Kripke Open Bisimulation

A Marriage of Game Semantics and Operational Techniques

Guilhem Jaber[1] and Nicolas Tabareau[2]

[1] Queen Mary University, London, UK
[2] Inria, Nantes, France

Abstract. Proving that two programs are contextually equivalent is notoriously hard, particularly for functional languages with references (i.e., local states). Many operational techniques have been designed to prove such equivalences, and fully abstract denotational model, using game semantics, have been built for such languages. In this work, we marry ideas coming from trace semantics, an operational variant of game semantics, and from Kripke logical relations, notably the notion of worlds as transition systems of invariants, to define a new operational technique: Kripke open bisimulations. It is the first framework whose completeness does not rely on any closure by contexts.

1 Introduction

Many operational methods have been designed to reason about contextual equivalence of stateful programs. This profusion comes mainly from the difficulty to know exactly what kind of equivalence can be proven or not by a particular method. Even if completeness have been stated for some of those methods, the proof of completeness always relies on a notion of closure by contexts, which prevents to conclude that all the proofs of equivalence can be performed. For instance, Kripke logical relations (KLR), one of the most popular (and complete) method, were in their first version insufficient to prove the equivalence of two simple programs, dubbed at the time the "awkward example". This is because the notion of worlds as invariants introduced in the seminal paper of Pitts and Stark [12] is too restricted. KLR have later been refined by Ahmed, Dreyer et al.[1,2], where a *transition system* between such invariant is used to overcome this restriction. A sibling relational method to reason about contextual equivalence is given by bisimulations. Environmental [16,13] and Normal Form (a.k.a. Open) Bisimulations [15,7,8] are (set of) relations on terms defined coinductively w.r.t. the operational reduction. Their underlying idea is that contextual equivalence can be seen as the greatest adequate bisimulation which is also a congruence. The issue with this approach is that building a bisimulation is a complex task, especially when contexts do not have control operators and thus are not powerful enough to discriminate terms. Recently, Relation Transition Systems [3] (RTS) have been introduced to take the best of the two approaches. While in the work on bisimulation, a class of bisimulation is defined and then shown to be a congruence, RTS provide a single bisimulation, whose definition is parametrized by a transition system—as it is done for KLR—plus a notion of *global knowledge*. This means that when proving equivalence of two terms, only the transition system of heap invariants and the global knowledge need to be constructed. Then it just remains to

show that the two terms are in the corresponding bisimulation. But RTS are not known to be complete (the first version do not cover, e.g., η-equivalence).

On another line of work, fully-abstract denotational model of higher-order references have been designed, in terms of trace semantics [6] or game semantics [9]. In theory, it is thus possible to prove equivalence of programs by computing their denotation in such models, then prove that the denotations are equal. This is however in general a really complex task. *Algorithmic game semantics* [10] has been designed to perform automatically this task, using an automaton representation of the denotation of a term. However, this can be done only for fragments of the language where the type of terms is restricted. Thus this methods cannot be applied to unrestricted terms.

Overall, transition systems constitute a central object in this area. One can wonder whether they have been used for the same purpose. In trace semantics [6], the *interactive reduction* which generates traces can be seen as a bipartite *Labeled Transition System* (LTS) between player (i.e., the term) and opponent (i.e., contexts) configurations. Such LTSs carry a lot of information: the control flow between the term and any context, on each transition the actions performed, and on each state the configuration that the interactive reduction has reached. In the work on KLR and RTS, the transition system is rather an abstraction of the control flow, which is shared between two terms, and states only provide invariants on heaps. But among these works, those that are complete all use a notion of closure by context at one point in their definition. The only exception is the work of Stovring and Lassen [15], but for an *untyped* λ-calculus with a control operator (contexts having access to such operators, they can discriminate more terms). It is interesting to notice that in [7], the completeness of their bisimulation, once references are added to their language, was conjectured (albeit for a continuation passing style calculus, where reasoning on divergence is easier).

In this paper, we propose *Kripke Open Bisimulations* (KOBs), which are derived from bisimulations on configurations of the LTS generating traces, but are rather defined directly on terms with the usual operational semantics. The motto of KOBs could be:

"*to prove equivalence of programs, only a transition system of invariants is needed*"

Indeed, its definitions can be carried on in a simple logic that does not make use of quantification over λ-terms nor of a notion of closure. So once the transition system of invariants on heaps has been provided, it is straightforward to conduct the proof of equivalence, by simply reducing the terms operationally and checking that we get synchronized behavior. Via the link to trace semantics, we prove full abstraction of KOBs, without relying on any kind of closure, which suggests that all the reasoning principles necessary to reason about equivalence of stateful programs are present in KOBs.

Reasoning Principles behind Kripke Open Bisimulations. When reasoning on contextual equivalence, the key notion is to determine what can be observed by a context. This, of course, depends on the programming language in which contexts are written. For example, when contexts have access to a mutable memory, they can store how many times a function (or callback) provided to a term is called. This means that two terms are equivalent only when they perform the same callbacks, e.g., $\lambda f.f(); f()$ is not equivalent in that case to $\lambda f.f()$. Moreover, with such a memory, contexts can also keep track

of the order in which arguments are applied to callbacks. Thus, $\lambda f.(f\ 1) + (f\ 2)$ is not equivalent to $\lambda f.(f\ 2) + (f\ 1)$ in this setting. To sum-up, when reasoning on a language with a mutable memory, two terms are contextually equivalent only if the control flow between the term and contexts are equal, such control flow taking into account the functional values provided by a term to the context via callbacks. This idea shows up in game semantics, where the intensional model is fully abstract for a language with store, without considering a quotient of the model.

But in this setting, contexts can also observe memory cells created or modified by a term. This is however only the case for languages with unrestricted memory management like C or assembly code, where pointer arithmetic is allowed. This is not the case for languages like ML, where memory is implemented with *references*, represented using locations, on which the typing system forbids any kind of arithmetic. Locations can yet be passed as arguments to functions. This means that a disclosure process of locations can happen between a term and the context. So part of the references created by a term can become observable by the context, as soon as the corresponding locations are disclosed. For example, $\texttt{let x} = \mathrm{ref}\ \texttt{0 in}\ \lambda \texttt{y} : (\mathrm{ref}\ \mathrm{Int}).\texttt{x} == \texttt{y}$ and $\lambda \texttt{y} : (\mathrm{ref}\ \mathrm{Int}).\mathbf{false}$ are equivalent. Indeed, the location stored in x has not been disclosed and remains private to the term, so that contexts have no access to it and cannot pass it as an argument to the λ-abstraction. This means that we need to keep track of such disclosure process of locations to reason about equivalence of programs for languages like ML. But, we must also keep track of the way references that remain private to a term evolve. Indeed, when a term recovers the control, after performing a callback or returning a higher-order value, its execution also depends on its private part of the heap, and not only on the values provided by the context (either directly as arguments or via the disclosed part of the heap).

Transition systems representing the control flow between a term and a context, together with labels on transitions representing the invariants on heaps and the disclosure of locations, are thus important pieces of information to reason about equivalence of programs in the presence of a mutable memory. Following the work on KLR [2], some states of transition systems are tagged as *inconsistent* to deal with the so-called deferred divergence examples. This technique corresponds to the restriction to *complete plays* in game semantics, and would not be necessary if the language we consider featured some simple notion of control flow operator to abort the reduction.

The goal of this paper is to marry the notion of worlds as evolving invariants of KLR to the direct style reasoning provided by open bisimulations to provide a framework in which the transition system is the only external information needed to decide the equivalence of two programs. The price to pay for this unified framework is a complex proof of soundness and completeness as it can no longer rely on a biorthogonality argument (because of direct-style reasoning) nor on a generic notion of bisimulation (because bisimulations are restricted to particular ones, specified by a *relational* transition system). The proof is performed using ideas coming from nominal game semantics [9] and its connection to trace semantics, an operational variant initiated by Laird [6].

All the detailed proofs appear in the technical appendix [5].

$$
\begin{array}{lll}
\tau, \sigma & \stackrel{def}{=} & \mathrm{Unit} \mid \mathrm{Bool} \mid \mathrm{Int} \mid \mathrm{ref}\,\tau \mid \mid \tau \times \sigma \mid \tau \to \sigma \\
u, u' & \stackrel{def}{=} & () \mid \mathbf{true} \mid \mathbf{false} \mid \widehat{n} \mid x \mid l \mid \langle u, u' \rangle \mid \lambda x.M \quad (\text{where } n \in \mathbb{Z}, x \in \mathrm{Var}, l \in \mathrm{Loc}) \\
M, M' & \stackrel{def}{=} & u \mid MM' \mid M + M' \mid \texttt{if } M \texttt{ then } M' \texttt{ else } M'' \mid M == M' \mid \\
 & & \quad \mathrm{ref}\, M \mid !M \mid M := M' \mid \mid \langle M, M' \rangle \mid \pi_1(M) \mid \pi_2(M) \mid \bot_\tau \\
C & \stackrel{def}{=} & \bullet \mid \lambda x.C \mid CM \mid MC \mid \mathrm{ref}\, C \mid C := M \mid M := C \mid !C \mid C == M \mid \ldots \\
K & \stackrel{def}{=} & \bullet \mid KM \mid uK \mid \mathrm{ref}\, K \mid K := M \mid u := K \mid !K \mid K == M \mid u == K \mid \ldots
\end{array}
$$

Fig. 1. Definition of RefML

2 RefML

The programming language considered in this paper is RefML, a typed call-by-value functional language with *nominal higher-order references*, which is a fragment of ML.

2.1 Syntax of RefML

The syntax of types τ, values u, terms M, contexts C and evaluation contexts K of RefML is defined in Figure 1. As usual, $\texttt{let } x \texttt{ = } N \texttt{ in } M$ is defined as $(\lambda x.M)N$ and $M; N$ is defined as $(\lambda x.N)M$ with x fresh in M. Evaluation contexts K are particular kinds of contexts, the ones that start by reducing terms that fill their hole $\bullet$. For each type τ, we use a special term $\bot_\tau$ that always diverges.

Heaps h are defined as finite partial maps $\mathrm{Loc} \rightharpoonup \mathrm{Val}$. The empty heap is written ε. Adding a new element to a partial map h is written $h \cdot [l \hookrightarrow v]$, and is defined only if $l \notin \mathrm{dom}(h)$. We also define $h[l \hookrightarrow v]$, for $l \in \mathrm{dom}(h)$, as the partial function h' which satisfies $h'(l') = h(l')$ when $l' \neq l$, and $h'(l) = u$. The restriction of a heap h to a set of locations L is written $h_{|L}$. A heap is said to be *closed* when, for all $l \in \mathrm{dom}(h)$, if $h(l)$ is itself a location then $h(l) \in \mathrm{dom}(h)$. Taking a set L of locations and a heap h, we define the image of L by h, written $h^*(L)$ as $h^*(L) \stackrel{def}{=} \cup_{j \geq 0} h^j(L)$ where $h^0(L) = L$ and $h^{j+1}(L) = h(h^j(L)) \cap \mathrm{Loc}$.

