# Characterizing Nondeterministic Union

Jacob Neumann
jacobneu@andrew.cmu.edu
April 19, 2021

**Abstract**

## 1 Introduction and Syntax

**Definition 1.1**

Given any set $\Pi$, define the following supersets of $\Pi$:

$$\begin{aligned}
\mathsf{or}^{\leq 0}(\Pi) &:= \Pi \\
\mathsf{or}^{\leq 1}(\Pi) &:= \Pi \cup \left\{\pi \text{ or } \pi' \;:\; \pi, \pi' \in \Pi\right\} \\
\mathsf{or}^{\leq n+1}(\Pi) &:= \Pi \cup \left\{\sigma \text{ or } \sigma' \;:\; \sigma, \sigma' \in \mathsf{or}^{\leq n}(\Pi)\right\} & (n \in \mathbb{N}) \\
\mathsf{or}^{<\omega}(\Pi) &:= \bigcup_{n\in\mathbb{N}} \mathsf{or}^{\leq n}(\Pi)
\end{aligned}$$

For any $n > 0$, we'll sometimes write $\mathsf{or}^{<n}(\Pi)$ for $\mathsf{or}^{\leq n-1}(\Pi)$.

**Definition 1.2**

Given a set $\Sigma$, let

$$\Sigma^{\dagger} := \Sigma \cup \{\mathsf{skip}, \mathsf{abort}\}.$$

**Definition 1.3**

For a fixed set $\Phi$ of atomic propositions and a set $\Sigma$ of program names, define the language $\mathcal{L}_{\Box\bigcirc}(\Sigma)$ by the grammar

$$\varphi, \psi ::= p \mid \neg\varphi \mid \varphi \wedge \psi \mid \Box\varphi \mid \bigcirc_{\sigma}\varphi. \qquad (p \in \Phi,\ \sigma \in \Sigma)$$

We'll make use of the standard (classical) abbreviations, e.g. $\varphi \to \psi$ for $\neg(\varphi \wedge \neg\psi)$, $\top$ for $\neg(p \wedge \neg p)$, and $\Diamond\varphi$ for $\neg\Box\neg\varphi$.

**Definition 1.4**

We'll use the abbreviations

$$\begin{aligned}
\mathsf{Maybe}_0(p, \pi_0, \pi_1) &:\equiv \bigcirc_{\pi_0} p \leftrightarrow \bigcirc_{\pi_0 \text{ or } \pi_1} p \\
\mathsf{Maybe}_1(p, \pi_0, \pi_1) &:\equiv \bigcirc_{\pi_1} p \leftrightarrow \bigcirc_{\pi_0 \text{ or } \pi_1} p
\end{aligned}$$

and

$$\begin{aligned}
\mathsf{Only}_0(p, \pi_0, \pi_1) &:\equiv (\bigcirc_{\pi_0} p \wedge \neg \bigcirc_{\pi_1} p \wedge \bigcirc_{\pi_0 \text{ or } \pi_1} p) \vee (\neg \bigcirc_{\pi_0} p \wedge \bigcirc_{\pi_1} p \wedge \neg \bigcirc_{\pi_0 \text{ or } \pi_1} p) \\
\mathsf{Only}_1(p, \pi_0, \pi_1) &:\equiv (\neg \bigcirc_{\pi_0} p \wedge \bigcirc_{\pi_1} p \wedge \bigcirc_{\pi_0 \text{ or } \pi_1} p) \vee (\bigcirc_{\pi_0} p \wedge \neg \bigcirc_{\pi_1} p \wedge \neg \bigcirc_{\pi_0 \text{ or } \pi_1} p)
\end{aligned}$$

## 2 Dynamic Topological Logic and Union Augmentation

**Definition 2.1**
A **topology** on a nonempty set $X$ is given in one of two equivalent ways:

- By a function
$$\mathsf{int} : \mathcal{P}(X) \to \mathcal{P}(X)$$
sending each subset $A \subseteq X$ to its **interior**, such that the following axioms are satisfied.

(Int1) $\mathsf{int}(X) = X$
(Int2) $\mathsf{int}(A) \subseteq A$ for all $A \subseteq X$
(Int3) $\mathsf{int}(\mathsf{int}(A)) = \mathsf{int}(A)$ for all $A \subseteq X$
(Int4) $\mathsf{int}(A \cap B) = \mathsf{int}(A) \cap \mathsf{int}(B)$ for all $A, B \subseteq X$

- By a collection $\tau$ of subsets of $X$ ($\tau \subseteq \mathcal{P}(X)$) satisfying

(Top1) $\emptyset, X \in \tau$
(Top2) If $A, B \in \tau$, then $A \cap B \in \tau$
(Top3) If $A_i \in \tau$ for all $i \in I$, $\left(\bigcup_{i \in I} A_i\right) \in \tau$

The elements of $\tau$ are known as **open sets**, or are said to be *open with respect to* $\tau$.

The equivalence of these definitions can be seen by putting $\mathsf{int}_X(A)$ to be the set of those $a \in A$ such that $a \in U \subseteq A$ for some $U \in \tau$, or conversely by defining $\tau$ to be the set of fixed points of $\mathsf{int}_X$ (those subsets $A \subseteq X$ such that $\mathsf{int}_X(A) = A$). Throughout, we shall use whichever form is most convenient.

**Definition 2.2**
Given sets $X, Y$ and topologies $\tau_X, \tau_Y$ on them, a relation $R \subseteq X \times Y$ is

- **open** if $A \in \tau_X$ implies $R(A) \in \tau_Y$
- **continuous** if $B \in \tau_Y$ implies $R^{-1}(B) \in \tau_X$

**Definition 2.3**
Let $X$ and $Y$ be sets, with topologies $\tau_X$ and $\tau_Y$. The **product topology of $\tau_X$ and $\tau_Y$** is the least[1] topology on $X \times Y$ containing all sets of the form

$$U \times V \qquad \text{for some } U \in \tau_X,\ V \in \tau_Y.$$

Equivalently, the subsets of $W \subseteq X \times Y$ which are open with respect to the product topology are those of the form
$$\bigcup_{i \in I} U_i \times V_i$$
where $U_i \in \tau_X$, $V_i \in \tau_Y$ for all $i \in I$.

[1]In the sense of containment: $\tau_1$ is "less than" $\tau_2$ if $A \in \tau_1$ implies $A \in \tau_2$, i.e. $\tau_1 \subseteq \tau_2$. This is often indicated by saying $\tau_1$ is *coarser* than $\tau_2$", or "$\tau_2$ is *finer* than $\tau_1$". We are defining the product topology to be the coarsest topologysatisfying the condition above.

**Definition 2.4**

For any set $X$, the **indiscrete** (or *trivial*) **topology** on $X$ is the topology

$$\tau = \{\emptyset, X\}\,.$$

**Lemma 2.1**

For sets $X$ and $Y$ equipped with topologies, the projection functions

$$\mathsf{pr}_1 : X \times Y \to X \qquad \mathsf{pr}_2 : X \times Y \to Y$$

defined by $\mathsf{pr}_1(x, y) = x$ and $\mathsf{pr}_2(x, y) = y$ are open and continuous.

**Definition 2.5**

For $\Sigma$ some set, a $\Sigma$ **dynamic topological model** ("$\Sigma$-DTM") $\mathfrak{M}$ consists of the following.

- A set $|\mathfrak{M}|$
- A topology $\mathsf{int}_{\mathfrak{M}} : \mathcal{P}(|\mathfrak{M}|) \to \mathcal{P}(|\mathfrak{M}|)$ on the set $|\mathfrak{M}|$
- For each $\sigma \in \Sigma$, a partial function
$$\|\sigma\|_{\mathfrak{M}} : |\mathfrak{M}| \rightharpoonup |\mathfrak{M}|$$
- A function
$$V_{\mathfrak{M}} : \Phi \to \mathcal{P}(|\mathfrak{M}|)$$
where $\Phi$ is some give set of *atomic propositions*.

**Definition 2.6**

For any set $\Sigma$ and any $\Sigma$-DTM $\mathfrak{M}$, define:

- $\|\mathsf{skip}\|_{\mathfrak{M}} : |\mathfrak{M}| \to |\mathfrak{M}|$ is the identity function taking $x \in |\mathfrak{M}|$ to itself;
- $\|\mathsf{abort}\|_{\mathfrak{M}} : |\mathfrak{M}| \rightharpoonup |\mathfrak{M}|$ is the function which is defined *nowhere*: $\|\mathsf{abort}\|_{\mathfrak{M}}(x)$ is undefined for all $x \in |\mathfrak{M}|$.

**Definition 2.7**

For a $\Sigma$-DTM $\mathfrak{M}$, define the **interpretation** of $\mathcal{L}_{\Box\bigcirc}\left(\Sigma^{\dagger}\right)$ in $\mathfrak{M}$,[2]

$$[\![-]\!]_{\mathfrak{M}} : \mathcal{L}_{\Box\bigcirc}\left(\Sigma^{\dagger}\right) \to \mathcal{P}(|\mathfrak{M}|)$$

by structural recursion on $\varphi$:

$$\begin{aligned} [\![p]\!]_{\mathfrak{M}} &= V_{\mathfrak{M}}(p) && (p \in \Phi) \\ [\![\neg\varphi]\!]_{\mathfrak{M}} &= |\mathfrak{M}| \setminus [\![\varphi]\!]_{\mathfrak{M}} \\ [\![\varphi \wedge \psi]\!]_{\mathfrak{M}} &= [\![\varphi]\!]_{\mathfrak{M}} \cap [\![\psi]\!]_{\mathfrak{M}} \\ [\![\Box\varphi]\!]_{\mathfrak{M}} &= \mathsf{int}_{\mathfrak{M}}([\![\varphi]\!]_{\mathfrak{M}}) \\ [\![\bigcirc_{\sigma}\varphi]\!]_{\mathfrak{M}} &= \|\sigma\|_{\mathfrak{M}}^{-1}([\![\varphi]\!]_{\mathfrak{M}}) \end{aligned}$$

For $x \in |\mathfrak{M}|$, write $(\mathfrak{M}, x) \models \varphi$ to mean that $x \in [\![\varphi]\!]_{\mathfrak{M}}$. For instance, $(\mathfrak{M}, x) \models \bigcirc_{\sigma}\varphi$ if and only if $\|\sigma\|_{\mathfrak{M}}(x)$ is defined and $(\mathfrak{M}, \|\sigma\|_{\mathfrak{M}}(x)) \models \varphi$. Furthermore, write $\mathfrak{M} \models \varphi$ to indicate that $[\![\varphi]\!]_{\mathfrak{M}} = |\mathfrak{M}|$.

We may omit the $\mathfrak{M}$ subscript when $\mathfrak{M}$ is clear from context.

[2] Note we are interpreting $\mathcal{L}_{\Box\bigcirc}\left(\Sigma^{\dagger}\right)$, not just $\mathcal{L}_{\Box\bigcirc}(\Sigma)$, care of Defn. 2.6.

**Definition 2.8**
Write $\Sigma$-$\mathsf{DTM}$ for the class of all $\Sigma$-DTMs.

For any $\Theta$ and any $\Sigma \subseteq \Theta$, we can view $\Theta$-DTMs as $\Sigma$-DTMs simply by "forgetting" about their interpretations $\|\theta\|$ for $\theta \in \Theta \setminus \Sigma$. We denote this as

$$(-)\restriction_\Sigma : \Theta\text{-}\mathsf{DTM} \to \Sigma\text{-}\mathsf{DTM}$$

i.e. writing $\mathfrak{M}\restriction_\Sigma \in \Sigma$-$\mathsf{DTM}$ for $\mathfrak{M} \in \Theta$-$\mathsf{DTM}$.

**Definition 2.9 (DTM-to-DTM Program Constructor)**
A **program constructor** $C$ consists of

- A rule (called the **syntactic component** of $C$) assigning to each set $\Pi$ a set $\Pi^c$. For all the examples we'll deal with here, $\Pi^c$ will be a superset of $\Pi$.
- For each $\Pi$, a function (called the **semantic component** of $C$)

  $$(-)^C : \Pi\text{-}\mathsf{DTM} \to \Pi^c\text{-}\mathsf{DTM}$$

  so, for each $\Pi$-DTM $\mathfrak{M}$, $\mathfrak{M}^C$ is a $\Pi^c$-DTM.

**Definition 2.10**
Define $\{0,1\}^{<\omega}$ to be the set of **finite binary strings** $s$,

$$s ::= \epsilon \mid 0s \mid 1s.$$

$\epsilon$ is called *the empty string*, and all other elements of $\{0,1\}^{<\omega}$ are called *nonempty*. We generally write nonempty strings with the $\epsilon$ at the end omitted.

Define some standard operations on strings.

- Define the *length* of a string $s \in \{0,1\}^{<\omega}$ recursively by:

  $$\begin{aligned}\mathsf{len}(\epsilon) &= 0\\ \mathsf{len}(0s) &= 1 + \mathsf{len}(s)\\ \mathsf{len}(1s) &= 1 + \mathsf{len}(s)\end{aligned}$$

  Write $\{0,1\}^n$ for the set of those $s \in \{0,1\}^{<\omega}$ such that $\mathsf{len}(s) = n$, and analogously for $\{0,1\}^{<n}$ and $\{0,1\}^{\leq n}$.
- For $s \in \{0,1\}^m$ and $t \in \{0,1\}^n$, write $s^\frown t \in \{0,1\}^{m+n}$ for the concatenation of $s$ and $t$.
- Define the **head** and **tail** of nonempty strings:

  $$\begin{aligned}\mathsf{hd}(0s) &= 0\\ \mathsf{hd}(1s) &= 1\\ \mathsf{tl}(0s) &= s\\ \mathsf{tl}(1s) &= s.\end{aligned}$$

  Clearly, $\mathsf{len}(\mathsf{tl}(s)) = \mathsf{len}(s) - 1$, so we could say

  $$\mathsf{hd} : \{0,1\}^{n+1} \to \{0,1\} \qquad \text{and} \qquad \mathsf{tl} : \{0,1\}^{n+1} \to \{0,1\}^n$$

  for each $n$.

**Definition 2.11**
Define $\{0,1\}^\omega$ to be the set of **infinite binary streams** $S$,

$$S : \mathbb{N} \to \{0,1\}\,.$$

Define the **head** and **tail** of a stream:

$$\begin{aligned}\mathsf{hd} &: \{0,1\}^\omega \to \{0,1\}\\ &: S \mapsto S(0)\\ \mathsf{tl} &: \{0,1\}^\omega \to \{0,1\}^\omega\\ &: S \mapsto (i \mapsto S(i+1)).\end{aligned}$$

The last line means that $\mathsf{tl}(S)(i) = S(i+1)$ for all $S \in \{0,1\}^\omega$ and all $i \in \mathbb{N}$.

