us suppose there exists $sf \in SF$ such that $sf \in b$. Since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, we then know that $\mathsf{SF}\langle w\rangle \in \pi$ and $\mathsf{B}\langle w\rangle \notin \pi$. On the other hand, since $\mathsf{SF}\langle w\rangle \prec_{\pi'} \mathsf{PFO}\langle e\rangle$ and $\mathsf{wf}(\Gamma', \pi')$, we have $\mathsf{B}\langle w\rangle \prec_{\pi'} \mathsf{PFO}\langle e\rangle$, i.e. $\mathsf{B}\langle w\rangle \in \pi$, leading to contradiction. Finally, let us assume there exists $\langle o, e'\rangle \in b$ such that $o \in \{\mathsf{fo}, \mathsf{fl}\}$ and $\mathtt{loc}(e') \in X$. Since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, we then know there exists $\lambda' \in \{\mathsf{FO}\langle e'\rangle, \mathsf{FL}\langle e'\rangle\}$ such that $\lambda' \in \pi$ and $\mathsf{B}\langle e'\rangle \notin \pi$. On the other hand, since $\lambda' \prec_{\pi'} \mathsf{PFO}\langle e\rangle$ and $\mathsf{wf}(\Gamma', \pi')$, we have $\mathsf{B}\langle e'\rangle \prec_{\pi'} \mathsf{PFO}\langle e\rangle$, i.e. $\mathsf{B}\langle e'\rangle \in \pi$, leading to contradiction.

The proof of cases (2) and (3) are analogous and thus omitted here.

For case (3), let $\mathtt{loc}(e) \in X$, $B(\tau) = b$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that there exist b_1, b_2 such that $B(\tau) = b_1.e.b_2$ and $(SF \cup W \cup FL \cup \{\langle \mathsf{fo}, e'\rangle \mid \mathtt{loc}(e') \in X\}) \cap b_1 = \emptyset$. From (AM-BPropW) we then have $M, PB, B \xrightarrow{\mathsf{B}\langle e\rangle} M, PB.e, B[\tau \mapsto b_1.b_2]$. As such, from (A-PropM) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB.e, B[\tau \mapsto b_1.b_2], \mathcal{H}, \pi.\lambda$$

That is, there exists $M' = M$, $PB' = PB.e$ and $B' = B[\tau \mapsto b_1.b_2]$ such that $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$, as required.

The proof of case (4) is analogous and thus omitted here.

For case (5), let $\mathtt{loc}(e) = x$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that there exist $\mathsf{PB}_1, \mathsf{PB}_2$ such that $\mathsf{PB} = \mathsf{PB}_1.e.\mathsf{PB}_2$ and $\mathsf{PB}_1 \cap (W_x \cup FO \cup FL) = \emptyset$. From (AM-PropW) we then have $M, PB, B \xrightarrow{\mathsf{PB}\langle e\rangle} M[x \mapsto e], PB_1.\mathsf{PB}_2, B$. As such, from (A-PropM) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M[x \mapsto e], PB_1.\mathsf{PB}_2, B, \mathcal{H}, \pi.\lambda$$

That is, there exists $M' = M[x \mapsto e]$, $PB' = PB_1.\mathsf{PB}_2$ and $B' = B$ such that $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$, as required.

The proof of case (6) is analogous and thus omitted here.

Case (G-Crash)
Let $\Gamma = (G_1, -).\cdots.(G_n, -)$. From (G-Crash) we know there exists π'' and G such that $\mathsf{P}' = \mathtt{rec}(\mathsf{P}_0, G)$, $\Gamma' = \Gamma.(G, (\pi, \pi''))$, $\pi' = \epsilon$, $\mathtt{comp}(\pi, \pi'')$ and $G \prec \mathtt{getG}(\mathtt{hist}(\Gamma), \pi, \pi'')$. Since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we know that for all events e:

- $e \in B(\mathtt{tid}(e)) \Leftrightarrow (\mathsf{B}\langle e\rangle \notin \pi \wedge (\mathsf{W}\langle e\rangle \in \pi \vee \mathsf{SF}\langle e\rangle \in \pi \vee \mathsf{FO}\langle e\rangle \in \pi \vee \mathsf{FL}\langle e\rangle \in \pi))$
- $\langle \mathsf{pfo}, e\rangle \in B(\mathtt{tid}(e)) \Leftrightarrow \mathsf{J}\langle e\rangle, \mathsf{D}\langle e\rangle \notin \pi \wedge \mathsf{PFO}\langle e\rangle \in \pi$
- $\langle \mathsf{pfl}, e\rangle \in B(\mathtt{tid}(e)) \Leftrightarrow \mathsf{J}\langle e\rangle, \mathsf{D}\langle e\rangle \notin \pi \wedge \mathsf{PFL}\langle e\rangle \in \pi$
- $\langle \mathsf{psf}, e\rangle \in B(\mathtt{tid}(e)) \Leftrightarrow \mathsf{J}\langle e\rangle, \mathsf{D}\langle e\rangle \notin \pi \wedge \mathsf{PSF}\langle e\rangle \in \pi$
- $e \in PB \Leftrightarrow \mathsf{PB}\langle e\rangle \notin \pi \wedge (\mathsf{B}\langle e\rangle \in \pi \vee \mathsf{U}\langle e, -\rangle \in \pi \vee \mathsf{PFO}\langle e\rangle \in \pi \vee \mathsf{PFL}\langle e\rangle \in \pi)$

As such, from the definition of $\mathtt{comp}(.,.)$, and since $\mathsf{norm}(\Gamma', \pi')$ holds (i.e. $\forall e.\ \mathsf{D}\langle e\rangle \notin \pi.\pi''$), we know for all events e:

- $\mathsf{D}\langle e\rangle \notin \pi''$
- $e \in B(\mathtt{tid}(e)) \Leftrightarrow \mathsf{B}\langle e\rangle \in \pi''$
- $\langle \mathsf{pfo}, e\rangle \in B(\mathtt{tid}(e)) \vee \langle \mathsf{pfl}, e\rangle \in B(\mathtt{tid}(e)) \vee \langle \mathsf{psf}, e\rangle \in B(\mathtt{tid}(e)) \Leftrightarrow \mathsf{J}\langle e\rangle \in \pi''$
- $e \in PB \Leftrightarrow \mathsf{PB}\langle e\rangle \in \pi''$

Moreover, from $\mathtt{getG}(\mathtt{hist}(\Gamma), \pi, \pi'')$ we have $\mathsf{wfp}(\pi.\pi'', \mathtt{hist}(\Gamma))$. As such, from the definition of $\rightarrow_{\mathsf{p}}$ and above we have $M, PB, B \xrightarrow{\pi''}_{\mathsf{p}} -, PB_0, B_0$.

Let $M'{=}M$, $PB'{=}PB_0$, $B'{=}B_0$ and $\mathcal{H}'{=}\mathcal{H}.(\pi, \pi'')$=$\mathtt{hist}(\Gamma')$. Since $\mathtt{comp}(\pi, \pi'')$ holds, by definition we also have $\mathsf{complete}(\pi.\pi'')$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma', \pi')$ hold, from their definitions we also know that $\mathsf{wf}(M', PB', B', \mathcal{H}', \pi')$ holds and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$. On the other hand, since $\mathsf{sim}_{\mathsf{rec}}(\mathbf{rec}, \mathtt{rec})$ holds and $(M, PB, B) \in \mathtt{confs}(\Gamma, \pi)$, it is straightforward to demonstrate that $\mathsf{sim}_{\mathsf{GM}}(G, M)$ and thus that $\mathbf{rec}(\mathsf{P}_0, M)$=$\mathtt{rec}(\mathsf{P}_0, G)$=$\mathsf{P}'$. Consequently, from (A-Crash) we have: $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}', M', PB', B', \mathcal{H}', \pi'$, as required.

Case (G-Step)
We know there exists e, r, u and $\lambda \in \{\mathsf{R}\langle r, e\rangle, \mathsf{W}\langle e\rangle, \mathsf{U}\langle u, e\rangle, \mathsf{MF}\langle e\rangle, \mathsf{SF}\langle e\rangle, \mathsf{FO}\langle e\rangle, \mathsf{FL}\langle e\rangle, \mathsf{J}\langle e\rangle\}$ such that $\pi'{=}\pi.\lambda$, $\mathsf{fresh}(\lambda, \pi)$, $\mathsf{fresh}(\lambda, \Gamma)$, $\Gamma'{=}\Gamma$ and $\mathsf{P} \xrightarrow{\lambda} \mathsf{P}'$. From the definition of $\mathsf{fresh}(.,.)$ we then know that $\mathsf{fresh}(\lambda, \mathcal{H})$ holds. There are now ten cases to consider:

(1) $\lambda = \mathsf{R}\langle r, e\rangle$
(2) $\lambda = \mathsf{W}\langle e\rangle$
(3) $\lambda = \mathsf{U}\langle u, e\rangle$
(4) $\lambda = \mathsf{MF}\langle e\rangle$
(5) $\lambda = \mathsf{SF}\langle e\rangle$
(6) $\lambda = \mathsf{FO}\langle e\rangle$
(7) $\lambda = \mathsf{FL}\langle e\rangle$
(8) $\lambda = \mathsf{J}\langle e\rangle$ and $e \in FO$
(9) $\lambda = \mathsf{J}\langle e\rangle$ and $e \in FL$
(10) $\lambda = \mathsf{J}\langle e\rangle$ and $e \in SF$

Case (1)*:* $\lambda = \mathsf{R}\langle r, e\rangle$
Let $\mathtt{tid}(r) = \tau$, $\mathtt{loc}(r) = x$ and $B(\tau) = b$. In what follows we demonstrate that $\mathtt{read}(M, PB, b, x) = e$. From (AM-Read) we then have $M, PB, B \xrightarrow{\mathsf{R}\langle r, e\rangle} M, PB, B$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB, B, \mathcal{H}, \pi.\lambda$$

That is, there exists $M'{=}M$, $PB'{=}PB$, $B'{=}B$ such that $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$, as required. We next demonstrate that $\mathtt{read}(M, PB, b, x) = e$.

From the definition of $\mathsf{wf}(\Gamma, \pi.\lambda)$ we know that $\mathsf{wfrd}(r, e, \pi, \pi_h)$, where $\pi_h = \pi_1.\cdots.\pi_n$, when $\Gamma = (-, (\pi_1, -)).\cdots.(-, (\pi_n, -))$. From the definition of $\mathsf{wfrd}(r, e, \pi, \pi_h)$ there are now four cases:

i) $\exists \pi_1, \pi_2.\ \pi = \pi_2.\mathsf{W}\langle e\rangle.\pi_1 \wedge \mathtt{tid}(e) = \mathtt{tid}(r) \wedge \mathsf{B}\langle e\rangle \notin \pi_1$
$\wedge \left\{\mathsf{W}\langle e'\rangle \in \pi_1 \,\middle|\, \mathtt{loc}(e'){=}\mathtt{loc}(r) \wedge \mathtt{tid}(e'){=}\mathtt{tid}(r)\right\} = \emptyset$
ii) $\exists \pi_1, \pi_2, \lambda_e.\ \pi = \pi_2.\lambda_e.\pi_1 \wedge (\lambda_e{=}\mathsf{B}\langle e\rangle \vee \lambda_e{=}\mathsf{U}\langle e, -\rangle)$
$\wedge \left\{\mathsf{B}\langle e'\rangle, \mathsf{U}\langle e', -\rangle \in \pi_1 \,\middle|\, \mathtt{loc}(e'){=}\mathtt{loc}(r)\right\} = \emptyset$
$\wedge \left\{e' \,\middle|\, \begin{array}{l}\mathsf{W}\langle e'\rangle \in \pi \wedge \mathsf{B}\langle e'\rangle \notin \pi \\ \wedge\, \mathtt{loc}(e'){=}\mathtt{loc}(r) \wedge \mathtt{tid}(e'){=}\mathtt{tid}(r)\end{array}\right\} = \emptyset$

iii) $\exists \pi_1, \pi_2.\ \pi_h = \pi_2.\mathsf{PB}\langle e\rangle.\pi_1$
$$\wedge \left\{ \begin{array}{l|l} \mathsf{B}\langle e'\rangle, \mathsf{U}\langle e', -\rangle \in \pi, & \mathtt{loc}(e')=\mathtt{loc}(r) \wedge \\ \mathsf{W}\langle e''\rangle \in \pi, & \mathtt{loc}(e'')=\mathtt{loc}(r) \wedge \\ \mathsf{PB}\langle e'\rangle \in \pi_1 & \mathtt{tid}(e'')=\mathtt{tid}(r) \end{array} \right\} = \emptyset$$
iv) $e = init_x \wedge \left\{ \begin{array}{l|l} \mathsf{B}\langle e'\rangle, \mathsf{U}\langle e', -\rangle \in \pi, & \mathtt{loc}(e')=\mathtt{loc}(r) \wedge \\ \mathsf{W}\langle e''\rangle \in \pi, & \mathtt{loc}(e'')=\mathtt{loc}(r) \wedge \\ \mathsf{PB}\langle e'\rangle \in \pi_h & \mathtt{tid}(e'')=\mathtt{tid}(r) \end{array} \right\} = \emptyset$

In case (i), since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we know there exists b_1, b_2 such that $b = b_2.e.b_1$ and $\forall e' \in b_1 \cap W.\ \mathtt{loc}(e') \neq x$. As such, by definition we have $\mathtt{read}(M, PB, b, x) = e$.

In case (ii), since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we know that for all $e' \in b \cap W$, $\mathtt{loc}(e') \neq x$; and that there exists PB_1, PB_2 such that $PB = PB_2.e.PB_1$, and for all $e' \in PB_1 \cap W$, $\mathtt{loc}(e') \neq x$. As such, by definition we have $\mathtt{read}(M, PB, b, x) = e$.

In case (iii), since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we know for all $e' \in (b \cup PB) \cap W$, $\mathtt{loc}(e') \neq x$; and that $M(x) = e$. As such, by definition we have $\mathtt{read}(M, PB, b, x) = e$.

In case (iv), since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we know for all $e' \in (b \cup PB) \cap W$, $\mathtt{loc}(e') \neq x$; and that $M(x) = init_x$. As such, by definition we have $\mathtt{read}(M, PB, b, x) = e$.

Case (2): $\lambda = \mathsf{W}\langle e\rangle$
Let $\mathtt{tid}(e)=\tau$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that $\left\{\langle \mathsf{pfl}, e_1\rangle, \langle \mathsf{pfo}, e_2\rangle \;\middle|\; \mathtt{loc}(e_2) \in X\right\} \cap B(\tau) = \emptyset$. From (AM-Write) we then have $M, PB, B \xrightarrow{\mathsf{W}\langle e\rangle} M, PB, B[\tau \mapsto B(\tau).e]$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB, B[\tau \mapsto B(\tau).e], \mathcal{H}, \pi.\lambda$$

That is, there exists $M'=M$, $PB'=PB$ and $B'=B[\tau \mapsto B(\tau).e]$ such that $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$, as required.

Case (3): $\lambda = \mathsf{U}\langle u, e\rangle$
Let $\mathtt{tid}(u)=\tau$ and $\mathtt{loc}(u)=x \in X$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that $B(\tau)=\epsilon$. In an analogous way to that in case (1) we can demonstrate that $\mathtt{read}(M, PB, b, x) = e$. From (AM-RMW) we then have $M, PB, B \xrightarrow{\mathsf{U}\langle u, e\rangle} M, PB.u, B$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB.u, B, \mathcal{H}, \pi.\lambda$$

That is, $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$, where $M'=M$, $PB'=PB.u$ and $B'=B$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\mathtt{confs}(.,.)$ we have $(M', PB', B') \in \mathtt{confs}(\Gamma, \pi')$, as required.

Case (4): $\lambda = \mathsf{MF}\langle e\rangle$
Let $\mathtt{tid}(e)=\tau$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that $B(\tau)=\epsilon$. From (AM-MFence) we then have $M, PB, B \xrightarrow{\mathsf{MF}\langle e\rangle} M, PB, B$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB, B, \mathcal{H}, \pi.\lambda$$

That is, $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$, when $M'=M$, $PB'=PB$ and $B'=B$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from

the definition of $\texttt{confs}(.,.)$ we have $(M', PB', B') \in \texttt{confs}(\Gamma, \pi')$, as required.

Case (5): $\lambda = \mathsf{SF}\langle e\rangle$
Let $\texttt{tid}(e)=\tau$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that $\forall e'.\ \forall o \in \{\mathsf{pfo}, \mathsf{pfl}\}.\ \langle o, e'\rangle \notin B(\tau)$. From (AM-SFence) we then have $M, PB, B \xrightarrow{\mathsf{SF}\langle e\rangle} M, PB, B[\tau \mapsto B(\tau).e]$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB, B[\tau \mapsto B(\tau).e], \mathcal{H}, \pi.\lambda$$

That is, $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$, when $M'=M$, $PB'=PB$ and $B'=B[\tau \mapsto B(\tau).e]$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\texttt{confs}(.,.)$ we have $(M', PB', B') \in \texttt{confs}(\Gamma, \pi')$, as required.

Case (6): $\lambda = \mathsf{FO}\langle e\rangle$
Let $\texttt{tid}(e)=\tau$ and $\texttt{loc}(e) \in X$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that $\forall e'.\ \texttt{loc}(e') \in X \Rightarrow \langle \mathsf{pfl}, e'\rangle, \langle \mathsf{pfo}, e'\rangle \notin b$. From (AM-FO) we then have $M, PB, B \xrightarrow{\mathsf{FO}\langle e\rangle} M, PB, B[\tau \mapsto B(\tau).e]$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB, B[\tau \mapsto B(\tau).e], \mathcal{H}, \pi.\lambda$$

That is, $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$, when $M'=M$, $PB'=PB$ and $B'=B[\tau \mapsto B(\tau).e]$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\texttt{confs}(.,.)$ we have $(M', PB', B') \in \texttt{confs}(\Gamma, \pi')$, as required.

The proof of case (7) is analogous and thus omitted here.