Typing Rules. Typing judgments are of the form $\Sigma; \Gamma \vdash M : \tau$, where Σ and Γ are respectively typing contexts for locations and variables. Such typing contexts are partial maps between locations or variables to types. The typing rules of RefML are standard, and given in Appendix A.We write $\Sigma; \Gamma \vdash C : \tau \rightsquigarrow \sigma$ when $\Sigma; \Gamma, x : \tau \vdash C[x] : \sigma$, with $x \notin \Gamma$. Then, we write $\Gamma \vdash h : \Sigma$ if $\mathrm{dom}(h) = \mathrm{dom}(\Sigma)$ and $\Sigma; \Gamma \vdash h(l) : \Sigma(l)$.

2.2 Operational Semantics

The small-step operational semantics of RefML, written $(M, h) \mapsto (M', h')$, is defined in Figure 2 [3]. We write $M\{v/x\}$ to represent the (capture-free) substitution of x by v in M. This reduction is deterministic, and in particular we suppose that the reduction $(K[\mathrm{ref}\, v], h) \mapsto (K[l], h \cdot [l \hookrightarrow v])$ chooses a location $l \notin \mathrm{dom}(h)$. Using higher-order references, usual fixpoints $\texttt{fix f(x).M}$ of type $\tau \to \sigma$ can be defined using the *Landin's Knot*: $\texttt{let y} = \texttt{ref}\ (\lambda\texttt{x}.\bot_\sigma)\ \texttt{in y} := (\lambda\texttt{x.let f} = !\texttt{y in M}); !\texttt{y}$.

[3] We also consider the non-deterministic reduction $\mapsto_{nd}$, defined in the same way but for the rule of allocation, which is defined as $(K[\mathrm{ref}\, v], h) \mapsto_{nd} (K[l], h \cdot [l \hookrightarrow v])$ for any $l \notin \mathrm{dom}(h)$.

$$
\begin{array}{llll}
(K[(\lambda x.M)u], h) \mapsto (K[M\,\{u/x\}], h) & (K[\widehat{n} + \widehat{m}], h) & \mapsto (K[\widehat{n+m}], h) & \\
(K[\widehat{n} == \widehat{n}], h) \mapsto (K[\mathbf{true}], h) & (K[\widehat{n} == \widehat{m}], h) & \mapsto (K[\mathbf{false}], h) & (n \neq m) \\
(K[l == l], h) \mapsto (K[\mathbf{true}], h) & (K[l == l'], h) & \mapsto (K[\mathbf{false}], h) & (l \neq l') \\
(K[\bot_\tau], h) \mapsto (K[\bot_\tau], h) & (K[!l], h) & \mapsto (K[h(l)], h) & \\
(K[\mathrm{ref}\, u], h) \mapsto (K[l], h \cdot [l \hookrightarrow u]) & (K[l := u], h) & \mapsto (K[()], h[l \hookrightarrow u]) & \\
(K[\pi_i \langle M_1, M_2 \rangle], h) & \mapsto (K[M_i], h) & & \\
(K[\texttt{if } b \texttt{ then } M_{\mathbf{true}} \texttt{ else } M_{\mathbf{false}}], h) \mapsto (K[M_b], h) & & &
\end{array}
$$

Fig. 2. Operational Semantics of RefML

In the following, we say that a pair (M, h) is irreducible, written $\mathbf{irred}(M, h)$, if it cannot be reduced anymore. Taking an irreducible pair formed by a well-typed term M and a closed well-typed heap, where all the free variables are of functional types, we get that M is either equal to a value or to a *callback*, i.e., a term of the form $K[f\ v]$ with f a free variable and v a value.

Contextual (a.k.a. observational) equivalence is defined as:

Definition 1. *Taking two terms M_1, M_2 of the same type τ in a context Σ, Γ, we say that M_1 and M_2 are* contextually equivalent*, written $\Sigma; \Gamma \vdash M_1 \simeq_{ctx} M_2 : \tau$, when $\forall \Sigma' \supseteq \Sigma. \forall h : \Sigma'$ closed. $\forall C$ s.t. $\Sigma'; \Gamma \vdash C : \tau \rightsquigarrow \mathrm{Unit}$. $(C[M_1], h) \Downarrow$ iff $(C[M_2], h) \Downarrow$, where $(C[M_1], h) \Downarrow$ means $(C[M_1], h) \mapsto^* ((), h')$.*

Using closed heaps containing Σ ensures that the reduction of $(C[M_i], h)$ cannot get stuck, i.e., either reduces to $()$ or diverges.

2.3 Abstract Values and Nominal Reasoning

In the following, we represent functional values (i.e., λ-abstractions) using *functional names* belonging to a set FN. Abstract values [4] v are then defined as:

$$v, v' \stackrel{def}{=} () \mid \mathbf{true} \mid \mathbf{false} \mid \widehat{n} \mid f \mid l \mid \langle v, v' \rangle \quad \text{with } n \in \mathbb{Z}, l \in \mathrm{Loc} \text{ and } f \in \mathrm{FN}.$$

To each type τ, we associate a set $[\![\tau]\!]$ formed by pairs (v, ϕ) of abstract values and typing function for functional names.

$$
\begin{array}{llll}
[\![\mathrm{Unit}]\!] \stackrel{def}{=} \{((), \varepsilon)\} & [\![\sigma \to \tau]\!] \stackrel{def}{=} \{(f, [f \mapsto (\sigma \to \tau)]) \mid f \in \mathrm{FN}\} \\
[\![\mathrm{Int}]\!] \stackrel{def}{=} \{(\widehat{n}, \varepsilon) \mid n \in \mathbb{Z}\} & [\![\theta_1 \times \theta_2]\!] \stackrel{def}{=} \{(\langle v_1, v_2 \rangle, \phi_1 \cdot \phi_2) \mid (v_i, \phi_i) \in [\![\theta_i]\!]\} \\
[\![\mathrm{Bool}]\!] \stackrel{def}{=} \{(\mathbf{true}, \varepsilon), (\mathbf{false}, \varepsilon)\} & [\![\mathrm{ref}\, \tau]\!] \stackrel{def}{=} \{(l, \varepsilon) \mid l \in \mathrm{Loc}\}
\end{array}
$$

Taking a typing context Γ, we define $[\![\Gamma]\!]$ as the set of pairs of substitution functions and typing function of functional names defined as

$$\{(\rho, \phi) \mid \mathrm{dom}(\rho) = \mathrm{dom}(\Gamma), \forall (x, \tau) \in \Gamma, \exists \phi_x. (\rho(x), \phi_x) \in [\![\tau]\!], \phi = \biguplus_{x \in \mathrm{dom}(\Gamma)} \phi_x\}.$$

Then, we need to reason up-to permutations of both functional names and locations. To do so, we use *nominal sets*, as introduced by Pitts [11]. Fixing a set of names $\mathbb{A}$, we

[4] By seeing functional names as variables, the operational semantics of RefML can be extended straightforwardly to abstract values.

consider the group of finite permutations $\mathrm{Perm}(\mathbb{A})$ of $\mathbb{A}$, i.e., the bijections π of $\mathbb{A}$ s.t. the set $\{a \in \mathbb{A} \mid \pi(a) \neq a\}$ is finite. Then an $\mathbb{A}$-*nominal set* is a set X equipped with a group action (noted $*$) on $\mathrm{Perm}(\mathbb{A})$. We omit to indicate $\mathbb{A}$ when it is clear from the context. A subset S of $\mathbb{A}$ is said to *support* an element t of a nominal set X when

$$\forall \pi \in \mathrm{Perm}(\mathbb{A}).\ (\forall a \in S.\ \pi(a) = a) \Rightarrow \pi * t = t.$$

The smallest subset of $\mathbb{A}$ which supports t is called the *support* of t, written $\nu_{\mathbb{A}}(t)$. Terms and heaps of RefML can be seen as a nominal set over both Loc and FN. Then, the support of a term is (i) its set of locations if it is seen as nominal over Loc, or (ii) its set of free functional names if it is seen as nominal over FN.

Two elements t, u of a nominal set X are said to be *nominally-equivalent*, written $t \sim_{\mathbb{A}} u$ if there exists π in $\mathrm{Perm}(\mathbb{A})$ s.t. $t = \pi * u$ holds. We sometimes need to be explicit in the permutation when working with two nominally equivalent elements t, u of a nominal set X. However, when this is the case, it is more convenient to work with (typed) *spans* rather than permutations because spans are easier to extend than permutations. Spans, which are equivalent to permutations, have already been used by Stark to reason about locations, when defining logical relations for the ν-calculus [14].

Definition 2. *A* span $\mathcal{S} : (\mathbb{A} \times \mathrm{Types}) \rightleftharpoons (\mathbb{A} \times \mathrm{Types})$ *is a pair of partial finite injections* $(\mathbb{A} \times \mathrm{Types}) \hookleftarrow \mathcal{S} \hookrightarrow (\mathbb{A} \times \mathrm{Types})$ *preserving types. We write* $\mathrm{Span}_{\mathbb{A}}$ *for the set of spans over* $\mathbb{A}$*.*

We write ε for the empty span.The image of a span $\mathcal{S}$ by the left (resp. right) injection is written $\mathcal{S}_1$ (resp. $\mathcal{S}_2$). Such images can be seen as typing contexts. Reciprocally, from a typing context Γ, we define the span $\widehat{\Gamma}$ as $\{(x, x, \tau) \mid (x, \tau) \in \Gamma\}$. The extension of a span $\mathcal{S}$ at type τ with $(a_1, a_2) \in \mathbb{A}$ is written $\mathcal{S} \cdot (a_1, a_2, \tau)$, when $a_1 \notin \mathcal{S}_1$ and $a_2 \notin \mathcal{S}_2$. We say that $\mathcal{S}'$ extends $\mathcal{S}$, written $\mathcal{S}' \sqsupseteq \mathcal{S}$, when $\mathcal{S}'$ is a span which includes $\mathcal{S}$ as a set. Two spans are disjoints, written $\mathcal{S} \# \mathcal{S}'$, when both $\mathcal{S}_i, \mathcal{S}'_i$ are disjoint sets. A span $\mathcal{S}$ induces a finite permutation $\pi_{\mathcal{S}} : \mathbb{A} \to \mathbb{A}$, using the so-called "Homogeneity Lemma" of [11] (Lemma 1.14). Then, we define a restriction of the nominal equivalence $\sim_{\mathbb{A}}$ with respect to a span $\mathcal{S}$, written $X \sim_{\mathcal{S}} Y$, if $X = \pi_{\mathcal{S}} * Y$. We usually write Φ and $\mathcal{D}$ for spans respectively over functional names and locations, and write $x \sim_{\Phi}^{\mathcal{D}} y$ for the nominal equivalence induced by those spans on a nominal set over both FN and Loc.