**Definition 2.12**
For each natural number $n$, the **$n$-fold nondeterministic union program constructor**, denoted $\mathsf{OR}\,\{n\}$, is a program constructor

- Whose syntactic component takes $\Pi$ to $\mathsf{or}^{\leq n}(\Pi^\dagger)$ (recall Defn. 1.1)
- Whose semantic component takes $\mathfrak{M}$ to $\mathfrak{M}^{\mathsf{OR}\{n\}}$, where $\mathfrak{M}^{\mathsf{OR}\{n\}}$ is defined by:
  - $\left|\mathfrak{M}^{\mathsf{OR}\{n\}}\right| = |\mathfrak{M}| \times \{0,1\}^{\leq n}$ (Recall Defn. 2.10);
  - $\mathsf{int}_{\mathfrak{M}^{\mathsf{OR}\{n\}}}$ is the product topology of $\mathsf{int}_{\mathfrak{M}}$ and the indiscrete topology on $\{0,1\}^{\leq n}$;
  - $\|\pi\|_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(x,s) = (\|\pi\|_{\mathfrak{M}}(x), s)$ for any $\pi \in \Pi^\dagger$;
  - for any $\sigma_0, \sigma_1 \in \mathsf{or}^{<n}(\Pi^\dagger)$,

$$\begin{aligned}&\|\sigma_0 \;\mathsf{or}\; \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(x,\epsilon) \text{ is undefined,}\\ &\|\sigma_0 \;\mathsf{or}\; \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(x,0s) = \|\sigma_0\|_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(x,s),\\ &\|\sigma_0 \;\mathsf{or}\; \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(x,1s) = \|\sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(x,s);\end{aligned}$$

  - and for $p \in \Phi$,

$$(x,s) \in V_{\mathfrak{M}^{\mathsf{OR}\{n\}}}(p) \qquad \Longleftrightarrow \qquad x \in V_{\mathfrak{M}}(p).$$

**Definition 2.13**
The **limited nondeterministic union program constructor**, denoted $\mathsf{OR}\,\{<\omega\}$, is a program constructor

- Whose syntactic component takes $\Pi$ to $\mathsf{or}^{<\omega}(\Pi^\dagger)$
- Whose semantic component takes $\mathfrak{M}$ to $\mathfrak{M}^{\mathsf{OR}\{<\omega\}}$, where $\mathfrak{M}^{\mathsf{OR}\{<\omega\}}$ is defined by:
  - $\left|\mathfrak{M}^{\mathsf{OR}\{<\omega\}}\right| = |\mathfrak{M}| \times \{0,1\}^{<\omega}$;
  - $\mathsf{int}_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}$ is the product topology of $\mathsf{int}_{\mathfrak{M}}$ and the indiscrete topology on $\{0,1\}^{<\omega}$;
  - $\|\pi\|_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(x,s) = (\|\pi\|_{\mathfrak{M}}(x), s)$ for any $\pi \in \Pi^\dagger$;
  - for any $\sigma_0, \sigma_1 \in \mathsf{or}^{<\omega}(\Pi^\dagger)$,

$$\begin{aligned}&\|\sigma_0 \;\mathsf{or}\; \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(x,\epsilon) \text{ is undefined,}\\ &\|\sigma_0 \;\mathsf{or}\; \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(x,0s) = \|\sigma_0\|_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(x,s),\\ &\|\sigma_0 \;\mathsf{or}\; \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(x,1s) = \|\sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(x,s);\end{aligned}$$

– and for $p \in \Phi$,

$$(x,s) \in V_{\mathfrak{M}^{\mathsf{OR}\{<\omega\}}}(p) \qquad \Longleftrightarrow \qquad x \in V_{\mathfrak{M}}(p).$$

The **unlimited nondeterministic union program constructor**, denoted $\mathsf{OR}\{\omega\}$, is a program constructor

- Whose syntactic component takes $\Pi$ to $\mathsf{or}^{<\omega}(\Pi^\dagger)$ (note that this still only allows for arbitrary *finite* nesting of $\mathsf{or}$s)
- Whose semantic component takes $\mathfrak{M}$ to $\mathfrak{M}^{\mathsf{OR}\{\omega\}}$, where $\mathfrak{M}^{\mathsf{OR}\{\omega\}}$ is defined by:
  - $\left|\mathfrak{M}^{\mathsf{OR}\{\omega\}}\right| = |\mathfrak{M}| \times \{0,1\}^\omega$;
  - $\mathsf{int}_{\mathfrak{M}^{\mathsf{OR}\{\omega\}}}$ is the product topology of $\mathsf{int}_{\mathfrak{M}}$ and the indiscrete topology on $\{0,1\}^\omega$
  - $\|\pi\|_{\mathfrak{M}^{\mathsf{OR}\{\omega\}}}(x,S) = (\|\pi\|_{\mathfrak{M}}(x), S)$ for $\pi \in \Pi^\dagger$;
  - for any $\sigma_0, \sigma_1 \in \mathsf{or}^{<\omega}(\Pi^\dagger)$,

$$\|\sigma_0 \text{ or } \sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{\omega\}}}(x,S) \quad = \quad \begin{cases} \|\sigma_0\|_{\mathfrak{M}^{\mathsf{OR}\{\omega\}}}(x, \mathsf{tl}\ S) & \text{if } \mathsf{hd}(S) = 0 \\ \|\sigma_1\|_{\mathfrak{M}^{\mathsf{OR}\{\omega\}}}(x, \mathsf{tl}\ S) & \text{if } \mathsf{hd}(S) = 1; \end{cases}$$

  - and for $p \in \Phi$,

$$(x,s) \in V_{\mathfrak{M}^{\mathsf{OR}\{\omega\}}}(p) \qquad \Longleftrightarrow \qquad x \in V_{\mathfrak{M}}(p).$$

**Note 2.1**

For $\beta = 0, 1, 2, \ldots, <\omega, \omega$, the definition of

$$\|\mathsf{skip}\|_{\mathcal{F}^{\mathsf{OR}\{\beta\}}} \qquad \text{and} \qquad \|\mathsf{abort}\|_{\mathcal{F}^{\mathsf{OR}\{\beta\}}}$$

given by Defn. 2.12/Defn. 2.13 matches the one given for arbitrary DTMs in Defn. 2.6.

# 3 Refinement

**Definition 3.1**

Write $\Sigma$-$\mathsf{Frame}$ for the class of all $\Sigma$-frames.

For any $\Sigma$, we have the class function

$$U : \Sigma\text{-}\mathsf{DTM} \to \Sigma\text{-}\mathsf{Frame}$$

sending each $\Sigma$-DTM to its underlying $\Sigma$-frame, i.e. "forgetting" about the valuation function $V_{\mathfrak{M}}$. The fibers of this function we'll denote as:

$$\mathsf{DTMs}(\mathcal{F}) := \{\mathfrak{M} \in \Sigma\text{-}\mathsf{DTM} \ : \ U(\mathfrak{M}) = \mathcal{F}\}.$$

**Definition 3.2 (Frame-to-Frame Program Constructor)**

A **program constructor (of frames)** $C$ consists of

- A rule (called the **syntactic component** of $C$) assigning to each set $\Pi$ a set $\Pi^c$.
- For each $\Pi$, a function (called the **semantic component** of $C$)

$$(-)^C : \Pi\text{-}\mathsf{Frame} \to \Pi^c\text{-}\mathsf{Frame}$$

  so, for each $\Pi$-frame $\mathcal{F}$, $\mathcal{F}^C$ is a $\Pi^c$-frame.

**Definition 3.3**

Given $\Pi \subseteq \Sigma$ and a $\Sigma$-frame $\mathcal{G}$, a **$\Pi$-refinement relation** on $\mathcal{G}$ is a binary relation $\mathcal{R} \subseteq |\mathcal{G}| \times |\mathcal{G}|$ such that the following hold.

(RR1) $\mathcal{R}$ is an equivalence relation.

(RR2) For all $\pi \in \Pi$, if $x, x'$ are points of $\mathcal{G}$ such that $x\mathcal{R}x'$, then

$$\|\pi\|_{\mathcal{G}}(x) \text{ is defined} \quad \Longleftrightarrow \quad \|\pi\|_{\mathcal{G}}(x') \text{ is defined}$$

and, if they are defined,

$$(\|\pi\|_{\mathcal{G}}(x), \|\pi\|_{\mathcal{G}}(x')) \in \mathcal{R}.$$

(RR3) If $A \subseteq |\mathcal{G}|$ is open with respect to $\mathsf{int}_{\mathcal{G}}$, then the set

$$\mathcal{R}(A) := \{x \in |\mathcal{G}| \ : \ \exists a \in A \text{ s.t. } x\mathcal{R}a\}$$

is also open with respect to $\mathsf{int}_{\mathcal{G}}$.

**Definition 3.4**

Given sets $\Pi \subseteq \Sigma$, a $(\Pi, \Sigma)$**-refined frame** is a pair $(\mathcal{G}, \mathcal{R})$ where $\mathcal{G}$ is a $\Sigma$-frame and $\mathcal{R}$ is a $\Pi$-refinement relation on $\mathcal{G}$.

A valuation function $V : \Phi \to \mathcal{P}(|\mathcal{G}|)$ is said to **respect** $\mathcal{R}$ if for any atomic proposition $p \in \Phi$ and any pair of $\mathcal{R}$-related $\mathcal{G}$-worlds, $(x, x') \in \mathcal{R}$, we have

$$x \in V(p) \quad \Longleftrightarrow \quad x' \in V(p).$$

Write $\mathsf{DTMs}(\mathcal{G}, \mathcal{R})$ for the set of $\Sigma$-DTMs $\mathfrak{N}$ such that $U(\mathfrak{N}) = \mathcal{G}$ and $V_{\mathfrak{N}}$ respects $\mathcal{R}$. A refined frame $(\mathcal{G}, \mathcal{R})$ is said to **satisfy** a formula $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Sigma)$ – written $(\mathcal{G}, \mathcal{R}) \models \varphi$ – if

$$\mathfrak{N} \models \varphi \qquad \text{for all } \mathfrak{N} \in \mathsf{DTMs}(\mathcal{G}, \mathcal{R}).$$

Finally, write

$$\mathrm{Th}_{\Box\bigcirc}(\Sigma; \mathcal{G}, \mathcal{R})$$

for the set $\{\varphi \in \mathcal{L}_{\Box\bigcirc}(\Sigma) \ : \ (\mathcal{G}, \mathcal{R}) \models \varphi\}$.

**Proposition 3.1**

Let $C$ be $\mathsf{OR}\{\beta\}$ for $\beta = 0, 1, 2, \ldots, {<}\omega, \omega$, and $\mathcal{F}$ an arbitrary $\Pi$-frame. Then the relation $\mathcal{R}^C_{\mathcal{F}}$ defined on $\left|\mathcal{F}^C\right|$ by

$$(x, \gamma)\mathcal{R}^C_{\mathcal{F}}(x', \gamma') \quad \Longleftrightarrow \quad x = x'$$

is a $\Pi$-refinement relation on $\mathcal{F}^C$.

**Lemma 3.2**

For $C$, $\mathcal{F}$ as in Prop. 3.1, the following equality holds.

$$\mathsf{DTMs}\left(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}\right) = \left\{\mathfrak{M}^C \ : \ \mathfrak{M} \in \mathsf{DTMs}(\mathcal{F})\right\}$$

**Definition 3.5**

Given a topology $\tau$ on a set $X$ and an equivalence relation $R \subseteq X \times X$, define the **quotient topology** to be the greatest topology on $X/R$ (the set of $R$-equivalence classes) such that the function

$$\begin{aligned} Q_R : X &\to X/R \\ x &\mapsto [x] = \left\{x' \in X \ : \ xRx'\right\} \end{aligned}$$

is continuous.

Equivalently, the quotient topology is uniquely identified by the following property: a set $V \subseteq X/R$ is open with respect to the quotient topology if and only if $Q_R^{-1}(V) \in \tau$.

**Lemma 3.3**

The quotient function $Q_R : X \to X/R$ is *open* with respect to the topology $\tau_X$ on $X$ and the quotient topology $\tau_{X/R}$ on $X/R$ iff the equivalence relation $R$ satisfies (RR3):

$$\text{if } A \in \tau_X, \text{ then } R(A) \in \tau_X.$$

**Definition 3.6**

Given a $(\Pi, \Sigma)$-refined frame $(\mathcal{G}, \mathcal{R})$, let the **quotient frame** of $\mathcal{G}$ by $\mathcal{R}$ be the $\Pi$-frame $\mathcal{G}/\mathcal{R}$ defined by

- $|\mathcal{G}/\mathcal{R}| = |\mathcal{G}| / \mathcal{R}$
- $\mathsf{int}_{\mathcal{G}/\mathcal{R}}$ is the quotient topology of $\mathsf{int}_{\mathcal{G}}$ by $\mathcal{R}$
- For each $\pi \in \Pi$ and $x \in |\mathcal{G}|$,

$$\|\pi\|_{\mathcal{G}/\mathcal{R}} ([x]) = \left[\|\pi\|_{\mathcal{G}} (x)\right]$$

  which is well-defined by (RR2).[3]

**Lemma 3.4**

For each $(\Pi, \Sigma)$-refined frame $(\mathcal{G}, \mathcal{R})$, there is a bijection

$$\kappa_{\mathcal{R}} : \mathsf{DTMs}(\mathcal{G}/\mathcal{R}) \to \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$$

such that

$$\begin{aligned} (\mathfrak{M}, [y]_{\mathcal{R}}) \models \varphi &\iff (\kappa_{\mathcal{R}}(\mathfrak{M}), y) \models \varphi \\ \mathfrak{M} \models \varphi &\iff \kappa_{\mathcal{R}}(\mathfrak{M}) \models \varphi \end{aligned}$$

for all $\mathfrak{M} \in \mathsf{DTMs}(\mathcal{G}/\mathcal{R})$, $y \in |\mathcal{G}|$, $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)$.