Case (8): $\lambda = \mathsf{J}\langle e\rangle$ *and* $e \in FO$
Let $\texttt{tid}(e) = \tau$ and $\texttt{loc}(e) \in X$. As $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ and $\mathsf{wf}(\Gamma, \pi)$ hold, it is straightforward to demonstrate that there exist b_1, b_2 such that $B(\tau)=b_1.\langle \mathsf{pfo}, e\rangle.b_2$ and $\forall e'.\ \texttt{loc}(e') \in X \Rightarrow \langle \mathsf{pfl}, e'\rangle, \langle \mathsf{pfo}, e'\rangle \notin b_1$. As such, from (AM-FO2) we have: $M, PB, B \xrightarrow{\mathsf{J}\langle e\rangle} M, PB, B[\tau \mapsto b_1.b_2]$. As such, from (A-Step) we have:

$$\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M, PB, B[\tau \mapsto b_1.b_2], \mathcal{H}, \pi.\lambda$$

That is, there exist $M'=M$, $PB'=PB$ and $B'=B[\tau \mapsto b_1.b_2]$ such that $\mathbf{rec} \vdash \mathsf{P}, M, PB, B, \mathcal{H}, \pi \Rightarrow \mathsf{P}, M', PB', B', \mathcal{H}, \pi'$. Moreover, since $\mathsf{wf}(M, PB, B, \mathcal{H}, \pi)$ holds, from its definition we also have $\mathsf{wf}(M', PB', B', \mathcal{H}, \pi')$ and thus from the definition of $\texttt{confs}(.,.)$ we have $(M', PB', B') \in \texttt{confs}(\Gamma, \pi')$, as required.

The proof of cases (9)-(10) are analogous and thus omitted here. □

Theorem 4 (Completeness). *For all* P, $\mathbf{rec}$, $\texttt{rec}$ *and all Px86$_{man}$-valid chains* $\mathcal{C}$ *of* P, *if* $\mathsf{sim}_{\texttt{rec}}(\mathbf{rec}, \texttt{rec})$ *then there exist* $M, \mathcal{H}$ *and* π *such that*

$$\mathbf{rec} \vdash \mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, M, PB_0, B_0, \mathcal{H}, \pi$$

Proof. Follows from Corollary 1, Lemma 4 and Lemma 7. □

A.4 Equivalence of Px86$_{\mathbf{man}}$ Operational and Intermediate Semantics

Let

$$R_l \triangleq \left\{ ((\tau : l), \lambda) \;\middle|\; \begin{array}{l} \mathtt{tid}(\lambda){=}\tau \wedge \exists e, x. \\ \quad \left(\begin{array}{l} (\mathtt{getE}(\lambda){=}e \wedge \lambda \neq \mathsf{J}\langle e\rangle \wedge \wedge \mathtt{lab}(e){=}l) \\ \vee(\lambda{=}\mathsf{J}\langle e\rangle \wedge e \in FO_x \wedge l{=}(\mathsf{FO}, x)) \\ \vee(\lambda{=}\mathsf{J}\langle e\rangle \wedge e \in FL_x \wedge l{=}(\mathsf{FL}, x)) \\ \vee(\lambda{=}\mathsf{J}\langle e\rangle \wedge e \in SF \wedge l{=}\mathsf{SF}) \\ \vee(\lambda \in \left\{\mathsf{D}\langle -\rangle, \mathcal{E}\langle\tau\rangle, \mathsf{B}\langle -\rangle, \mathsf{PB}\langle -\rangle, \mathsf{PFO}\langle -\rangle, \mathsf{PFL}\langle -\rangle, \mathsf{PSF}\langle -\rangle\right\} \wedge l = \epsilon) \end{array} \right) \end{array} \right\}$$

Lemma 8. *For all* P, P'*:*

- *for all* τ, l*, if* $\mathsf{P} \xrightarrow{\tau:l} \mathsf{P}'$*, then there exists* λ *such that:* $((\tau, l), \lambda) \in R_l$ *and* $\mathsf{P} \xrightarrow{\lambda} \mathsf{P}'$
- *for all* λ*, if* $\mathsf{P} \xrightarrow{\lambda} \mathsf{P}'$*, then there exists* τ, l *such that:* $((\tau, l), \lambda) \in R_l$ *and* $\mathsf{P} \xrightarrow{\tau:l} \mathsf{P}'$

Proof. By straightforward induction on the structures of $\xrightarrow{\tau:l}$ and $\xrightarrow{\lambda}$. □

Let

$$R_m \triangleq \left\{ \begin{array}{l|l} ((\mathsf{M}, \mathsf{PB}, \mathsf{B}), & (\mathsf{M}, \mathsf{PB}, \mathsf{B}) \in \textsc{Mem} \times \textsc{PBuff} \times \textsc{BMap} \\ (M, PB, B)) & \wedge\ (M, PB, B) \in \textsc{AMem} \times \textsc{APBuff} \times \textsc{ABMap} \\ & \wedge\ \forall x, v.\ \mathsf{M}(x) = v \Leftrightarrow \mathtt{val}_{\mathsf{w}}(M(x)) = v \\ & \wedge\ \mathsf{sim}_{\mathsf{pb}}(\mathsf{PB}, PB) \wedge \mathsf{sim}_{\mathsf{b}}(\mathsf{B}, B) \end{array} \right\}$$

$$\begin{array}{rl} \mathsf{sim}_{\mathsf{pb}}(\mathsf{PB}, PB) \overset{\text{def}}{\Leftrightarrow} & \mathsf{PB} = PB = \epsilon \\ & \vee\ \exists \mathsf{PB}', PB', x, v, e.\ \mathsf{PB}{=}\langle x, v\rangle.\mathsf{PB}' \wedge PB{=}e.PB' \wedge \mathtt{loc}(e){=}x \wedge \mathtt{val}_{\mathsf{w}}(e){=}v \\ & \vee\ \exists \mathsf{PB}', PB', x, e.\ \mathsf{PB}{=}\langle \mathsf{per}, x\rangle.\mathsf{PB}' \wedge PB{=}e.PB' \wedge \mathtt{loc}(e){=}x \wedge e \in FO \cup FL \end{array}$$

$$\mathsf{sim}_{\mathsf{b}}(\mathsf{B}, B) \overset{\text{def}}{\Leftrightarrow} dom(\mathsf{B}){=}dom(B) \wedge \forall \tau \in dom(\mathsf{B}).\ \mathsf{sim}_{\mathsf{b}}(\mathsf{B}(\tau), B(\tau))$$

$$\begin{array}{rl} \mathsf{sim}_{\mathsf{b}}(\mathsf{b}, b) \overset{\text{def}}{\Leftrightarrow} & (\mathsf{b}{=}b{=}\epsilon) \\ & \vee\ \exists \mathsf{b}', b', x, v, e.\ \mathsf{b}{=}\langle x, v\rangle.\mathsf{b}' \wedge b{=}e.b' \wedge \mathtt{val}_{\mathsf{w}}(e){=}v \wedge e \in W_x \wedge \mathsf{sim}_{\mathsf{b}}(\mathsf{b}', b') \\ & \vee\ \exists \mathsf{b}', b', e.\ \mathsf{b}{=}\langle \mathsf{sf}\rangle.\mathsf{b}' \wedge b{=}\langle \mathsf{sf}, e\rangle.b' \wedge e \in SF \wedge \mathsf{sim}_{\mathsf{b}}(\mathsf{b}', b') \\ & \vee\ \exists \mathsf{b}', b', e.\ \mathsf{b}{=}\langle \mathsf{psf}\rangle.\mathsf{b}' \wedge b{=}\langle \mathsf{psf}, e\rangle.b' \wedge e \in SF \wedge \mathsf{sim}_{\mathsf{b}}(\mathsf{b}', b') \\ & \vee\ \exists \mathsf{b}', b', x, e, o.\ o \in \{\mathsf{fo}, \mathsf{pfo}\} \wedge \mathsf{b}{=}\langle o, x\rangle.\mathsf{b}' \wedge b{=}\langle o, e\rangle.b' \wedge e \in FO_x \wedge \mathsf{sim}_{\mathsf{b}}(\mathsf{b}', b') \\ & \vee\ \exists \mathsf{b}', b', x, e, o.\ o \in \{\mathsf{fl}, \mathsf{pfl}\} \wedge \mathsf{b}{=}\langle o, x\rangle.\mathsf{b}' \wedge b{=}\langle o, e\rangle.b' \wedge e \in FL_x \wedge \mathsf{sim}_{\mathsf{b}}(\mathsf{b}', b') \end{array}$$

Lemma 9. *For all* $M, PB, B, M, PB, B, M', PB', B'$*:*

- $((\mathsf{M}_0, \mathsf{PB}_0, \mathsf{B}_0), (M_0, PB_0, B_0)) \in R_m$
- *for all* $\mathsf{M}', \mathsf{PB}', \mathsf{B}', \tau, l$ *such that* $(\mathsf{M}, \mathsf{PB}, \mathsf{B}) \xrightarrow{\tau:l} (\mathsf{M}', \mathsf{PB}', \mathsf{B}')$*:*
  if $((\mathsf{M}, \mathsf{PB}, \mathsf{B}), (M, PB, B)) \in R_m$
  then there exist M', PB', B', λ *such that* $((\tau, l), \lambda) \in R_l$*,* $((\mathsf{M}', \mathsf{PB}', \mathsf{B}'), (M', PB', B')) \in R_m$ *and* $(M, PB, B) \xrightarrow{\lambda} (M', PB', B')$
- *for all* M', PB', B', λ *such that* $(M, PB, B) \xrightarrow{\lambda} (M', PB', B')$*:*
  if $((\mathsf{M}, \mathsf{PB}, \mathsf{B}), (M, PB, B)) \in R_m$
  then there exist $\mathsf{M}', \mathsf{PB}', \mathsf{B}', \tau, l$ *such that* $((\tau, l), \lambda) \in R_l$*,* $((\mathsf{M}', \mathsf{PB}', \mathsf{B}'), (M', PB', B')) \in R_m$ *and* $(\mathsf{M}, \mathsf{PB}, \mathsf{B}) \xrightarrow{\tau:l} (\mathsf{M}', \mathsf{PB}', \mathsf{B}')$

Proof. The first part follows immediately from the definitions of M_0, PB_0, B_0, M_0, PB_0, B_0. The last two parts follow from straightforward induction on the structures of $\xrightarrow{\tau:l}$ and $\xrightarrow{\lambda}$. □

Let

$$R \triangleq \left\{ \begin{array}{l} ((\mathsf{P}, \mathsf{M}, \mathsf{PB}, \mathsf{B}), \\ (\mathsf{P}, M, PB, B, \mathcal{H}, \pi)) \end{array} \middle| \begin{array}{l} \mathsf{P} \in \textsc{Prog} \wedge \mathcal{H} \in \textsc{Hist} \wedge \pi \in \textsc{Path} \\ \wedge\ ((\mathsf{M}, \mathsf{PB}, \mathsf{B}), (M, PB, B)) \in R_m \end{array} \right\}$$

Lemma 10. *For all* $\mathsf{P}, \mathsf{M}, \mathsf{PB}, \mathsf{B}, M, PB, B, M', PB', B', \mathcal{H}, \pi$*:*

- $((\mathsf{P}, \mathsf{M}_0, \mathsf{PB}_0, \mathsf{B}_0), (\mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon)) \in R$
- *for all* $\mathsf{P}', \mathsf{M}', \mathsf{PB}', \mathsf{B}'$ *such that* $(\mathsf{P}, \mathsf{M}, \mathsf{PB}, \mathsf{B}) \Rightarrow (\mathsf{P}', \mathsf{M}', \mathsf{PB}', \mathsf{B}')$*:*
  if $((\mathsf{P}, \mathsf{M}, \mathsf{PB}, \mathsf{B}), (\mathsf{P}, M, PB, B, \mathcal{H}, \pi)) \in R$
  then there exist $M', PB', B', \mathcal{H}', \pi'$ *such that* $((\mathsf{P}', \mathsf{M}', \mathsf{PB}', \mathsf{B}'), (\mathsf{P}', M', PB', B', \mathcal{H}', \pi')) \in R$ *and* $(\mathsf{P}, M, PB, B, \mathcal{H}, \pi) \Rightarrow (\mathsf{P}', M', PB', B', \mathcal{H}', \pi')$.
- *for all* $\mathsf{P}', M', PB', B', \mathcal{H}', \pi'$ *such that* $(\mathsf{P}, M, PB, B, \mathcal{H}, \pi) \Rightarrow (\mathsf{P}', M', PB', B', \mathcal{H}', \pi')$*:*
  if $((\mathsf{P}, \mathsf{M}, \mathsf{PB}, \mathsf{B}), (\mathsf{P}, M, PB, B, \mathcal{H}, \pi)) \in R$
  then there exist $\mathsf{M}', \mathsf{PB}', \mathsf{B}'$ *such that* $((\mathsf{P}', \mathsf{M}', \mathsf{PB}', \mathsf{B}'), (\mathsf{P}', M', PB', B', \mathcal{H}', \pi')) \in R$ *and* $(\mathsf{P}, \mathsf{M}, \mathsf{PB}, \mathsf{B}) \Rightarrow (\mathsf{P}', M', PB', B')$.

Proof. The proof of the first part follows immediately from the definition of R and Lemma 9. The proofs of the last two parts follow from straightforward induction on the structures of $\xrightarrow{\tau:l}$, $\xrightarrow{\lambda}$, Lemma 8 and Lemma 9. □

Theorem 5 (Intermediate and operational semantics equivalence)**.** *For all* P*:*

- *for all* M*:*
  if $\mathsf{P}, \mathsf{M}_0, \mathsf{PB}_0, \mathsf{B}_0 \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, \mathsf{M}, \mathsf{PB}_0, \mathsf{B}_0$,
  then there exist $M, \mathcal{H}, \pi$ *such that* $\mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, M, PB_0, B_0, \mathcal{H}, \pi$ *and* $((\mathsf{M}, \mathsf{PB}_0, \mathsf{B}_0), (M, PB_0, B_0)) \in R_m$
- *for all* $M, \mathcal{H}, \pi$*:*
  if $\mathsf{P}, M_0, PB_0, B_0, \epsilon, \epsilon \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, M, PB_0, B_0, \mathcal{H}, \pi$,
  then there exists M *such that* $\mathsf{P}, \mathsf{M}_0, \mathsf{PB}_0, \mathsf{B}_0 \Rightarrow^* \mathsf{P}_{\mathbf{skip}}, \mathsf{M}, \mathsf{PB}_0, \mathsf{B}_0$ *and* $((\mathsf{M}, \mathsf{PB}_0, \mathsf{B}_0), (M, PB_0, B_0)) \in R_m$.

Proof. Follows from Lemma 10 and straightforward induction on the length of $\Rightarrow^*$. □

B A CORRECT PSER IMPLEMENTATION IN Px86

We briefly describe the PSER model developed by Raad et al. [2019c]. We then develop a sound PSER implementation in Px86, thus demonstrating that PSER correctly compiles to Px86.

PSER Programming Language. For simplicity, Raad et al. [2019c] assume that the (sequential) programs in each thread comprise a sequence of PSER *transactions*. That is, the set of *PSER programs*, $\textsc{Prog}_{\textsc{pser}} \subseteq \textsc{Prog}$, are defined by the following grammar:

$$\textsc{Prog}_{\textsc{pser}} \ni \mathsf{P} ::= \textsc{TId} \xrightarrow{\text{fin}} \textsc{Com}_{\textsc{pser}} \qquad \textsc{Com}_{\textsc{pser}} \ni \mathsf{C}_{\textsc{pser}} ::= [\mathsf{T}] \mid \mathsf{C}_{\textsc{pser}}; \mathsf{C}_{\textsc{pser}}$$
$$\mathsf{T} ::= \mathsf{e} \mid \textbf{load}(x) \mid \textbf{store}(x, \mathsf{e}) \mid \textbf{let } \mathsf{a} := \mathsf{C} \textbf{ in } \mathsf{C} \mid \textbf{if } (\mathsf{C}) \textbf{ then } \mathsf{C} \textbf{ else } \mathsf{C} \mid \textbf{repeat } \mathsf{C}$$

PSER Labels and Events. In order to distinguish the events of one transaction from another, Raad et al. [2019c] assume a finite set of *transaction identifiers*, TXId, ranged over by ξ. A PSER label is then either: (1) a *read* label $(\mathtt{R}, x, v, \xi)$, for reading v from x in ξ; or (2) a *write* label $(\mathtt{W}, x, v, \xi)$, for writing v to x in ξ; or (3) a *begin* label $(\mathtt{B}, \xi)$, marking the beginning of ξ; or (4) an *end* label $(\mathtt{E}, \xi)$, marking the end of ξ. A *PSER event* is an event (Def. 1) with a PSER label. *PSER read* and *write events* comprise events with read and write labels, respectively. *PSER durable events* coincide with PSER write events. The function $\mathtt{tx}$ returns the transaction identifier of a PSER label or event.

Given an execution G, the *'same-transaction' relation*, $\mathsf{st} \in G.E \times G.E$, is the equivalence relation given by $\mathsf{st} \triangleq \left\{(a, b) \in G.E \times G.E \;\middle|\; \mathtt{tx}(a){=}\mathtt{tx}(b)\right\}$. Given a relation r on $G.E$, r_{T} denotes *lifting* r to (equivalence) classes: $\mathsf{r}_{\mathsf{T}} \triangleq \mathsf{st}; (\mathsf{r} \setminus \mathsf{st}); \mathsf{st}$, and $[a]_{\mathsf{st}}$ denotes the st class that contains a, i.e. $[a]_{\mathsf{st}} \triangleq \left\{e \in G.E \;\middle|\; (a, e) \in \mathsf{st}\right\}$. Note that a class without an end event denotes a transaction whose execution was rendered *incomplete* by a crash. The events of *complete transactions* in G are denoted by $G.T$; i.e. those events whose associated end events are in G: $G.T \triangleq \left\{a \in G.E \;\middle|\; \exists e \in [a]_{\mathsf{st}}.\ \mathtt{lab}(e){=}(\mathtt{E}, -)\right\}$.

PSER Executions. An execution G is a *PSER execution* if: (1) $G.E$ are PSER events; (2) each transaction class contains *exactly one* begin event; (3) each transaction class contains *at most one* end event; (4) each begin (resp. end) event is the first (resp. last) event (in po) within its transaction; and (5) only the last (po-maximal) transaction in each thread may be incomplete (due to a crash).

Definition 11 (PSER-consistency). A PSER execution $(E, I, P, \mathsf{po}, \mathsf{rf}, \mathsf{mo}, \mathsf{nvo})$ is *PSER-consistent* iff:

- $(\mathsf{rf} \cup \mathsf{mo} \cup \mathsf{rb}) \cap \mathsf{st} \subseteq \mathsf{po}$ where $\mathsf{rb} \triangleq (\mathsf{rf}^{-1}; \mathsf{mo}) \setminus \mathsf{id}$ (SER1)
- $\mathsf{hb}_{\mathsf{ser}}$ is irreflexive, where $\mathsf{hb}_{\mathsf{ser}} \triangleq (\mathsf{po}_{\mathsf{T}} \cup \mathsf{rf}_{\mathsf{T}} \cup \mathsf{mo}_{\mathsf{T}} \cup \mathsf{rb}_{\mathsf{T}})^{+}$ (SER2)
- $\mathsf{hb}_{\mathsf{ser}}|_D \subseteq \mathsf{nvo}$ (PSER-NVO)
- $dom([D]; \mathsf{st}; [P]) \subseteq P \subseteq G.T$ (PSER-ATOMIC1)
- $\mathsf{acyclic}(\mathsf{nvo}_{\mathsf{T}})$ (PSER-ATOMIC2)

The (SER1) and (SER2) axioms are those of serialisability [?] adapted to declarative consistency models as done e.g. in [Raad et al. 2018, 2019b]. The '*reads-before*' relation, rb, relates a read r to all writes that are mo-after the write r reads from. The (SER1) ensures that e.g. a transaction observes its own writes by requiring $\mathsf{rf} \cap \mathsf{st} \subseteq \mathsf{po}$ (i.e. intra-transactional reads respect po). The (SER2) guarantees the existence of a total sequential order in which all concurrent transactions appear to execute atomically one after another. This total order is obtained by an arbitrary extension of the (partial) 'happens-before' relation $\mathsf{hb}_{\mathsf{ser}}$, which captures synchronisation resulting from transactional orderings imposed by program order (po_{T}) or conflict ($\mathsf{rf}_{\mathsf{T}} \cup \mathsf{mo}_{\mathsf{T}} \cup \mathsf{rb}_{\mathsf{T}}$).