3 Trace Semantics

This section presents a fully abstract model of RefML, based on a trace representation of game semantics which will be used to prove soundness and completeness of Kripke open bisimulations and at the same time to shed light on the intuitions coming from game semantics that have been used to define Kripke open bisimulations. Rather than working with the fully abstract game model of RefML defined by Murawski and Tzevelekos [9], it appears to be more convenient to work with a typed variant [4] of the trace model introduced by Laird [6]. This is because trace semantics, which provides as well a fully-abstract model of RefML, has a strong operational flavor, since it is generated via an *interactive reduction*, representing exactly all the possible interactions between terms and contexts.

$$\textbf{Intern}\quad \langle (M,\tau) :: \mathcal{S}, \gamma, \phi, h, D\rangle \longrightarrow \langle (M',\tau) :: \mathcal{S}, \gamma, \phi, h', D\rangle$$
(when $(M,h) \mapsto_{nd} (M',h')$)

$$\textbf{P-Ans}\quad \langle (u,\tau) :: \mathcal{S}, \gamma, \phi, h, D\rangle \xrightarrow{\langle \bar{v}\rangle, h'_{|D'}} \langle \mathcal{S}, \gamma', \phi', h[h'], D'\rangle$$

$$\textbf{P-Quest}\ \langle (K[f\ u],\sigma) :: \mathcal{S}, \gamma, \phi, h, D\rangle \xrightarrow{\bar{f}\langle v\rangle, h'_{|D'}} \langle (K[\bullet_{\tau'}],\sigma) :: \mathcal{S}, \gamma', \phi', h[h'], D'\rangle$$
(with $\phi(f) = \tau \to \tau'$)

in all P-rules: $(v, \gamma_v, \phi_v) \in \mathbf{AVal}_u(\tau), D' = \mathrm{discl}(u,h,D)$,
$(h', \gamma_h, \phi_h) \in \mathbf{AHeap}_{D'}(h'), \gamma' = \gamma \cdot \gamma_v \cdot \gamma_h, \phi' = \phi \cdot \phi_v \cdot \phi_h$

$$\textbf{O-Ans}\quad \langle (K[\bullet_\tau],\sigma) :: \mathcal{S}, \gamma, \phi, h, D\rangle \xrightarrow{\langle v\rangle, h'_{|D'}} \langle (K[v],\sigma) :: \mathcal{S}, \gamma, \phi', h[h'], D'\rangle$$

$$\textbf{O-Quest}\ \langle \mathcal{S}, \gamma, \phi, h, D\rangle \xrightarrow{f\langle v\rangle, h'_{|D'}} \langle (u\,v,\sigma) :: \mathcal{S}, \gamma, \phi', h[h'], D'\rangle$$
(with $\gamma(f) = u$)

in all O-Rules: $(v,\phi_v) \in [\![\tau]\!]$, $(h',\phi_h) \in [\![D']\!]$, $\phi' = \phi \cdot \phi_v \cdot \phi_h$, $D' = \mathrm{discl}(v,h,D)$

Fig. 3. Definition of the interaction reduction

3.1 Interactive reduction

The denotation of terms is defined as set of traces, whose basic blocks are *actions* a, of four kinds (following game semantics terminology, actions of terms and contexts are respectively called Player and Opponent actions):

- a question of Player (resp. Opponent) via a functional name f with argument an abstract value v, represented by the action $\bar{f}\,\langle v\rangle$ (resp. $f\,\langle v\rangle$);
- an answer by Player (resp. Opponent) with the abstract value v, represented by the action $\langle \bar{v}\rangle$ (resp. $\langle v\rangle$).

A *trace* is then defined as a sequence of actions-with-heap (a,h), where a is an action and h is a closed abstract heap. An important point is that h represents the disclosed part of the heap, common to the term and the context.

Traces are generated using an *interactive reduction*. This reduction is defined on "*evaluation stacks*" $\mathcal{S}$, which are either

- *passive*, $(K^n[\bullet_{\sigma_n}],\tau_n) :: \ldots :: (K^1[\bullet_{\sigma_1}],\tau_1)$, formed by typed evaluations contexts, for *Opponent configurations*,
- or active, $(M,\theta) :: \mathcal{S}'$ formed by a term M of type θ and a passive stack $\mathcal{S}'$, for *Player configurations*.

The empty stack is simply written $\Diamond$. When Player provides a higher-order value to Opponent, either via a callback (i.e., a question) or directly when reducing to a λ-abstraction (i.e., an answer), it is stored in an environment γ, which is a partial maps from FN to Val. Then Opponent can recover what is stored in γ, by asking a question. We associate a type to every functional names using a typing function $\phi : \mathrm{FN} \rightharpoonup \mathrm{Types}$, such that $\mathrm{dom}(\gamma) \subseteq \mathrm{dom}(\phi)$. Functional names in $\mathrm{dom}(\phi)\backslash\mathrm{dom}(\gamma)$ are the one provided by Opponent, which can then be used by Player. To represent disclosure of locations, we use a typing function $D : \mathrm{Loc} \rightharpoonup \mathrm{Types}$, that we often see as a relation, which grows as the term or the context discloses new locations.

Definition 3. *The disclosed locations coming from a value v and a heap h and already disclosed locations in D is computed using the fonction* $\mathrm{discl}(v,h,D)$*, defined as a typing function D' such that $(l,\tau) \in D'$ iff $l \in h^*(\nu_{\mathrm{Loc}}(v,D))$ and $D'; \phi \vdash h(l) : \tau$.*

The interactive reduction is defined in Figure 3 as a bipartite LTS between Player and Opponent configurations $\langle \mathcal{S}, \gamma, \phi, h, D \rangle$, where labels are actions-with-heap. The basic idea is that if a term reduces:

- to a callback $K[f\ u]$, the corresponding Player configuration performs a question $\bar{f}\,\langle v \rangle$, reducing to an Opponent configuration with K on top of the evaluation stack,
- to a value u, the corresponding Player configuration performs an answers $\langle \bar{v} \rangle$, reducing to an Opponent configuration where the head of the evaluation stack has been popped,

where v is an abstract values which, together with an environment γ' mapping its functional names to values, represents u. This γ' is added to the player environment. An opponent configuration can perform a question $f\,\langle v \rangle$ by interrogating a functional name f in γ, or, if its evaluation stack is non-empty, it can perform an answer $\langle v \rangle$, filling the hole of the first context of the stack.

The representation of a value u of type τ as a triple (v, ϕ, γ) formed by an abstract value, and two functions mapping its fresh functional names to values and types, is defined via the following set $\mathbf{AVal}_u(\tau)$:

$$\begin{array}{ll}
\mathbf{AVal}_v(\iota) & \stackrel{def}{=} \{(v, \varepsilon, \varepsilon)\} \text{ for } \iota = \mathrm{Unit}, \mathrm{Bool}, \mathrm{Int}, \mathrm{ref}\,\tau \\
\mathbf{AVal}_{\langle u_1, u_2 \rangle}(\tau_1 \times \tau_2) & \stackrel{def}{=} \{(\langle v_1, v_2 \rangle, \gamma_1 \cdot \gamma_2, \phi_1 \cdot \phi_2) \mid (v_i, \gamma_i, \phi_i) \in \mathbf{AVal}_{u_i}(\tau_i)\} \\
\mathbf{AVal}_u(\tau \to \sigma) & \stackrel{def}{=} \{(f, [f \mapsto u], [f \mapsto (\tau \to \sigma)]) \mid f \in \mathrm{FN}\}
\end{array}$$

We also define a function $\mathbf{AHeap}_D(h)$ to transform a heap h into a triple (h', γ, ϕ) formed by an abstract heap, and two functions mapping its fresh functional names to values and types, defined, using the typing information on locations contains in D, as:

$$\begin{array}{ll}
\mathbf{AHeap}_D(\varepsilon) & \stackrel{def}{=} \{(\varepsilon, \varepsilon, \varepsilon)\} \\
\mathbf{AHeap}_D(h \cdot [l \mapsto u]) & \stackrel{def}{=} \{(h' \cdot [l \mapsto v], \gamma \cdot \gamma', \phi \cdot \phi') \mid (h', \gamma', \phi') \in \mathbf{AHeap}_D(h), \\
& \qquad (v, \gamma, \phi) \in \mathbf{AVal}_u(\tau) \text{ with } (l, \tau) \in D\}
\end{array}$$

We write $C \stackrel{a}{\Rightarrow} C'$ when, if C is a Player configuration then there exists a Player configuration C'' such that $C \to C'' \stackrel{a}{\to} C'$, otherwise if C is an Opponent configuration then $C \stackrel{a}{\to} C'$. A trace T is *generated* by a configuration C when it can be written as a sequence $a_1 \cdots a_n$ of actions-with-heap s.t. $C \stackrel{a_1}{\Rightarrow} C_1 \stackrel{a_2}{\Rightarrow} \ldots \stackrel{a_n}{\Rightarrow} C_n$, in which case we write $C \stackrel{T}{\Rightarrow} C_n$. The set of traces generated by C is written $\mathrm{Tr}(C)$. A trace $T \in \mathrm{Tr}(C)$ is said to be *complete* if the number of answers occurring in the trace is greater than its number of questions plus the length of the evaluation stack of C. They can also be seen as the traces for which C reduces to a *final* Opponent configuration, that is one with an empty stack. The set of complete traces of a configuration C is written $\mathbf{comp}(\mathrm{Tr}(C))$. To define the denotation associated to an *open* term M, an extra action $?\,\langle v \rangle$, the *initial Opponent question*, is added to fix the choice of abstract values for the free variables of M.

Definition 4. *The set of complete traces generated by M, written $[\![\Sigma; \Gamma \vdash M : \tau]\!]$, is*

$$\bigcup \mathbf{comp}(\{?\,\langle \mathrm{codom}(\rho) \rangle \cdot \mathrm{Tr}(\langle (\rho(M), \tau), \varepsilon, \phi_\Gamma \cdot \phi_\Sigma, h, \Sigma' \rangle \mid (\rho, \phi_\Gamma) \in [\![\Gamma]\!], \\ \Sigma' \supseteq \Sigma, (h, \phi_\Sigma) \in [\![\Sigma']\!], \nu_{\mathrm{Loc}}(\rho) \subseteq \mathrm{dom}(\Sigma')\}).$$

As proven by Laird in [6] for closed, and more generally for open terms in [4], we get a full abstraction result:

Theorem 1. $\Sigma; \Gamma \vdash M_1 \simeq_{ctx} M_2 : \tau \Leftrightarrow [\![\Sigma; \Gamma \vdash M_1 : \tau]\!] = [\![\Sigma; \Gamma \vdash M_2 : \tau]\!]$.

3.2 Nominal equivalence of Traces

In the following, we decompose traces forming the denotation of terms, thus loosing the initial Opponent question which fixes the choice of names. To overtake this problem, we reason up to nominal equivalence of traces, with permutations which fix these names via two spans Φ and $\mathcal{D}$ on Loc and FN. We write $T \simeq^{\mathcal{D}}_{\Phi} T'$ if $T = a_1 \cdot \ldots \cdot a_n, T' = a'_1 \cdot \ldots \cdot a'_n$ and there exist two spans $\Phi' \sqsupseteq \Phi$ and $\mathcal{D}' \sqsupseteq \mathcal{D}$ such that for all $i, a_i \sim^{\mathcal{D}'}_{\Phi'} a'_i$. We then apply such nominal reasoning on *compatible configurations*

Definition 5. *Two configurations C_1, C_2 are* compatible *for $\Phi, \mathcal{D}$ when, writing C_i as $\langle \mathcal{S}_i, \gamma_i, \phi_i, h_i, D_i \rangle$, we have $\Phi_i = \phi_i$, $\mathcal{D}_i = D_i$, there exists a subspan $\Phi_P \sqsubseteq \Phi$ such that $\mathrm{dom}(\gamma_i) = \Phi_i$, and writing n_i for the evaluation stack $\mathcal{S}_i$, for all $j \in \{1, \ldots, \mathbf{min}(n_1, n_2)\}$, the j-th elements of $\mathcal{S}_1$ and $\mathcal{S}_2$ are of the same type, and $n_1 = 0$ iff $n_2 = 0$ (i.e., C_1 is a final configuration iff C_2 is).*

Taking two compatible configurations C_1, C_2 for $\Phi, \mathcal{D}$, we write $C_1 \simeq^{\mathcal{D}}_{\Phi} C_2$ when for all $T_1 \in \mathbf{comp}(\mathrm{Tr}(C_1))$, there exists $T_2 \in \mathbf{comp}(\mathrm{Tr}(C_2))$ such that $T_1 \simeq^{\mathcal{D}}_{\Phi} T_2$, and for all $T_2 \in \mathbf{comp}(\mathrm{Tr}(C_2))$, there exists $T_1 \in \mathbf{comp}(\mathrm{Tr}(C_1))$ such that $T_1 \simeq^{\mathcal{D}}_{\Phi} T_2$.