**Definition 3.7**

Given $\Sigma$-frames $\mathcal{F}$ and $\mathcal{G}$, a $\Sigma$**-isomorphism** is a function $I : |\mathcal{F}| \to |\mathcal{G}|$ which

- is bijective;
- respects each $\sigma \in \Sigma$: if $I(x) = y$ then $\|\sigma\|_{\mathcal{F}} (x)$ is defined iff $\|\sigma\|_{\mathcal{G}} (y)$ is defined, and, if both are defined,

$$I(\|\sigma\|_{\mathcal{F}} (x)) = \|\sigma\|_{\mathcal{G}} (y);$$

- is continuous and open with respect to $\mathsf{int}_{\mathcal{F}}$ and $\mathsf{int}_{\mathcal{G}}$.

We write $\mathcal{F} \cong_\Sigma \mathcal{G}$ to mean that there is such a $\Sigma$-isomorphism between $\mathcal{F}$ and $\mathcal{G}$.

**Proposition 3.5**

Let $C = \mathsf{OR}\{\beta\}$ for $\beta = 0, 1, 2, \ldots, {<}\omega, \omega$ and $\mathcal{F}$ be any $\Pi$-frame.

$$\mathcal{F} \cong_\Pi \mathcal{F}^C / \mathcal{R}^C_{\mathcal{F}}.$$

[3] If $\|\pi\|_{\mathcal{G}} (x)$ is undefined for some $x$, then, by (RR2), $\|\pi\|_{\mathcal{G}} (x')$ is undefined for all $x' \in [x]$. In this case, Defn. 3.6 specifies $\|\pi\|_{\mathcal{G}/\mathcal{R}}$ to be undefined at $[x]$.

# 4 Bisimulation

**Definition 4.1**

Given $\Sigma$-frames $\mathcal{F}$ and $\mathcal{G}$, a binary relation $s \subseteq |\mathcal{F}| \times |\mathcal{G}|$ is said to be a **$\Sigma$-bisimulation** (of frames) if

- $s$ is *total*: for each $x \in |\mathcal{F}|$ there exists $y \in |\mathcal{G}|$ such that $xsy$.
- $s$ is open and continuous with respect to $\mathsf{int}_{\mathcal{F}}$ and $\mathsf{int}_{\mathcal{G}}$
- $s$ respects each $\sigma \in \Sigma$: if $xsy$, then $\|\sigma\|_{\mathcal{F}}(x)$ is defined iff $\|\sigma\|_{\mathcal{G}}(y)$ is defined, and, if both are defined,
$$(\|\sigma\|_{\mathcal{F}}(x), \|\sigma\|_{\mathcal{G}}(y)) \ \in \ s.$$

We write $s : \mathcal{F} \nrightarrow_{\Sigma} \mathcal{G}$ to indicate this symbolically.

If $s : \mathcal{F} \nrightarrow_{\Sigma} \mathcal{G}$ and, moreover,

- $s$ is *surjective*: for each $y \in |\mathcal{G}|$ there exists $x \in |\mathcal{F}|$ such that $xsy$,

then we say that $s$ is a **$\Sigma$-equivalence** and write $s : \mathcal{F} \simeq_{\Sigma} \mathcal{G}$. We'll write $\mathcal{F} \simeq_{\Sigma} \mathcal{G}$ to mean that there exists such a $\Sigma$-equivalence.

**Proposition 4.1**

If $C$ is $\mathsf{OR}\{\beta\}$ (as in the previous section) and $\mathcal{F}$ any $\Pi$-frame, then

$$\mathcal{F} \simeq_{\Pi} \mathcal{F}^{C}\restriction_{\Pi}.$$

In particular, the relation $T^{C}_{\mathcal{F}}$ witnessing this equivalence is given by

$$(x, (x', \gamma)) \in T^{C}_{\mathcal{F}} \qquad \Longleftrightarrow \qquad x = x'.$$

I.e. $x$ is related by $T^{C}_{\mathcal{F}}$ to all of its "copies" $(x, \gamma)$, $(x, \gamma')$, $(x, \gamma'')$, $\ldots$ in $\mathcal{F}^{C}$ (considered as a $\Pi$-frame).

**Definition 4.2**

Given sets $X$ and $Y$, equivalence relations $R$ and $S$ on $X$ and $Y$, respectively, and a binary relation $t \subseteq X \times Y$, define the relation $\%t \subseteq (X/R) \times (Y/S)$ by

$$[x]_R \ \%t \ [y]_S \qquad \Longleftrightarrow \qquad \exists x_0 \in [x]_R,\ y_0 \in [y]_S \text{ s.t. } (x_0, y_0) \in t.$$

Notice: the notation suppresses the fact that $\%t$ depends on $R$ and $S$. Whenever we work with $\%t$, we'll need to make $R$ and $S$ clear from context.

**Proposition 4.2**

If $(\mathcal{G}, \mathcal{R})$ and $(\mathcal{H}, \mathcal{S})$ are $(\Pi, \Sigma)$-refined frames and

$$t : \mathcal{G} \nrightarrow_{\Sigma} \mathcal{H}$$

is a $\Sigma$-bisimulation, then

$$\%t : \mathcal{G}/\mathcal{R} \nrightarrow_{\Pi} \mathcal{H}/\mathcal{S}$$

is a $\Pi$-bisimulation. If $t$ is surjective, then so too is $\%t$.[4]

[4] But not the converse: $\%t$ can be surjective without $t$ being surjective.

**Definition 4.3**
Given $(\Pi, \Sigma)$-refined frames $(\mathcal{G}, \mathcal{R})$ and $(\mathcal{H}, \mathcal{S})$, a **strong bisimulation of refined frames** is a relation $t \subseteq |\mathcal{G}| \times |\mathcal{H}|$ where

- $t$ is a $\Sigma$-bisimulation from $\mathcal{G}$ to $\mathcal{H}$
- $\%t$ is a $\Pi$-*isomorphism* from $\mathcal{G}/\mathcal{R}$ to $\mathcal{H}/\mathcal{S}$.

We'll write $t : (\mathcal{G}, \mathcal{R}) \nrightarrow (\mathcal{H}, \mathcal{S})$ to indicate this. If $t$ furthermore is surjective, we'll call it an **strong equivalence of refined frames** and write $t : (\mathcal{G}, \mathcal{R}) \simeq (\mathcal{H}, \mathcal{S})$.

**Definition 4.4**
Given $(\Pi, \Sigma)$-refined frames $(\mathcal{G}, \mathcal{R})$ and $(\mathcal{H}, \mathcal{S})$, a relation $t \subseteq |\mathcal{G}| \times |\mathcal{H}|$ is said to constitute a **strong embedding (of refined frames)** if

- $t$ is total
- $t$ respects each $\sigma \in \Sigma$: if $ytz$, then $\|\sigma\|_{\mathcal{G}}(y)$ is defined iff $\|\sigma\|_{\mathcal{H}}(z)$ is defined, and, if both are defined,
$$(\|\sigma\|_{\mathcal{G}}(y), \|\sigma\|_{\mathcal{H}}(z)) \;\in\; t.$$
- $\%t$ is a $\Pi$-isomorphism.

Write $t : (\mathcal{G}, \mathcal{R}) \hookrightarrow (\mathcal{H}, \mathcal{S})$ to indicate that $t$ is a strong embedding of refined frames.

**Lemma 4.3**
If $\mathfrak{M}$, $\mathfrak{N}$ are $\Sigma$-DTMs and $t : U(\mathfrak{M}) \nrightarrow_{\Sigma} U(\mathfrak{N})$ a $\Sigma$-bisimulation between their underlying frames which satusfies the (Base) condition:

$$x \in V_{\mathfrak{M}}(p) \iff y \in V_{\mathfrak{N}}(p) \qquad \text{for all } x \in |\mathfrak{M}|,\ y \in t(x),\ p \in \Phi$$

then for all $(x, y) \in t$,

$$(\mathfrak{M}, x) \models \varphi \iff (\mathfrak{N}, y) \models \varphi \qquad \text{for all } \varphi \in \mathcal{L}_{\Box\bigcirc}(\Sigma).$$

**Theorem 4.4**
If $t : (\mathcal{G}, \mathcal{R}) \hookrightarrow (\mathcal{H}, \mathcal{S})$, then there is a bijection

$$\tau : \mathsf{DTMs}(\mathcal{G}, \mathcal{R}) \to \mathsf{DTMs}(\mathcal{H}, \mathcal{S})$$

such that for all $\mathfrak{N} \in \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$

- $t$ preserves and reflects $\mathcal{L}_{\Box\bigcirc}(\Pi)$ theories (pointwise and globally):
$$\begin{aligned} (\mathfrak{N}, y) \models \varphi &\iff (\tau(\mathfrak{N}), z) \models \varphi && (y \in |\mathcal{G}|,\ z \in t(y),\ \varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)) \\ \mathfrak{N} \models \varphi &\iff \tau(\mathfrak{N}) \models \varphi && (\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)) \end{aligned}$$
- $t$ preserves and reflects $\mathcal{L}_{\bigcirc}(\Sigma)$ theories (pointwise):
$$(\mathfrak{N}, y) \models \varphi \iff (\tau(\mathfrak{N}), z) \models \varphi \qquad (y \in |\mathcal{G}|,\ z \in t(y),\ \varphi \in \mathcal{L}_{\bigcirc}(\Sigma).)$$

## 5 Characterization

**Definition 5.1**

Define

$$\{0,1\}^{\leq\omega} = \{0,1\}^{<\omega} \cup \{0,1\}^{\omega}$$

and then let $\mathsf{OR}\,\{\leq\omega\}$ be the program constructor

- whose syntactic component takes $\Pi$ to $\mathsf{or}^{<\omega}(\Pi^\dagger)$
- whose semantic component takes $\mathcal{F}$ to $\left(\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\leq\omega\}}\right)$, where $\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}$ is defined by:
  - $\left|\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}\right| = |\mathcal{F}| \times \{0,1\}^{\leq\omega}$;
  - $\mathsf{int}_{\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}}$ is the product topology of $\mathsf{int}_{\mathcal{F}}$ and the indiscrete topology on $\{0,1\}^{\leq\omega}$;
  - for $\pi \in \Pi^\dagger$,
  $$\|\pi\|_{\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}}(x,S) \quad = \quad (\|\pi\|_{\mathcal{F}}(x), S);$$
  - for $\sigma_0, \sigma_1 \in \mathsf{or}^{<\omega}(\Pi^\dagger)$,
  $$\|\sigma_0 \mathbin{\mathsf{or}} \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}}(x,S) \quad = \quad \begin{cases} \text{undefined} & \text{if } S = \epsilon \\ \|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}}(x, \mathsf{tl}\ S) & \text{if } \mathsf{hd}(S) = 0 \\ \|\sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}}(x, \mathsf{tl}\ S) & \text{if } \mathsf{hd}(S) = 1; \end{cases}$$
  - and $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\leq\omega\}}$ is given as in Prop. 3.1.

**Proposition 5.1**

For any $\Pi$-frame $\mathcal{F}$,

$$\mathrm{Th}_{\Box\bigcirc}\left(\mathsf{or}^{<\omega}(\Pi)\,;\mathcal{F}^{\mathsf{OR}\{<\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{<\omega\}}\right) \quad = \quad \mathrm{Th}_{\Box\bigcirc}\left(\mathsf{or}^{<\omega}(\Pi)\,;\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\leq\omega\}}\right)$$

i.e. the language $\mathcal{L}_{\Box\bigcirc}(\mathsf{or}^{<\omega}(\Pi))$ cannot distinguish the results of the program constructors $\mathsf{OR}\,\{<\omega\}$ and $\mathsf{OR}\,\{\leq\omega\}$.

**Proposition 5.2**

For each $\Pi$-frame $\mathcal{F}$, the inclusion function

$$\begin{array}{ll} |\mathcal{F}| \times \{0,1\}^{<\omega} & \to |\mathcal{F}| \times \{0,1\}^{\leq\omega} \\ (x,s) & \mapsto (x,s) \end{array}$$

is a strong embedding

$$\left(\mathcal{F}^{\mathsf{OR}\{<\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{<\omega\}}\right) \hookrightarrow \left(\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\leq\omega\}}\right).$$

**Definition 5.2**

A $(\Pi,\Sigma)$-refined frame $(\mathcal{G},\mathcal{R})$ is said to **satisfy** a set $\Delta$ of $\mathcal{L}_{\Box\bigcirc}(\Sigma)$ formulas – written $(\mathcal{G},\mathcal{R}) \models \Delta$ – if $(\mathcal{G},\mathcal{R}) \models \varphi$ for all $\varphi \in \Delta$.