The (PSER-NVO), (PSER-ATOMIC1) and (PSER-ATOMIC2) axioms describe the persistency semantics of PSER. The (PSER-NVO) stipulates that transactional writes persist in the $\mathsf{hb}_{\mathsf{ser}}$ order. This in turn preserves inter-transactional synchronisation orderings across crashes. For instance, if ξ_2 reads from ξ_1, then ξ_1 persists before ξ_2; as such, upon recovery we never encounter the erroneous

```
0. [T]PSER→Px86 ≜
1.  LS:=∅;
2.  RS:=∅; WS:=∅;
3.  τ:=getTID(); ξ:=getTxID();
4.  log[τ]:=fo ξ;  w :=fo new-array();
5.  (|T|); sfence;
6.  ws[ξ]:=fo w;
7.  for (x∈WS) {
8.   if (promote(x)) LS.add(x);
9.   else {
10.    for (x∈LS) w-unlock(x);
11.    for (x∈(WS∪RS)\LS) r-unlock(x);
12.    goto line 1; } }
13.  for (x∈WS) {
14.   a:= w[x];
15.   x :=fo a;
16.  }
17.  sfence;
18.  for (x∈WS) w-unlock(x);
19.  for (x∈RS\WS) r-unlock(x);
```

```
(|x:=a|) ≜ if (x ∉ RS∪WS) {
             r-lock(x);
             l[x]:=fo ξ;
           } WS.add(x);
           w[x]:=fo a;

(|a:=x|) ≜ if (x ∉ RS∪WS) {
             r-lock(x);
             l[x]:=fo ξ;
           } RS.add(x);
           if (x∉WS)
             a:=x;
           else
             a:=w[x];

(|T1;T2|) ≜ (|T1|);(|T2|)

          ...
```

```
20. recover(P) ≜
21.  for (x∈dom(l))
22.   w-unlock(x);
23.  for (τ∈dom(P)) {
24.   ξ:=log[τ];
25.   w:=ws[ξ];
26.   if (w=⊥)
27.    P'[τ]:=sub(P[τ],ξ);
28.   else {
29.    P'[τ]:=sub(P[τ],ξ+1);
30.    if (!committed(w,ξ)) {
31.     for (x∈dom(w))
32.      x :=fo w[x];
33.    }
34.   }
35.  }
36.  sfence;
37.  run(P');
```

where committed(w, ξ) $\overset{\text{def}}{\Leftrightarrow} dom(w)=\emptyset \vee \exists x, \xi'.\ x \in dom(w) \wedge \xi' \neq \xi \wedge l[x]=\xi'$

Fig. 12. PSER implementation of transaction [T] in Px86 (left¦middle) where the grey code ensures deadlock avoidance and the highlighted code ensures persistency; PSER recovery implementation in Px86 (right).

scenario where ξ_2 has persisted, whilst the transaction it read from (ξ_1) has not. (PSER-ATOMIC1) and (PSER-ATOMIC2) ensure that transactions *persist atomically*: (1) only complete transactions persist ($P \subseteq G.T$); (2) either all or none of the (durable) events in a transaction persist ($dom([D]; \mathsf{st}; [P]) \subseteq P$); and (3) the persists of a transaction are not interleaved by those of others (acyclic(nvo_T)).

B.1 A PSER Implementation in Px86

In Fig. 12 we present a sound implementation of PSER and its recovery mechanism in Px86, thus demonstrating correct PSER-toPx86 compilation. As we often need to explicitly persist writes, we write $x :=_{\mathsf{fo}} e$ as a shorthand for $x := e; \mathbf{flush}_{\text{opt}}\ x$.

MRSW Locks. As we describe shortly, our PSER implementation in Fig. 12 uses *locks* to synchronise concurrent accesses to shared data. As serialisability allows concurrent transactions to read from the same memory location simultaneously, for better performance we use MRSW (multiple-readers-single-writer) locks. We thus assume that each location x is associated with an MRSW lock which can be acquired by either (i) multiple threads reading from x simultaneously; or (ii) a single thread writing to x. A reader (resp. writer) lock on x is acquired by calling r-lock(x) (resp. w-lock(x)), and released by calling r-unlock(x) (resp. w-unlock(x)). Moreover, a reader lock on x can be *promoted* to a writer one by calling promote(x). As two distinct reader locks on x may simultaneously attempt to promote their locks, promotion is done on a 'first-come-first-served' basis. A call to promote(x) thus returns a boolean denoting either (i) successful promotion (true); or (ii) failed promotion as another reader lock on x is currently being promoted (false). A call to promote(x) returns successfully once all other readers have released their locks on x and thus the calling reader can safely assume exclusive ownership of the lock (in write mode). Our MRSW lock implementation is straightforward, and is provided in Fig. 13.

```
r-lock(x) ≜                          can-promote(x) ≜
  start: a := xl;                      start: a := xl;
         if (is-odd a)                        if (is-odd a)
           goto start;                          return false;
         if (!CAS(xl, a, a+2))                if (!CAS(xl, a, a−1))
           goto start;                          goto start;
                                              repeat (xl == 1);
r-unlock(x) ≜ FAA(xl, −2);                    return true;

w-lock(x) ≜ repeat (CAS(xl, 0, 1))   w-unlock(x) ≜ xl := 0;
```

Fig. 13. MRSW lock implementation in Px86

Serialisability of Our PSER Implementation. Given a transaction [T], our PSER implementation of T in Px86, written $[\mathsf{T}]_{\text{PSER}\to\text{Px86}}$, is given in Fig. 12 (left). Ignoring the code in grey (lines 1, 8–12), and the highlighted code, $[\mathsf{T}]_{\text{PSER}\to\text{Px86}}$ describes a serialisable implementation of T using MRSW locks. Let RS and WS respectively denote the *read set* and *write set* of T, i.e. the locations read and written by T. Conceptually, a serialisable implementation of T would: (i) acquire the locks on all locations in RS ∪ WS; (ii) execute T *locally* where the reads in T are carried out in place (read directly from memory), while the writes are recorded tentatively in a log w; (iii) commit the *effect* of T (in w) by propagating the writes in w to memory; and (iv) release the acquired locks.

Note that the locations accessed by a transaction are not known in advance; i.e. the RS and WS are not known beforehand. As such, we cannot acquire all necessary locks at the beginning as stated in step (i) above. Instead, we compute RS and WS incrementally, acquiring the necessary locks *on the fly*, by combining steps (i)-(ii) above. Moreover, to reduce lock contention as much as possible, we acquire all necessary locks in read mode, and promote the locks on WS just before committing. Our serialisable implementation thus proceeds as follows. Starting with empty RS and WS (line 2), and an empty write log w (line 4), we execute T locally (as described above) whilst acquiring the necessary locks on the fly. This is denoted by $(\!|\mathsf{T}|\!)$ on line 5, as described shortly. Once the local execution $(\!|\mathsf{T}|\!)$ is completed, we promote the locks on WS (lines 7–8), commit the writes recorded in w to memory (lines 13–15), and finally release all acquired locks (lines 18–19).

The local execution $(\!|\mathsf{T}|\!)$ is given in Fig. 12 (middle), and is obtained from T as follows. For each write operation $x := \mathsf{a}$, the WS is extended with x, and the written value is logged in $w[x]$. Recall that to reduce lock contention, for each written location x, our implementation first acquires a reader lock on x, and subsequently promotes it to a writer lock. As such, the local execution of $x := \mathsf{a}$ first checks if a reader lock for x has been acquired (i.e. $x \in \mathsf{RS} \cup \mathsf{WS}$) and obtains one if this is not the case. Analogously, for each read operation $\mathsf{a} := x$, a reader lock is acquired if necessary and RS is extended with x. Moreover, as each transaction must observe its own writes, the local execution of $\mathsf{a} := x$ first checks if x has been written to by itself (i.e. $x \in \mathsf{WS}$). If this is not the case the value of x is read from the memory; otherwise, the value of x is read from the log w. The local execution of the remaining inductive cases (e.g. $\mathsf{T}_1; \mathsf{T}_2$) is defined by straightforward induction on the structure of commands (e.g. $(\!|\mathsf{T}_1; \mathsf{T}_2|\!) \triangleq (\!|\mathsf{T}_1|\!); (\!|\mathsf{T}_2|\!)$), and is omitted here.

Avoiding Deadlocks. Recall that a call to promote(x) by reader r returns false when another reader r' is in the process of promoting a lock on x. When this is the case, r must release its reader lock on x to ensure the successful promotion of x by r' and thus avoid deadlocks. To this end, our implementation includes a deadlock avoidance mechanism (lines 8–12) as follows. We record a set

LS (initialised with $\emptyset$ on line 1) of those locks on the write set that have been successfully promoted so far. When promoting a lock on x succeeds (line 8), then LS is extended with x. On the other hand, when promoting x fails (line 9), all those locks promoted so far (i.e. in LS) as well as the other reader locks acquired thus far (i.e. in $\mathtt{WS} \cup \mathtt{RS} \setminus \mathtt{LS}$) are released and the transaction is restarted.

Persistency of PSER Implementation. Recall that given $\mathsf{P} \in \textsc{Prog}_{\textsc{PSER}}$, the sequential program in each thread $\tau_i \in dom(\mathsf{P})$ comprises a sequence of transactions, i.e. $\mathsf{P}(\tau_i)=[\mathsf{T}_i^1];\cdots;[\mathsf{T}_i^n]$. We thus represent $\mathsf{P}(\tau_i)$ as an array T_i such that $T_i[j] = [\mathsf{T}_i^j]$. We further assume that the context of each thread τ_i is set up such that: (1) a call to `getTID()` returns i; and (2) a call to `getTxID()` returns j when executing $[\mathsf{T}_i^j]$. A program P is executed by calling `run(P)`.

To ensure correct recovery, our implementation must account for the possibility of a crash at each program point. To do this, we record the metadata for tracking the progress of each thread in *log*, *ws* and *l*, as follows. For each thread τ, $log[\tau]$ records the last executed transaction; for each transaction ξ, $ws[\xi]$ records the effect of ξ; and for each location x, $l[x]$ records the last transaction that acquired a lock on x. As such, when thread τ executes transaction ξ (line 3) with transaction code given by T, our implementation logs ξ in $log[\tau]$ (line 4); records the transaction's effect in $ws[\xi]$ (line 6); and records ξ in $l[x]$ for each location x accessed in T (via $(\!|\mathsf{T}|\!)$ on line 5).

Recall that the transaction effect is computed in w via $(\!|\mathsf{T}|\!)$. For correct recovery, we must ensure that the transaction effect is persisted *fully* and *not partially* in case of a crash. To achieve this, before recording the effect w in $ws[\xi]$ on line 6, we insert an **sfence** instruction (line 5) to ensure that all pending writes, including those of w, are persisted before the write on line 6.

Observe that our implementation adheres to the following pattern: (1) it updates the metadata for tracking the thread progress (lines 3–4); (2) executes an **sfence** (line 5); (3) executes the transaction (lines 7–15); and (4) executes an **sfence** (line 17). The first two steps ensure that the recovery metadata of each thread does not lag behind its progress; conversely, the last two steps ensure that the progress of each thread does not lag behind its recovery metadata. Therefore, in case of a crash, the persisted progress of each thread τ may at most be one step behind its persisted metadata.

PSER Recovery Implementation. After a crash, a program P is restored by calling `recover(P)` in Fig. 12 (right), which releases all locks to avoid deadlocks (lines 2–3); restores the progress of threads by generating a new program P' (lines 4–17); and ultimately runs P' (line 18).

Recall that the persisted progress of each thread is at most one step behind its persisted metadata. As such, it suffices to check whether the effect of the *last* recorded transaction for τ has persisted, and to resume the execution of τ accordingly. More concretely, let the last transaction executed by τ be ξ (line 5) and let us read the effect of ξ in the local variable w (line 6). Then, either (i) the effect has not persisted before the crash (i.e. the crash occurred before line 6) and thus $\mathtt{w}=\bot$ and $\mathsf{P}[\tau]$ is resumed from ξ (line 8), or (ii) the effect has persisted (i.e. the crash occurred after line 6) and thus $\mathsf{P}[\tau]$ is advanced to $\xi{+}1$ (line 10), where $\mathsf{sub}(\mathsf{P}[\tau], n)$ denotes the subarray of $\mathsf{P}[\tau]$ at n.

Note that in case (ii), the effect of ξ (in w) may not have fully committed or persisted to memory (e.g. if the crash occurred before line 13), and we must thus commit the transaction effect (lines 12–16). This is ascertained via `committed(w, ξ)` on line 11, checking if the writes of ξ in w have fully persisted. The `committed(w, ξ)` predicate is defined in Fig. 12. When $dom(\mathtt{w})=\emptyset$, the transaction is read-only and w is vacuously persisted. When $dom(\mathtt{w})\neq\emptyset$ and $x \in dom(\mathtt{w})$, we can safely assume w has persisted if another transaction $\xi'\neq\xi$ is the *last* transaction to acquire the lock on x (i.e. $l[x]=\xi'$). More concretely, since w has persisted, the crash must have occurred after line 6. That is, the $(\!|\mathsf{T}|\!)$ on line 5 has fully persisted and thus the lock on x was acquired by ξ (as $x \in dom(\mathtt{w})$). Consequently, as ξ' is the last transaction to acquire the x lock, then ξ must have released the lock on x (line 18),

i.e. ξ has fully committed and persisted. Finally, the **sfence** on line 17 ensures that the committed writes are persisted before subsequent writes in the restarted program P'.

Theorem 6 (Soundness). *The PSER implementation and its recovery mechanism in Fig. 12 are sound.*

Proof. The full proof is given in the next section (§C). □

C SOUNDNESS OF PSER IMPLEMENTATION IN Px86

For an arbitrary program P and a Px86-valid execution chain $C = G_1;\cdots;G_n$ of P with $G_i = (E_i, I_i, P_i, \mathsf{po}_i, \mathsf{rf}_i, \mathsf{mo}_i, \mathsf{nvo}_i)$, observe that when P comprises k threads, the trace of each execution era (via `start()` or `recover()`) comprises two stages: i) the trace of the *initialisation* stage by the master thread τ_0 performing initialisation or recovery, prior to the call to `run(P)`; followed (in po order) by ii) the trace of each of the constituent program threads $\tau_1 \cdots \tau_k$, provided that the execution did not crash during the initialisation stage.

Note that as the execution is Px86-valid, thanks to the placement of **sfence** instructions, for each thread τ_j, we know that the set of persistent events in execution era i, namely P_i, contains roughly a *prefix* (in po order) of thread τ_j's trace. More concretely, for each constituent thread $\tau_j \in \{\tau_1 \cdots \tau_k\} = dom(\mathsf{P})$, there exist $p^j_1 \cdots p^j_n, q^j_1 \cdots q^j_n, w^j_1, \cdots, w^j_n$ such that:

(1) $\mathsf{P}[\tau_j] = \mathsf{T}^0_j;\cdots;\mathsf{T}^{p^j_1}_j;\mathsf{T}^{p^j_1+1}_j;\cdots\mathsf{T}^{p^j_2}_j;\cdots;\mathsf{T}^{p^j_{n-1}+1}_j;\cdots;\mathsf{T}^{p^j_n}_j$, where each T^k_j denotes the k^{th} transaction of thread τ_j; and $\mathsf{T}^{p^j_i}_j$ denotes the last transaction of τ_j *logged* in the i^{th} era, i.e. the i^{th} crash occurred when $\mathtt{log}[\tau_j] = \xi^{p^j_i}_j$.

(2) At the beginning of each execution era $i \in \{1\cdots n\}$, for all j, the program executed by thread τ_j (calculated in P' and subsequently executed by calling `run(P')`) is that of $\mathtt{sub}(\mathsf{P}[\tau_j], q^i_j)$, such that either $q^i_j = p^{i-1}_j + 1$ when $w^j_i \neq \bot$, or $q^i_j = p^{i-1}_j$ when $w^j_i = \bot$, where $p^0_j = 0$.

(3) In each execution era $i \in \{1\cdots n\}$, the trace of the program is of the form $\theta^p_{\text{init}(i)} \xrightarrow{\text{po}} (\theta_{(i,1)}||\cdots||\theta_{(i,k)})$, where $\theta^p_{\text{init}(i)}$ denotes a (potentially full) prefix of $\theta_{\text{init}(i)}$; $\theta_{\text{init}(i)}$ denotes the execution of the initialisation or recovery mechanism defined shortly; and $\theta_{(i,j)}$ denotes the trace of the j^{th} constituent thread $\tau_j \in dom(\mathsf{P})$ and is defined as follows:

$$\theta_{(i,j)} \triangleq \begin{cases} \theta_i(\xi^{q^i_j}_j) \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} \theta^p_i(\xi^{p^i_j}_j) & \text{if } \theta'_{\text{init}_i} = \theta_{\text{init}_i} \\ \\ \emptyset & \text{otherwise} \end{cases}$$

More concretely, whenever $\theta^p_{\text{init}_i} = \theta_{\text{init}_i}$, i.e. no crash occurred during the execution of $\theta^p_{\text{init}_i}$, then $\theta_{(i,j)}$ denotes the execution of the $(q^i_j)^{\text{th}}$ to o^{th} transactions of thread τ_j, with $\theta_i(\xi)$ defined shortly. We write T^i for the set of all transactions executed in the i^{th} era.

Moreover, due to the placement of **sfence** instructions, before crashing and proceeding to the next era, *all* durable events in $\theta_i(\xi^{q^i_j}_j) \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} \theta_i(\xi^{p^i_j-1}_j)$ have persisted, and a *subset* of the durable events in $\theta_i(\xi^{p^i_j}_j)$ have persisted. Note that this subset may be equal to $\theta_i(\xi^{p^i_j}_j)$, in which case all its durable events have persisted.