Theorem 2. *Suppose that $\Sigma; \Gamma \vdash M_1, M_2 : \tau$, then $\Sigma; \Gamma \vdash M_1 \simeq_{ctx} M_2 : \tau$ if and only if for all $(\rho, \phi_\Gamma) \in [\![\Gamma]\!]$, $\Sigma' \supset \Sigma$ and $(h, \phi_\Sigma) \in [\![\Sigma']\!]$ closed s.t. $\nu_{\mathrm{Loc}}(\rho) \subseteq \mathrm{dom}(\Sigma')$, we have $C_1 \simeq^{\widehat{\Sigma'}}_{\widehat{\phi}} C_2$, where $C_i = \langle (\rho(M_i), \tau), \varepsilon, \phi, h, \Sigma' \rangle$ with $\phi = \phi_\Gamma \cdot \phi_\Sigma$.*

3.3 A simple bisimulation on traces

One can see the LTS that generates traces as a (possibly infinite) automaton, where the final states correspond to opponent configurations with an empty evaluation stack. Then, bisimulations on this automaton can be defined in a standard way in order to capture the equality of the two languages recognized from two states (i.e. two configurations).

Using the fact that the LTS is bipartite, deterministic, and that a Player configuration can generate at most one action (up to nominal equivalence), we introduce a notion of bisimilation on traces as a family of pairs of relations $(\mathcal{P}_{\Phi,\mathcal{D}}, \mathcal{O}_{\Phi,\mathcal{D}})$ on *compatible* Player and Opponent configurations for Φ and $\mathcal{D}$ two spans respectively on functional names and locations, whose mutual coinductive definitions is given in Figure 4. Its definition is somehow complicated by the fact that the LTS is not complete, since for any configuration there exists some action a such that C does not produce a. This is particularly the case of diverging Player configurations, which simply do not produce any actions. We cannot complete the LTS by adding a unique "garbage state", since this state would not be compatible with the other diverging states. So for an Opponent (non-final) configuration C and two spans $\Phi, \mathcal{D}$, we consider the associated diverging compatible state $C^{\lightning i}_{\Phi,\mathcal{D}}$ defined as $\langle \mathcal{S}^{\lightning i}, \gamma^{\lightning i}, \Phi_i, h, \mathcal{D}_i \rangle$, where we write $\mathcal{S}^{\lightning i}$ for the evaluation stack $(\lambda\_.\bot_\tau)\bullet_\sigma, \tau)$ such that the top element of the evaluation stack of C is of type $\sigma \rightsquigarrow \tau$ and $\gamma^{\lightning 1}$ is defined as $\{(f, \lambda\_ : \sigma.\bot_{\sigma'}) \mid \exists f' \in \mathrm{dom}(C.\gamma).(f', f, \sigma \to \sigma') \in \Phi\}$ (the symmetric definitions applies for $i = 2$).

This notion of bisimulation captures equality of complete traces in the following sense (the proof can be found in Appendix B).

Theorem 3. *Taking C_1, C_2 be two configurations of polarity $X \in \{O, P\}$, we have $C_1 \simeq^{\mathcal{D}}_{\Phi} C_2$ iff $(C_1, C_2) \in X_{\Phi,\mathcal{D}}$.*

$$
\begin{aligned}
\mathcal{O}_{\Phi,\mathcal{D}} &\stackrel{def}{=} \Big\{(C_1,C_2) \mid \forall \Phi' \sqsupseteq \Phi, \forall \mathcal{D}' \sqsupseteq \mathcal{D} \forall a_1 \sim^{\mathcal{D}'}_{\Phi'} a_2 . \exists (C_1', C_2') \in \mathcal{P}_{\Phi',\mathcal{D}'} . \\
&\qquad \big((C_1 \stackrel{a_1}{\Longrightarrow} C_1') \Leftrightarrow (C_2 \stackrel{a_1}{\Longrightarrow} C_2')\big)\Big\} \\
\mathcal{P}_{\Phi,\mathcal{D}} &\stackrel{def}{=} \{(C_1,C_2) \mid (\forall i \in \{1,2\}. C_i \in \mathcal{P}^{\natural\, i}) \vee (\exists \Phi' \sqsupseteq \Phi . \exists \mathcal{D}' \sqsupseteq \mathcal{D} . \\
&\qquad \exists (C_1', C_2') \in \mathcal{O}_{\Phi',\mathcal{D}'} . \exists a_1 \sim^{\mathcal{D}'}_{\Phi'} a_2 . (\forall i \in \{1,2\}. C_i \stackrel{a_i}{\Longrightarrow} C_i')\} \\
\mathcal{P}^{\natural\, i} &\stackrel{def}{=} \{C \mid C \Uparrow \vee \exists C' \in \mathcal{O}^{\natural\, i} . \exists a . C \stackrel{a}{\Rightarrow} C'\} \\
\mathcal{O}^{\natural\, 1} &\stackrel{def}{=} \{C \mid \exists \Phi, \mathcal{D} . \Phi_1 = C.\phi \wedge \mathcal{D}_1 = C.D \wedge (C, C^{\natural\, 1}_{\Phi,\mathcal{D}}) \in \mathcal{O}_{\Phi,\mathcal{D}}\} \\
\mathcal{O}^{\natural\, 2} &\stackrel{def}{=} \{C \mid \exists \Phi, \mathcal{D} . \Phi_2 = C.\phi \wedge \mathcal{D}_2 = C.D \wedge (C^{\natural\, 2}_{\Phi,\mathcal{D}}, C) \in \mathcal{O}_{\Phi,\mathcal{D}}\}
\end{aligned}
$$

Fig. 4. Bisimulations on traces

4 Kripke Open Bisimulations

Bisimulations on traces can be somehow difficult to use as the LTS they are defined on is in most cases infinite. Indeed, Opponent has always the possibility to question a function f in γ as many times as he wants. The interaction generated by this question depends on both the value and the heap provided by Opponent. It is possible to characterizes them by knowing what are the disclosed locations (living in D) and the private part of the heap ($h_{\overline{D}}$), at any point after the introduction of f. To do so, we use a notion of world w, formed by such invariants on private heaps and a span on disclosed locations, and a transition system $\mathcal{A}$ describing how these worlds evolve. One can check the equivalence of two functional values disclosed by Player by checking their equivalence for any "future" world. This is the basic reasoning principle of Kripke Open Bisimulations, which is in fact taken from Kripke Logical Relations.

4.1 Transition Systems and Worlds

As in the work on RTS, we choose to work with "small" worlds, which only states *local* invariants relevant to the terms we reason on, but nothing about the invariants of the global contexts. But compared to the worlds used in RTS, we choose to do not incorporate the transition system inside the definition of worlds, but to use instead an external definition of transition system which dictates the evolution of worlds. Doing so, we can see transitions as pairs of pre- and post-conditions on heaps. We call them *World Transition Systems* (WTS, defined in Figure 5), since they are simply transition functions between worlds. Worlds w are tuples formed by a state s from an abstract set State, two heaps (describing the private part of the heap) h_1, h_2, a typed span on locations $\mathcal{D}$ and a boolean indicating if the world is inconsistent or not. We suppose that for $i \in \{1,2\}, \mathrm{dom}(h_i) \cap \mathcal{D}_i = \varnothing$. In practice, State can simply be taken as natural numbers. For a world $w = (s, h_1, h_2, \mathcal{D}, b)$ we define the predicates $\mathbf{cons}(w)$ and $\mathbf{incons}(w)$ respectively as $b = \mathbf{false}$ and $b = \mathbf{true}$. WTS are formed by a pair $(\delta, \delta_{\mathbf{pub}})$ respectively for private and public transitions, which are simply relations between worlds. Since worlds do not fully specify the disclosed part of heaps, there can be some branching on the values stored inside, which explains the non-deterministic representation of transitions, rather than just using a partial function. Private transitions

represent transitions that only terms can take, while public ones can be taken by both terms and contexts. This explains the condition $\delta_{\mathbf{pub}} \subseteq \delta^*_{\mathbf{priv}}$. Moreover, private transitions cannot transform an inconsistent world into a consistent one .

Worlds specify heaps precisely, since there is no freedom on the private part of the heap, while on the public part, the span is used to induce a nominal equivalence. But depending on whether the disclosed part is an abstract heap or a usual heap, we use two different predicates, defined in Figure 5:

- $\mathbf{P}_\Phi(w)$, which characterizes tuples $(h_1, h_2, \mathcal{D}, \Phi')$ of heaps together with a span on disclosed locations and a span on functional names Φ' that extends Φ, and which is used to collect the functional names used as abstract values on the $h_{i|\mathcal{D}_i}$
- $\mathbf{Q}_\Phi(w)$, which characterizes tuples $(h_1, h_2, \mathcal{D})$, where the $h_{i|\mathcal{D}_i}$ can contain λ-abstraction on which $\mathcal{V}_\mathcal{A} [\![\tau]\!]_\Phi\, w$, introduced in the next section, is used to reason about (via a mutual definition).

Transitions of a WTS $\mathcal{A}$ are used to define private and public notions of future worlds.

Definition 6. *Let $\mathcal{A}$ be a WTS and w_1, w_2 two worlds. We say that w_2 is a future (w.r.t. $\mathcal{A}$) of w_1, written $w_2 \sqsupseteq w_1$ if either $w_1 = w_2$ or $\delta_{\mathbf{priv}}(w_1, w_2)$. Note that strictly speaking, $\sqsupseteq$ depends on $\mathcal{A}$ but it is not explicit in the notation as $\mathcal{A}$ is always clear from context. Public futures (noted with $\sqsupseteq_{\mathbf{pub}}$) are defined similarly using $\delta_{\mathbf{pub}}$.*

Because contexts may create fresh disclosed locations during execution, we also introduce a notion of *freshened* extension $\mathcal{F}(w)$ of a world w which forces the existence of a state creating an arbitrary number of fresh disclosed locations. $\mathcal{F}(w)$ is defined as $\{(s, h_1, h_2, \mathcal{D}) \mid s = w.s, h_1 = w.h_1, h_2 = w.h_2, \exists \mathcal{D}'.\mathcal{D} = \mathcal{D}' \uplus w.\mathcal{D}\}$. We then write $w' \sqsupseteq^\mathcal{F} w$ (resp. $w' \sqsupseteq^\mathcal{F}_{\mathbf{pub}} w$) when there exists w'' such that $w'' \sqsupseteq w$ (resp. $w'' \sqsupseteq_{\mathbf{pub}} w$) and $w' \in \mathcal{F}(w'')$. We write $\sqsupseteq^{\mathcal{F}*}$ and $\sqsupseteq^{\mathcal{F}*}_{\mathbf{pub}}$ respectively for the transitive closure of $\sqsupseteq^\mathcal{F}$ and $\sqsupseteq^\mathcal{F}_{\mathbf{pub}}$.