**Definition 5.3**
Given a program constructor $C$, a set $\Delta$ of $\mathcal{L}_{\Box\bigcirc}(\Pi^c)$ formulas is said to **left-characterize** $C$ if

- $(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}) \models \Delta$ for all $\Pi$-frames $\mathcal{F}$
- For all $(\Pi, \Pi^c)$-refined frames $(\mathcal{G}, \mathcal{R})$, if $(\mathcal{G}, \mathcal{R}) \models \Delta$ then there exists a $\Pi$-frame $\mathcal{F}$ such that
$$\left(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}\right) \hookrightarrow (\mathcal{G}, \mathcal{R})$$

**Definition 5.4**
Given a program constructor $C$, a set $\Delta$ of $\mathcal{L}_{\Box\bigcirc}(\Pi^c)$ formulas is said to **right-characterize** $C$ if

- $(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}) \models \Delta$ for all $\Pi$-frames $\mathcal{F}$
- For all $(\Pi, \Pi^c)$-refined frames $(\mathcal{G}, \mathcal{R})$, if $(\mathcal{G}, \mathcal{R}) \models \Delta$ then there exists a $\Pi$-frame $\mathcal{F}$ such that
$$(\mathcal{G}, \mathcal{R}) \hookrightarrow \left(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}\right)$$

**Definition 5.5**
Given a program constructor $C$, a set $\Delta$ of $\mathcal{L}_{\Box\bigcirc}(\Pi^c)$ formulas is said to **bicharacterize** $C$ if

- $(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}) \models \Delta$ for all $\Pi$-frames $\mathcal{F}$
- For all $(\Pi, \Pi^c)$-refined frames $(\mathcal{G}, \mathcal{R})$, if $(\mathcal{G}, \mathcal{R}) \models \Delta$ then there exists a $\Pi$-frame $\mathcal{F}$ such that
$$\left(\mathcal{F}^C, \mathcal{R}^C_{\mathcal{F}}\right) \simeq (\mathcal{G}, \mathcal{R})$$

**Proposition 5.3**
$\chi_{\mathsf{OR}\{\omega\}}$ (as given in Fig. 5.1) *right-characterizes* $\mathsf{OR}\{\omega\}$.

| | | |
|---|---|---|
| (OR-Typ) | $\mathsf{Maybe}_0(p, \pi_0, \pi_1) \vee \mathsf{Maybe}_1(p, \pi_0, \pi_1)$ | Typicality |
| (OR-Reg0) | $\mathsf{Only}_0(p, \pi_0, \pi_1) \to \mathsf{Maybe}_0(q, \pi_2, \pi_3)$ | Regularity 0 |
| (OR-Reg1) | $\mathsf{Only}_1(p, \pi_0, \pi_1) \to \mathsf{Maybe}_1(q, \pi_2, \pi_3)$ | Regularity 1 |
| (OR-Real0) | $p \wedge \mathsf{Only}_1(q, \pi_0, \pi_1) \to \Diamond(p \wedge \mathsf{Only}_0(q, \pi_0, \pi_1))$ | Realization 0 |
| (OR-Real1) | $p \wedge \mathsf{Only}_0(q, \pi_0, \pi_1) \to \Diamond(p \wedge \mathsf{Only}_1(q, \pi_0, \pi_1))$ | Realization 1 |
| (OR-Refresh) | $p \to \bigcirc_{(\mathsf{skip\ or\ skip})} p$ | Refresh |
| (OR-Nest0) | $\mathsf{Only}_0(p, \pi_0, \pi_1) \to \bigcirc_{\mathsf{skiporskip}} \bigcirc_{\sigma_0} \varphi \to \bigcirc_{\sigma_0 \mathsf{or} \sigma_1} \varphi$ | Nesting 0 |
| (OR-Nest1) | $\mathsf{Only}_1(p, \pi_0, \pi_1) \to \bigcirc_{\mathsf{skiporskip}} \bigcirc_{\sigma_1} \varphi \to \bigcirc_{\sigma_0 \mathsf{or} \sigma_1} \varphi$ | Nesting 1 |
| (OR-PrimInd) | $\bigcirc_{(\mathsf{skip\ or\ skip})} \bigcirc_{\pi_0} \varphi \leftrightarrow \bigcirc_{\pi_0} \bigcirc_{(\mathsf{skip\ or\ skip})} \varphi$<br>$\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort}) \to \neg \bigcirc_{\pi_0} \neg \mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$<br>$\mathsf{Only}_1(\top, \mathsf{skip}, \mathsf{abort}) \to \neg \bigcirc_{\pi_0} \neg \mathsf{Only}_1(\top, \mathsf{skip}, \mathsf{abort})$ | Primitive Independence |

Figure 5.1: $\chi_{\mathsf{OR}\{\omega\}}$. $p, q$ vary over $\Phi \cup \{\top, \bot\}$, $\pi_0, \pi_1, \pi_2, \pi_3$ vary over $\Pi^\dagger$, $\sigma_0, \sigma_1$ vary over $\mathsf{or}^{<\omega}(\Pi^\dagger)$, and $\varphi$ varies over $\mathcal{L}_{\Box\bigcirc}(\mathsf{or}^{<\omega}(\Pi^\dagger))$. Recall the abbreviations of Defn. 1.4.

TODO: Left-Characterization of $\mathsf{OR}\{<\omega\}$

TODO: Bi-Characterization of $\mathsf{OR}\{n\}$

# 6 Conclusion

# A Proofs

(Lemma 3.4)
For each $(\Pi, \Sigma)$-refined frame $(\mathcal{G}, \mathcal{R})$, there is a bijection

$$\kappa_{\mathcal{R}} : \mathsf{DTMs}(\mathcal{G}/\mathcal{R}) \to \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$$

such that

$$\begin{aligned} (\mathfrak{M}, [y]_{\mathcal{R}}) \models \varphi &\iff (\kappa_{\mathcal{R}}(\mathfrak{M}), y) \models \varphi \\ \mathfrak{M} \models \varphi &\iff \kappa_{\mathcal{R}}(\mathfrak{M}) \models \varphi \end{aligned}$$

for all $\mathfrak{M} \in \mathsf{DTMs}(\mathcal{G}/\mathcal{R})$, $y \in |\mathcal{G}|$, $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)$.

*Proof.* —
Given $\mathfrak{M} \in \mathsf{DTMs}(\mathcal{G}/\mathcal{R})$, define $\kappa_{\mathcal{R}}(\mathfrak{M})$ by the valuation

$$V_{\kappa_{\mathcal{R}}(\mathfrak{M})}(p) = \{y \in |\mathcal{G}| \ : \ [y]_{\mathcal{R}} \in V_{\mathfrak{M}}(p)\}$$

for each $p \in \Phi$. The right-hand side can alternatively be written as

$$\bigcup_{[y]_{\mathcal{R}} \in V_{\mathfrak{M}}(p)} [y]_{\mathcal{R}}$$

so each $V_{\kappa_{\mathcal{R}}(\mathfrak{M})}(p)$ is the union of $\mathcal{R}$-equivalence classes, hence $V_{\kappa_{\mathcal{R}}(\mathfrak{M})}$ respects $\mathcal{R}$.

To see this is a bijection, supposed we have $\mathfrak{N} \in \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$. Then we can define the quotient of $V_{\mathfrak{N}}$ by $\mathcal{R}$ to be the valuation on $\mathcal{G}/\mathcal{R}$ given by

$$[y]_{\mathcal{R}} \in V_{\mathfrak{N}/\mathcal{R}}(p) \iff y \in V_{\mathfrak{N}}(p).$$

This is well-defined because $V_{\mathfrak{N}}$ respects $\mathcal{R}$. So we'll write $\mathfrak{N}/\mathcal{R}$ for the $\Pi$-DTM on the frame $\mathcal{G}/\mathcal{R}$ with this valuation. Check that

$$\kappa_{\mathcal{R}}(\mathfrak{N}/\mathcal{R}) = \mathfrak{N} \quad \text{and} \quad \kappa_{\mathcal{R}}(\mathfrak{M})/\mathcal{R} = \mathfrak{M}$$

for any $\mathfrak{N} \in \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$ and $\mathfrak{M} \in \mathsf{DTMs}(\mathcal{G}/\mathcal{R})$, and conclude $\kappa_{\mathcal{R}}$ is a bijection.

The claim that $(\mathfrak{M}, [y]_{\mathcal{R}})$ and $(\kappa_{\mathcal{R}}(\mathfrak{M}), y)$ satisfy all the same $\mathcal{L}_{\Box\bigcirc}(\Pi)$ formulas can be established by induction similar to the one used in the proof of Lemma 4.3. This uses the fact that $Q_{\mathcal{R}} : |\mathcal{G}| \to |\mathcal{G}|/\mathcal{R}$ is open (by Lemma 3.3) and continuous (by definition of the quotient topology), that $(\mathfrak{M}, [y]_{\mathcal{R}})$ and $(\kappa_{\mathcal{R}}(\mathfrak{M}), y)$ agree on atomic propositions (by definition of $\kappa_{\mathcal{R}}$), and the definition of $\|\pi\|_{\mathcal{G}/\mathcal{R}}$ (for $\pi \in \Pi$) given in Defn. 3.6. Note that this result only holds (and only makes sense) for $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)$, not all $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Sigma)$, since $\mathcal{G}/\mathcal{R}$ doesn't have interpretations for $\sigma \in \Sigma \setminus \Pi$.

This can then quickly be used to show that the global theories of $\mathfrak{M}$ and $\kappa_{\mathcal{R}}(\mathfrak{M})$ agree as well: if $\mathfrak{M} \models \varphi$, i.e. $(\mathfrak{M}, [y]_{\mathcal{R}}) \models \varphi$ for all $[y]_{\mathcal{R}}$, then, by the previous paragraph, $(\kappa_{\mathcal{R}}(\mathfrak{M}), y) \models \varphi$ for all $y$, i.e. $\kappa_{\mathcal{R}}(\mathfrak{M}) \models \varphi$. And vice versa.

$\square$

(**Prop. 3.5**)
Let $C = \mathsf{OR}\{\beta\}$ for $\beta = 0, 1, 2, \ldots, <\omega, \omega$ and $\mathcal{F}$ be any $\Pi$-frame.

$$\mathcal{F} \cong_\Pi \mathcal{F}^C / \mathcal{R}^C_\mathcal{F}.$$

*Proof.* —
Observe that the $\mathcal{R}^C_\mathcal{F}$-equivalence classes are of the form

$$\{x\} \times \{0,1\}^\beta$$

for each $x \in |\mathcal{F}|$. So the $\Pi$-isomorphism from $I : |\mathcal{F}| \to \left|\mathcal{F}^C\right| / \mathcal{R}^C_\mathcal{F}$ will send $x$ to $\{x\} \times \{0,1\}^\beta$. This is clearly a bijection.

If $U \subseteq |\mathcal{F}|$ is open (with respect to the topology of $\mathcal{F}$), then, by definition of the product topology, so too is $U \times \{0,1\}^\beta$ (with respect to the topology of $\mathcal{F}^C$). But

$$U \times \{0,1\}^\beta = Q^{-1}_{\mathcal{R}^C_\mathcal{F}}\left(\left\{\{u\} \times \{0,1\}^\beta \ : \ u \in U\right\}\right) = Q^{-1}_{\mathcal{R}^C_\mathcal{F}}(I(U)).$$

which tells us that $I(U)$ must also be an open subset of $\mathcal{F}^C / \mathcal{R}^C_\mathcal{F}$. Thus $I$ is an open function.

For continuity, $V \subseteq \mathcal{F}^C / \mathcal{R}^C_\mathcal{F}$ is open just in case $Q^{-1}_{\mathcal{R}^C_\mathcal{F}}(V)$ is open. Then check that

$$Q^{-1}_{\mathcal{R}^C_\mathcal{F}}(V) = \left\{x \in |\mathcal{F}| \ : \ \left(\{x\} \times \{0,1\}^\beta\right) \in V\right\} \times \{0,1\}^\beta = I^{-1}(V) \times \{0,1\}^\beta.$$

Since $I^{-1}(V) \times \{0,1\}^\beta$ is open, conclude by Lemma 2.1 that $I^{-1}(V)$ is open.

$I$ respects each $\pi \in \Pi$ by definition of the $\mathsf{OR}\{\beta\}$ program constructors and the quotient frame: $\|\pi\|_\mathcal{F}(x)$ is defined iff $\|\pi\|_{\mathcal{F}^C}(x, \gamma)$ is defined for all $\gamma$ iff $\|\pi\|_{\mathcal{F}^C/\mathcal{R}^C_\mathcal{F}}(I(x))$ is defined. If it is defined, then $\|\pi\|_{\mathcal{F}^C/\mathcal{R}^C_\mathcal{F}}(I(x))$ is $I(\|\pi\|_\mathcal{F}(x))$.

□

(**Prop. 4.1**)
If $C$ is $\mathsf{OR}\{\beta\}$ (as in the previous section) and $\mathcal{F}$ any $\Pi$-frame, then

$$\mathcal{F} \simeq_\Pi \mathcal{F}^C{\restriction}_\Pi.$$

In particular, the relation $T^C_\mathcal{F}$ witnessing this equivalence is given by

$$(x, (x', \gamma)) \in T^C_\mathcal{F} \quad \iff \quad x = x'.$$

I.e. $x$ is related by $T^C_\mathcal{F}$ to all of its “copies” $(x, \gamma)$, $(x, \gamma')$, $(x, \gamma'')$, …in $\mathcal{F}^C$ (considered as a $\Pi$-frame).

*Proof.* —
Let $T^C_\mathcal{F}$ be as defined in the statement of the claim. It is immediate to see that $T^C_\mathcal{F}$ is total. Surjectivity follows from the fact that every element of $\left|\mathcal{F}^C{\restriction}_\Pi\right|$ is of the form $(x, \gamma)$ for some $x \in |\mathcal{F}|$. For any $U \subseteq |\mathcal{F}|$ and $V \subseteq \left|\mathcal{F}^C{\restriction}_\Pi\right|$, check that

$$T^C_\mathcal{F}(U) = \mathsf{pr}_1^{-1}(U) \qquad \text{and} \qquad \left(T^C_\mathcal{F}\right)^{-1}(V) = \mathsf{pr}_1(V)$$

so it's quick to check (using Lemma 2.1) that $T^C_\mathcal{F}$ is open and continuous. Finally, recall that

$$\|\pi\|_{\mathcal{F}^C{\restriction}_\Pi}(x, S) = (\|\pi\|_\mathcal{F}(x), S)$$

where the left-hand side is defined iff the right-hand side is. From this, it follows that $T^C_\mathcal{F}$ respects $\pi$.

□

(**Prop. 4.2**)

If $(\mathcal{G}, \mathcal{R})$ and $(\mathcal{H}, \mathcal{S})$ are $(\Pi, \Sigma)$-refined frames and

$$t : \mathcal{G} \nrightarrow_{\Sigma} \mathcal{H}$$

is a $\Sigma$-bisimulation, then

$$\%t : \mathcal{G}/\mathcal{R} \nrightarrow_{\Pi} \mathcal{H}/\mathcal{S}$$

is a $\Pi$-bisimulation. If $t$ is surjective, then so too is $\%t$.[5]

*Proof.* —

The totality of $\%t$ follows from the totality of $t$.

To see that $\%t$ is open, observe that

$$\%t(U) = Q_{\mathcal{S}}(t(Q_{\mathcal{R}}^{-1}(U)))$$

so, by the continuity of $Q_{\mathcal{R}}$, the openness of $t$, and the openness of $Q_{\mathcal{S}}$, if $U$ is open, so too is $\%t(U)$. Likewise, for continuity, observe

$$\%t^{-1}(V) = Q_{\mathcal{R}}(t^{-1}(Q_{\mathcal{S}}^{-1}(V)))$$

so concatenate the continuity of $Q_{\mathcal{S}}$, the continuity of $t$, and the openness of $Q_{\mathcal{R}}$.