In the very first era ($i = 1$) we have $\theta_{\text{init}(1)} = \emptyset$, and when $i > 1$, the $\theta_{\text{init}(i)}$ is of the form: $Us \xrightarrow{\text{po}} C(i,1) \xrightarrow{\text{po}} W(i,1) \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} C(i,k) \xrightarrow{\text{po}} W(i,k) \xrightarrow{\text{po}} sf$, where Us denotes the sequence of events releasing all locks, $\mathtt{lab}(sf)$=SF, and for all $i \in \{1\cdots n\}$ and $j \in \{1\cdots k\}$:

$$C(i{+}1,j) \triangleq rlog_{(i+1,j)} \xrightarrow{\text{po}} rwmap_{(i+1,j)} \xrightarrow{\text{po}} wp'_{(i+1,j)}$$

where $\texttt{lab}(rlog_{(i+1,j)}) = (\texttt{R}, \texttt{log}[\tau_j], \xi_j^{p_j^i})$, $\texttt{lab}(rwmap_{(i+1,j)}) = (\texttt{R}, \texttt{ws}[\xi_j^{p_j^i}], w_j^{i+1})$, $\texttt{lab}(wp'_{(i+1,j)}) = (\texttt{W}, \texttt{P'}[\tau_j], q_j^{i+1})$, and when $dom(w_j^{i+1}) = x_1 \cdots x_m$:

$$W(i{+}1, j) \triangleq W_1^{(i+1,j)} \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} W_m^{(i+1,j)}$$

and for all $t \in \{1 \cdots m\}$:

$$W_t^{(i+1,j)} \triangleq \begin{cases} wx_t^{(i+1,j)} \xrightarrow{\text{po}} fox_t^{(i+1,j)} & \text{if } q_j^{i+1}{=}p_j^i{+}1 \text{ and } \neg\texttt{committed}(w_j^{i+1}, \xi_j^{p_j^i}) \\ \emptyset & \text{otherwise} \end{cases}$$

such that $\texttt{lab}(wx_t^{(i+1,j)}) = (\texttt{W}, x_t, w_j^{i+1}[x_t])$ and $\texttt{lab}(fox_t^{(i+1,j)}) = (\texttt{FO}, x_t)$.

We write T_{rec}^i for the set of all transactions recovered in the i^{th} era:

$$T_{rec}^i \triangleq \left\{ \xi \;\middle|\; \exists j.\ \texttt{lab}(rlog_{(i,j)}) = (\texttt{R}, \texttt{log}[\tau_j], \xi) \wedge W(i,j) \neq \emptyset \right\}$$

Let $RS_\xi^0 = WS_\xi^0 = \emptyset$. When ξ is a transaction of thread τ with body $\texttt{T}$, then the trace $\theta_i(\xi)$ is of the form:

$$Fs \xrightarrow{\text{po}} Ts \xrightarrow{\text{po}} sf_1 \xrightarrow{\text{po}} log \xrightarrow{\text{po}} logfo \xrightarrow{\text{po}} PLs \xrightarrow{\text{po}} Ws \xrightarrow{\text{po}} sf_2 \xrightarrow{\text{po}} WUs \xrightarrow{\text{po}} RUs$$

where $\texttt{lab}(sf_1) = \texttt{lab}(sf_2) = \texttt{SF}$, and :

- Fs denotes the sequence of events failing to obtain the necessary locks, i.e. those iterations that do not succeed in promoting the writer locks;
- Ts denotes the sequence of events corresponding to the execution of $(\!|\texttt{T}|\!)$ and is of the form $t_1 \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} t_k$, where for $m \in \{1 \cdots k\}$ each t_m is either of the form $rd(x_m, v_m, RS_{m-1}, WS_{m-1})$ or $wr(x_m, v_m, RS_{m-1}, WS_{m-1})$, with:

$$rd(x_m, v_m, RS_{m-1}, WS_{m-1}) \triangleq \begin{cases} frl_m & \text{if } x_m \notin RS_{m-1} \cup WS_{m-1} \\ \xrightarrow{\text{po}} rl_{x_m}^0 \xrightarrow{\text{po}} rl_{x_m} & \\ \xrightarrow{\text{po}} wlog_{x_m} \xrightarrow{\text{po}} wrs_{x_m} & \\ \xrightarrow{\text{po}} r_{x_m} & \\ wrs_{x_m} \xrightarrow{\text{po}} r_{x_m} & \text{otherwise} \end{cases}$$

$$wr(x_m, v_m, RS_{m-1}, WS_{m-1}) \triangleq \begin{cases} fs_m & \text{if } x_m \notin RS_{m-1} \cup WS_{m-1} \\ \xrightarrow{\text{po}} rl_{x_m}^0 \xrightarrow{\text{po}} rl_{x_m} & \\ \xrightarrow{\text{po}} wlog_{x_m} \xrightarrow{\text{po}} wws_{x_m} & \\ \xrightarrow{\text{po}} lw_{x_m} \xrightarrow{\text{po}} lfo_{x_m} & \\ wws_{x_m} \xrightarrow{\text{po}} lw_{x_m} \xrightarrow{\text{po}} lfo_{x_m} & \text{otherwise} \end{cases}$$

where frl_m denotes the sequence of events attempting (but failing) to acquire the read lock on x_m, $\texttt{lab}(rl_{x_m}^0) = (\texttt{R}, xl_m, a)$, for some even value a, $\texttt{lab}(rl_{x_m}) = (\texttt{U}, xl_m, a, a+2)$, $\texttt{lab}(wlog_{x_m}) = (\texttt{W}, \texttt{l}[x_m], \xi)$, $\texttt{lab}(wrs_{x_m}) = (\texttt{W}, \texttt{RS}, RS_m)$, $\texttt{lab}(r_{x_m}) = (\texttt{R}, x_m, v_m)$ if $x_m \notin WS_{m-1}$; and $\texttt{lab}(r_{x_m}) = (\texttt{R}, \texttt{w}[x_m], v_m)$ otherwise, $\texttt{lab}(wws_{x_m}) = (\texttt{W}, \texttt{WS}, WS_m)$, $\texttt{lab}(lw_{x_m}) = (\texttt{W}, \texttt{w}[x_m], v_m)$, $\texttt{lab}(lfo_{x_m}) = (\texttt{FO}, \texttt{w}[x_m])$, and for all $m > 0$:

$$RS_{m+1} \triangleq \begin{cases} RS_m \cup \{x_m\} & \text{if } t_m{=}rd(x_m, v_m, -, -) \\ RS_m & \text{otherwise} \end{cases}$$

$$WS_{m+1} \triangleq \begin{cases} WS_m \cup \{x_m\} & \text{if } t_m{=}wr(x_m, v_m, -, -) \\ WS_m & \text{otherwise} \end{cases}$$

Let $\mathtt{RS}_\xi = RS_m$ and $\mathtt{WS}_\xi = WS_m$; let $\mathtt{RS}_\xi \cup \mathtt{WS}_\xi$ be enumerated as $\{x_1 \cdots x_i\}$ for some i.

- $\mathtt{lab}(log) = (\mathtt{W}, \mathtt{ws}[\xi], \mathtt{w})$, and $\mathtt{lab}(logfo) = (\mathtt{FO}, \mathtt{ws}[\xi])$.
- *PLs* denotes the sequence of events promoting the reader locks to writer ones (when the given location is in the write set), and is of the form $PL_{x_1} \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} PL_{x_i}$, where for all $n \in \{1 \cdots i\}$:

$$PL_{x_n} = \begin{cases} plw_{x_n} \xrightarrow{\text{po}} spl_{x_n} \xrightarrow{\text{po}} pl_{x_n} & \text{if } x_n \in \mathtt{WS}_\xi \\ \emptyset & \text{otherwise} \end{cases}$$

and $\mathtt{lab}(plw_{x_i}) = (\mathtt{U}, xl_i, v_i, v_i{-}1)$ for some even value v_i; pls_{x_i} denotes the sequence of reads waiting for the lock to be available (spinning), and $\mathtt{lab}(pl_{x_i}) = (\mathtt{R}, xl_i, 1)$:

- *Ws* denotes the sequence of events committing the writes of $(\!|\mathtt{T}|\!)$ and is of the form $c_{x_1} \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} c_{x_i}$, where for all $n \in \{1 \cdots i\}$:

$$c_{x_n} = \begin{cases} lr_{x_n} \xrightarrow{\text{po}} w_{x_n} \xrightarrow{\text{po}} fo_{x_n} & \text{if } x_n \in \mathtt{WS}_\xi \\ \emptyset & \text{otherwise} \end{cases}$$

and $\mathtt{lab}(lr_{x_n}) = (\mathtt{R}, \mathtt{w}[x_n], v_n)$, $\mathtt{lab}(w_{x_n}) = (\mathtt{W}, x_n, v_n)$, $\mathtt{lab}(fo_{x_n}) = (\mathtt{FO}, x_n)$, for some v_n.

- *WUs* denotes the sequence of events releasing the writer locks and is of the form $WU_{x_1} \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} WU_{x_i}$, where for all $n \in \{1 \cdots i\}$:

$$WU_{x_n} = \begin{cases} wu_{x_n} & \text{if } x_n \in \mathtt{WS}_\xi \\ \emptyset & \text{otherwise} \end{cases}$$

where $\mathtt{lab}(wu_{x_n}) = (\mathtt{W}, xl_n, 0)$.

- *RUs* denotes the sequence of events releasing the reader locks (when the given location is in the read set only) and is of the form $RU_{x_1} \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} RU_{x_i}$, where for all $n \in \{1 \cdots i\}$:

$$RU_{x_n} = \begin{cases} ru_{x_n} & \text{if } x_n \notin \mathtt{WS}_\xi \\ \emptyset & \text{otherwise} \end{cases}$$

where $\mathtt{lab}(ru_{x_n}) = (\mathtt{U}, xl_n, v_n, v_n{-}2)$ for some v_n.

Note that for all $\xi_1, \xi_2 \in T^i_{rec}$, if $\xi_1 \neq \xi_2$, then $\mathtt{WS}_{\xi_1} \cap \mathtt{WS}_{\xi_2} = \emptyset$. As such, for each location x, there is at most one write to x during the execution of the recovery $\theta_{\text{init}(i)}$. We denote this write by rec_x.

For each location $x \in \mathtt{WS}_\xi$, let fw_x denote the maximal write (in po order) logging a write for x in $\mathtt{w}[x\ ]$. That is, when $Ts = t_1 \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} t_m$, let $fw_x = \mathtt{wmax}(x, [t_1 \cdots t_m])$, where:

$$\mathtt{wmax}(x, [\,]) \text{ undefined}$$
$$\mathtt{wmax}(x, L.[t]) \triangleq \begin{cases} t.lw_x & \text{if } t{=}wr(x, -, -, -) \\ \mathtt{wmax}(x, L) & \text{otherwise} \end{cases}$$

Note that if an execution is Px86-consistent, then $(fw_{x_n}, lr_{x_n}) \in \mathrm{rf}$, for all $x_n \in \mathtt{WS}_\xi$.

In order to establish the soundness of our implementation, it suffices to show that given an Px86-consistent execution graph G of the implementation, we can construct a corresponding PSER-consistent execution graph G' with the same outcome. In era i, given a transaction ξ of thread τ_j

with code T, $\mathtt{RS}_\xi \cup \mathtt{WS}_\xi = \{x_1 \cdots x_i\}$ and trace $\theta_i(\xi)$ as above with $\theta_i(\xi).Ts = t_1 \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} t_k$, we construct the corresponding PSER execution trace $\theta'_i(\xi)$ as follows:

$$\theta'_i(\xi) \triangleq t'_1 \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} t'_k$$

where for all $m \in \{1 \cdots k\}$:

$$\begin{array}{lll} \mathtt{lab}(t'_m){=}(\mathsf{R}, x_m, v_m, \xi) & \text{when} & t_m = rd(x_m, v_m, -, -) \\ \mathtt{lab}(t'_m){=}(\mathsf{W}, x_m, v_m, \xi) & \text{when} & t_m = wr(x_m, v_m, -, -) \end{array}$$

and in the first case the identifier of t'_m is that of $\theta_i(\xi).r_{x_m}$; and in the second case the identifier of t'_m is that of $\theta_i(\xi).lw_{x_m}$. We thus define a function, $\mathtt{imp}(.)$, mapping each PSER event t'_m to its corresponding Px86 event: $\theta_i(\xi).r_{x_m}$ when $\mathtt{lab}(t'_m){=}(\mathsf{R}, x_m, v_m, \xi)$, or $\theta_i(\xi).lw_{x_m}$ when $\mathtt{lab}(t'_m){=}(\mathsf{W}, x_m, v_m, \xi)$.

We are now in a position to demonstrate the soundness of our implementation. Given an Px86-consistent execution graph G_i of the implementation in the i^{th} era, we construct a PSER execution graph G'_i as follows and demonstrate that it is PSER-consistent:

- $G'_i.E{=}G'_i.I \cup Rec \cup Run$, with $Rec \triangleq \bigcup_{\xi \in T^i_{rec}} \theta'_{i-1}(\xi).E$, $\theta'_0(-){=}\emptyset$ and $Run \triangleq \bigcup_{\xi \in T^i} \theta'_i(\xi).E$.
- $G'_i.I = \left\{ (\mathsf{W}, x, v, 0) \;\middle|\; \begin{array}{l} x \in \text{Loc} \wedge (i = 0 \Rightarrow v = 0) \wedge \\ (i > 0 \Rightarrow \exists e \in \max\left(\text{nvo}_i|_{G'_{i-1}.P \cap W_x}\right).\ \mathtt{val_w}(e){=}v; \end{array} \right\}$
- $G'_i.P = G'_i.I \cup PRec \cup \bigcup_{\xi \in T^i} \mathsf{p}(\xi)$, where:

$$PRec \triangleq \begin{cases} Rec & \theta^p_{\text{init}_i}{=}\theta_{\text{init}_i} \wedge \theta_{\text{init}_i}.E \cap D \subseteq G_i.P \\ \emptyset & \text{otherwise} \end{cases}$$

$$\mathsf{p}(\xi) \triangleq \begin{cases} \theta'_i(\xi).E & \text{if } \theta_i(\xi).E \cap D \subseteq G_i.P \\ \emptyset & \text{otherwise} \end{cases}$$

- $G'_i.\text{po} = G'_i.I \times (G'_i.E \setminus G'_i.I) \cup (Rec \times Run)_i \cup G.\text{po}|_{G'.E}$
- $G'_i.\text{rf} = \bigcup_{\xi \in T^i} \mathsf{RF}_\xi \cup \bigcup_{\xi \in T^i_{rec}} \mathsf{RF}'_\xi$
- $G'_i.\text{mo} = \left(G'_i.I \times ((G'_i.E \setminus G'_i.I) \cap W)\right)_{loc} \cup ((Rec \cap W) \times (Run \cap W))_{loc} \cup \left\{(e, e') \;\middle|\; \exists x.\ e, e' \in W_x \cap Rec \wedge \mathtt{tx}(e){=}\mathtt{tx}(e') \wedge (e, e') \in G'_i.\text{po}\right\} \cup \text{MO}$
- $G'_i.\text{nvo} = G'_i.I \times ((G'_i.E \setminus G'_i.I) \cap D) \cup \left\{(e, e') \;\middle|\; e, e' \in G'_i.I \cap D \wedge \mathtt{id}(e) < \mathtt{id}(e')\right\} \cup ((Rec \cap D) \times (Run \cap D)) \cup \left\{(e, e') \;\middle|\; e, e' \in G'_i.D \cap Rec \wedge (e, e') \in G'_i.\text{st} \cap \text{po}\right\} \cup \left\{(e, e') \;\middle|\; e, e' \in G'_i.Rec \cap D \wedge (e, e') \notin G'_i.\text{st} \wedge (e, e') \in G'_i.\text{hb}\right\} \cup \left\{(e, e') \;\middle|\; e, e' \in G'_i.Rec \cap D \wedge (e, e') \notin G'_i.\text{st} \cup \text{hb} \wedge \mathtt{tx}(e) \prec \mathtt{tx}(e')\right\} \cup \text{NVO}$

where $\prec$ denotes a strict total order on transaction identifiers (e.g. natural number ordering), and:

$$\mathsf{RF}_\xi \triangleq \left\{(t'_k, t'_j) \;\middle|\; \begin{array}{l}\exists x, v, \xi.\ \mathtt{lab}(t'_j){=}(\mathsf{R}, x, v, \xi) \wedge \mathtt{lab}(t'_k){=}(\mathsf{W}, x, v, \xi)\\ \wedge (t_k.wl_x, t_j.r_x) \in G.\mathsf{rf}\end{array}\right\}$$
$$\cup \left\{(t'_k, t'_j) \;\middle|\; \begin{array}{l}\exists x, v, \xi, \xi'.\ \mathtt{lab}(t'_j){=}(\mathsf{R}, x, v, \xi) \wedge \mathtt{lab}(t'_k){=}(\mathsf{W}, x, v, \xi') \wedge \xi \neq \xi'\\ \wedge\ t_k = \theta_i(\xi').fw_x \wedge (\theta_i(\xi').w_x, \theta_i(\xi).t_j.r_x) \in G.\mathsf{rf}\end{array}\right\}$$

$$\mathsf{RF'}_\xi \triangleq \left\{(w, r) \;\middle|\; \mathtt{tx}(r){=}\xi \wedge (w, r) \in G'_{i-1}.\mathsf{rf} \wedge \mathtt{tx}(w){=}\mathtt{tx}(r)\right\}$$
$$\cup \left\{(w_0, r) \;\middle|\; \begin{array}{l}\mathtt{tx}(r){=}\xi \wedge \mathtt{loc}(r){=}\mathtt{loc}(w_0) \wedge w_0 \in G'_i.I\\ \wedge\ \exists w.\ (w, r) \in G'_{i-1}.\mathsf{rf} \wedge \mathtt{tx}(w) \neq \mathtt{tx}(r)\end{array}\right\}$$

$$\mathsf{MO} \triangleq \left\{(t'_k, t'_j) \;\middle|\; \mathtt{tx}(t'_k) = \mathtt{tx}(t'_j) \wedge \mathtt{loc}(t'_k){=}\mathtt{loc}(t'_j) \wedge t'_k, t'_j \in W \wedge (t_k, t_j) \in G.\mathrm{po}\right\}$$
$$\cup \left\{(t'_k, t'_j) \;\middle|\; \begin{array}{l}t'_k, t'_j \in W \wedge \exists x, \xi_k, \xi_j.\ \mathtt{loc}(t'_k){=}\mathtt{loc}(t'_j){=}x\\ \wedge\ t_k \in \theta_i(\xi_k) \wedge t_j \in \theta_i(\xi_j) \wedge (\theta_i(\xi_k).c_x, \theta_i(\xi_j).c_x) \in G.\mathsf{mo}\end{array}\right\}$$

$$\mathsf{NVO} \triangleq \left\{(t'_k, t'_j) \;\middle|\; \mathtt{tx}(t'_k) = \mathtt{tx}(t'_j) \wedge t'_k, t'_j \in D \wedge (t_k, t_j) \in G.\mathrm{po}\right\}$$
$$\cup \left\{(t'_k, t'_j) \;\middle|\; \begin{array}{l}t'_k, t'_j \in W \wedge \exists x, y, \xi_k, \xi_j.\ \mathtt{loc}(t'_k){=}x \wedge \mathtt{loc}(t'_j){=}y\\ \wedge\ t_k \in \theta_i(\xi_k) \wedge t_j \in \theta_i(\xi_j) \wedge (\theta_i(\xi_k).c_x, \theta_i(\xi_j).c_y) \in G.\mathsf{nvo}\end{array}\right\}$$

Lemma 11. *Given an Px86-consistent execution graph G of the implementation and its corresponding PSER execution graph G' constructed as above, for all a, b, ξ_a, ξ_b, x:*

$$\xi_a \neq \xi_b \wedge \xi_a \neq 0 \wedge \xi_a \notin T_{rec} \wedge a \in \theta'(\xi_a) \wedge b \in \theta'(\xi_b) \wedge \mathtt{loc}(a) = \mathtt{loc}(b) = x \Rightarrow$$

$$((a, b) \in G'.\mathsf{rf} \Rightarrow \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x) \tag{98}$$

$$\wedge\ ((a, b) \in G'.\mathsf{mo} \Rightarrow \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x) \tag{99}$$

$$\wedge\ \big((a, b) \in G'.\mathsf{rb} \Rightarrow (x \in \mathtt{WS}_{\xi_a} \wedge \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x) \tag{100}$$
$$\vee\ (x \notin \mathtt{WS}_{\xi_a} \wedge \theta(\xi_a).ru_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x)\big)$$

Proof. Pick an arbitrary Px86-consistent execution graph G of the implementation and its corresponding PSER execution graph G' constructed as above. Pick an arbitrary a, b, ξ_a, ξ_b, x such that $\xi_a \neq \xi_b$, $\xi_a \neq 0$, $\xi_a \notin T_{rec}$, $a \in \theta'(\xi_a)$, $b \in \theta'(\xi_b)$, and $\mathtt{loc}(a) = \mathtt{loc}(b) = x$.