4.2 Definition of KOBs

This section introduces Kripke open bisimulations. For space limitation, we have illustrated in Appendix F,on well-known examples of the literature, how to use direct-style reasoning, spans of names, WTSs and reasoning about divergence—which constitute the main concepts of KOBs.

Kripke open bisimulations, defined via a mutual coinduction in Figure 5, are a family of relations on values ($\mathcal{V}_\mathcal{A} [\![\tau]\!]_\Phi\, w$), evaluation contexts[5] ($\mathcal{K}_\mathcal{A} [\![\tau, \sigma]\!]_\Phi\, w$) and terms ($\mathcal{E}_\mathcal{A} [\![\tau]\!]_\Phi\, w$), that represents a particular kind of bisimulation, indexed by a world w of the WTS $\mathcal{A}$ and by a span on functional names Φ.

Compared to the bisimulations on traces, here we do not reason anymore on configurations, but simply on terms. The bisimulation on Player configurations corresponds to $\mathcal{E}_\mathcal{A} [\![\tau]\!]$, while the bisimulation on Opponent configurations corresponds to $\mathcal{V}_\mathcal{A} [\![\tau]\!]\, w$ for the questions, and $\mathcal{K}_\mathcal{A} [\![\sigma, \tau]\!]$ for the answers.

Forgetting a moment about the necessary predicative reasoning principle for diverging terms, Kripke open bisimulations mainly guarantee that, once reducing two terms

[5] Even if we use a relation $\mathcal{K}_\mathcal{A} [\![\sigma, \tau]\!]$ on evaluation contexts, our definition does not make any use of biorthogonality.

$$
\begin{array}{l}
\mathrm{World} \stackrel{def}{=} \mathrm{State} \times \mathrm{Heap}^2 \times \mathrm{Span}_{\mathrm{Loc}} \times \mathrm{Bool} \\
\mathrm{WTS} \stackrel{def}{=} \{(\delta_{\mathbf{priv}}, \delta_{\mathbf{pub}}) \mid \delta_{priv}, \delta_{pub} \subseteq \mathcal{P}(\mathrm{World} \times \mathrm{World}), \delta_{\mathbf{pub}} \subseteq \delta^*_{\mathbf{priv}}, \\
\qquad\qquad \forall (w, w') \in \delta_{\mathbf{priv}}.\mathbf{cons}(w') \Rightarrow \mathbf{cons}(w)\}
\end{array}
$$

$$
\begin{array}{l}
\mathbf{P}_{\Phi}(w) \stackrel{def}{=} \big\{(h_1, h_2, \mathcal{D}, \Phi') \mid \exists \Phi''. \exists h_1^d, h_2^d. \Phi' = \Phi \cdot \Phi'' \wedge h_1^d \sim^{\mathcal{D}}_{\Phi''} h_2^d \wedge \mathcal{D} = w.\mathcal{D} \\
\qquad\qquad \wedge \forall i \in \{1,2\}. h_i = w.h_i \cdot h_i^d \wedge h_i^d \in [\![\mathcal{D}_i]\!]\big\} \\
\mathbf{Q}_{\Phi}(w) \stackrel{def}{=} \big\{(h_1, h_2, \mathcal{D}) \mid \forall (l_1, l_2, \tau) \in \mathcal{D}. (h_1(l_1), h_2(l_2)) \in \mathcal{V}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi}\, w \\
\qquad\qquad \wedge \mathcal{D} = w.\mathcal{D} \wedge \forall i \in \{1,2\}. h_i = w.h_i \cdot h_i^d \wedge \mathrm{dom}(h_i^d) = \mathcal{D}_i\big\}
\end{array}
$$

$$
\begin{array}{l}
\mathcal{V}_{\mathcal{A}}\,[\![\iota]\!]_{\Phi}\, w \stackrel{def}{=} \{(v_1, v_2) \mid v_1, v_2 \in [\![\iota]\!], v_1 \sim_{w.D} v_2\} \\
\mathcal{V}_{\mathcal{A}}\,[\![\tau_1 \times \tau_2]\!]_{\Phi}\, w \stackrel{def}{=} \{(\langle v_1, v_2\rangle, \langle v_1', v_2'\rangle) \mid \forall i \in \{1,2\}. (v_i, v_i') \in \mathcal{V}_{\mathcal{A}}\,[\![\tau_i]\!]_{\Phi}\, w\} \\
\mathcal{V}_{\mathcal{A}}\,[\![\tau \to \sigma]\!]_{\Phi}\, w \stackrel{def}{=} \{((u_1, u_2) \mid \forall w' \sqsupseteq^{\mathcal{F}*} w. \forall \Phi' \# \Phi. \forall (v_1, \Phi_1'), (v_2, \Phi_2') \in [\![\tau]\!]. \\
\qquad\qquad v_1 \sim_{\Phi', w'.\mathcal{D}} v_2 \Rightarrow (u_1\ v_1, u_2\ v_2) \in \mathcal{E}_{\mathcal{A}}\,[\![\sigma]\!]_{\Phi.\Phi'}\,(w', w')\} \\
\mathcal{G}_{\mathcal{A}}\,[\![\Phi_P]\!]_{\Phi}\, w \stackrel{def}{=} \{(\gamma_1, \gamma_2) \mid \forall (f_1, f_2, \tau) \in \Phi_P. (\gamma_1(f_1), \gamma_2(f_2)) \in \mathcal{V}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi}\, w\}
\end{array}
$$

$$
\begin{array}{l}
\mathcal{K}_{\mathcal{A}}\,[\![\tau, \sigma]\!]_{\Phi}\,(w, w_0) \stackrel{def}{=} \{(K_1, K_2) \mid \forall w' \sqsupseteq^{\mathcal{F}*}_{\mathbf{pub}} w. \forall \Phi' \# \Phi. \forall (v_1, \Phi_1'), (v_2, \Phi_2') \in [\![\tau]\!]. \\
\qquad\qquad v_1 \sim_{\Phi', w'.\mathcal{D}} v_2 \Rightarrow (K_1[v_1], K_2[v_2]) \in \mathcal{E}_{\mathcal{A}}\,[\![\sigma]\!]_{\Phi.\Phi'}\,(w', w_0)\}
\end{array}
$$

$$
\begin{array}{l}
\mathcal{E}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi}\,(w, w_0) \stackrel{def}{=} \Big\{(M_1, M_2) \mid \forall (h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi}(w). \\
\quad \Big(\exists M_1', M_2'. \exists w' \sqsupseteq w. \exists (h_1', h_2', \mathcal{D}') \in \mathbf{Q}_{\Phi'}(w'). \\
\qquad \forall i \in \{1,2\}. \big((M_i, h_i) \mapsto^* (M_i', h_i') \wedge \mathbf{irred}(M_i', h_i')\big) \\
\qquad \wedge \Big(\big(\exists (u_1, u_2) \in \mathcal{V}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi'}\, w' \wedge w' \sqsupseteq^*_{\mathbf{pub}} w_0 \wedge \forall i \in \{1,2\}. M_i' = u_i \wedge \mathrm{discl}(u_i, h_i', \mathcal{D}_i) \subseteq \mathcal{D}_i'\big) \\
\qquad\quad \vee \big(\exists (f_1, f_2, \sigma \to \sigma') \in \Phi'. \exists (u_1, u_2) \in \mathcal{V}_{\mathcal{A}}\,[\![\sigma]\!]_{\Phi'}\, w'. \exists \big(K_1, K_2) \in \mathcal{K}_{\mathcal{A}}\,[\![\sigma', \tau]\!]_{\Phi'}\,(w', w_0) \\
\qquad\qquad \forall i \in \{1,2\}. M_i' = K_i[f_i\ u_i] \mathrm{discl}(u_i, h_i', \mathcal{D}_i) \subseteq \mathcal{D}_i'\big)\Big)\Big) \\
\quad \vee \big(\forall i \in \{1,2\}. (M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_{\mathcal{A}}\,[\![\tau]\!]_{\Phi_i'}\,(w, w_0)\big)\Big\}
\end{array}
$$

$$
\begin{array}{l}
\mathcal{E}^i_{\mathcal{A}}\,[\![\tau]\!]_{\phi}\,(w, w_0) \stackrel{def}{=} \Big\{(M, h, D) \mid (M, h) \Uparrow \vee \Big(\exists M'. \exists w' \sqsupseteq w. \exists (h', D') \in \mathbf{Q}^i_{\phi}(w'). \\
\quad (M, h) \mapsto^* (M', h') \wedge \mathbf{irred}(M', h') \wedge \mathbf{incons}(w') \wedge \\
\qquad \Big(\big(\exists u \in \mathcal{V}^i_{\mathcal{A}}\,[\![\tau]\!]_{\phi}\, w'. M' = u \wedge \mathrm{discl}(u, h', D) \subseteq D' \wedge w' \sqsupseteq^*_{\mathbf{pub}} w_0\big) \vee (\exists (f, \sigma \to \sigma') \in \phi. \\
\qquad\quad \exists u \in \mathcal{V}^i_{\mathcal{A}}\,[\![\sigma]\!]_{\phi}\, w'. \exists K \in \mathcal{K}^i_{\mathcal{A}}\,[\![\sigma', \tau]\!]_{\phi}\,(w', w_0). M' = K[f\ u]) \wedge \mathrm{discl}(u, h', D') \subseteq D'\Big)\Big)\Big\}
\end{array}
$$

$$
\begin{array}{l}
\Sigma; \Gamma \vdash M_1 \simeq_{kob} M_2 : \tau \stackrel{def}{=} \exists \mathcal{A} \in \mathrm{WTS}. \exists s \in \mathrm{State}. \forall (\rho, \phi) \in [\![\Gamma]\!]. \forall \Sigma' \supseteq \Sigma. \\
\qquad\qquad \nu_{\mathrm{Loc}}(\rho) \subseteq \mathrm{dom}(\Sigma') \Rightarrow (\rho(M_1), \rho(M_2)) \in \mathcal{E}_{\mathcal{A}}\,[\![\tau]\!]_{\widehat{\phi}}\,(w_0, w_0) \\
\quad \text{where } w_0 = (s, \varepsilon, \varepsilon, \widehat{\Sigma'}, \mathbf{false}). \text{ and for all } w \sqsupseteq^*_{\mathbf{pub}} w_0.\mathbf{cons}(w)
\end{array}
$$

Fig. 5. Definition of Kripke Open Bisimulations for RefML.

with heaps satisfying the invariants of the current world w, they either diverge, or there exists a future world w' of w such that the heaps produced by the reduction satisfy its invariants, and if the resulting terms are values, they are related, otherwise the resulting terms are callbacks which are synchronized, with the evaluation contexts surrounding them being related. The span on functional names Φ is used to keep track of functional names given by the context to the terms. Indeed, compared to logical relations, when τ is of functional type, the definition of $\mathcal{V}_{\mathcal{A}}[\![\tau \to \sigma]\!]_{\Phi} w$ does not quantify over related values v_1, v_2 of type τ, but uses instead fresh functional names f_1, f_2, remembering in Φ that they are related.