To see that $\%t$ respects each $\pi \in \Pi$, pick arbitrary $(x_0, y_0) \in t$ witnessing $([x_0]_{\mathcal{R}}, [y_0]_{\mathcal{S}}) \in \%t$. Then,

$$\begin{aligned}
\|\pi\|_{\mathcal{G}/\mathcal{R}}([x_0]) \text{ is defined} &\iff \|\pi\|_{\mathcal{G}}(x_0) \text{ is defined} && \text{(Defn. 3.6)}\\
&\iff \|\pi\|_{\mathcal{H}}(y_0) \text{ is defined} && (t \text{ bisimulation})\\
&\iff \|\pi\|_{\mathcal{H}/\mathcal{S}}([y_0]) \text{ is defined} && \text{(Defn. 3.6.)}
\end{aligned}$$

If $\|\pi\|$ *is* defined,

$$\begin{aligned}
\|\pi\|_{\mathcal{G}/\mathcal{R}}([x_0]) &= \big[\, \|\pi\|_{\mathcal{G}}(x_0)\big]\\
\|\pi\|_{\mathcal{H}/\mathcal{S}}([y_0]) &= \big[\, \|\pi\|_{\mathcal{H}}(y_0)\big]
\end{aligned}$$

Since $t$ is a bisimulation and $\|\pi\|_{\mathcal{G}}(x_0)$ is defined,

$$\|\pi\|_{\mathcal{H}}(y_0) \in t(\|\pi\|_{\mathcal{G}}(x_0))$$

and it then follows from the definition of $\%t$ that

$$\big([\|\pi\|_{\mathcal{G}}(x_0)], [\|\pi\|_{\mathcal{H}}(y_0)]\big) \in \%t$$

so

$$\Big(\|\pi\|_{\mathcal{G}/\mathcal{R}}([x_0]), \|\pi\|_{\mathcal{H}/\mathcal{S}}([y_0])\Big) \in \%t,$$

as desired.

Finally, if $t$ is surjective, then

$$\begin{aligned}
\%t(|\mathcal{G}/\mathcal{R}|) &= Q_{\mathcal{S}}(t(Q_{\mathcal{R}}^{-1}(|\mathcal{G}/\mathcal{R}|))) && \text{(above)}\\
&= Q_{\mathcal{S}}(t(|\mathcal{G}|)) && (Q_{\mathcal{R}} \text{ total})\\
&= Q_{\mathcal{S}}(|\mathcal{H}|) && (t \text{ surjective})\\
&= |\mathcal{H}/\mathcal{S}| && (Q_{\mathcal{S}} \text{ surjective})
\end{aligned}$$

so $\%t$ is surjective too.

[5] But not the converse: $\%t$ can be surjective without $t$ being surjective.

□

(**Lemma 4.3**)

If $\mathfrak{M}$, $\mathfrak{N}$ are $\Sigma$-DTMs and $t : U(\mathfrak{M}) \nrightarrow_\Sigma U(\mathfrak{N})$ a $\Sigma$-bisimulation between their underlying frames which satusfies the (Base) condition:

$$x \in V_{\mathfrak{M}}(p) \iff y \in V_{\mathfrak{N}}(p) \qquad \text{for all } x \in |\mathfrak{M}|,\ y \in t(x),\ p \in \Phi$$

then for all $(x, y) \in t$,

$$(\mathfrak{M}, x) \models \varphi \iff (\mathfrak{N}, y) \models \varphi \qquad \text{for all } \varphi \in \mathcal{L}_{\Box\bigcirc}(\Sigma).$$

*Proof.* —

By structural induction on $\varphi$. The base case – atomic propositions – is exactly the (Base) condition. The $\wedge$ and $\neg$ inductive steps are trivial.

If $(\mathfrak{M}, x) \models \Box\varphi$, then $x \in \mathsf{int}_{\mathfrak{M}}([\![\varphi]\!]_{\mathfrak{M}})$, i.e. there's an open set $U \subseteq |\mathfrak{M}|$ such that $x \in U \subseteq [\![\varphi]\!]$. Since $t$ is open, we have that $t(U)$ is open. Since $x \in U$ and $(x, y) \in t$, we have $y \in t(U)$. By the inductive hypothesis,

$$t(U) \subseteq [\![\varphi]\!]_{\mathfrak{N}}$$

so conclude $(\mathfrak{N}, y) \models \Box\varphi$. The proof in the other direction proceeds similarly, using the continuity of $t$ rather than its openness.

If $(\mathfrak{M}, x) \models \bigcirc_\sigma\varphi$ for some $\sigma \in \Sigma$, then we know $\|\sigma\|_{\mathfrak{M}}(x)$ is defined and is a $\varphi$-world. Since $t$ respects $\sigma$ and $(x, y) \in t$, we have that $\|\sigma\|_{\mathfrak{N}}(y)$ is defined and

$$(\|\sigma\|_{\mathfrak{M}}(x), \|\sigma\|_{\mathfrak{N}}(y)) \in t.$$

By the inductive hypothesis, $\|\sigma\|_{\mathfrak{N}}(y)$ must validate $\varphi$, hence

$$(\mathfrak{N}, y) \models \bigcirc_\sigma\varphi$$

as desired. Again, the other direction proceeds similarly. By induction, we have that the result holds for all $\varphi$.

□

(**Theorem 4.4**)

If $t : (\mathcal{G}, \mathcal{R}) \hookrightarrow (\mathcal{H}, \mathcal{S})$, then there is a bijection

$$\tau : \mathsf{DTMs}(\mathcal{G}, \mathcal{R}) \to \mathsf{DTMs}(\mathcal{H}, \mathcal{S})$$

such that for all $\mathfrak{N} \in \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$

- $t$ preserves and reflects $\mathcal{L}_{\Box\bigcirc}(\Pi)$ theories (pointwise and globally):

$$(\mathfrak{N}, y) \models \varphi \iff (\tau(\mathfrak{N}), z) \models \varphi \qquad (y \in |\mathcal{G}|,\ z \in t(y),\ \varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi))$$
$$\mathfrak{N} \models \varphi \iff \tau(\mathfrak{N}) \models \varphi \qquad (\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi))$$

- $t$ preserves and reflects $\mathcal{L}_{\bigcirc}(\Sigma)$ theories (pointwise):

$$(\mathfrak{N}, y) \models \varphi \iff (\tau(\mathfrak{N}), z) \models \varphi \qquad (y \in |\mathcal{G}|,\ z \in t(y),\ \varphi \in \mathcal{L}_{\bigcirc}(\Sigma).)$$

*Proof.* —

To define $\tau$, first begin by noticing that $\%t$ induces a theory-preserving bijection

$$\lambda : \mathsf{DTMs}(\mathcal{G}/\mathcal{R}) \to \mathsf{DTMs}(\mathcal{H}/\mathcal{S})$$

given by

$$V_{\lambda(\mathfrak{M})}(p) = \%t(V_{\mathfrak{M}}(p)).$$

Check that this is indeed a bijection (since $\%t$ is a bijection) and that for any $(y, z) \in t$,

$$(\mathfrak{M}, [y]_{\mathcal{R}}) \models \varphi \quad \Longleftrightarrow \quad (\lambda(\mathfrak{M}), [z]_{\mathcal{S}}) \models \varphi$$

for any $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)$. This is an application of Lemma 4.3, since $\%t$ is a $\Pi$-bisimulation which, by definition of $V_{\lambda(\mathfrak{M})}$, satisfies the (Base) condition.

Furthermore, note that

$$\mathfrak{M} \models \varphi \quad \Longleftrightarrow \quad \lambda(\mathfrak{M}) \models \varphi.$$

This is due to the totality and surjectivity of $\%t$: if $(\mathfrak{M}, [y]_{\mathcal{R}}) \models \varphi$ for all $[y]_{\mathcal{R}}$, then use the result above to conclude that $(\lambda(\mathfrak{M}), [z]_{\mathcal{S}}) \models \varphi$ for all $[z]_{\mathcal{S}}$ in the image of $\%t$, which is to say, all $[z]_{\mathcal{S}}$. Similarly for the other direction.

Now, recalling Lemma 3.4, define

$$\tau = \kappa_{\mathcal{S}} \circ \lambda \circ \kappa_{\mathcal{R}}^{-1}.$$

The right-hand side is the composition of three bijections, and is thus a bijection. So we have that $\tau$ is a bijection. Moreover, it preserves theories: for any $(y, z) \in t$ and any $\varphi \in \mathcal{L}_{\Box\bigcirc}(\Pi)$,

$$\begin{aligned}
(\mathfrak{N}, y) \models \varphi &\Longleftrightarrow (\kappa_{\mathcal{R}}^{-1}(\mathfrak{N}), [y]_{\mathcal{R}}) \models \varphi && \text{(Lemma 3.4)}\\
&\Longleftrightarrow (\lambda(\kappa_{\mathcal{R}}^{-1}(\mathfrak{N})), [z]_{\mathcal{S}}) \models \varphi && \text{(above)}\\
&\Longleftrightarrow (\kappa_{\mathcal{S}}(\lambda(\kappa_{\mathcal{R}}^{-1}(\mathfrak{N}))), z) \models \varphi && \text{(Lemma 3.4)}\\
&\Longleftrightarrow (\tau(\mathfrak{N}), z) \models \varphi && \text{(defn. } \tau\text{.)}
\end{aligned}$$

as desired. We can also obtain

$$\mathfrak{N} \models \varphi \quad \Longleftrightarrow \quad \tau(\mathfrak{N}) \models \varphi$$

by concatenating the above result with Lemma 3.4:

$$\begin{aligned}
\mathfrak{N} \models \varphi &\Longleftrightarrow \kappa_{\mathcal{R}}^{-1}(\mathfrak{N}) \models \varphi && \text{(Lemma 3.4)}\\
&\Longleftrightarrow \lambda(\kappa_{\mathcal{R}}^{-1}(\mathfrak{N})) \models \varphi && \text{(above)}\\
&\Longleftrightarrow \kappa_{\mathcal{S}}(\lambda(\kappa_{\mathcal{R}}^{-1}(\mathfrak{N}))) \models \varphi && \text{(Lemma 3.4)}
\end{aligned}$$

as desired.

Finally, to see that $t$ preserves and reflects $\mathcal{L}_{\bigcirc}(\Sigma)$ theories, proceed by structural induction. The base case is already covered above, and, as usual, the $\wedge$ and $\neg$ inductive steps are trivial. So it suffices to prove the $\bigcirc_\sigma$ inductive step.

For some $\psi \in \mathcal{L}_{\bigcirc}(\Sigma)$, suppose for any $(y, z) \in t$ that

$$(\mathfrak{N}, y) \models \psi \quad \Longleftrightarrow \quad (\tau(\mathfrak{N}), z) \models \psi.$$

Now suppose $(\mathfrak{N}, y) \models \bigcirc_\sigma \psi$ for some $\sigma \in \Sigma$. So $\|\sigma\|_{\mathfrak{N}}(y)$ is defined and is a $\psi$-world. Since $t$ respects $\sigma$, we get that $\|\sigma\|_{\tau(\mathfrak{N})}(z)$ exists, is $t$-related to $\|\sigma\|_{\mathfrak{N}}(y)$, and, by the inductive hypothesis, is a $\psi$-world. So $(\tau(\mathfrak{N}), z) \models \bigcirc_\sigma \psi$, as desired. The other direction proceeds similarly: $(\tau(\mathfrak{N}), z) \models \bigcirc_\sigma \psi$ implies that $\|\sigma\|_{\tau(\mathfrak{N})}(z)$ must exist and be a $\psi$-world, so, since $t$ respects $\sigma$, we have by the inductive hypothesis that $\|\sigma\|_{\mathfrak{N}}(y)$ must exist and be a $\psi$-world, so $(\mathfrak{N}, y) \models \bigcirc_\sigma \psi$, and we're done.

$\square$

(**Prop. 5.1**)
For any $\Pi$-frame $\mathcal{F}$,

$$\mathrm{Th}_{\square\bigcirc}\left(\mathsf{or}^{<\omega}(\Pi)\,;\mathcal{F}^{\mathsf{OR}\{<\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{<\omega\}}\right) \quad = \quad \mathrm{Th}_{\square\bigcirc}\left(\mathsf{or}^{<\omega}(\Pi)\,;\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\leq\omega\}}\right)$$

i.e. the language $\mathcal{L}_{\square\bigcirc}(\mathsf{or}^{<\omega}(\Pi))$ cannot distinguish the results of the program constructors $\mathsf{OR}\{<\omega\}$ and $\mathsf{OR}\{\leq\omega\}$.

*Proof.* —

$\square$

(**Prop. 5.2**)
For each $\Pi$-frame $\mathcal{F}$, the inclusion function

$$\begin{aligned} |\mathcal{F}| \times \{0,1\}^{<\omega} &\to |\mathcal{F}| \times \{0,1\}^{\leq\omega} \\ (x, s) &\mapsto (x, s) \end{aligned}$$

is a strong embedding

$$\left(\mathcal{F}^{\mathsf{OR}\{<\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{<\omega\}}\right) \hookrightarrow \left(\mathcal{F}^{\mathsf{OR}\{\leq\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\leq\omega\}}\right).$$

*Proof.* —

$\square$

# B Characterizations

**Definition B.1**
Given a $(\Pi, \Sigma)$-refined frame $(\mathcal{G}, \mathcal{R})$ and worlds $w, w'$, write

$$w \approx_{\mathcal{R}} w'$$

to mean that $w$ and $w'$ are $\mathcal{R}$-related. We then extend this to allow either side to be undefined: for $\sigma, \sigma' \in \Sigma$, write

$$\|\sigma\|_{\mathcal{G}}(w) \approx_{\mathcal{R}} \|\sigma'\|(w')$$

to mean that either

- $\|\sigma\|_{\mathcal{G}}$ is undefined at $w$ and $\|\sigma'\|_{\mathcal{G}}$ is undefined at $w'$

| | | |
|---|---|---|
| (OR-Typ) | $\mathsf{Maybe}_0(p, \pi_0, \pi_1) \vee \mathsf{Maybe}_1(p, \pi_0, \pi_1)$ | Typicality |
| (OR-Reg0) | $\mathsf{Only}_0(p, \pi_0, \pi_1) \rightarrow \mathsf{Maybe}_0(q, \pi_2, \pi_3)$ | Regularity 0 |
| (OR-Reg1) | $\mathsf{Only}_1(p, \pi_0, \pi_1) \rightarrow \mathsf{Maybe}_1(q, \pi_2, \pi_3)$ | Regularity 1 |
| (OR-Real0) | $p \wedge \mathsf{Only}_1(q, \pi_0, \pi_1) \rightarrow \Diamond(p \wedge \mathsf{Only}_0(q, \pi_0, \pi_1))$ | Realization 0 |
| (OR-Real1) | $p \wedge \mathsf{Only}_0(q, \pi_0, \pi_1) \rightarrow \Diamond(p \wedge \mathsf{Only}_1(q, \pi_0, \pi_1))$ | Realization 1 |
| (OR-Refresh) | $p \rightarrow \bigcirc_{(\mathsf{skip\ or\ skip})} p$ | Refresh |
| (OR-Nest0) | $\mathsf{Only}_0(p, \pi_0, \pi_1) \rightarrow \bigcirc_{\mathsf{skiporskip}} \bigcirc_{\sigma_0} \varphi \rightarrow \bigcirc_{\sigma_0 \mathsf{or} \sigma_1} \varphi$ | Nesting 0 |
| (OR-Nest1) | $\mathsf{Only}_1(p, \pi_0, \pi_1) \rightarrow \bigcirc_{\mathsf{skiporskip}} \bigcirc_{\sigma_1} \varphi \rightarrow \bigcirc_{\sigma_0 \mathsf{or} \sigma_1} \varphi$ | Nesting 1 |
| (OR-PrimInd) | $\bigcirc_{(\mathsf{skip\ or\ skip})} \bigcirc_{\pi_0} \varphi \leftrightarrow \bigcirc_{\pi_0} \bigcirc_{(\mathsf{skip\ or\ skip})} \varphi$<br>$\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort}) \rightarrow \neg \bigcirc_{\pi_0} \neg \mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$<br>$\mathsf{Only}_1(\top, \mathsf{skip}, \mathsf{abort}) \rightarrow \neg \bigcirc_{\pi_0} \neg \mathsf{Only}_1(\top, \mathsf{skip}, \mathsf{abort})$ | Primitive Independence |

Figure B.1: $\chi_{\mathsf{OR}\{\omega\}}$. $p, q$ vary over $\Phi \cup \{\top, \bot\}$, $\pi_0, \pi_1, \pi_2, \pi_3$ vary over $\Pi^\dagger$, $\sigma_0, \sigma_1$ vary over $\mathsf{or}^{<\omega}(\Pi^\dagger)$, and $\varphi$ varies over $\mathcal{L}_{\Box\bigcirc}(\mathsf{or}^{<\omega}(\Pi^\dagger))$. Recall the abbreviations of Defn. 1.4.