RTS. (98)
Assume $(a, b) \in G'.\mathsf{rf}$. Since $\xi_a \neq 0$, we know that $\xi_b \notin T_{rec}$. As such, from the definition of $G'.\mathsf{rf}$ we then know $(\theta(\xi_a).w_x, \theta(\xi_b).r_x) \in G.\mathsf{rf}$. On the other hand, from the properties of MRSW locks we know that either i) $x \in \mathtt{WS}_{\xi_b}$ and $\xi_b.wu_x \xrightarrow{G.\mathsf{tso}} \xi_a.rl_x$; or ii) $x \notin \mathtt{WS}_{\xi_b}$ and $\xi_b.ru_x \xrightarrow{G.\mathsf{tso}} \xi_a.pl_x$; or iii) $\xi_a.wu_x \xrightarrow{G.\mathsf{tso}} \xi_b.rl_x$.

In case (i) we then have $\xi_a.w_x \xrightarrow{G.\mathsf{rf}} \xi_b.r_x \xrightarrow{G.\mathrm{po}} \xi_b.wu_x \xrightarrow{G.\mathsf{tso}} \xi_a.rl_x \xrightarrow{G.\mathrm{po}} \xi_a.w_x$. From the Px86-consistency of the execution we have $G.\mathsf{rf} \subseteq G.\mathrm{po} \cup G.\mathsf{tso}$. There are now two cases to consider: a) ξ_a and ξ_b are in the same thread; or b) ξ_a and ξ_b are in the different threads. In case (i.a) from the Px86-consistency of the execution we have $\xi_a.w_x \xrightarrow{G.\mathrm{po}} \xi_b.r_x \xrightarrow{G.\mathrm{po}} \xi_b.wu_x \xrightarrow{G.\mathrm{po}} \xi_a.rl_x \xrightarrow{G.\mathrm{po}} \xi_a.w_x$. That is, we have $\xi_a.w_x \xrightarrow{G.\mathrm{po}} \xi_a.w_x$, contradicting the assumption that G is Px86-consistent. In case (i.b) from the Px86-consistency of the execution we have $\xi_a.w_x \xrightarrow{G.\mathsf{tso}} \xi_b.r_x \xrightarrow{G.\mathsf{tso}} \xi_b.wu_x \xrightarrow{G.\mathsf{tso}} \xi_a.rl_x \xrightarrow{G.\mathsf{tso}} \xi_a.w_x$. That is, we have $\xi_a.w_x \xrightarrow{G.\mathsf{tso}} \xi_a.w_x$, contradicting the assumption that G is Px86-consistent.

Similarly in case (ii) we have $\xi_a.w_x \xrightarrow{G.\mathsf{rf}} \xi_b.r_x \xrightarrow{G.\mathrm{po}} \xi_b.ru_x \xrightarrow{G.\mathsf{tso}} \xi_a.pl_x \xrightarrow{G.\mathrm{po}} \xi_a.w_x$. Again there are now two cases to consider: a) ξ_a and ξ_b are in the same thread; or b) ξ_a and ξ_b are in the

different threads. In case (ii.a) from the Px86-consistency of the execution we have $\xi_a.w_x \xrightarrow{G.\mathsf{po}} \xi_b.r_x \xrightarrow{G.\mathsf{po}} \xi_b.ru_x \xrightarrow{G.\mathsf{po}} \xi_a.pl_x \xrightarrow{G.\mathsf{po}} \xi_a.w_x$. That is, we have $\xi_a.w_x \xrightarrow{G.\mathsf{po}} \xi_a.w_x$, contradicting the assumption that G is Px86-consistent. In case (ii.b) from the Px86-consistency of the execution we have $\xi_a.w_x \xrightarrow{G.\mathsf{tso}} \xi_b.r_x \xrightarrow{G.\mathsf{tso}} \xi_b.ru_x \xrightarrow{G.\mathsf{tso}} \xi_a.pl_x \xrightarrow{G.\mathsf{tso}} \xi_a.w_x$ That is, we have $\xi_a.w_x \xrightarrow{G.\mathsf{tso}} \xi_a.w_x$, contradicting the assumption that G is Px86-consistent.

In case (iii) the desired result holds immediately.

RTS. (99) and (100)

The proofs of these parts are analogous and are omitted here. □

Lemma 12. *Given an Px86-consistent execution graph G of the implementation and its corresponding PSER execution graph G' constructed as above, for all a, b:*

$$(a,b) \in G'.\mathsf{hb} \wedge a \notin G'.I \cup Rec \Rightarrow (\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{tso}$$

Proof. Let $G'.\mathsf{hb}^1 \triangleq G'.\mathsf{po}_\mathsf{T} \cup \mathsf{rf}_\mathsf{T} \cup \mathsf{mo}_\mathsf{T} \cup \mathsf{rb}_\mathsf{T}$, and $G'.\mathsf{hb}^{n+1} \triangleq G'.\mathsf{hb}^1; G'.\mathsf{hb}^n$, for all $n > 1$. We then show the following equivalent result:

$$\forall n \in \mathbb{N}^+.\ (a,b) \in G'.\mathsf{hb}^n \wedge a \notin G'.I \cup Rec \Rightarrow (\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{tso}$$

We proceed by induction on n.

Base case $n = 1$

Pick arbitrary a, b such that $(a,b) \in G'.\mathsf{hb}^1$ and $a \notin G'.I \cup Rec$. Given the definition of hb^1, we thus know that either: i) $(a,b) \in G'.\mathsf{po}_\mathsf{T}$; or ii) $(a,b) \in G'.\mathsf{rf}_\mathsf{T}$; or iii) $(a,b) \in G'.\mathsf{mo}_\mathsf{T}$; or iv) $(a,b) \in G'.\mathsf{rb}_\mathsf{T}$.

In case (i), we know that $a, b \in W \cup R$ and thus $\mathtt{imp}(a), \mathtt{imp}(b) \in W \cup R$. There are two cases to consider: a) $(a,b) \notin W \times R$; or b) $(a,b) \in W \times R$. In case (i.a) we have $(\mathtt{imp}(a), \mathtt{imp}(b)) \notin W \times R$. From the construction of G' we have $(\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{po}$ and thus since $(\mathtt{imp}(a), \mathtt{imp}(b)) \notin W \times R$, from Px86-consistency of G we have $(\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{tso}$.

In case (i.b) let $\mathtt{loc}(a)=x$ and $\mathtt{loc}(b)=y$. We then know $(\mathtt{imp}(a), \mathtt{imp}(b)) \in W \times R$,$\mathtt{loc}(\mathtt{imp}(a))=x$ and $\mathtt{loc}(\mathtt{imp}(b))=y$. From the structure of G we then know that there exists ξ_a, ξ_b such that $\mathtt{imp}(a) \xrightarrow{G.\mathsf{po}} \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{po}} \theta(\xi_b).rl_y \xrightarrow{G.\mathsf{po}} \mathtt{imp}(b)$. Moreover, since $\theta(\xi_a).wu_x \in W$, $\theta(\xi_b).rl_y \in U$ and $\mathtt{imp}(b) \in R$, from Px86-consistency of G we have $\mathtt{imp}(a) \xrightarrow{G.\mathsf{tso}} \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_y \xrightarrow{G.\mathsf{tso}} \mathtt{imp}(b)$. That is, we have $(\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{tso}$, as required.

In case (ii), we know there exists ξ_a, ξ_b such that $\xi_a \neq \xi_b$, $\xi_a \neq 0$, $\xi_a \notin T_{rec}$, $a \in \theta'(\xi_a)$ and $b \in \theta'(\xi_b)$. As such, from Lemma 11 we have $\theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x$. We thus have $\mathtt{imp}(a) \xrightarrow{G.\mathsf{po}} \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x \xrightarrow{G.\mathsf{po}} \mathtt{imp}(b)$. As such, since $\xi_a.wu_x \in W$, $\mathtt{imp}(a), \mathtt{imp}(b) \in R \cup W$ and $\theta(\xi_b).rl_x \in U$, we have $\mathtt{imp}(a) \xrightarrow{G.\mathsf{tso}} \theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x \xrightarrow{G.\mathsf{tso}} \mathtt{imp}(b)$. That is, we have $(\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{tso}$.

The proof of cases (iii-iv) cases are analogous and are omitted here.

Inductive case $n = m+1$ **for** $m > 0$

Pick arbitrary a, b such that $(a,b) \in G'.\mathsf{hb}^n$ and $a \notin G'.I \cup Rec$. That is, there exists c, ξ_c such that $(a,c) \in G'.\mathsf{hb}^1$, $(c,b) \in G'.\mathsf{hb}^m$ and $c \in \theta'(\xi_c)$. From the proof of the base case we then have $(\mathtt{imp}(a), \mathtt{imp}(c)) \in G.\mathsf{tso}$. Moreover, given the construction of G' and since $\xi_a \neq 0$, and $\xi_a \notin T_{rec}$, we know that $\xi_c \neq 0$, and $\xi_c \notin T_{rec}$. As such, from the inductive hypothesis we have $(\mathtt{imp}(c), \mathtt{imp}(b)) \in G.\mathsf{tso}$. As $(\mathtt{imp}(a), \mathtt{imp}(c)) \in G.\mathsf{tso}$ and $(\mathtt{imp}(c), \mathtt{imp}(b)) \in G.\mathsf{tso}$, we thus have $(\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{tso}$, as required.

□

Lemma 13 (Implementation soundness). *For all Px86-consistent execution graphs G of the implementation and their counterpart PSER execution graphs G' constructed as above:*

$$G'.\mathsf{hb} \text{ is irreflexive} \tag{101}$$
$$G'.\mathsf{hb} \cap (D \times D) \subseteq G'.\mathsf{nvo} \tag{102}$$
$$dom(G'.[D]; \mathsf{st}; [P]) \subseteq G'.P \subseteq G'.T \tag{103}$$
$$G'.\mathsf{nvo}_{\mathsf{T}} \text{ is acyclic} \tag{104}$$

Proof. Pick an arbitrary Px86-consistent execution G of the implementation and its counterpart PSER execution graphs G' constructed as above.

Parts (103) and (104) follow from the construction of G'.

RTS. (101)
We proceed by contradiction. Let assume that there exists a such that $(a, a) \in G'.\mathsf{hb}$. Note that given the construction of G', we know that the initialisation events in $G'.I$ have no incoming $G'.\mathsf{po} \cup \mathsf{rf} \cup \mathsf{mo} \cup \mathsf{rb}$ edges, and as such this cycle contains *no initialisation events in* $G'.I$; in particular, $a \notin G'.I$ and thus $\mathtt{tx}(a) \neq 0$. Moreover, since the only incoming $G'.\mathsf{po} \cup \mathsf{rf} \cup \mathsf{mo} \cup \mathsf{rb}$ edges to the events in $G'.Rec$ are those from the initialisation events in $G'.I$, and since this cycle contains no initialisation events, we also know that this cycle contains no events from $G'.Rec$. That is, $a \notin G'.Rec$. As such, from Lemma 12 we have $(\mathtt{imp}(a), \mathtt{imp}(a)) \in G.\mathsf{tso}$, contradicting our assumption that G is Px86-consistent.

RTS. (102)
Pick an arbitrary a, b such that $(a, b) \in G'.\mathsf{hb}$ and $a, b \in G'.D$; that is, $a, b \in W$. Let $\mathtt{loc}(a) = x$ and $\mathtt{loc}(b) = y$. There are now three cases to consider: i) $a \in G'.I$; or ii) $a \in G'.Rec$; or iii) $a \in G'.Run$.

In case (i), given the construction of G', we know that the initialisation events in $G'.I$ have no incoming $G'.\mathsf{po} \cup \mathsf{rf} \cup \mathsf{mo} \cup \mathsf{rb}$ edges, and thus we know that $b \notin G'.I$. Consequently, from the construction of G' we have $(a, b) \in G'.\mathsf{nvo}$.

In case (ii), given the construction of G', we know that the only outgoing $G'.\mathsf{po} \cup \mathsf{rf} \cup \mathsf{mo} \cup \mathsf{rb}$ edges of events in Rec is to events in $Rec \cup Run$. As such, we know that $b \in G'.Rec \cup Run$. Consequently, from the construction of G' we have $(a, b) \in G'.\mathsf{nvo}$.

In case (iii), given the construction of G', we know that the only outgoing $G'.\mathsf{po} \cup \mathsf{rf} \cup \mathsf{mo} \cup \mathsf{rb}$ edges of events in Run is to events in Run. As such, we know that $b \in G'.Run$. It is then straightforward to demonstrate from part (101) that $\mathtt{tx}(a) \neq \mathtt{tx}(b)$. That is, there exists ξ_a, ξ_b such that $\xi_a \neq \xi_b$, $a \in \theta'(\xi_a)$ and $b \in \theta'(\xi_b)$. There are now four cases to consider: a) $(a, b) \in G'.\mathsf{po}$; or b) $(a, b) \in G'.\mathsf{rf}$; or c) $(a, b) \in G'.\mathsf{mo}$; or d) $(a, b) \in G'.\mathsf{rb}$.

In case (a) we know there exist $sf \in SF, fo \in FO$ such that $\mathtt{loc}(fo) = \mathtt{loc}(\mathtt{imp}(a))$, and $\mathtt{imp}(a) \xrightarrow{G.\mathsf{po}} fo \xrightarrow{G.\mathsf{po}} sf \xrightarrow{G.\mathsf{po}} \mathtt{imp}(b)$; thus from the Px86-consistency of G we have: $(\mathtt{imp}(a), \mathtt{imp}(b)) \in G.\mathsf{nvo}$. Consequently, from the definition of G' we have $(a, b) \in G'.\mathsf{nvo}$.

In case (b) from Lemma 11 we have $\theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x$. Moreover, we know there exist $sf \in SF, fo \in FO$ such that $\mathtt{loc}(fo) = \mathtt{loc}(\mathtt{imp}(a))$, and $\mathtt{imp}(a) \xrightarrow{G.\mathsf{po}} fo \xrightarrow{G.\mathsf{po}} sf \xrightarrow{G.\mathsf{po}} \theta(\xi_a).wu_x$. As such, from the Px86-consistency of G we have: $(\mathtt{imp}(a), \theta(\xi_a).wu_x) \in G.\mathsf{nvo}$. Moreover, from the Px86-consistency of G and since $\theta(\xi_a).wu_x \xrightarrow{G.\mathsf{tso}} \theta(\xi_b).rl_x$, we have $\theta(\xi_a).wu_x \xrightarrow{G.\mathsf{mo}} \theta(\xi_b).rl_x$ and thus $\theta(\xi_a).wu_x \xrightarrow{G.\mathsf{nvo}} \theta(\xi_b).rl_x$. As such, we have $(\mathtt{imp}(a), \theta(\xi_a).wu_x) \in G.\mathsf{nvo}$. Consequently, from the definition of G' we have $(a, b) \in G'.\mathsf{nvo}$.

Proof of cases (c-d) are analogous and are omitted here. □

D PERSISTENT MICHAEL–SCOTT QUEUE LIBRARY

In Fig. 14 we present a *persistent* variant of the lock free Michael–Scott (MS) queue [Michael and Scott 1996] implementation (left) and its recovery mechanism (right) in the PTSO language. For simplicity, in our variant of the Michael–Scott queue we do not track the *tail* pointer.

For simplicity, the queue contents are stored as an array that may grow dynamically. A queue at q comprises two components, represented as two adjacent cells: (i) the queue contents at q, written $q.\texttt{data}$, recording the location of the contents array; and (ii) the queue head at q+1, written $q.\texttt{head}$.

We assume that client programs are of the form $\texttt{C}_0||\cdots||\texttt{C}_k$; that each $\texttt{C}_i$ is of the form $o_0^i;\cdots;o_l^i$, where each o_j^i is a library operation (enq or deq); We thus represent each $\texttt{C}_i$ as an array $\texttt{C}_i$ of length l+1, with each $\texttt{C}_i[j] = o_j^i$. We then represent P as an array of length k+1 at location P, with $\texttt{P}[i] = \texttt{C}_i$.[2] A client program P is executed by calling run(P). A call to run(P) spawns k+1 threads $\tau_0\cdots\tau_k$ and sets up their contexts, with each τ_i executing $\texttt{C}_i$. We further assume that the context of each thread τ_i is set up such that: (1) a call to getTID()returns i; and (2) a call to getPC()returns the *'progress counter'* (or 'program counter'), namely the index of the counter operation in $\texttt{C}_i$ currently under execution (i.e. j when executing o_j^i). To ensure correct recovery, the metadata for tracking the progress of each thread is recorded in a map at *map*.