The definition of $\mathcal{E}_{\mathcal{A}}[\![\tau]\!]_{\Phi}(w, w_0)$ is indexed by an extra world w_0, corresponding to the initial world where the reduction of the two terms has been considered, and which is thus freshened in the definition of $\mathcal{V}_{\mathcal{A}}[\![\tau \to \sigma]\!]_{\Phi} w$. We enforce the existence of a *public* transition between a future world w' of w, and w_0 when terms have been reduced to values (but not callbacks). This corresponds to a well-bracketed behavior, where the question, which happens in the world w_0, is answered in the world w'.

Finally, the "full" KOB $\Sigma; \Gamma_f, \Gamma_g \vdash M_1 \simeq_{kob} M_2 : \tau$ is defined for terms with open ground variables, that must be substituted by ground values. All the futures worlds of the initial world w_0 used in its definition must be consistent. The main difference with Kripke logical relations is that there is an existential quantification over the WTS $\mathcal{A}$ which fixes the possible futures instead of a universal over all possible world extensions.

Predicative Reasoning. When considering diverging terms, synchronization of callbacks is no longer valid, since the two terms can diverge at different time during the execution. This is taken into account by a predicative reasoning involving:

- $\mathbf{Q}^i_{\phi}(w)$ defined as $\{(h, D) \mid h = (w.h_i) \cdot h^d \wedge D = (w.\mathcal{D})_i \wedge \mathrm{dom}(h^d) = D \wedge \forall (l, \tau) \in D, h(l) \in \mathcal{V}^i_{\mathcal{A}}[\![\tau]\!]_{\phi} w\}$
- $\mathcal{V}^1_{\mathcal{A}}[\![\iota]\!]_{\Phi} w$, defined as the set of closed values of type ι, for ι a ground type,
- $\mathcal{V}^1_{\mathcal{A}}[\![\tau \to \sigma]\!]_{\Phi} w$, defined as the set of values $\{v \mid \exists \Phi \in \mathrm{Span}_{\mathrm{FN}}.\Phi_1 = \phi \wedge (v, \lambda x.\bot_{\sigma}) \in \mathcal{V}_{\mathcal{A}}[\![\tau \to \sigma]\!]_{\Phi} w\}$,
- $\mathcal{K}^1_{\mathcal{A}}[\![\tau, \sigma]\!]_{\phi}(w, w_0)$, defined as the set of contexts $\{K \mid \exists \Phi \in \mathrm{Span}_{\mathrm{FN}}.\Phi_1 = \phi \wedge (K, (\lambda\_.\bot_{\sigma})\bullet) \in \mathcal{K}_{\mathcal{A}}[\![\tau, \sigma]\!]_{\Phi}(w, w_0)\}$

and $\mathcal{V}^2_{\mathcal{A}}[\![\tau]\!]_{\Phi} w$, $\mathcal{K}^2_{\mathcal{A}}[\![\tau, \sigma]\!]_{\Phi}(w, w_0)$ defined in a symmetric way. Then, we use inconsistent states to allow predicative reasoning, however this is not allowed in a public future of the initial world used in the definition $\Sigma; \Gamma_f, \Gamma_g \vdash M_1 \simeq_{kob} M_2 : \tau$, to avoid having unrelated "final" answers. This condition is also present in the definition of KLR [2].

4.3 An Example: Well-Bracketed State Change

We now see how KOBs work, using the WTS in Figure 6. on the "well-bracketed state change" example:

$$M_1 = \texttt{let x} = \mathrm{ref}\ \texttt{0}\ \texttt{in}\ \lambda \texttt{f.x} := \texttt{0}; \texttt{f}(); \texttt{x} := \texttt{1}; \texttt{f}();\ !x$$
$$M_2 = \lambda \texttt{f.f}(); \texttt{f}();\ 1$$

The transitions from the left to the middle state and from the middle to the right state are public, so that both the term and contexts can take it. The other one is only private to the term. Then, to prove the equivalence, we begin in the left state, and we

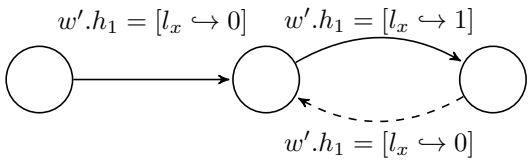


Fig. 6. WTS for the well-bracketed state change example.

reduce (M_1, ε) to $(v_1, h_1) = (\lambda \mathtt{f}.\mathtt{l_x} := \mathtt{0}; \mathtt{f}(); \mathtt{l_x} := \mathtt{1}; \mathtt{f}();\ !l_x\ , [\mathtt{l_x} \mapsto \mathtt{0}])$. Then, to prove the equivalence of v_1 and M_2, we must reason about all the future state of the middle one. This correspond to the fact that such λ-abstractions can be called at any point of the execution by the context, via nested calls.

Suppose we are in the right state. Then we know that $l_x \hookrightarrow 1$ and to prove that the two λ-abstractions are equivalent in this state, we directly reason on the corresponding open terms where the bindings of f have been removed. We reduce them, and since l_x is set to 0, we go back to the middle state, which is (privately) accessible. They both perform the same callback with the same value $()$, so so far they are related. Then, we must prove that the two contexts $\bullet; \mathtt{l_x} := \mathtt{1}; \mathtt{f}();\ !l_x\ $ and $\bullet; \mathtt{f}(); \mathtt{1}$ are related. To do so, we can go to any state *publicly* accessible from the current one. That is, we can be in the middle or right state as both are publicly accessible. Moreover, the contexts we consider have a hole of type Unit, so we just have to prove that the two terms $(); \mathtt{l_x} := \mathtt{1}; \mathtt{f}();\ !l_x\ $ and $(); \mathtt{f}(); \mathtt{1}$ are related. Reducing them, the callbacks are again related, and we must find a state where the post-condition $l_x \mapsto 1$ is valid, i.e., we move to the right state. Finally, we travel to any public future state from this one, so we stay in the same place, to prove that the contexts $\bullet;\ !l_x\ $ and $\bullet; 1$ are equivalent, which is straightforward since we know that l_x points to 1. The reasoning for the state where $l_x \hookrightarrow 0$ is similar. We see that the proof is done via a simple reasoning on the transition system, reducing the terms step by step. We have not simplified it in any way.

5 Soundness

We now prove a correspondence between bisimulations on traces and KOBs (the complete proofs are given in Appendix D).To do so, since KOBs are defined using the usual operational semantics, we need a lemma to validate the transformation of values into abstract values and functional environments defined via $\mathbf{AVal}_u(\tau)$

Lemma 1. *Let us consider* $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} w$. *Taking* $(v_1, \phi_1, \gamma_1) \in \mathbf{AVal}_{u_1}(\tau)$ *and* $(v_2, \phi_2, \gamma_2) \in \mathbf{AVal}_{u_2}(\tau)$ *such that* $\mathrm{dom}(\phi_i) \cap \mathrm{dom}(\Phi_{O,i}) = \varnothing$, *there exists a span* Φ_P *satisfying* $\Phi_{P,i} = \phi_i$ *such that* $v_1 \sim_{\Phi_P}^{w.\mathcal{D}} v_2$ *and* $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$.

To relate the evaluation stacks of the two considered configurations, it is necessary to relate their j-th elements at world w_j corresponding to the invariant when these evaluation stacks have been pushed.

Theorem 4. *Let* $n \in \mathbb{N}$ *and* $n + 2$ *world* $w_{n+1} \sqsupseteq^{\mathcal{F}*} w_n \sqsupseteq^{\mathcal{F}*} \ldots \sqsupseteq^{\mathcal{F}*} w_0$ *such that*
– $\forall w' \sqsupseteq^*_{\mathbf{pub}} w_0.\mathbf{cons}(w')$,

– $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} (w_{n+1}, w_n)$,
– *for all* $j \in \{1, \ldots, n\}$, $(K_1^j, K_2^j) \in \mathcal{K}_{\mathcal{A}} [\![\sigma_j, \tau_j]\!]_{\Phi_O} (w_{n+1}, w_{j-1})$,
– $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w_{n+1}$.

Then for all $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi_P \cdot \Phi_O}(w_{n+1})$*, writing* $\mathcal{S}_i$ *for* $(K_i^n[\bullet_{\sigma_n}], \tau_n) :: \ldots :: (K_i^1[\bullet_{\sigma_1}], \tau_1)$, $(\langle (M_1, \tau) :: \mathcal{S}_1, \gamma_1, \Phi'_1, h_1, \mathcal{D}_1 \rangle, \langle (M_2, \tau) :: \mathcal{S}_2, \gamma_2, \Phi'_2, h_2, \mathcal{D}_2 \rangle) \in \mathcal{P}_{\Phi', \mathcal{D}}$.

From Theorems 2, 3 and 4, we get the wanted result.

Corollary 1. *Suppose that* $\Sigma; \Gamma \vdash M_1 \simeq_{kob} M_2 : \tau$*, then* $\Sigma; \Gamma \vdash M_1 \simeq_{ctx} M_2 : \tau$.

6 Completeness

As opposed to KLR, completeness of KOBs can no longer be proven "for free" using biorthogonality. The proof needs to be more constructive and relies crucially on the connection to the fully-abstract trace semantics introduced in Section 3. However, it is not possible to use directly bisimulations on traces as they do not enforce the existence of a WTS $\mathcal{A}$ and a world w validating the equivalence. We introduce instead a variant notion of bisimulation on traces—*faithful Kripke bisimulation on traces*—whose definition is indexed by a WTS $\mathcal{A}$ and a list of world L, and which satisfies the property that being related for these new bisimulations implies being related for KOBs. We conclude by constructing an *exhaustive WTS* associated to a pair of configurations in the bisimulation on traces, which shows that two equivalent programs produce traces that are related by a faithful Kripke bisimulation (the complete proofs are given in Appendix E).

6.1 Faithful Kripke Bisimulations on Traces

To prove completeness of KOBs, we introduce an intermediate notion—between bisimulations on traces and KOBs: *faithful Kripke bisimulations on traces*, defined in Figure 7. They are pairs of relations $(\overline{\mathcal{P}}_{\mathcal{A}}(\Phi, L), \overline{\mathcal{O}}_{\mathcal{A}}(\Phi, L))$ on *partial configurations*, that is pairs formed by an evaluation stack and a functional environment, whose definitions is indexed by a span Φ on functional names and by stack of worlds, of size those of the evaluation stacks plus two, $L = w_n :: \ldots :: w_1$ such that $w_n \sqsupseteq^{\mathcal{F}*} \ldots \sqsupseteq^{\mathcal{F}*} w_1$. Restriction to *partial configurations* is harmless since we can always complete them using the span Φ and the top element of L. Faithful Kripke bisimulations on traces are used to enforce two main properties on a WTS $\mathcal{A}$: (i) the existence of accessible worlds validating the possible heaps obtained from reachable configurations, (ii) from $(C_1, C_2) \in \mathcal{O}_{\Phi, w}$ and $w' \sqsupseteq^* w$, the existence of equivalent execution of C_1, C_2 to configurations which satisfies the invariants of w'. Their definition can be seen as a mix between the bisimulations on traces introduced in Section 3.3, since they are defined on (partial) configurations and use interactive reduction, and the KOBs, since they use worlds and WTS to deduce and enforce invariants on heaps. However, there is a crucial distinction in the use of the WTS, that is the enforcement of *faithfulness* via the two predicates $\mathbf{Faitful}_{\Phi}(L), \mathbf{Faitful}_{\mathbf{pub}, \Phi}(L)$ respectively on private and public transitions. Indeed, these predicates enforce that all the transition of the WTS can be taken by some reduction of the LTS generating the traces (notice that $\mathbf{Faitful}_{\mathbf{pub}, \Phi}(L)$ enforces a stronger condition that this reduction should not change the stack of the configurations). These properties are not enforce by the KOBs, and it is indeed possible to