- both $\|\sigma\|_{\mathcal{G}}(w)$ and $\|\sigma'\|_{\mathcal{G}}(w')$ are defined, and moreover they are $\mathcal{R}$-related:

$$\big(\|\sigma\|_{\mathcal{G}}(w), \|\sigma'\|_{\mathcal{G}}(w')\big) \in \mathcal{R}.$$

**(Prop. 5.3)**

$\chi_{\mathsf{OR}\{\omega\}}$ (as given in Fig. B.1) *right-characterizes* $\mathsf{OR}\{\omega\}$.

*Proof.* —

We'll address each axiom scheme in turn, proving that (a) the refined frames produced by $\mathsf{OR}\{\omega\}$ satisfy the axioms, and (b) that any $(\Pi, \mathsf{or}^{<\omega}(\Pi^\dagger))$-refined frame $(\mathcal{G}, \mathcal{R})$ validating the axioms must be structured like an $\mathsf{OR}\{\omega\}$-augmented frame. The precise meaning of "structured *like*" will ultimately be the existence of a strong embedding of refined frames.

Start with (OR-Typ). The central function of the or construct is to select one of two programs. The main thing we'll be doing in this proof is codifying in the object language what it means to "select" between programs, and the high-level process for how this selection takes place in a $\mathsf{OR}\{\omega\}$-augmented frame. For the moment, we'll suppose the programs we're selecting between are primitives, $\pi_0$ and $\pi_1$. For some atomic proposition $p$, the formula $\mathsf{Maybe}_0(p, \pi_0, \pi_1)$ asserts that, as far as the truth or falsity of $p$ is concerned, $\pi_0 \mathsf{\ or\ } \pi_1$ *might be* $\pi_0$. Notably, it could be the case that $\pi_0 \mathsf{\ or\ } \pi_1$ *actually* ends up being $\pi_1$, but the world resulting from $\pi_0$ and the world resulting from $\pi_1$ just happen to either both satisfy or both refute $p$ (or both $\pi_0$ and $\pi_1$ are undefined). The formula $\mathsf{Maybe}_1(p, \pi_0, \pi_1)$, of course, encodes the analogous statement between $\pi_1$ and $\pi_0 \mathsf{\ or\ } \pi_1$.

Requiring this axiom for all $p$ and all $\pi_0, \pi_1$ begins to formalize our intuition that or "selects". Consider the following claim.

**Claim 1**

Given a refined frame $(\mathcal{G}, \mathcal{R})$ validating all instances of (OR-Typ), a world $w$ of $\mathcal{G}$, and $\pi_0, \pi_1 \in \Pi^\dagger$,

$$\|\pi_0 \mathsf{\ or\ } \pi_1\|_{\mathcal{G}}(w) \approx_{\mathcal{R}} \|\pi_0\|_{\mathcal{G}}(w) \qquad \text{or} \qquad \|\pi_0 \mathsf{\ or\ } \pi_1\|_{\mathcal{G}}(w) \approx_{\mathcal{R}} \|\pi_1\|_{\mathcal{G}}(w).$$

So, up to $\mathcal{R}$-equivalence, $\pi_0 \mathsf{\ or\ } \pi_1$ takes you to the same place as either $\pi_0$ or $\pi_1$. Note that this 'or' is *not* exclusive: the $\approx_{\mathcal{R}}$ relation is an equivalence relation, so if it happens to be the case that $\|\pi_0\|_{\mathcal{G}}(x) \approx_{\mathcal{R}} \|\pi_1\|(x)$, then both disjuncts of Claim 1 will hold. To express this more concisely, define a function

$$\mathsf{PreType} : |\mathcal{G}| \times \Pi^{\dagger} \times \Pi^{\dagger} \to \mathcal{P}(\{0,1\})$$

such that

$$j \in \mathsf{PreType}(x, \pi_0, \pi_1) \quad \iff \quad \|\pi_0 \text{ or } \pi_1\|_{\mathcal{G}}(x) \approx_{\mathcal{R}} \|\pi_j\|_{\mathcal{G}}(x) \qquad (j \in \{0,1\})$$

The point of (OR-Typ) is to guarantee $\mathsf{PreType}(x, \pi_0, \pi_1) \neq \emptyset$.

Refined frames produced by $\mathsf{OR}\{\omega\}$ obey this requirement. Pick a world $(x, \gamma)$ of $\mathcal{F}^{\mathsf{OR}\{\omega\}}$ and consider $\mathsf{hd}(\gamma)$. Since $\gamma \in \{0,1\}^{\omega}$, $\mathsf{hd}(\gamma)$ must either be 0 or 1. Suppose $\mathsf{hd}(\gamma) = 0$. Then,

$$\begin{aligned}
\|\pi_0 \text{ or } \pi_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(x, \gamma) &\approx_{\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}} (\|\pi_0\|_{\mathcal{F}}(x), \mathsf{tl}(\gamma)) && \text{(Defn of } \mathsf{OR}\{\omega\}) \\
&\approx_{\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}} (\|\pi_0\|_{\mathcal{F}}(x), \gamma) && \text{(Defn of } \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}) \\
&\approx_{\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}} \|\pi_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(x, \gamma) && \text{(Defn of } \mathsf{OR}\{\omega\})
\end{aligned}$$

Identical logic with $\mathsf{hd}(\gamma) = 1$ will yield $\|\pi_0 \text{ or } \pi_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(x, \gamma) \approx_{\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}} \|\pi_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(x, \gamma)$. Either way, by some simple reasoning with $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}$-respecting valuations, this implies that $\left(\mathcal{F}^{\mathsf{OR}\{\omega\}}, \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}\right)$ validates all instances of (OR-Typ).

So (OR-Typ) guarantees that $\mathsf{PreType} : |\mathcal{G}| \times \Pi^{\dagger} \times \Pi^{\dagger} \to \mathcal{P}(\{0,1\})$ is well-defined and never returns $\emptyset$. But notice that PreType depends on its primitive program arguments: *a priori*, there's nothing preventing, say,

$$\mathsf{PreType}(w, \pi_0, \pi_1) = \{0\} \qquad \text{and} \qquad \mathsf{PreType}(w, \pi_2, \pi_3) = \{1\}.$$

I.e. in world $w$, $\pi_0 \text{ or } \pi_1$ is interpreted as $\pi_0$ but $\pi_2 \text{ or } \pi_3$ is interpreted as $\pi_3$. But this is not how $\mathsf{OR}\{\omega\}$-augmented frames operate: in a world $(x, \gamma)$, either the first "disjunct" is chosen (i.e. $\mathsf{hd}(\gamma) = 0$, so $\pi_0 \text{ or } \pi_1$ is $\pi_0$ and $\pi_2 \text{ or } \pi_3$ is $\pi_2$) or the second one (i.e. $\mathsf{hd}(\gamma) = 1$, so $\pi_0 \text{ or } \pi_1$ is $\pi_1$ and $\pi_2 \text{ or } \pi_3$ is $\pi_3$). The choice of program is indifferent two *which* programs are being chosen between.

So now consider (OR-Reg0): it asserts that if a world validates $\mathsf{Only}_0(p, \pi_0, \pi_1)$ for some $p, \pi_0, \pi_1$, then it validates $\mathsf{Maybe}_0(q, \pi_2, \pi_3)$ for arbitrary $q, \pi_2, \pi_3$. And then (OR-Reg1) makes the analogous assertion for $\mathsf{Only}_1$. Assuming all instances of these axioms makes $\mathsf{Only}_0$ and $\mathsf{Only}_1$ into powerful enough assertions to prove what we want.

**Claim 2**

Suppose $(\mathcal{G}, \mathcal{R})$ is a refined frame validating all instances of (OR-Typ), (OR-Reg0), and (OR-Reg1). Then, for each world $w \in |\mathcal{G}|$, exactly one of the following holds:

- $w$ is a **0-world**: for all $\pi, \pi' \in \Pi^{\dagger}$,

$$0 \in \mathsf{PreType}(w, \pi, \pi')$$

- $w$ is a **1-world**: for all $\pi, \pi' \in \Pi^{\dagger}$,

$$1 \in \mathsf{PreType}(w, \pi, \pi')$$

*Proof.* —

First we show that each world must either be a 0-world or a 1-world. Suppose not. Then we have a world $w$ and primitive programs $\pi_0, \pi_1, \pi_2, \pi_3$ such that

$$\mathsf{PreType}(w, \pi_0, \pi_1) = \{0\} \qquad \text{and} \qquad \mathsf{PreType}(w, \pi_2, \pi_3) = \{1\}\,.$$

So

$$\begin{aligned}
\|\pi_0 \text{ or } \pi_1\|(w) &\approx_{\mathcal{R}} \|\pi_0\|(w)\\
\|\pi_0 \text{ or } \pi_1\|(w) &\not\approx_{\mathcal{R}} \|\pi_1\|(w)\\
\|\pi_2 \text{ or } \pi_3\|(w) &\approx_{\mathcal{R}} \|\pi_3\|(w)\\
\|\pi_2 \text{ or } \pi_3\|(w) &\not\approx_{\mathcal{R}} \|\pi_2\|(w)
\end{aligned}$$

Pick $p \neq q$ and define an $\mathcal{R}$-respecting valuation by:

$$V(p) = \begin{cases} \mathcal{R}(\|\pi_0\|(w)) & \text{if } \|\pi_0\|(w) \text{ is defined}\\ \mathcal{R}(\|\pi_1\|(w)) & \text{otherwise}\end{cases}$$
$$V(q) = \begin{cases} \mathcal{R}(\|\pi_2\|(w)) & \text{if } \|\pi_2\|(w) \text{ is defined}\\ \mathcal{R}(\|\pi_2 \text{ or } \pi_3\|(w)) & \text{otherwise}\end{cases}$$

These cases are exhaustive: $\|\pi_0\|(w) \not\approx_{\mathcal{R}} \|\pi_1\|(w)$, so if $\|\pi_0\|(w)$ is undefined, $\|\pi_1\|(w)$ must be. $\|\pi_2\|(w) \not\approx_{\mathcal{R}} \|\pi_2 \text{ or } \pi_3\|(w)$, so if $\|\pi_2\|(w)$ is undefined, then $\|\pi_2 \text{ or } \pi_3\|(w)$ must be.

Observe that, under this valuation, $w$ validates $\mathsf{Only}_0(p, \pi_0, \pi_1)$: if $\|\pi_0\|(w)$ is defined, then $\|\pi_1\|(w)$ is either undefined or not in $V(p)$, so the first disjunct of $\mathsf{Only}_0(p, \pi_0, \pi_1)$ is satisfied. If $\|\pi_0\|(w)$ is not defined, then neither is $\|\pi_0 \text{ or } \pi_1\|(w)$, but $\|\pi_1\|(w)$ is defined and in $V(p)$ by definition, validating the second disjunct of $\mathsf{Only}_0(p, \pi_0, \pi_1)$.

However, $w$ refutes $\mathsf{Maybe}_0(q, \pi_2, \pi_3)$. If $\|\pi_2\|(w)$ is defined, then by definition it is in $V(q)$, so $w$ satisfies $\bigcirc_{\pi_2} q$. But $\|\pi_2 \text{ or } \pi_3\|(w) \not\approx_{\mathcal{R}} \|\pi_2\|(w)$, so $w$ refutes $\bigcirc_{\pi_2 \text{or} \pi_3} q$, defeating $\mathsf{Maybe}_0(q, \pi_2, \pi_3)$. On the other hand, if $\|\pi_2\|(w)$ is undefined, $w$ refutes $\bigcirc_{\pi_2} q$ automatically. But in that case $\|\pi_2 \text{ or } \pi_3\|(w)$ is defined and is a $q$-world by definition of $V$, again refuting $\mathsf{Maybe}_0(q, \pi_2, \pi_3)$. So, no matter what, we have refuted (OR-Reg0). By contradiction, conclude that every world must either be a 0-world or a 1-world.

Finally, let us see that $w$ cannot be both a 0-world and a 1-world (i.e. that it cannot be that $\mathsf{PreType}(w, \pi, \pi') = \{0, 1\}$ *for all* $\pi, \pi' \in \Pi^\dagger$). To see this, it suffices to consider the program

$$\mathsf{skip\ or\ abort}$$

If $w$ is a 0-world, then it "selects" the first disjunct and

$$\|\mathsf{skip\ or\ abort}\|(w) = w.$$

On the other hand, if $w$ is a 1-world, then it "selects" the second disjunct and

$$\|\mathsf{skip\ or\ abort}\|(w) \text{ is undefined.}$$

Clearly, these are incompatible, so $w$ cannot be both.