Initialisation. The start() commences the execution of the client program stored at location P by initialising the metadata necessary for crash recovery. It thus creates a new (empty) queue at q, together with a recovery map of the relevant size (the number of threads in P) at *map*, and launches the execution by calling run(P). When the i^{th} thread contains l+1 instructions ($\texttt{P}[i].\texttt{size} = l$+1), then its associated *map* entry (i.e. $map[i]$) is an array of length l+1, with one entry per instruction. For each i^{th} thread τ_i the $map[i]$ entry is initialised with a $\bot$-instantiated array of the appropriate size (i.e. $\texttt{P}[i].\texttt{size}$) to denote that τ_i has made no progress as of yet. The **sfence** on line 46 ensures that if the execution of start() crashes, then recovery does not observe a partially initialised *map*.

Queue Operations. A call to enq(v) creates a new node n with value v, traverses the queue starting at the head q.head until it finds an empty (null) entry, and inserts the new node n at this location using an atomic **CAS**. Analogously, a call to deq() retrieves the head entry at $q.\texttt{head}$ (which may hold nullwhen the queue is empty) in n and returned. If n is not null (the queue is not empty), the head index is duly incremented by one.

Persistence of Queue Operations. Recall that we track the progress of each thread in *map* to ensure correct crash recovery. In particular, when τ_i executes its j^{th} operation, *prior* to carrying out the relevant queue update, it updates $map[i][j].\texttt{node}$ to n, where n denotes the node being added or removed. This is done on lines 4 and 14 of enq and deq, where the subsequent **sfence** instructions (lines 4 and 18) ensure that the thread metadata does not lag behind its progress.

Upon recovery, the progress of thread τ is assessed by calling $\texttt{getProgress}(\tau)$ on line 53. A call to $\texttt{getProgress}(\tau)$ traverses the array at $map[\tau]$ in order to locate the latest non-$\bot$ value. That is, if $\texttt{getProgress}(\tau)$ returns (j,n,a) then: (1) the effects of the first pc−1 operations of τ have persisted prior to the last crash; (2) the pc^{th} operation of τ was attempting to enqueue/dequeue node n; and (3) the effect of this pc^{th} operation may or may not have persisted prior to the last crash. As such, if $\texttt{getProgress}(\tau)$ returns $(j,\texttt{n})$ and o_j^τ (the j^{th} operation of τ) is a deq, node n may or may not have been removed by τ when the crash occurred. One can then inspect the queue to ascertain whether the execution of o_j^i was completed and persisted. If n is in the queue, then the

[2] Note that we do not make assumptions about the thread *IDs*; nor do we assume that recovery restores the same threads (with same IDs). Rather, as the number of threads in P is known in advance, each thread is distinguished by its index in P.

```
1. q.enq(v) ≜
2.   pc:=getPC(); τ:=getTID();
3.   n:=newNode(v,τ,pc);
4.   map[τ][pc].node:=fo n; sfence;
5.   h:=q.head;
6.   while(q.data[h] != null)
7.     h:=h+1;
8.   if (!CASfo(q.data[h],null,n))
9.     goto line 6;
10.  sfence;

11. q.deq() ≜
12.  pc:=getPC(); τ:=getTID();
13.  h:=q.head; n:=q.data[h];
14.  map[τ][pc].node:=fo n;
15.  if (n != null) {
16.    τ':=n.t; pc':=n.pc;
17.    map[τ'][pc'].done:=fo ⊤;
18.  } sfence;
19.  if (n != null) {
20.    if (!CASfo(q.head,h,h+1))
21.      goto line 13;
22.    sfence;
23.    map[τ][pc].done:=fo ⊤; sfence
24.  } return n;

25. rem(n) ≜
26.  for(τ ∈ P){
27.    pc:=0
28.    while(map[τ][pc].node!=⊥){
29.      m:=map[τ][pc].node;
30.      a:=map[τ][pc].done;
31.      if (n=m&&a=⊤) return 1;
32.      pc++;
33.  } }
34.  return 0;

35. isIn(q,n) ≜
36.  h:=q.head; c:=q.data[h];
37.  while(c != null) {
38.    if (n=c) return true;
39.    else { h:=h+1; c:=q.data[h]; }
40.  } return false;

41. start(P) ≜
42.  lq:=newQueue();
43.  s:=P.size; lmap:=newMap(s);
44.  for(τ ∈ P)
45.    lmap[t]:=newArray(P[τ].size,⊥);
46.  sfence;
47.  q:=lq; map:=lmap; run(P);

48. recover(P) ≜
49.  if (q=null || map=null)
50.    start();
51.  for(τ ∈ P) enq[τ]:=-1;
52.  for(τ ∈ P) {
53.    (pc,n,a):=getProgress(τ);
54.    if (pc>=0 && isDeq(P[τ][pc])) {
55.      if (n=null)
56.        P'[τ]:=sub(P[τ],pc+1);
57.      else {
58.        if (a=⊤)
59.          P'[τ]:=sub(P[τ],pc+1);
60.        else if (inIn(q,n)||rem(n))
61.          P'[τ]:=sub(P[τ],pc);
62.        else {
63.          P'[τ]:=sub(P[τ],pc+1);
64.          map[τ][pc].done:=fo ⊤}
65.        τ':=n.t; pc':=n.pc;
66.        enq[τ']:=max(enq[τ'],pc'+1);}
67.    } else if (pc<0) P'[τ]:=P[τ]; }
68.  for(τ ∈ P) {
69.    (pc,n,a):=getProgress(τ);
70.    if (pc>=0 && isEnq(P[τ][pc])) {
71.      if (pc < enq[τ])
72.        P'[τ]:=sub(P[τ],enq[τ]);
73.      else if (a==⊤||isIn(q,n))
74.        P'[τ]:=sub(P[τ],pc+1);
75.      else
76.        P'[τ]:=sub(P[τ],pc); }
77.  } sfence;
78.  run(P');

79. getProgress(τ) ≜
80.  pc:=-1; n:=⊥; a:=⊥;
81.  while(map[τ][pc+1].node!=⊥) pc++;
82.  if (pc>=0) {
83.    n:=map[τ][pc].node;
84.    a:=map[τ][pc].done;
85.  } return (pc,n,a);
```

Fig. 14. A persistent Michael–Scott queue implementation and its recovery mechanism in Px86

crash occurred before the removal of n was persisted and thus recovery must resume executing τ_i from o^i_j. On the other hand, if n is not in the queue, then recovery must resume τ_i from o^i_{j+1}. Similarly, if o^i_j is an enq, one can *in most cases* determine the progress of τ_i by inspecting the queue. If n is in the queue, then the crash occurred after the insertion of n was persisted and thus recovery must resume τ_i from o^i_{j+1}. However, if n is not in the queue, it may be the case that τ_i added n to the queue, while another thread later removed n from the queue, prior to the crash.

To understand this better, consider P=q.enq(v)||(q.deq();o^1_1;o^1_2). Let us suppose thread τ_0 executing enq(v) adds v to the queue and thus sets *map*[0][0].node to n for some n with value v. Thread τ_1 later executes deq() and removes n from the queue, and subsequently crashes while executing o^1_2. Let us assume that all writes persisted before the crash, i.e. *map*[0][0].node=n. In this scenario, even though the execution of τ_0 was finalised and fully persisted, we cannot ascertain this by simply inspecting the queue, as n is removed by τ_1.

To remedy this, the deq operations must *help* advance the progress of enq operations. That is, when removing a node n, we can confirm that n was indeed added to the queue, and thus the progress of the thread responsible for inserting it must be advanced accordingly. To this end, for each node n added to the queue, the representation of n additionally records the metadata of the thread responsible for adding it to the queue. More concretely, when the j^{th} operation of τ adds node n to the queue, as part of its representation n records: 1) the thread τ at location n+1, written n.t; and 2) the operation index j at location n+2, written n.pc. When removing n via deq, the implementation updates the current progress of the thread responsible for inserting n (i.e. n.t) in *map* if necessary (lines 15-17). That is, when n.t = τ and n.pc = j, as τ has successfully enqueued n via its j^{th} operation, its current recorded progress in *map*[i][j].done is updated to the designated value $\top$, to indicate that the insertion of n is indeed successful. As we describe shortly, upon recovery, when *map*[τ][j].done = $\top$ and o^i_j (the j^{th} operation of τ) is an enqueue operation, we can infer that the effect of o^i_j has persisted successfully and can thus advance the progress of τ accordingly. In the example above, this ensures that τ_1 sets *map*[0][0].done to $\top$ when removing n, thus ensuring that recovery realises the completion of τ_0 operations.

Lastly, the **sfence** instructions on lines 10 and 23 ensure that the thread progress does not lag behind its recovery metadata in *map*.

Recovery. The recovery mechanism of a queue client program at location P is triggered by calling recover(P). The first two lines ensure that q and *map* have been initialised; otherwise start(P) is called. As discussed above, the deq calls help advance the progress of their counterpart enq calls. Analogously, the recovery program can also use the progress of deq calls prior to crash to restore the progress of enq calls correctly. To this end, the enq array (initialised on line 51) tracks the progress of enq calls as observed by deq calls. The recovery mechanism then restores the progress of threads by generating a new program P', where each P'[τ] entry is a *suffix* of the original program in P[τ]. This restoration is done in two passes: first for threads executing a deq operation prior to crash (lines 52-67), and then for those executing an enq (68-77).

Recall that the progress of thread τ prior to crash can be ascertained by calling getProgress(τ). For each dequeuing thread τ, when getProgress(τ) returns (pc,n), if n=null (the queue was empty when τ attempted a deq) then its effect has (trivially) persisted and thus its progress can be advanced to pc+1. This is done on line 56 by setting P[τ] to sub(P[τ],pc+1), i.e. the subarray of P[τ] starting at pc+1. On the other hand if n≠null, then the effect of τ (removing n) may or may not have persisted. Recall that to determine the progress of τ one can inspect the queue to ascertain whether it contains n. This is done by calling isIn(q,n). As discussed above, the τ progress can be restored accordingly to either pc when n is still in the queue (line 61), or pc+1 when n is not in the queue (line 63). In both cases, we can confirm that the thread responsible for enqueuing n has

persisted past the operation inserting n. When n.t=τ' and n.pc=pc', the enq[τ'] entry is thus set to the maximum value observed for τ' so far, i.e. max(enq[τ'],pc'+1) – see line 66.

For each enqueuing thread τ, when getProgress(τ) returns (pc,n,a), if the progress recorded for τ lags behind that observed by dequeuing operations (pc<enq[τ]), then progress is duly set to enq[τ] on line 72. On the other hand, if the progress is not lagging, then the effect of τ (adding n) may or may not have persisted. Inspecting the queue, one can then restore the τ progress accordingly to either pc+1 when n is in the queue (line 74), or pc when n is not in the queue (line 76). Moreover, recall that dequeuing threads help advance the progress of enqueuing threads by updating the relevant entry to the designated value $\top$. As such, when a=$\top$ (line 73), we can deduce that the node inserted by the pc^{th} operation has been removed by a dequeuing thread prior to the crash, and thus the progress of τ can be advanced to pc+1 accordingly.

Lastly, for each thread τ, when getProgress(τ) returns (pc,n,a), observe that when pc<0 then τ has made no progress prior to the crash and hence it must execute P[τ] from the start (line 67).

Persistent Linearisability of the Implementation in Fig. 14. The linearisation point of enq is on line 8; the deq has two linearisation points depending on q.data: (i) if q.data is empty, the linearisation point is on line 13; (ii) if q.data is not empty, the linearisation point is on line 20. To show that an execution era G of our implementation is persistently linearisable, we construct the E_c and E_t sets using the linearisation.

Note that the linearisation points of enq operations, as well as those of deq in case (ii) above, are *write* and *update* instructions and are thus ordered by the total-store-order G.tso. We can then construct a sequential history θ as an enumeration of the library events such that the order between their linearisation points is respected. That is, θ is of the form $inv_1; ack_1; \cdots; inv_m; ack_m$, where for all $i, j \in \{1 \cdots m\}$ we have: $i < j$ iff the linearisation point associated with (inv_i, ack_i) is tso-ordered before that of (inv_j, ack_j).

Lastly, we demonstrate that the combined histories of execution eras form a legal queue history as given in [Raad and Vafeiadis 2018]. We present the persistent linearisability of our implementation in Thm. 7 below together with its full proof.

D.1 Soundness of the Persistent Michael–Scott Queue Library

For an arbitrary program P and a Px86-valid execution $C = G_1, \cdots, G_n$ of P with $G_i = (E, I, P, \mathrm{po}, \mathrm{rf}, \mathrm{mo}, \mathrm{nvo})$, let G_i.tso=tso. Observe that when P comprises k threads, the trace of each execution era comprises two stages: i) the trace of the *setup* stage by the master thread τ_0 performing initialisation or recovery, prior to the call to run(P); followed (in po order) by ii) the trace of each of the constituent program threads $\tau_1 \cdots \tau_k$, provided that the execution did not crash during the setup stage.

Thanks to the placement of **sfence** instructions, for each thread τ_j, we know that the set of persistent events in execution era i, namely P_i, contains roughly a *prefix* (in po order) of thread τ_j's trace. More concretely, for each constituent thread $\tau_j \in \{\tau_1 \cdots \tau_k\} = dom(\mathsf{P})$, there exist $P_j^1 \cdots P_j^n$ such that:

1) $\mathsf{P}[\tau_j] = o_j^0; \cdots; o_j^{P_j^1}; o_j^{P_j^1+1}; \cdots o_j^{P_j^2}; \cdots; o_j^{P_j^{n-1}+1}; \cdots; o_j^{P_j^n}$, comprising enq and deq operations;

2) at the beginning of each execution era $i \in \{1 \cdots n\}$, the program executed by thread τ_j (calculated in P' and subsequently executed by calling run(P')) is that of sub(P[τ_j],P_j^{i-1}+1), where $P_j^0 = -1$, for all j; and

3) in each execution era $i \in \{1 \cdots n\}$, the trace $\theta_{(i,j)}$ of each constituent thread $\tau_j \in dom(\mathrm{P})$ is of the following form:

$$\begin{aligned}\theta_{(i,j)} \triangleq\ & \theta(o_j^{P_j^{i-1}+1}, \tau_j, P_j^{i-1}{+}1, n_j^{P_j^{i-1}+1}, e_j^{P_j^{i-1}+1}) \\ & \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} \theta(o_j^{P_j^{i}}, \tau_j, P_j^{i}, n_j^{P_j^{i}}, e_j^{P_j^{i}}) \\ & \xrightarrow{\text{po}} \theta(o_j^{P_j^{i}+1}, \tau_j, P_j^{i}{+}1, n_j^{P_j^{i}+1}, e_j^{P_j^{i}+1}) \\ & \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} \theta(o_j^{m_j^{i}-1}, \tau_j, m_j^{i}{-}1, n_j^{m_j^{i}-1}, e_j^{m_j^{i}-1}) \\ & \xrightarrow{\text{po}} \theta'(o_j^{m_j^{i}}, \tau_j, m_j^{i}, n_j^{m_j^{i}}, e_j^{m_j^{i}})\end{aligned}$$

for some $m_j^i, n_j^{P_j^{i-1}+1}, \cdots, n_j^{P_j^i}, n_j^{P_j^i+1}, \cdots, n_j^{m_j^i}, e_j^{P_j^{i-1}+1}, \cdots, e_j^{P_j^i}, e_j^{P_j^i+1}, \cdots, e_j^{m_j^i}$ where:

- The first two lines denote the execution of the $(P_j^{i-1}{+}1)^{\text{st}}$ to $(P_j^i)^{\text{th}}$ library calls of thread τ_j, with $\theta(o, \tau, p, n, e)$ defined shortly. Moreover, before crashing and proceeding to the next era, *all* durable events in $\theta(o_j^{P_j^{i-1}+1}, \cdots) \xrightarrow{\text{po}} \cdots \xrightarrow{\text{po}} \theta(o_j^{P_j^{i-1}}, \cdots)$ have persisted, and a *prefix* (in po order) of the durable events in $\theta(o_j^{P_j^i}, \tau_j, P_j^i, n_j^{P_j^i}, e_j^{P_j^i})$ have persisted. Note that this prefix may be equal to $\theta(o_j^{P_j^i}, \tau_j, P_j^i, n_j^{P_j^i}, e_j^{P_j^i})$, in which case all its events have persisted.
- The next two lines denote the execution of the subsequent library calls of thread τ_j where $m_j^i \leq P_j^n$, with *none* of their durable events having persisted.
- The last line denotes the execution of the $(m_j^i)^{\text{th}}$ call of thread τ_j ($m_j^i \leq P_j^n$), during which the program crashed and thus the execution of era i ended. The $\theta'(o, \tau, p, n, e)$ denotes a (potentially full) prefix of $\theta(o, \tau, p, n, e)$.