$$\overline{\mathcal{O}}_{\mathcal{A}}(\Phi, L) \stackrel{def}{=} \Big\{(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \mid \forall (h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w), \forall \Phi'' \sqsupseteq \Phi,$$
$$\forall w' \in \mathcal{F}(w), \forall a_1, a_2. a_1 \sim_{\Phi''}^{w', \mathcal{D}} a_2 \Rightarrow \exists \mathcal{S}_1', \mathcal{S}_2'. (\langle \mathcal{S}_1', \gamma_1\rangle, \langle \mathcal{S}_2', \gamma_2\rangle) \in \mathcal{P}_{\mathcal{A}}(\Phi', L')$$
$$\wedge \big((C_1 \stackrel{a_1}{\Rightarrow} \langle \mathcal{S}_1', \gamma_1, \Phi_1'', h_1', \mathcal{D}_1'\rangle) \Leftrightarrow (C_2 \stackrel{a_2}{\Rightarrow} \langle \mathcal{S}_2', \gamma_2, \Phi_2'', h_2', \mathcal{D}_2'\rangle)\big)$$
$$\wedge (\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \in (\mathbf{Faitful}_\Phi(L) \cap \mathbf{Faitful}_{\mathbf{pub},\Phi}(L))\Big\}$$

$$\overline{\mathcal{P}}_{\mathcal{A}}(\Phi, L) \stackrel{def}{=} \Big\{(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \mid \forall (h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w). (\forall i \in \{1,2\}. C_i \in \overline{\mathcal{P}}_{\mathcal{A}}^{\natural\, i}(\Phi_i', L))$$
$$\vee \big(\exists w' \sqsupseteq w. \quad \exists (h_1', h_2', \mathcal{D}', \Phi'') \in \mathbf{P}_{\Phi''}(w'). \exists (\langle \mathcal{S}_1', \gamma_1'\rangle, \langle \mathcal{S}_2', \gamma_2'\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', L').$$
$$\exists (a_1 \sim_{\Phi''}^{\mathcal{D}'} a_2). \forall i \in \{1,2\}. \big(C_i \stackrel{a_i}{\Rightarrow} \langle \mathcal{S}_i', \gamma_i', \phi_i'', h_i', D_i'\rangle\big) \wedge (a_1, a_2 \text{ answer} \Rightarrow w' \sqsupseteq_{\mathbf{pub}} w)\big)\Big\}$$

where (in both definitions) $C_i = \langle \mathcal{S}_i, \gamma_i, \Phi_i', h_i, \mathcal{D}_i\rangle, L = w :: L''$
$L' = w' :: L$ if both a_i are questions, otherwise $L' = w' :: L''$

$$\overline{\mathcal{P}}_{\mathcal{A}}^{\natural\, i}(\phi, L)) \stackrel{def}{=} \{C \mid C \uparrow \vee \exists w' \sqsupseteq w. \exists (h', D', \phi') \in \mathbf{P}_\phi^i(w'). \exists \langle \mathcal{S}', \gamma'\rangle \in \overline{\mathcal{O}}_{\mathcal{A}}^{\natural\, i}(\phi', L').$$
$$\exists a. (C \stackrel{a}{\Rightarrow} \langle \mathcal{S}', \gamma', \phi', h', D'\rangle) \wedge \mathbf{incons}(w') \wedge (a \text{ answer} \Rightarrow w' \sqsupseteq_{\mathbf{pub}} w)\}$$

where $L = w :: L''$ and $L' = w' :: L$ if both a_i are questions, otherwise $L' = w' :: L''$

$$\overline{\mathcal{O}}_{\mathcal{A}}^{\natural\, 1}(\phi, L) \stackrel{def}{=} \{\langle \mathcal{S}, \gamma\rangle \mid \exists \Phi. \Phi_1 = \phi \wedge (\langle \mathcal{S}, \gamma\rangle, \langle \mathcal{S}^{\natural\, i}, \gamma^{\natural\, i}\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi, L)\}$$

$$\mathbf{Faitful}_\Phi(w :: L) \stackrel{def}{=} \{(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \mid \forall w' \sqsupseteq^{\mathcal{F}*} w. \forall (h_1', h_2', \mathcal{D}', \Phi'') \in \mathbf{P}_\Phi(w'),$$
$$\exists (h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w), \exists T_1, T_2. \exists (\langle \mathcal{S}_1', \gamma_1'\rangle, \langle \mathcal{S}_2', \gamma_2'\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', (w' :: L')).$$
$$\langle \mathcal{S}_i, \gamma_i, \Phi_i', h_i, \mathcal{D}_i\rangle \stackrel{T_i}{\Rightarrow} \langle \mathcal{S}_i', \gamma_i', \Phi_i'', h_i', \mathcal{D}_i'\rangle\}$$

$$\mathbf{Faitful}_{\mathbf{pub},\Phi}(w :: L) \stackrel{def}{=} \{(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \mid \forall w' \sqsupseteq_{\mathbf{pub}}^{\mathcal{F}*} w. \forall (h_1', h_2', \mathcal{D}', \Phi'') \in \mathbf{P}_\Phi(w'),$$
$$\exists (h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w), \exists T_1, T_2. \exists \gamma_1', \gamma_2'. (\langle \mathcal{S}_1, \gamma_1'\rangle, \langle \mathcal{S}_2, \gamma_2'\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', (w' :: L)) \wedge$$
$$\langle \mathcal{S}_i, \gamma_i, \Phi_i', h_i, \mathcal{D}_i\rangle \stackrel{T_i}{\Rightarrow} \langle \mathcal{S}_i, \gamma_i', \Phi_i'', h_i', \mathcal{D}_i'\rangle\}$$

Fig. 7. Faithful Kripke bisimulations

use them with some WTS where they are not true. Thanks to those properties, faithful Kripke bisimulations on traces implies KOBs in the following sense.

Theorem 5. *Let M_1, M_2 two terms, w and w_0 two worlds, $\Phi = \Phi_P \cdot \Phi_O$ a span on functional names such that $(w.\mathcal{D})_i; \Phi_{O,i} \vdash M_i : \tau$, and γ_1, γ_2 two functional environments with $\mathrm{dom}(\gamma_i) = \Phi_{P,i}$. If $(\langle M_1, \gamma_1\rangle, \langle M_2, \gamma_2\rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi, (w, w_0))$, then $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}}[\![\tau]\!]_{\Phi_O}(w, w_0)$.*

6.2 Exhaustive WTS

It remains to construct the exhaustive relational WTS, which can be seen as the merge of two WTS coming from trace semantics. Its construction is obfuscated by nominal reasoning and diverging terms and requires some basic operations on WTSs:

- Add a private transition r: $\mathcal{A} \overset{\mathbf{priv}}{\oplus} r \stackrel{def}{=} (\delta_{\mathbf{priv}} \cup \{r\}, \delta_{\mathbf{pub}})$
- Add a public transition r: $\mathcal{A} \overset{\mathbf{pub}}{\oplus} r \stackrel{def}{=} (\delta_{\mathbf{priv}}, \delta_{\mathbf{pub}} \cup \{r\})$
- Union of two transition systems: $\mathcal{A}_1 \sqcup \mathcal{A}_2 \stackrel{def}{=} (\delta_{1,\mathbf{priv}} \cdot \delta_{2,\mathbf{priv}}, \delta_{1,\mathbf{pub}} \cdot \delta_{2,\mathbf{pub}})$.

The exhaustive WTS (for terms) $\mathbf{SE}_\Phi^L$ is defined by mutual coinduction with its corresponding WTS (for contexts) $\mathbf{SK}_\Phi^L$, where L is a list of worlds whose head corresponds

$$\mathbf{SE}^{w::L}_{\Phi}(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \stackrel{def}{=} \bigsqcup_{(h_1,h_2,\mathcal{D},\Phi')\in \mathbf{P}_\Phi(w)} \Big(\bigsqcup_{a_1 \sim^{\mathcal{D}'}_{\Phi''} a_2} \mathbf{SK}^{L'}_{\Phi'}(\langle \mathcal{S}'_1, \gamma'_1\rangle, \langle \mathcal{S}'_2, \gamma'_2\rangle) \overset{\mathbf{priv}}{\oplus} (w, w') \underbrace{\overset{\mathbf{pub}}{\oplus} (w_0, w')}_{\text{only if } a_i\text{'s are Player answers}} \Big) \bigsqcup \Big(\bigsqcup_i \mathbf{SE}^{\not{\iota}_i, w::L}(C_i) \Big)$$

$$\mathbf{SE}^{\not{\iota}_i, w::L}(\langle \mathcal{S}, \gamma, \phi, h, D\rangle) \stackrel{def}{=} \bigsqcup_a \mathbf{SK}^{\not{\iota}_i, L}_{\phi'}(\langle \mathcal{S}', \gamma'\rangle) \overset{\mathbf{priv}}{\oplus} (w, w') \underbrace{\overset{\mathbf{pub}}{\oplus} (w_0, w')}_{\text{only if } a \text{ is a Player answer}}$$

$$\mathbf{SK}^{w::L}_{\Phi}(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle) \stackrel{def}{=} \bigsqcup_{a_1 \sim^{\mathcal{D}'}_{\Phi'} a_2} \Big(\mathbf{SE}^{L'}_{\Phi'}(\langle \mathcal{S}'_1, \gamma_1\rangle, \langle \mathcal{S}'_2, \gamma_2\rangle) \Big)$$

$$\mathbf{SK}^{\not{\iota}_1, L}_{\phi}(\langle \mathcal{S}, \gamma\rangle) \stackrel{def}{=} \mathbf{SK}^{L}_{\Phi}(\langle \mathcal{S}, \gamma\rangle, \langle \mathcal{S}^{\not{\iota}}, \gamma^{\not{\iota}}\rangle)$$

Fig. 8. The exhaustive relational WTS

to the current one, while its tail corresponds to the public transitions that must be added once a value is reached in the interactive reduction. The definition is given in Figure 8.

The definition of $\mathbf{SE}^{w::L}_{\Phi}$ is done on Player evaluation stacks $\mathcal{S}_1, \mathcal{S}_2$. In the definition, writing C_i for $\langle \mathcal{S}_i, \gamma_i, \Phi'_i, h_i, \mathcal{D}_i\rangle$, we have $C_i \stackrel{a_i}{\Rightarrow} \langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}'_i\rangle$ and w' is equal to $(s, h'_{1|\overline{\mathcal{D}'_1}}, h'_{2|\overline{\mathcal{D}'_1}}, \mathcal{D}'', w.b)$ with s a fresh state. $L' = w' :: L''$ where, if the a_i are Player questions, then $L'' = L$, otherwise, $w_0 :: L'' = L$. To deal with divergence, we use the auxilliary definition $\mathbf{SE}^{\not{\iota}_i, w::L}$ to consider actions a such that $C \stackrel{a}{\Rightarrow} \langle \mathcal{S}', \gamma', \phi'_1, h'_1, D'_1\rangle$, In these case, $w' = (s, h'_{1|\overline{D'_1}}, w.h_2, \mathcal{D}', \mathbf{true})$ with $\mathcal{D}'$ any span s.t. $\mathcal{D}'_i = D'_i$. $L' = w' :: L''$ such that, if a is a Player questions, then $L = L''$, otherwise, $L = w_0 :: L'$. The definition of $\mathbf{SK}^{L}_{\Phi}$ is done on two Opponent evaluation stacks, with $\Phi' \sqsupseteq \Phi$, $w' \in \mathcal{F}(w)$ and both $\langle \mathcal{S}_i, \gamma_i, \Phi'_i, h_i, (w.\mathcal{D})_i\rangle \stackrel{a_i}{\Rightarrow} \langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i, (w'.\mathcal{D})_i\rangle$. $L' = w' :: L''$ and, if the a_i are Opponent questions, $L'' = w :: L$, otherwise, $L'' = L$.