□(Claim 2)

This guarantees that the following function is well-defined and total.

$$\begin{aligned}\mathsf{Type} &: |\mathcal{G}| \to \{0,1\}\\ &: w \mapsto \begin{cases} 0 & \text{if } \|\pi \text{ or } \pi'\|_{\mathcal{G}}(w) \approx_{\mathcal{R}} \|\pi\|_{\mathcal{G}}(w) \text{ for all } \pi, \pi' \in \Pi^{\dagger} \\ 1 & \text{if } \|\pi \text{ or } \pi'\|_{\mathcal{G}}(w) \approx_{\mathcal{R}} \|\pi'\|_{\mathcal{G}}(w) \text{ for all } \pi, \pi' \in \Pi^{\dagger} \end{cases}\end{aligned}$$

So if $w$ is a 0-world, $\mathsf{Type}(w) = 0$; if $w$ is a 1-world, $\mathsf{Type}(w) = 1$. Let us state one helpful lemma.

**Lemma B.1**

Let $(\mathcal{G}, \mathcal{R})$ be a refined frame validating all instances of (OR-Typ), (OR-Reg0), and (OR-Reg1). Then, for any $w \in |\mathcal{G}|$ and any $q \in \Phi \cup \{\top, \bot\}$, the following are equivalent.

(1) $\mathsf{Type}(w) = 0$

(2) There exists $\pi_0, \pi_1 \in \Pi$ and an $\mathcal{R}$-respecting valuation $V$ such that

$$((\mathcal{G}, V), w) \models \mathsf{Only}_0(q, \pi_0, \pi_1).$$

(3) For all $\mathfrak{N} \in \mathsf{DTMs}(\mathcal{G}, \mathcal{R})$,

$$(\mathfrak{N}, w) \models \bigcirc_{(\mathsf{skip\ or\ abort})} \top.$$

And analogously for $\mathsf{Type}(w) = 1$, $\mathsf{Only}_1(q, \pi_0, \pi_1)$, and $\neg\bigcirc_{(\mathsf{skip\ or\ abort})}\top$ (or $\bigcirc_{(\mathsf{abort\ or\ skip})}\top$).

As mentioned, $\mathsf{OR}\,\{\omega\}$-augmented refined frames will satisfy this. Worlds of the form $(x, 0\gamma)$ will be 0-worlds and worlds of the form $(x, 1\gamma)$ will be 1-worlds, essentially by definition. It's easy to check that 0-worlds will validate every $\mathsf{Maybe}_0$ formula, and 1-worlds every $\mathsf{Maybe}_1$. Furthermore, in light of Lemma B.1, we see that *only* the 0-worlds will validate $\mathsf{Only}_0$ formulas and *only* the 1-worlds will validate $\mathsf{Only}_1$ formulas. So every instance of the Regularity axioms will be satisfied on $\mathsf{OR}\,\{\omega\}$-augmented frames.

So we have that every world is either a 0-world or a 1-world, and not both. The next feature of $\mathsf{OR}\,\{\omega\}$-augmented frames we'll need to encode is that 0-worlds and 1-worlds "come in pairs": the world $(x, 0\gamma)$ is a 0-world, and its "twin" $(x, 1\gamma)$ is a 1-world. The existence of both "possibilities" is the central feature of these frames. The (OR-Real) axioms will guarantee that every $\mathcal{R}$-equivalence class contains both a 0-world and a 1-world.

**Claim 3**

If $(\mathcal{G}, \mathcal{R})$ is a refined frame satisfying all instances of the (OR-Typ), (OR-Reg), and (OR-Real) axioms, then every $\mathcal{R}$-equivalence class $U$ contains both a 0-world and a 1-world.

*Proof.* —

Pick arbitrary $U$. By the previous lemmas and claims about (OR-Typ) and (OR-Reg), we have that every world $w \in U$ is either a 0-world or a 1-world. We'll suppose we have $w \in U$ with $\mathsf{Type}(w) = 0$ and use (OR-Real1) to find a $w' \in U$ with $\mathsf{Type}(w') = 1$. An identical argument can be made using (OR-Real0) to obtain a 0-world in $U$ from the existence of a 1-world in $U$, completing the argument.

Let $w$ be some arbitrary 0-world in $U$, $p$ some atomic proposition. Let $V$ be an $\mathcal{R}$-respecting valuation on $\mathcal{G}$ which puts

$$V(p) = U.$$

So all (and only) the worlds in $U$ validate $p$. Then, observe that the following is an instance of (OR-Real1)

$$p \wedge \mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort}) \rightarrow \Diamond(p \wedge \mathsf{Only}_1(\top, \mathsf{skip}, \mathsf{abort})).$$

Now, we can see that, under the valuation $V$, $w$ validates the antecedent: we stipulated $V$ to be such that $w$ validates $p$ and $\|\mathsf{skip\ or\ abort}\|(w) = w$ because $w$ is a 0-world, so we can see that the first disjunct of $\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$ is satisfied at $w$. Therefore $w$ must validate the consequent of this instance of (OR-Real1):

$$((\mathcal{G}, V), w) \models \Diamond(p \wedge \mathsf{Only}_1(\top, \pi_0, \pi_1)).$$

So $w$ must be in the closure of $[\![p]\!] \cap [\![\mathsf{Only}_1(\top, \pi_0, \pi_1)]\!]$, which implies that the latter is nonempty. So we obtain a world

$$w' \in [\![p]\!] \cap [\![\mathsf{Only}_1(\top, \pi_0, \pi_1)]\!].$$

However, notice that $[\![p]\!] = U$, so $w' \in U$. Since $w'$ validates $\mathsf{Only}_1(\top, \pi_0, \pi_1)$ for some $\pi_0, \pi_1$ and some valuation $V$, we obtain from Lemma B.1

$$\mathsf{Type}(w') = 1$$

as desired.

$\square$(Claim 3)

So every $\mathcal{R}$ equivalence class contains a world $w$ such that $\mathsf{Type}(w) = 0$ and a world $w'$ such that $\mathsf{Type}(w') = 1$. As mentioned, $\mathsf{OR}\{\omega\}$-augmented frames will possess this property: the $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}$-equivalence classes are all of the form

$$\{(x, S) \ : \ S \in \{0,1\}^{\omega}\}$$

for each $x \in |\mathcal{F}|$. This equivalence class contains a multitude of 0-worlds (all those $(x, S)$ such that $\mathsf{hd}(S) = 0$) and 1-worlds (all those $(x, S)$ such that $\mathsf{hd}(S) = 1$). And so if $p, q, \pi_0, \pi_1$, and the valuation are such that some 0-world $(x, 0\gamma)$ validates $p \wedge \mathsf{Only}_0(q, \pi_0, \pi_1)$, it is quick to check that $(x, 1\gamma)$ will validate $p \wedge \mathsf{Only}_1(q, \pi_0, \pi_1)$, and vice versa. So the Realization axioms are satisfied as well.

So far, we have only been discussing or-ing together primitive programs (plus skip and abort), and have obtained a fairly robust description of how these frames resolve a single or. In order to extend this to nested or's, start by considering the program (skip or skip). In a $\mathsf{OR}\{\omega\}$-augmented frame, executing this program takes one from $(x, \gamma)$ to $(x, \mathsf{tl}(\gamma))$, using up a "bit" from $\gamma$, but staying within the same $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}$-equivalence class. The axiom scheme (OR-Refresh) encodes this for an arbitrary refined frames.

**Claim 4**

If $(\mathcal{G}, \mathcal{R})$ is a refined frame satisfying (OR-Refresh), then for every $w \in |\mathcal{G}|$,

$$(w, \|\mathsf{skip\ or\ skip}\|(w)) \in \mathcal{R}.$$

Notice that, as a corollary of this claim, we have that $\|\mathsf{skip\ or\ skip}\|$ is total – this will be relevant momentarily. So in any refined frame validating all the axioms so far, we have an assignment of a "type" (0 or 1) to each world, and a function $\|\mathsf{skip\ or\ skip}\|$ which permutes each $\mathcal{R}$-equivalence class. This allows us to make the following definition.

**Definition B.2**

Given $(\mathcal{G}, \mathcal{R})$ satisfying (OR-Typ), (OR-Reg), (OR-Real), and (OR-Refresh), define

$$\mathsf{FullType} : |\mathcal{G}| \to \{0,1\}^\omega$$

by

$$\mathsf{FullType}(w)(n) = \mathsf{Type}(\|\mathsf{skip\ or\ skip}\|^n (w))$$

So for each world $w$ of such a frame, $\mathsf{FullType}(w)$ is an infinite[6] sequence of 0's and 1's, spelling out what type of worlds will be encountered by repeatedly executing $\mathsf{skip\ or\ skip}$. Of course, for $\mathsf{OR}\{\omega\}$-augmented frames, the $\mathsf{FullType}$ of a world is just its program constructor state:

**Proposition B.2**

For any $\Pi$-frame $\mathcal{F}$, any $x \in |\mathcal{F}|$, and any $S \in \{0,1\}^\omega$,

$$\mathsf{FullType}(x, S) = S.$$

Now, all that's left is for us to require that the $\mathsf{FullType}$ of a world in an arbitrary refined frame specifies the behavior of nested $\mathsf{or}$'s the same way as it does in an $\mathsf{OR}\{\omega\}$-augmented frame. This will be the role of (OR-Nest) and (OR-PrimInd).

**Lemma B.3**

Suppose $(\mathcal{G}, \mathcal{R})$ validates all of $\chi_{\mathsf{OR}}$, and let $\sigma_0, \sigma_1 \in \mathsf{or}^{<\omega}(\Pi^\dagger)$. Then,

- for all 0-worlds $w_0$ of $\mathcal{G}$,

$$\|\sigma_0 \mathsf{\ or\ } \sigma_1\|_\mathcal{G}(w_0) \approx_\mathcal{R} \|\sigma_0\|_\mathcal{G}(\|\mathsf{skip\ or\ skip}\|_\mathcal{G}(w_0));$$

- and for all 1-worlds $w_1$ of $\mathcal{G}$,

$$\|\sigma_0 \mathsf{\ or\ } \sigma_1\|_\mathcal{G}(w_1) \approx_\mathcal{R} \|\sigma_1\|_\mathcal{G}(\|\mathsf{skip\ or\ skip}\|_\mathcal{G}(w_1)).$$

So what this lemma says is that, like in $\mathsf{OR}\{\omega\}$-augmented frames, the execution of arbitrary $\sigma_0 \mathsf{\ or\ } \sigma_1$ from $w$ consists of (a) reading off the first bit of $\mathsf{FullType}(w)$ to see whether $w$ is a 0-world or a 1-world, (b) "throwing out" that bit (taking us to $\|\mathsf{skip\ or\ skip}\|(w)$, which has $\mathsf{FullType}$ equal to the tail of $\mathsf{FullType}(w)$), and then (c) executing $\sigma_0$ or $\sigma_1$ accordingly. So we've *almost* required all the salient properties. But we need two more requirements: (1) execution of primitives doesn't touch the $\mathsf{FullType}$ at all, and (2) executing $\sigma_0 \mathsf{\ or\ } \sigma_1$ uses up exactly 1 "bit" (to select $\sigma_0$ or $\sigma_1$), plus however many bits are required to execute whichever of the two we select. This is formally stated in the following lemma.

**Lemma B.4**

For any $(\mathcal{G}, \mathcal{R})$ validating all of $\chi_{\mathsf{OR}}$, any world $w$ of $\mathcal{G}$, and any $\pi \in \Pi^\dagger$ such that $\|\pi\|_\mathcal{G}(w)$ is defined,

$$\mathsf{FullType}(\|\pi\|_\mathcal{G}(w)) = \mathsf{FullType}(w).$$

For any $\sigma_0, \sigma_1 \in \mathsf{or}^{<\omega}(\Pi^\dagger)$,

$$\mathsf{FullType}(\|\sigma_0 \mathsf{\ or\ } \sigma_1\|_\mathcal{G}(w)) = \begin{cases} \mathsf{FullType}(\|\sigma_0\|_\mathcal{G}(\|\mathsf{skip\ or\ skip}\|_\mathcal{G}(w))) & \text{if } \mathsf{Type}(w) = 0 \\ \mathsf{FullType}(\|\sigma_1\|_\mathcal{G}(\|\mathsf{skip\ or\ skip}\|_\mathcal{G}(w))) & \text{if } \mathsf{Type}(w) = 1 \end{cases}$$

if the worlds in question are defined.

[6] This is where we use the totality of $\|\mathsf{skip\ or\ skip}\|$.

*Proof.* —

Let's begin with the first claim, about how $\mathsf{FullType}$ is preserved in the execution of a primitive program. To begin with, the claim that

$$\mathsf{Type}(\|\pi\|_{\mathcal{G}}(w)) = \mathsf{Type}(w)$$

whenever $\|\pi\|(w)$ is defined is (almost) directly asserted by the latter two ($\mathsf{OR\text{-}PrimInd}$) axioms. They say that execution of $\pi \in \Pi^{\dagger}$ cannot take one from a 0-world to a 1-world, or vice versa. So execution of primitives preserves $\mathsf{Type}$. To extend this to $\mathsf{FullType}$, suppose $\mathsf{FullType}(\|\pi\|(w)) \neq \mathsf{FullType}(w)$. Then there must be some $n > 0$[7] such that, without loss of generality,

$$\mathsf{FullType}(\|\pi\|(w))(n) = 1 \qquad \text{and} \qquad \mathsf{FullType}(w)(n) = 0.$$

So then let $\varphi$ be

$$\underbrace{\bigcirc_{(\mathsf{skip\ or\ skip})} \bigcirc_{(\mathsf{skip\ or\ skip})} \cdots \bigcirc_{(\mathsf{skip\ or\ skip})}}_{n-1} \mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort}).$$

We'll refute the instance of the first ($\mathsf{OR\text{-}PrimInd}$) axiom with this as our $\varphi$.

Regardless of the valuation, $w$ will validate $\bigcirc_{(\mathsf{skip\ or\ skip})} \bigcirc_{\pi} \varphi$. Since $\mathsf{FullType}(w)(n) = 0$, we have by definition of $\mathsf{FullType}$ that $\|\mathsf{skip\ or\ skip}\|^n(w)$ is a 0-world, i.e.

$$\mathsf{Type}(\|\mathsf{skip\ or\ skip}\|^n(w)) = 0.$$

By the above, this tells us that

$$\mathsf{Type}(\|\pi\|(\|\mathsf{skip\ or\ skip}\|^n(w))) = 0.$$

And, therefore $\|\pi\|(\|\mathsf{skip\ or\ skip}\|^n(w))$ validates $\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$, and we conclude that $w$ validates $\bigcirc_{(\mathsf{skip\ or\ skip})} \bigcirc_{\pi} \varphi$.