The trace $\theta(o, \tau, p, n, e)$ of each library call is defined as follows:

$$\begin{aligned}\theta(\mathsf{deq}(), \tau, p, n, h) \triangleq\ & \mathit{inv}{=}(\mathtt{I}, \iota_p, \mathsf{deq}, ()) \xrightarrow{\text{po}} \mathit{FD} \\ & \xrightarrow{\text{po}} (\mathtt{R}, \mathit{pc}, p) \xrightarrow{\text{po}} (\mathtt{R}, \mathsf{tid}_\tau, \tau) \\ & \xrightarrow{\text{po}} r_h{=}(\mathtt{R}, q.\mathsf{head}, h) \xrightarrow{\text{po}} r{=}(\mathtt{R}, q.\mathsf{data}[h], n) \\ & \xrightarrow{\text{po}} \mathit{lin}_1{=}(\mathtt{W}, \mathit{map}[\tau][p].\mathsf{node}, n) \xrightarrow{\text{po}} S_1 \xrightarrow{\text{po}} \mathtt{SF} \xrightarrow{\text{po}} S_2 \\ & \xrightarrow{\text{po}} \mathit{ack}{=}(\mathtt{A}, \iota_p, \mathsf{deq}, n)\end{aligned}$$

where *FD* denotes the sequence of events, attempting but failing to dequeue, with

$$S_1 = \begin{cases} \emptyset & \text{if } n = \mathtt{null} \\ (\mathtt{R}, n.\mathtt{t}, \tau') \xrightarrow{\text{po}} (\mathtt{R}, n.\mathtt{pc}, p') \xrightarrow{\text{po}} (\mathtt{W}, \mathit{map}[\tau'][p'].\mathsf{done}, \top) & \text{otherwise} \end{cases}$$

$$S_2 = \begin{cases} \emptyset & \text{if } n = \mathtt{null} \\ \mathit{lin}_2{=}(\mathtt{U}, q.\mathsf{head}, h, h{+}1) \xrightarrow{\text{po}} \mathtt{SF} \xrightarrow{\text{po}} c{=}(\mathtt{W}, \mathit{map}[\tau][p].\mathsf{done}, \top) \xrightarrow{\text{po}} \mathtt{SF} & \text{otherwise} \end{cases}$$

for some τ', p'; and

$$\begin{aligned}\theta(\mathsf{enq}(v), \tau, p, n, e) \triangleq\ & \mathit{inv}{=}(\mathtt{I}, \iota_p, \mathsf{enq}, n) \xrightarrow{\text{po}} (\mathtt{R}, \mathit{pc}, p) \xrightarrow{\text{po}} (\mathtt{R}, \mathsf{tid}_\tau, \tau) \\ & \xrightarrow{\text{po}} (\mathtt{W}, n.\mathsf{val}, v) \xrightarrow{\text{po}} (\mathtt{W}, n.\mathsf{tid}, \tau) \xrightarrow{\text{po}} (\mathtt{W}, n.\mathtt{pc}, p) \\ & \xrightarrow{\text{po}} (\mathtt{W}, \mathit{map}[\tau][p].\mathsf{node}, n) \xrightarrow{\text{po}} \mathtt{SF} \xrightarrow{\text{po}} (\mathtt{R}, \mathtt{q}.\mathsf{head}, h) \\ & \underbrace{\xrightarrow{\text{po}} (\mathtt{R}, q.\mathsf{data}[h], v_0) \xrightarrow{\text{po}} A_0 \xrightarrow{\text{po}} \cdots (\mathtt{R}, q.\mathsf{data}[h{+}s{-}1], v_{s-1}) \xrightarrow{\text{po}} A_{s-1}}_{s \text{ times}} \\ & \xrightarrow{\text{po}} (\mathtt{R}, q.\mathsf{data}[h{+}s], \mathtt{null}) \xrightarrow{\text{po}} \mathit{lin}{=}(\mathtt{U}, q.\mathsf{data}[h{+}s], \mathtt{null}, n) \\ & \xrightarrow{\text{po}} \mathtt{SF} \xrightarrow{\text{po}} \mathit{ack}{=}(\mathtt{A}, \iota_p, \mathsf{enq}, ())\end{aligned}$$

for some $s \geq 0$ such that $h{+}s = e$, and for all $k \in \{0 \cdots s{-}1\}$, either 1) $v_k \neq \texttt{null}$ and $A_k = \emptyset$; or $v_k = \texttt{null}$ and $A_k = (\mathsf{R}, \texttt{q.data[}h{+}k\texttt{]}, v'_k)$ with $v'_k \neq \texttt{null}$. In the above traces, for brevity we have omitted the thread identifiers (τ_j) and event identifiers and represent each event with its label only. We use the $\theta(\texttt{enq(-)}, \tau, p, n, e)$ prefix to extract its specific events, e.g. $\theta(\texttt{enq(-)}, \tau, p, n, e).inv$.

Let us write `q.tail` to denote the index of the last entry in the queue. Observe that each `enq` operation leaves the `q.head` value unchanged while increasing `q.tail` by 1. Similarly, each `deq` operation leaves `q.tail` unchanged while increasing `q.head` by one. Note that in each $\theta(\texttt{enq}(v), \tau, p, n, e)$, the $e{-}1$ denotes the value of `q.tail` immediately before the insertion of node n by $\theta(\texttt{enq}(v), \tau, p, n, e)$, i.e. the e denotes the value of `q.tail` immediately after the insertion of node n by $\theta(\texttt{enq}(v), \tau, p, n, e)$. Similarly, in each $\theta(\texttt{deq()}, \tau, p, n, h)$, the h denotes the value of `q.head` immediately before the removal of node n by $\theta(\texttt{deq()}, \tau, p, n, h)$.
Let:

$$\texttt{lp}(\theta(o, \tau, p, n, e)) \triangleq \begin{cases} \theta(o, \tau, p, n, e).lin & \text{if } o{=}\texttt{enq}(v) \\ \theta(o, \tau, p, n, e).lin_1 & \text{if } o{=}\texttt{deq()} \text{ and } \theta(o, \tau, p, n, e).S_2{=}\emptyset \\ \theta(o, \tau, p, n, e).lin_2 & \text{if } o{=}\texttt{deq()} \text{ and } \theta(o, \tau, p, n, e).S_2{\neq}\emptyset \end{cases}$$

For each $\tau_j \in dom(\mathsf{P})$ let:

$$P_{(i,j)} = P_i \cap \left\{ e \;\middle|\; \texttt{tid}(e) = \tau_j \right\} \qquad E'_{(i,j)} = P_{(i,j)} \cup S_{(i,j)}$$

where

$$\begin{aligned} S_{(i,j)} \triangleq & \left\{ (\mathsf{A}, \iota, \mathsf{enq}, ()) \;\middle|\; \begin{array}{l} \exists o, p, n, inv, e. \\ \quad inv = (\mathsf{I}, \iota, \mathsf{enq}, n) = \max\left(\mathsf{nvo}|_{P_{(i,j)} \cap I}\right) \\ \quad \wedge\, inv \in \theta(o, \tau_j, p, n, e) \wedge \forall r'.\ (\mathsf{A}, \iota, \mathsf{enq}, r') \notin P_{(i,j)} \\ \quad \wedge\, \texttt{lp}(\theta(o, \tau_j, p, n, e)) \in P_{(i,j)} \end{array} \right\} \\ & \cup \left\{ (\mathsf{A}, \iota, \mathsf{deq}, n) \;\middle|\; \begin{array}{l} \exists o, p, inv, e. \\ \quad inv = (\mathsf{I}, \iota, \mathsf{deq}, ()) = \max\left(\mathsf{nvo}|_{P_{(i,j)} \cap I}\right) \\ \quad \wedge\, inv \in \theta(o, \tau_j, p, n, e) \wedge \forall r'.\ (\mathsf{A}, \iota, \mathsf{deq}, r') \notin P_{(i,j)} \\ \quad \wedge\, \texttt{lp}(\theta(o, \tau_j, p, n, e)) \in P_{(i,j)} \wedge (n \neq \texttt{null} \Rightarrow \theta(o, \tau_j, p, n, e).c \in P_{(i,j)}) \end{array} \right\} \\ & \cup \left\{ (\mathsf{A}, \iota, \mathsf{deq}, n) \;\middle|\; \begin{array}{l} n \neq \texttt{null} \wedge \exists o, p, inv, e. \\ \quad inv = (\mathsf{I}, \iota, \mathsf{deq}, ()) = \max\left(\mathsf{nvo}|_{P_{(i,j)} \cap I}\right) \\ \quad \wedge\, inv \in \theta(o, \tau_j, p, n, e) \wedge \forall r'.\ (\mathsf{A}, \iota, \mathsf{deq}, r') \notin P_{(i,j)} \\ \quad \wedge\, \theta(o, \tau_j, p, n, e).lin_1 \in P_{(i,j)} \\ \quad \wedge \forall k < j.\ \forall p', e'.\ \theta(\texttt{deq()}, \tau_k, p', n, e').lin_1 \notin P_{(i,k)} \\ \quad \wedge \exists k, p', e'.\ k \geq j \wedge \theta(\texttt{deq()}, \tau_k, p', n, e').lin_2 \in P_{(i,k)} \\ \qquad \wedge \theta(\texttt{deq()}, \tau_k, p', n, e').c \notin P_{(i,k)}) \end{array} \right\} \end{aligned}$$

Let $E'_i = \bigcup_{\tau_j \in dom(\mathsf{P})} E'_{(i,j)}$. From the definition of each $E'_{(i,j)}$ and $P_{(i,j)}$ we then know that $P_i \subseteq E'_i$ and $E'_i \in \texttt{comp}(P_i)$. Let $T_i = \texttt{trunc}(E'_i)$.

Let C_i denote an enumeration of $\bigcup_{\tau_j \in dom(\mathsf{P})} \{\theta(o_j^{P_j^{i-1}+1}, \tau_j, P_j^{i-1}{+}1, n_j^{P_j^{i-1}+1}) \cdots \theta(o_j^{P_j^i}, \tau_j, P_j^i, n_j^{P_j^i}\}$ that respects *memory order (in* tso_i*) of linearisation points*. That is, for all $\theta(o, \tau_j, p, n, e)$, $\theta(o', \tau_{j'}, p', n', e')$, if $\texttt{lp}(\theta(o, \tau_j, p, n, e)) \xrightarrow{\mathsf{tso}_i} \texttt{lp}(\theta(o', \tau_{j'}, p', n', e'))$, then $\theta(o, \tau_j, p, n, e) \prec_{C_i} \theta(o', \tau_{j'}, p', n', e')$.

When C_i is enumerated as $C_i = \theta(c_i^1, \tau_i^1, p_i^1, n_i^1, e_i^1). \cdots . \theta(c_i^{t_i}, \tau_i^{t_i}, p_i^{t_i}, n_i^{t_i}, e_i^{t_i})$, let us define

$$\begin{aligned} \theta_i = \theta(c_i^1, \tau_i^1, p_i^1, n_i^1, e_i^1).inv\ .\ \theta(c_i^1, \tau_i^1, p_i^1, n_i^1, e_i^1).ack \\ . \cdots . \theta(c_i^{t_i}, \tau_i^{t_i}, p_i^{t_i}, n_i^{t_i}, e_i^{t_i}).inv\ .\ \theta(c_i^{t_i}, \tau_i^{t_i}, p_i^{t_i}, n_i^{t_i}, e_i^{t_i}).ack \end{aligned}$$

Lemma 14. *Given a Px86-valid execution $C = G_1, \cdots, G_n$, let for all $i \in \{1 \cdots n\}$, C_i be as defined above. Then, for all i, $\theta(o,\tau,p,n,e)$, $\theta(o',\tau',p',n',e')$, a,b,c,d, if $a \in \theta(o,\tau,p,n,e)$ and $b \in \theta(o',\tau',p',n',e')$, $C_i|_c = \theta(o,\tau,p,n,e)$, $C_i|_d = \theta(o',\tau',p',n',e')$ and $(a,b) \in \mathsf{hb} \triangleq G_i.\mathrm{po} \cup G_i.\mathrm{rf}^+$, then either 1) $c = d$ and $(a,b) \in G_i.\mathrm{po}$; or 2) $c < d$.*

Proof. Pick an arbitrary Px86-valid execution $C = G_1, \cdots, G_n$, and let for all $i \in \{1 \cdots n\}$, C_i be as defined above. Pick arbitrary i. Since G_i is Px86-consistent we know there exists a total store order tso that satisfies the conditions of Px86-consistency. As G_i is Px86-consistent, we know that $G_i.\mathrm{rf} \subseteq G_i.\mathrm{po} \cup G_i.\mathrm{rf}_e$. That is, $\mathsf{hb} \triangleq (G_i.\mathrm{po} \cup G_i.\mathrm{rf}_e)^+$. From the definition of transitive closure it is then straightforward to show that $\mathsf{hb} \triangleq \bigcup_{j \in \mathbb{N}} \mathsf{hb}^j$, where $\mathsf{hb}^0 \triangleq G_i.\mathrm{po} \cup G_i.\mathrm{rf}_e$ and $\mathsf{hb}^{k+1} \triangleq \mathsf{hb}^0; \mathsf{hb}^k$, for all $k \in \mathbb{N}$. We thus demonstrate the following instead:

For all $j \in \mathbb{N}$, and for all $\theta(o,\tau,p,n,e)$, $\theta(o',\tau',p',n',e')$, a,b,c,d, if $a \in \theta(o,\tau,p,n,e)$ and $b \in \theta(o',\tau',p',n',e')$, $C_i|_c = \theta(o,\tau,p,n,e)$, $C_i|_d = \theta(o',\tau',p',n',e')$ and $(a,b) \in \mathsf{hb}^j$, then either 1) $c = d$ and $(a,b) \in \mathrm{po}_i$; or 2) $c < d$.

We proceed by induction on j.

Base case: j=0

We have $(a,b) \in \mathsf{hb}^0 = G_i.\mathrm{po} \cup G_i.\mathrm{rf}_e$. There are seven cases to consider: 1) c=d and $(a,b) \in G_i.\mathrm{po}$ in which case the desired result holds immediately; 2) c=d and $(a,b) \in G_i.\mathrm{rf}_e$ which immediately leads to a contradiction as c=d; 3) $c \neq d$ and $(a,b) \in G_i.\mathrm{po}$; 4) $c \neq d$, $(a,b) \in G_i.\mathrm{rf}_e$, o=$\mathtt{enq}(v)$ and o'=$\mathtt{enq}(v')$ for some v, v'; 5) $c \neq d$, $(a,b) \in G_i.\mathrm{rf}_e$, o=$\mathtt{enq}(v)$ and o'=$\mathtt{deq}()$ for some v; 6) $c \neq d$, $(a,b) \in G_i.\mathrm{rf}_e$, o=$\mathtt{deq}()$ and o'=$\mathtt{enq}(v)$ for some v; 7) $c \neq d$, $(a,b) \in G_i.\mathrm{rf}_e$, o=$\mathtt{deq}()$ and o'=$\mathtt{deq}()$.

In case 3 we then have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{G_i.\mathrm{po}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. As such, since G_i is Px86-consistent and linearisation points are in $W \cup U$ (see $\mathtt{lp}(.)$ definition), we have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required.

In case 4, note that the only location written by o that may be read externally by other queue operations is that of its linearisation point; i.e. a=$\mathtt{lp}(\theta(o,\tau,p,n,e))$ – the *map* entry written by o is never read by other queue operations. Similarly, the only locations that o' reads externally from another enq is from $q.\mathtt{data}$ either before its linearisation point (while traversing for an empty slot) or at its linearisation point (when inserting via **CAS**). That is, $b \xrightarrow{G_i.\mathrm{po}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Moreover, since $\mathtt{lp}(\theta(o',\tau',p',n',e')) \in U$, we have $b \xrightarrow{\mathrm{tso}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. We then have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{G_i.\mathrm{rf}_e} b \xrightarrow{\mathrm{tso}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. From Px86-consistency of G_i we thus have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} b \xrightarrow{\mathrm{tso}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. That is, $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required.

Similarly, in case 5 as in 4 we know a=$\mathtt{lp}(\theta(o,\tau,p,n,e))$. Moreover, the only locations that o' reads externally from another enq is from $q.\mathtt{data}$ which is before its linearisation point. That is, $b \xrightarrow{G_i.\mathrm{po}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. As $b \in R$ and $\mathtt{lp}(\theta(o',\tau',p',n',e')) \in W \cup U$, from Px86-consistency of G_i we have $b \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. We then have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{G_i.\mathrm{rf}_e} b \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. From Px86-consistency of G_i we thus have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} b \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. That is, $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required.

In case 6, note that the only location written by o that may be read externally by other queue operations is that of its linearisation point when incrementing the $q.\mathtt{head}$ value; i.e. a=$\mathtt{lp}(\theta(o,\tau,p,n,e)) \in$

U – the *map* entries written by o is never read by other queue operations. Moreover, the only locations that o' reads externally from another deq is from $q.\mathtt{head}$ which is before its linearisation point. That is, $b \xrightarrow{G_i.\mathrm{po}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. As $b \in R$ and $\mathtt{lp}(\theta(o',\tau',p',n',e')) \in U$, from Px86-consistency of G_i we have $b \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. We then have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{G_i.\mathrm{rf}_e} b \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. From Px86-consistency of G_i we thus have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} b \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. That is, $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required.

In case 7, as in 6 we know $a{=}\mathtt{lp}(\theta(o,\tau,p,n,e)) \in U$. Moreover, the only locations that o' reads externally from another deq is from $q.\mathtt{head}$ which is either before or at its linearisation point when incrementing the $q.\mathtt{head}$ value; i.e. $b \xrightarrow{G_i.\mathrm{po}^?} \mathtt{lp}(\theta(o',\tau',p',n',e')) \in U$. As $\mathtt{lp}(\theta(o',\tau',p',n',e')) \in U$ from Px86-consistency we have $b \xrightarrow{G_i.\mathrm{tso}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. We then have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{G_i.\mathrm{rf}_e} b \xrightarrow{\mathrm{tso}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. From Px86-consistency of G_i we thus have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} b \xrightarrow{\mathrm{tso}^?} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. That is, $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required.

Inductive case $j{=}k{+}1$

Either $(a,b) \in \mathrm{hb}^j \cap G_i.\mathrm{po}$; or there exists at least one $G_i.\mathrm{rf}_e$ edge between a,b: there exists f,g such that $a \xrightarrow{G_i.\mathrm{po}^?} f \xrightarrow{G_i.\mathrm{rf}_e} g \xrightarrow{\mathrm{hb}^k} b$. In the former case the desired result follows from the base case. In the latter case we then know there exists $\theta(o_1,\tau_1,p_1,n_1,e_1)$ and $\theta(o_2,\tau_2,p_2,n_2,e_2)$ such that $f \in \theta(o_1,\tau_1,p_1,n_1,e_1)$ and $g \in \theta(o_2,\tau_2,p_2,n_2,e_2)$. Since $f \xrightarrow{G_i.\mathrm{rf}_e} g$, following similar steps as in the base case we then know $\mathtt{lp}(\theta(o_1,\tau_1,p_1,n_1,e_1)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o_2,\tau_2,p_2,n_2,e_2))$. Now either 1) $\theta(o_1,\tau_1,p_1,n_1,e_1){=}\theta(o,\tau,p,n,e)$ or 2) $\theta(o_1,\tau_1,p_1,n_1,e_1) \neq \theta(o,\tau,p,n,e)$. In case (1) we thus have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o_2,\tau_2,p_2,n_2,e_2))$. In case (2) we thus have $a \xrightarrow{G_i.\mathrm{po}} f$. As such, since G_i is Px86-consistent and linearisation points are in $W \cup U$ (see $\mathtt{lp}(.)$ definition), we have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \theta(o_1,\tau_1,p_1,n_1,e_1)$. From the transitivity of tso we then have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \theta(o_2,\tau_2,p_2,n_2,e_2)$. That is, in both cases we have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o_2,\tau_2,p_2,n_2,e_2))$.

On the other hand, either a) $\theta(o_2,\tau_2,p_2,n_2,e_2){=}\theta(o',\tau',p',n',e')$ or b) $\theta(o_2,\tau_2,p_2,n_2,e_2) \neq \theta(o',\tau',p',n',e')$. In case (a) we thus have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required.