However, $w$ refutes $\bigcirc_{\pi} \bigcirc_{(\mathsf{skip\ or\ skip})} \varphi$. By the hypothesis above, we know that

$$\mathsf{FullType}(\|\pi\|(w))(n) = 1,$$

i.e.

$$\mathsf{Type}(\|\mathsf{skip\ or\ skip}\|^n(\|\pi\|(w))) = 1.$$

Thus, $\|\mathsf{skip\ or\ skip}\|^n(\|\pi\|(w))$ is a 1-world and must refute $\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$. Therefore, we get that $w$ refutes

$$\bigcirc_{\pi} \bigcirc_{(\mathsf{skip\ or\ skip})} \varphi \equiv \bigcirc_{\pi} \underbrace{\bigcirc_{(\mathsf{skip\ or\ skip})} \bigcirc_{(\mathsf{skip\ or\ skip})} \cdots \bigcirc_{(\mathsf{skip\ or\ skip})}}_{n} \mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort}).$$

So the first ($\mathsf{OR\text{-}PrimInd}$) axiom is refuted at $w$, contrary to our assumption.

Let's move on to the second claim. Again, the valuation is unimportant here. Suppose without loss of generality that $\mathsf{Type}(w) = 0$ and assume for contradiction that $\sigma_0, \sigma_1$ are such that

$$\mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|(w)) \neq \mathsf{FullType}(\|\sigma_0\|(\|\mathsf{skip\ or\ skip}\|(w))).$$

[7] We know $n$ cannot be 0, since $\mathsf{FullType}(w)(0) = \mathsf{Type}(w)$.

Again, there must be some $n \in \mathbb{N}$ at which index the two sequences differ. Assume without loss of generality that

$$\mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|(w))(n) = 0 \qquad \text{and} \qquad \mathsf{FullType}(\|\sigma_0\|(\|\mathsf{skip} \text{ or } \mathsf{skip}\|(w)))(n) = 1.$$

We'll refute the following instance of (OR-Nest0):

$$\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort}) \rightarrow \bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})} \bigcirc_{\sigma_0} \varphi \rightarrow \bigcirc_{\sigma_0 \text{or} \sigma_1} \varphi$$

where $\varphi$ is given as

$$\underbrace{\bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})} \bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})} \cdots \bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})}}_{n} \mathsf{Only}_1(\top, \mathsf{skip}, \mathsf{abort}).$$

First, observe that $w$ validates $\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$, since it is a 0-world. Next, we see that $w$ validates $\bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})} \bigcirc_{\sigma_0} \varphi$. To see this, note that

$$\mathsf{FullType}(\|\sigma_0\|(\|\mathsf{skip} \text{ or } \mathsf{skip}\|(w)))(n) = 1$$

means $\|\mathsf{skip} \text{ or } \mathsf{skip}\|^n(\|\sigma_0\|(\|\mathsf{skip} \text{ or } \mathsf{skip}\|(w)))$ is a 1-world, and therefore satisfies $\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$. It follows that $\|\sigma_0\|(\|\mathsf{skip} \text{ or } \mathsf{skip}\|(w))$ validates $\varphi$, so $w$ validates $\bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})} \bigcirc_{\sigma_0} \varphi$.

Finally, we show that $w$ *refutes* $\bigcirc_{\sigma_0 \text{or} \sigma_1} \varphi$. This follows from the assumption that $\mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|(w))(n) = 0$. Again, we unfold this assumption to see that

$$\|\mathsf{skip} \text{ or } \mathsf{skip}\|^n(\|\sigma_0 \text{ or } \sigma_1\|(w)) \text{ is a 0-world}$$

and thus $\|\sigma_0 \text{ or } \sigma_1\|(w)$ cannot validate $\varphi$.

So we have that $w$ validates $\mathsf{Only}_0(\top, \mathsf{skip}, \mathsf{abort})$ and $\bigcirc_{(\mathsf{skip}\text{ or }\mathsf{skip})} \bigcirc_{\sigma_0} \varphi$ but not $\bigcirc_{\sigma_0 \text{or} \sigma_1} \varphi$, contrary to our assumption of (OR-Nest0). So conclude

$$\mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|(w)) = \mathsf{FullType}(\|\sigma_0\|(\|\mathsf{skip} \text{ or } \mathsf{skip}\|(w))).$$

□

This marks the last structural lemma we need to prove, and we can now finally conclude with our main result. To finish off the "soundness" portion of our proof – that any $\mathsf{OR}\{\omega\}$-augmented frame validates $\chi_{\mathsf{OR}}$ – it just remains to check (OR-Nest) and (OR-PrimInd). But this is pretty quick: for a 0-world $(x, 0\gamma)$ in an $\mathsf{OR}\{\omega\}$-augmented frame,

$$\|\sigma_0 \text{ or } \sigma_1\|(x, 0\gamma) = \|\sigma_0\|(x, \gamma) = \|\sigma_0\|(\|\mathsf{skip} \text{ or } \mathsf{skip}\|(x, 0\gamma))$$

so (OR-Nest0) will automatically be satisfied, and likewise for (OR-Nest1). For the latter two (OR-PrimInd) axioms: the fact that $\|\pi\|(x, S)$ is defined as $(\|\pi\|(x), S)$, i.e. the $S$ is left alone completely, means that executing primitives from a 0-world will always land you in a 0-world, and from a 1-world will always land in a 1-world. For the first (OR-PrimInd) axiom, one can easily verify that, in a $\mathsf{OR}\{\omega\}$-augmented frame,

$$\|\pi\| \circ \|\mathsf{skip} \text{ or } \mathsf{skip}\| = \|\mathsf{skip} \text{ or } \mathsf{skip}\| \circ \|\pi\|$$

satisfying the axiom. We have therefore proved the soundness part.

The "completeness" part – that every $\chi_{\mathsf{OR}}$-satisfying $(\mathcal{G}, \mathcal{R})$ can be embedded into an $\mathsf{OR}\{\omega\}$-augmented frame, is stated as the following claim.

**Claim 5**

Given $(\mathcal{G},\mathcal{R})$ validating all of $\chi_{\mathsf{OR}}$, the function

$$\begin{aligned} t : |\mathcal{G}| &\to \left|(\mathcal{G}/\mathcal{R})^{\mathsf{OR}\{\omega\}}\right| \\ &: w \mapsto ([w]_{\mathcal{R}}, \mathsf{FullType}(w)) \end{aligned}$$

is a strong embedding of refined frames

$$(\mathcal{G},\mathcal{R}) \hookrightarrow \left((\mathcal{G}/\mathcal{R})^{\mathsf{OR}\{\omega\}}, \mathcal{R}^{\mathsf{OR}\{\omega\}}_{\mathcal{G}/\mathcal{R}}\right)$$

*Proof.* —

For convenience of notation, let $\mathcal{F}$ denote $\mathcal{G}/\mathcal{R}$.

First of all, the totality of $t$ follows from the (already proven) totality of $\mathsf{FullType}$: each world $w$ of $\mathcal{G}$ has a well-defined $\mathsf{FullType}(w) \in \{0,1\}^{\omega}$, hence $t$ is well-defined on every $w$.

To see that $\%t : \mathcal{F} \to \mathcal{F}^{\mathsf{OR}\{\omega\}}/\mathcal{R}^{\mathsf{OR}\{\omega\}}_{\mathcal{F}}$ is a $\Pi$-iso, observe that $\%t$ is just the isomorphism of Prop. 3.5: each world $[w]_{\mathcal{R}} \in |\mathcal{F}| = |\mathcal{G}|/\mathcal{R}$ is identified with the $\mathcal{R}^{\mathsf{OR}\{\omega\}}_{\mathcal{F}}$-equivalence class of worlds of the form $([w]_{\mathcal{R}}, S)$ for $S \in \{0,1\}^{\omega}$.

Finally, we must prove that $t$ respects all $\sigma \in \mathsf{or}^{<\omega}(\Pi^{\dagger})$. We do this by structural induction on $\sigma$. As our base case, pick $\pi \in \Pi^{\dagger}$ and $w \in |\mathcal{G}|$. By definition of quotient, $\|\pi\|_{\mathcal{G}/\mathcal{R}}([w]_{\mathcal{R}})$ is defined iff $\|\pi\|_{\mathcal{G}}(w)$ is defined. By definition of $\mathsf{OR}\{\omega\}$-augmentation, $\|\pi\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}([w]_{\mathcal{R}}, S)$ is defined (for any $S \in \{0,1\}^{\omega}$) iff $\|\pi\|_{\mathcal{G}/\mathcal{R}}([w]_{\mathcal{R}})$ is defined. If these are all defined, then

$$\begin{aligned} t(\|\pi\|_{\mathcal{G}}(w)) &= \left(\left[\,\|\pi\|_{\mathcal{G}}(w)\right]_{\mathcal{R}}, \mathsf{FullType}(\|\pi\|_{\mathcal{G}}(w))\right) && \text{(Defn. } t) \\ &= \left(\left[\,\|\pi\|_{\mathcal{G}}(w)\right]_{\mathcal{R}}, \mathsf{FullType}(w)\right) && \text{(Lemma B.4)} \\ &= (\|\pi\|_{\mathcal{F}}([w]_{\mathcal{R}}), \mathsf{FullType}(w)) && \text{(Defn. quotient)} \\ &= \|\pi\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}([w]_{\mathcal{R}}, \mathsf{FullType}(w)) && \text{(Defn. } \mathsf{OR}\{\omega\}) \\ &= \|\pi\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)) && \text{(Defn. } t) \end{aligned}$$

as desired.

Now inductively suppose for some $\sigma_0$ that for any $w \in |\mathcal{G}|$ that $\|\sigma_0\|_{\mathcal{G}}$ is defined at $w$ iff $\|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}$ is defined at $t(w)$, and if they are defined,

$$t(\|\sigma_0\|_{\mathcal{G}}(w)) = \|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))$$

and likewise for $\sigma_1$. Then pick an arbitrary $w \in |\mathcal{G}|$ and assume without loss of generality that $w$ is a 0-world. So then, by Lemma B.3,

$$\|\sigma_0 \mathbin{\mathsf{or}} \sigma_1\|_{\mathcal{G}}(w) \quad \approx_{\mathcal{R}} \quad \|\sigma_0\|_{\mathcal{G}}(\|\mathsf{skip} \mathbin{\mathsf{or}} \mathsf{skip}\|_{\mathcal{G}}(w)). \tag{*}$$

For convenience, write $w'$ for $\|\mathsf{skip} \mathbin{\mathsf{or}} \mathsf{skip}\|_{\mathcal{G}}(w)$. Then notice by definition of $\mathsf{FullType}$ that

$$\mathsf{FullType}(w') = \mathsf{tl}(\mathsf{FullType}(w))$$

and thus, recalling that $w$ and $w'$ must be $\mathcal{R}$-related,

$$t(w') = ([w]_{\mathcal{R}}, \mathsf{tl}(\mathsf{FullType}(w))) = \|\mathsf{skip} \mathbin{\mathsf{or}} \mathsf{skip}\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)). \tag{**}$$

So then we have the following chain of reasoning: $\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w)$ is defined iff $\|\sigma_0\|_{\mathcal{G}}(w')$ is defined (by (*)); $\|\sigma_0\|_{\mathcal{G}}(w')$ is defined iff $\|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w'))$ is defined (by inductive hypothesis); and $\|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w'))$ is defined iff $\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))$ is defined (by (**) and definition of $\mathsf{OR}\{\omega\}$). So all that remains to show is that $t(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w))$ is $\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))$ if both are indeed defined.

First we show that they're in the same $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}$-equivalence class. To begin, note

$$\begin{aligned}
&\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)) \\
&= \|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(\|\mathsf{skip} \text{ or } \mathsf{skip}\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))) \\
&= \|\sigma_0\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w')) && (**) \\
&= t(\|\sigma_0\|_{\mathcal{G}}(w')). && \text{(IH)}
\end{aligned}$$

Now, $\|\sigma_0\|_{\mathcal{G}}(w')$ and $\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w)$ are $\mathcal{R}$-related (*), and thus $t$ will send them into the same $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}$ equivalence class. Thus,

$$\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)) \quad \mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}} \quad t(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w)).$$

So, in order to show that $\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)) = t(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w))$, it suffices to show

$$\mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))) = \mathsf{FullType}(t(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w))).$$

Notice that, by definition, $t$ preserves $\mathsf{FullType}$: $\mathsf{FullType}(t(w)) = \mathsf{FullType}(w)$. Note that the $\mathsf{FullType}$ on the right-hand side of this equation is calculated according to how $\mathsf{or}$'s are resolved in $\mathcal{G}$, whereas the left-hand $\mathsf{FullType}$ is calculated according to how $\mathcal{F}^{\mathsf{OR}\{\omega\}}$ resolves $\mathsf{or}$ (see Prop. B.2). Then,

$$\begin{aligned}
&\mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))) \\
&= \mathsf{FullType}(t(\|\sigma_0\|_{\mathcal{G}}(w'))) && (\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)) = t(\|\sigma_0\|_{\mathcal{G}}(w'))) \\
&= \mathsf{FullType}(\|\sigma_0\|_{\mathcal{G}}(w'))) && (t \text{ preserves } \mathsf{FullType}) \\
&= \mathsf{FullType}(\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{G}}(w)) && (\text{Lemma B.4})
\end{aligned}$$

So $\|\sigma_0 \text{ or } \sigma_1\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))$ are $\mathcal{R}_{\mathcal{F}}^{\mathsf{OR}\{\omega\}}$-related and have the same $\mathsf{FullType}$. By how $\mathsf{OR}\{\omega\}$-augmented frames are structured, this implies they are equal.

So conclude that for every $w$ and every $\sigma \in \mathsf{or}^{<\omega}(\Pi^\dagger)$, $\|\sigma\|_{\mathcal{G}}(w)$ is defined iff $\|\sigma\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w))$ is defined and, if both are defined,

$$t(\|\sigma\|_{\mathcal{G}}(w)) = \|\sigma\|_{\mathcal{F}^{\mathsf{OR}\{\omega\}}}(t(w)).$$

So $t$ respects each $\sigma$, and therefore constitutes an embedding of refined frames.

$\square$(Claim 5)

And, at long last, we're done.

$\square$