In case (b), let $C_i|_r = \theta(o_2,\tau_2,p_2,n_2,e_2)$. From the inductive hypothesis we then have $r < d$. As such, from the definition of C_i we have $\mathtt{lp}(\theta(o_2,\tau_2,p_2,n_2,e_2)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. As we also have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o_2,\tau_2,p_2,n_2,e_2))$, from the transitivity of tso we have $\mathtt{lp}(\theta(o,\tau,p,n,e)) \xrightarrow{\mathrm{tso}} \mathtt{lp}(\theta(o',\tau',p',n',e'))$. Consequently, from the definition of C_i we have $c < d$, as required. □

Lemma 15. *Given a Px86-valid execution $C = G_1,\cdots,G_n$, let for all $i \in \{1\cdots n\}$, θ_i be defined as above with $C_i = \theta(c_i^1,\tau_i^1,p_i^1,n_i^1,e_i^1).\cdots.\theta(c_i^{t_i},\tau_i^{t_i},p_i^{t_i},n_i^{t_i},e_i^{t_i})$. For all $i \in \{1\cdots n\}$, and a,b, let $O_a^b = \theta(c_i^a,\tau_i^a,p_i^a,n_i^a,e_i^a).inv.\theta(c_i^a,\tau_i^a,p_i^a,n_i^a,e_i^a).ack.\cdots.\theta(c_i^b,\tau_i^b,p_i^b,n_i^b,e_i^b).inv.\theta(c_i^b,\tau_i^b,p_i^b,n_i^b,e_i^b).ack$.*

For all $G_i = (E_i, I_i, P_i, \mathrm{po}_i, \mathrm{rf}_i, \mathrm{mo}_i, \mathrm{nvo}_i)$, *for all* θ_i, *for all* Q_i^0 *and for all* $l \in \{0 \cdots t_i\}$, $k{=}t_i{-}l$, $E_i^k = P_i \setminus \bigcup_{x=k+1}^{t_i} \theta(c_i^x, \tau_i^x, p_i^x, n_i^x, e_i^x).E$, *and* Q_i^k:

$$\begin{aligned}&\mathtt{getQ}(Q_i^0, O_1^k) = Q_i^k \wedge \mathrm{isQ}(q, Q_i^k, \mathrm{nvo}_i, I_i, E_i^k) \Rightarrow\\ &\quad \exists Q_i^t.\ \mathtt{getQ}(Q_i^k, O_{k+1}^{t_i}) = Q_i^t \wedge \mathrm{isQ}(q, Q_i^t, \mathrm{nvo}_i, I_i, P_i)\end{aligned}$$

where:

$$\begin{aligned}\mathrm{isQ}(q, Q, \mathrm{nvo}, I, P) \triangleq\ & (\mathit{init}_q = \max\left(\mathrm{nvo}|_{P\cap(W\cup U)_q}\right) \wedge Q{=}\epsilon)\\ &\vee(\exists h, s.\ |Q| {=} s \wedge \forall v \in Q.\ v \neq \mathtt{null}\\ &\quad \wedge \mathtt{val_w}(\max\left(\mathrm{nvo}|_{P\cap(W\cup U)_{q.\mathtt{head}}}\right)){=}h\\ &\quad \wedge \forall k \in \{0 \cdots s{-}1\}.\\ &\qquad \mathtt{val_w}(\max\left(\mathrm{nvo}|_{P\cap(W\cup U)_{q.\mathtt{data}[h+k]}}\right)){=} Q|_k\\ &\quad \wedge \forall k \geq s.\\ &\qquad \mathtt{val_w}(\max\left(\mathrm{nvo}|_{I\cap(W\cup U)_{q.\mathtt{data}[h+k]}}\right)){=}\mathtt{null}\\ &\qquad \wedge (P \setminus I) \cap (W \cup U)_{q.\mathtt{data}[h+k]} = \emptyset)\end{aligned}$$

and

$$\mathtt{getQ}(s, \theta) \triangleq \begin{cases} s & \mathit{if}\ \theta{=}\epsilon\\ \mathtt{getQ}(s; n, \theta') & \mathit{if}\ \exists n, \theta', \iota.\ n{\neq}\mathtt{null} \wedge \theta{=}(\mathtt{I}, \iota, \mathtt{enq}, n).(\mathtt{A}, \iota, \mathtt{enq}, ()).\theta'\\ \mathtt{getQ}(s', \theta') & \mathit{if}\ \exists n, \theta', \iota, s'.\ n{\neq}\mathtt{null} \wedge s{=}n; s'\\ & \qquad\qquad \wedge\ \theta{=}(\mathtt{I}, \iota, \mathtt{deq}, ()).(\mathtt{A}, \iota, \mathtt{deq}, n).\theta'\\ \mathtt{getQ}(s, \theta') & \mathit{if}\ \exists \theta', \iota.\ s{=}\epsilon \wedge \theta{=}(\mathtt{I}, \iota, \mathtt{deq}, ()).(\mathtt{A}, \iota, \mathtt{deq}, \mathtt{null}).\theta'\\ \mathit{undefined} & \mathit{otherwise}\end{cases}$$

Proof. Pick an arbitrary Px86-valid execution $C = G_1, \cdots, G_n$. Let θ_i and C_i be as defined as above for all $i \in \{1 \cdots n\}$. Pick an arbitrary $i \in \{1 \cdots n\}$, $G_i = (E_i, I_i, P_i, \mathrm{po}_i, \mathrm{rf}_i, \mathrm{mo}_i, \mathrm{nvo}_i)$ and θ_i. Let $G_i.\mathrm{tso}{=}\mathrm{tso}_i$ We proceed by induction on l.

Base case $l = 0$, $k = t_i$
Pick arbitrary Q_i^0 and Q_i^k such that $\mathtt{getQ}(Q_i^0, O_1^k) = Q_i^k$ and $\mathrm{isQ}(q, Q_i^k, \mathrm{nvo}_i, I_i, E_i^k)$. As $k = t_i$, we have $\mathrm{isQ}(q, Q_i^k, \mathrm{nvo}_i, I_i, P_i)$. As $O_{k+1}^{t_i}{=}\epsilon$, we have $\mathtt{getQ}(Q_i^k, O_{k+1}^{t_i}) = Q_i^k$, as required.

Inductive case $0 < l \leq t_i$

$$\begin{aligned}&\forall Q.\ \forall k' > k.\ \mathtt{getQ}(Q_i^0, O_1^{k'}) = Q \wedge \mathrm{isQ}(q, Q, \mathrm{nvo}_i, I_i, E_i^{k'}) \Rightarrow\\ &\quad \exists Q_i^t.\ \mathtt{getQ}(Q, O_{k'+1}^{t_i}) = Q_i^t \wedge \mathrm{isQ}(q, Q_i^t, \mathrm{nvo}_i, I_i, P_i)\end{aligned} \tag{I.H.}$$

Pick arbitrary Q_i^0 and Q_i^k such that $\mathtt{getQ}(Q_i^0, O_1^k) = Q_i^k$ and $\mathrm{isQ}(q, Q_i^k, \mathrm{nvo}_i, I_i, E_i^k)$. We are then required to show that there exists Q_i^t such that $\mathtt{getQ}(Q_i^k, O_{k+1}^{t_i}) = Q_i^t$ and $\mathrm{isQ}(q, Q_i^t, \mathrm{nvo}_i, I_i, P_i)$. We then know:

$$O_{k+1}^{t_i} = \theta(c_i^{k+1}, \tau_i^{k+1}, p_i^{k+1}, n_i^{k+1}, e_i^{k+1}).\mathit{inv}.\theta(c_i^{k+1}, \tau_i^{k+1}, p_i^{k+1}, n_i^{k+1}, e_i^{k+1}).\mathit{ack}.O_{k+2}^{t_i}$$

There are now three cases to consider: 1) there exists m such that $c_i^{k+1}{=}\mathtt{enq}(m)$ and $n_i^{k+1}{=}m$; or 2) there exists $m \neq \mathtt{null}$ such that $c_i^{k+1}{=}\mathtt{deq}()$ and $n_i^{k+1}{=}m$; or 3) $c_i^{k+1}{=}\mathtt{deq}()$ and $n_i^{k+1}{=}\mathtt{null}$.

In case (1), as $\mathtt{getQ}(Q_i^0, O_1^k) = Q_i^k$, from its definition we have $\mathtt{getQ}(Q_i^0, O_1^{k+1}) = Q_i^k.m$. Let $Q_i^{k+1} = Q_i^k.m$. Given the trace $\theta(c_i^{k+1}, \tau_i^{k+1}, p_i^{k+1}, n_i^{k+1}, e_i^{k+1})$, since from the Px86-validity of G_i

we have $I_i \times (P_i \setminus I_i) \subseteq \mathsf{nvo}_i$ and as $\mathsf{isQ}(q, Q_i^k, \mathsf{nvo}_i, I_i, E_i^k)$ holds, from its definition we have $\mathsf{isQ}(q, Q_i^{k+1}, \mathsf{nvo}_i, I_i, E_i^{k+1})$. From (I.H.) we know there exists Q_i^t such that $\mathtt{getQ}(Q_i^{k+1}, O_{k+2}^{t_i}) = Q_i^t$ and $\mathsf{isQ}(q, Q_i^t, \mathsf{nvo}_i, I_i, P_i)$. As $\mathtt{getQ}(Q_i^{k+1}, O_{k+2}^{t_i}) = Q_i^t$, by definition we also have $\mathtt{getQ}(Q_i^k, O_{k+1}^{t_i}) = Q_i^t$, as required.

In case (2), given the trace of $\theta(c_i^{k+1}, \tau_i^{k+1}, p_i^{k+1}, n_i^{k+1})$ we know that there exists w, r, a such that $w \in U$, $\mathtt{loc}(w)$=$\mathtt{q.data[}a\mathtt{]}$, $\mathtt{val_w}(w)$=m, $r = \theta(c_i^{k+1}, \tau_i^{k+1}, p_i^{k+1}, n_i^{k+1}).r$ and $(w, r) \in \mathsf{rf}_i$. Since G_i is Px86-valid, we know either:

i) $w \in I_i$ and for all $j \in \{1 \cdots k\}$ $\theta(c_i^j, \tau_i^j, p_i^j, n_i^j, e_i^j).E \cap (W \cup U)_{\mathtt{q.data[}a\mathtt{]}} = \emptyset$; or

ii) there exists j such that $1 \leq j \leq k$ and w=$\theta(c_i^j, \tau_i^j, p_i^j, n_i^j, e_i^j).lin$ and $c_i^j = \mathtt{enq}(m)$.

As $I_i \subseteq P_i$ and the events of $\theta(c_i^j, \tau_i^j, p_i^j, n_i^j, e_i^j)$ are persistent (discussed above in the construction of θ_i), in both cases we know that $w \in E_i^k$.

It is straightforward to demonstrate that each $\mathtt{enq}$ operation in θ_i writes to a unique index in $q.\mathtt{data}$. I case (ii) we thus know for all $j' \in \{1 \cdots k\} \setminus \{j\}$, $\theta(c_i^{j'}, \tau_i^{j'}, p_i^{j'}, n_i^{j'}, e_i^{j'}).E \cap (W \cup U)_{\mathtt{q.data[}a\mathtt{]}} = \emptyset$. That is, $\max\left(\mathsf{nvo}|_{E_i^k \cap (W \cup U)_{\mathtt{q.data[}a\mathtt{]}}}\right) = w$. Consequently, in both cases we have $\max\left(\mathsf{nvo}|_{E_i^k \cap (W \cup U)_{\mathtt{q.data[}a\mathtt{]}}}\right) = w$. On the other hand, since $\mathsf{isQ}(q, Q_i^k, \mathsf{nvo}_i, I_i, E_i^k)$ holds, from its definition we know $\mathtt{val_w}(\max\left(\mathsf{nvo}|_{E_i^k \cap (W \cup U)_{\mathtt{q.data[}a\mathtt{]}}}\right)) = Q_i^k|_0$. We thus have $Q_i^k|_0 = m$.

Let $Q_i^k = m.Q'$ for some Q' and let $Q_i^{k+1} = Q'$. As $\mathtt{getQ}(Q_i^0, O_1^k)$ holds, from its definition we also have $\mathtt{getQ}(Q_i^0, O_1^{k+1}) = Q_i^{k+1}$. Given the trace $\theta(c_i^{k+1}, \tau_i^{k+1}, p_i^{k+1}, n_i^{k+1}, e_i^{k+1})$, as $\mathsf{isQ}(q, Q_i^k, \mathsf{nvo}_i, I_i, E_i^k)$ holds, from its definition we have $\mathsf{isQ}(q, Q_i^{k+1}, \mathsf{nvo}_i, I_i, E_i^{k+1})$. From (I.H.) we then know there exists Q_i^t such that $\mathtt{getQ}(Q_i^{k+1}, O_{k+2}^{t_i}) = Q_i^t$ and $\mathsf{isQ}(q, Q_i^t, \mathsf{nvo}_i, I_i, P_i)$. As $\mathtt{getQ}(Q_i^{k+1}, O_{k+2}^{t_i}) = Q_i^t$, from its definition we also have $\mathtt{getQ}(Q_i^k, O_{k+1}^{t_i}) = Q_i^t$, as required.

Case (3) is analogous to that of case (2) and is omitted here. □

Corollary 2. *Given a Px86-valid execution $C = G_1; \cdots; G_n$, let for all $i \in \{1 \cdots n\}$, θ_i be defined as above. For all $G_i = (I_i, P_i, E_i, \mathrm{po}_i, \mathsf{rf}_i, \mathsf{tso}_i, \mathsf{nvo}_i)$, θ_i and for all Q_i^0:*

$$\mathsf{isQ}(q, Q_i^0, \mathsf{nvo}_i, I_i, I_i) \Rightarrow$$
$$\exists Q_i^t.\ \mathtt{getQ}(Q_i^0, \theta_i) = Q_i^t \wedge \mathsf{isQ}(q, Q_i^t, \mathsf{nvo}_i, I_i, P_i)$$

Proof. Follows immediately from the previous lemma when $k = 0$. □

Lemma 16. *Given a Px86-valid execution $C = G_1, \cdots, G_n$, if $\theta = \theta_1.\cdots.\theta_n$ with θ_i defined as above for all $i \in \{1 \cdots n\}$, then:*

$$\exists Q.\ \mathtt{getQ}(\epsilon, \theta) = Q$$

Proof. Pick an arbitrary Px86-valid execution $C = G_1, \cdots, G_n$, with $\theta = \theta_1.\cdots.\theta_n$ and θ_i defined as above for all $i \in \{1 \cdots n\}$. Let $Q_1^0 = \epsilon$. By definition we then have $\mathsf{isQ}(q, Q_1^0, \mathsf{nvo}_1, E_1^0, E_1^0)$. On the other hand from Corollary 2 we have:

$$\begin{array}{l}
\exists Q_1^t.\ \mathtt{getQ}(Q_1^0, \theta_1) = Q_1^t \wedge \mathsf{isQ}(q, Q_1^t, \mathsf{nvo}_1, E_1^0, E_1^P) \\
\forall Q_2^0.\ \mathsf{isQ}(q, Q_2^0, \mathsf{nvo}_2, E_2^0, E_2^0) \Rightarrow \\
\quad \exists Q_2^t.\ \mathtt{getQ}(Q_2^0, \theta_2) = Q_2^t \wedge \mathsf{isQ}(q, Q_2^t, \mathsf{nvo}_2, E_2^0, E_2^P) \\
\cdots \\
\forall Q_n^0.\ \mathsf{isQ}(q, Q_n^0, \mathsf{nvo}_n, E_n^0, E_n^0) \Rightarrow \\
\quad \exists Q_n^t.\ \mathtt{getQ}(Q_n^0, \theta_n) = Q_n^t \wedge \mathsf{isQ}(q, Q_n^t, \mathsf{nvo}_n, E_n^0, E_n^P)
\end{array}$$

For all $j \in \{2 \cdots n\}$, let $Q_j^0 = \texttt{getQ}(Q_{j-1}^0, \theta_{j-1})$. From above we then have :

$$\exists Q_1^t, \cdots, Q_n^t.\\ \texttt{getQ}(Q_1^0, \theta_1) = Q_1^t \wedge \texttt{getQ}(Q_1^t, \theta_2) = Q_2^t \wedge \cdots \wedge \texttt{getQ}(Q_{n-1}^t, \theta_n) = Q_n^t$$

From its definition we thus know there exists Q_n^t such that $\texttt{getQ}(Q_1^0, \theta_1.\cdots.\theta_n) = Q_n^t$. That is, there exists Q such that $\texttt{getQ}(\epsilon, \theta) = Q$, as required. □

Theorem 7. *For all client programs* P *of the queue library (comprising calls to* enq *and* deq *only) and all Px86-valid executions C of* P*, C is persistently linearisable.*

Proof. Pick an arbitrary program P and a Px86-valid execution $C = G_1, \cdots, G_n$ of P. For each $i \in \{1 \cdots n\}$, construct T_i and θ_i as above. It then suffices to show that:

$$\forall i \in \{1 \cdots n\}.\ \forall a, b \in T_i.\ (a, b) \in G_i.\mathsf{hb} \Rightarrow a \prec_{\theta_i} b \tag{105}$$

$$\mathsf{fifo}(\epsilon, \theta) \text{ holds when } \theta = \theta_1.\cdots.\theta_n \tag{106}$$

where $G_i.\mathsf{hb} \triangleq (G_i.\mathsf{po} \cup G_i.\mathsf{rf})^+$.

TS. (105)

Pick arbitrary $i \in \{1 \cdots n\}, a, b \in T_i$ such that $(a, b) \in \mathsf{hb}_i$. We then know there exist $c, \tau, p, n, e, c', \tau', p', n', e'$ such that $a \in \theta(c, \tau, p, n, e)$, $b \in \theta(c', \tau', p', n', e')$ and either:

1) $\theta(c, \tau, p, n, e) = \theta(c', \tau', p', n', e')$, $a = \theta(c, \tau, p, n, e).inv$ and $b = \theta(c, \tau, p, n, e).ack$; or

2) $\theta(c, \tau, p, n, e) = \theta(c', \tau', p', n', e')$, $a = \theta(c, \tau, p, n, e).ack$ and $b = \theta(c, \tau, p, n, e).inv$; or

3) $\theta(c, \tau, p, n, e) \neq \theta(c', \tau', p', n', e')$.

In case (1) the desired result holds immediately from the definition of θ_i.

In case (2) we have $b \xrightarrow{G_i.\mathsf{po}} a$. On the other hand from Lemma 14 we have $a \xrightarrow{G_i.\mathsf{po}} b$. That is, we have $(a, a) \in G_i.\mathsf{po}$, leading to a contradiction.

In case (3) from Lemma 14 and the definition of θ_i we have $a \prec_{\theta_i} b$, as required.

TS. (106)

From Lemma 16 we know there exists Q such that $\texttt{getQ}(\epsilon, \theta) = Q$. From the definition of fifo(.,.) we know $\mathsf{fifo}(\epsilon, \theta)$ holds if and only if there exists Q such that $\texttt{getQ}(\epsilon, \theta) = Q$. As such we have $\mathsf{fifo}(\epsilon, \theta)$, as required.

□