Using the tree structure of the exhaustive WTS, we can prove the following theorem, which, combined with Theorem 2,3,5, allows to conclude on completeness of KOBs.

Theorem 6. *Let $\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle$ be two Player reduced configurations such that both $\mathcal{S}_1, \mathcal{S}_2$ have the same size n, Φ a spans on functional names, L a list of n worlds whose top element is w, and $(h_1, h_2, \mathcal{D}, \Phi) \in \mathbf{P}_\Phi(w)$. Writing C_i for $\langle \mathcal{S}_i, \gamma_i, \Phi_i, h_i, \mathcal{D}_i\rangle$, if $(C_1, C_2) \in \mathcal{P}_{\Phi,\mathcal{D}}$ then $(\langle \mathcal{S}_1, \gamma_1\rangle \langle \mathcal{S}_2, \gamma_2\rangle) \in \overline{\mathcal{P}}_{\mathbf{SE}^L_\Phi(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle)}(\Phi, L)$.*

7 Future Work

Toward Automation of Proofs of Equivalence. The ultimate goal of this work is to reason automatically on contextual equivalence. That is, given two terms M_1, M_2 and supposing that a WTS $\mathcal{A}$ is provided, we would like to prove automatically that $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_e w_0$. This is why we have removed quantification over "complex" objects in the definition of KOBs. By introducing a symbolic execution for fragments of the language (without higher-order references), one can automatically check whether two terms of these fragmets are in $\mathcal{E}_{\mathcal{A}} [\![\tau]\!]_e w_0$. This can be seen as *model-checking* equivalence of programs w.r.t. a WTS. Going further, we want to study fragments of

the language where some WTS $\mathcal{A}$, being of course a lot more compact than the exhaustive one, can be built automatically. Doing so, we should be able to *decide* equivalence of programs. We have begun to implement these ideas, using an SMT-solver. It gives promising results, being able to decide automatically the (in)-equivalence of many examples from the literature. It would then be interesting to compare such results from the one from *algorithmic game semantics* [10].

Semantic Cube. One of the most impressive result of game semantics is the characterization of various imperative features via constraint on strategies (e.g., first-order references = visibility condition), coined the "semantic cube" by Abramsky. Following this idea, Dreyer et al. [2] give a characterization of such imperative features via constraints on the shape of worlds and on the way we can reason about them. The restriction to first-order references corresponds to the possibility to backtrack in the world. In our framework, it should be possible to modify the definition of $\mathcal{E}_{\mathcal{A}} [\![\tau]\!]_e (w, w_0)$ so that future worlds of successive callbacks would be branching over w_0 instead of being linearly related—branching corresponds to backtracking. Finally, adding a control operator corresponds to removing the distinction between private and public transitions (and inconsistent states), since the restriction to complete traces is not necessary, which would lead to an interesting comparison with the work of Støvring and Lassen [15].

Compositionality. Because we use "small worlds", so that the *frame rule* is not baked in the definitions of KOBs, we cannot get compositionality results for free. It should however be possible to prove it, by defining a product $\mathcal{A}_1 \otimes \mathcal{A}_2$ of two LTSs, with an associated weakening lemma on LTS stating that if $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}_1} [\![\tau]\!]_\Phi w_1$, then $(M_1, M_2) \in \mathcal{E}_{(\mathcal{A}_1 \otimes \mathcal{A}_2)} [\![\tau]\!]_\Phi (w_1 \otimes w_2)$. The crucial point is that we should only require $\mathrm{discl}(u_i, h_i', \mathcal{D}_i) \subseteq \mathcal{D}_i'$ in the definition of $\mathcal{E}_{\mathcal{A}} [\![\tau]\!]_\Phi w$, instead of equality. This should allow to prove the composition theorem: if $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}_1} [\![\tau \to \sigma]\!]_\Phi w_1$ and $(N_1, N_2) \in \mathcal{E}_{\mathcal{A}_2} [\![\tau]\!]_\Phi w_2$ then $(M_1\ N_1, M_2\ N_2) \in \mathcal{E}_{(\mathcal{A}_1 \otimes \mathcal{A}_2)} [\![\tau]\!]_\Phi (w_1 \otimes w_2)$. Its proof should follows quite closely the proof of compositionality for RTS [3].

References

1. A. Ahmed, D. Dreyer, and A. Rossberg. State-dependent representation independence. In *Proceedings of POPL*, 2009.
2. D. Dreyer, G. Neis, and L. Birkedal. The impact of higher-order state and control effects on local relational reasoning. *Journal of Functional Programming*, 22:477–528, 9 2012.
3. C.-K. Hur, D. Dreyer, G. Neis, and V. Vafeiadis. The marriage of bisimulations and Kripke logical relations. In *Proceedings of POPL*, volume 47, pages 59–72, 2012.
4. G. Jaber. Operational nominal game semantics. In *Proceedings of FoSSaCS*. Springer, 2015.
5. G. Jaber and N. Tabareau. Kripke open bisimulation, a marriage of game semantics and operational techniques, 2015. Technical Appendix `http://guilhem.jaber.fr/aplas2015-full.pdf`.
6. J. Laird. A fully abstract trace semantics for general references. In *Proceedings of ICALP*, pages 667–679. Springer, 2007.
7. S. Lassen and P. Levy. Typed normal form bisimulation. In *Proceedings of CSL'07*, pages 283–297. Springer, 2007.
8. S. Lassen and P. Levy. Typed normal form bisimulation for parametric polymorphism. In *Proceedings of LICS*, pages 341–352. IEEE, 2008.

9. A. Murawski and N. Tzevelekos. Game semantics for good general references. In *Proceedings of LICS*, pages 75–84. IEEE, 2011.
10. A. Murawski and N. Tzevelekos. Algorithmic games for full ground references. In *Proceedings of ICALP*, pages 312–324, Berlin, Heidelberg, 2012. Springer-Verlag.
11. A. Pitts. Nominal logic, a first order theory of names and binding. *Information and computation*, 186(2):165–193, 2003.
12. A. Pitts and I. Stark. Operational reasoning for functions with local state. In *Higher Order Operational Techniques in Semantics*. CUP, 1998.
13. D. Sangiorgi, N. Kobayashi, and E. Sumii. Environmental bisimulations for higher-order languages. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 2011.
14. I. Stark. Names, equations, relations: Practical ways to reason about new. *Fundamenta Informaticae*, 33(4):369–396, 1998.
15. K. Støvring and S. Lassen. A complete, co-inductive syntactic theory of sequential control and state. In *Proceedings of POPL*, pages 161–172. ACM, 2007.
16. E. Sumii. A complete characterization of observational equivalence in polymorphic λ-calculus with general references. In *Proceedings of CSL*, pages 455–469. Springer, 2009.

A RefML

A.1 Syntax of RefML

$$
\begin{array}{lll}
\tau, \sigma & \stackrel{def}{=} & \mathrm{Unit} \mid \mathrm{Bool} \mid \mathrm{Int} \mid \mathrm{ref}\,\tau \mid \ \mid \tau \times \sigma \mid \tau \to \sigma \\
u, u' & \stackrel{def}{=} & () \mid \mathbf{true} \mid \mathbf{false} \mid \widehat{n} \mid x \mid l \mid \langle u, u' \rangle \mid \lambda x.M \quad (\text{where } n \in \mathbb{Z}, x \in \mathrm{Var}, l \in \mathrm{Loc}) \\
M, M' & \stackrel{def}{=} & u \mid MM' \mid M + M' \mid \texttt{if } M \texttt{ then } M' \texttt{ else } M'' \mid M == M' \mid \\
 & & \mathrm{ref}\, M \mid !M \mid M := M' \mid \ \mid \langle M, M' \rangle \mid \pi_1(M) \mid \pi_2(M) \mid \bot_\tau \\
C & \stackrel{def}{=} & \bullet \mid \lambda x.C \mid CM \mid MC \mid \mathrm{ref}\, C \mid C := M \mid M := C \mid !C \mid C + M \mid M + C \mid \\
 & & \texttt{if } C \texttt{ then } M \texttt{ else } M' \mid \texttt{if } M \texttt{ then } C \texttt{ else } M' \mid \texttt{if } M \texttt{ then } M' \texttt{ else } C \mid \\
 & & C == M \mid M == C \mid \langle C, M \rangle \mid \langle M, C \rangle \mid \pi_1 C \mid \pi_2 C \\
K & \stackrel{def}{=} & \bullet \mid KM \mid vK \mid \mathrm{ref}\, K \mid K := M \mid v := K \mid !K \mid K + M \mid v + K \mid \\
 & & \texttt{if } K \texttt{ then } M \texttt{ else } M' \mid K == M \mid v == K \mid \langle v, K \rangle \mid \langle K, v \rangle \mid \pi_1 K \mid \pi_2 K
\end{array}
$$

A.2 Typing Rules

$$\frac{}{\Sigma; \Gamma \vdash () : \mathrm{Unit}} \qquad \frac{}{\Sigma; \Gamma \vdash \mathbf{true} : \mathrm{Bool}} \qquad \frac{}{\Sigma; \Gamma \vdash \mathbf{false} : \mathrm{Bool}}$$

$$\frac{}{\Sigma; \Gamma \vdash \widehat{n} : \mathrm{Int}} \qquad \frac{}{\Sigma; \Gamma \vdash \bot_\tau : \tau} \qquad \frac{(x, \tau) \in \Gamma}{\Sigma; \Gamma \vdash x : \tau} \qquad \frac{(l, \tau) \in \Sigma}{\Sigma; \Gamma \vdash l : \mathrm{ref}\,\tau}$$

$$\frac{\Sigma; \Gamma \vdash M : \tau \qquad \Sigma; \Gamma \vdash N : \sigma}{\Sigma; \Gamma \vdash \langle M, N \rangle : \tau \times \sigma} \qquad \frac{\Sigma; \Gamma \vdash M : \tau_1 \times \tau_2}{\Sigma; \Gamma \vdash \pi_i M : \tau_i}$$

$$\frac{\Sigma; \Gamma, x : \tau \vdash M : \sigma}{\Sigma; \Gamma \vdash \lambda x.M : \tau \to \sigma} \qquad \frac{\Sigma; \Gamma \vdash M : \tau \to \sigma \qquad \Sigma; \Gamma \vdash N : \tau}{\Sigma; \Gamma \vdash MN : \sigma}$$

$$\frac{\Sigma; \Gamma \vdash M : \tau}{\Sigma; \Gamma \vdash \mathrm{ref}\, M : \mathrm{ref}\,\tau} \qquad \frac{\Sigma; \Gamma \vdash M : \mathrm{ref}\,\tau}{\Sigma; \Gamma \vdash !M : \tau} \qquad \frac{\Sigma; \Gamma \vdash M : \mathrm{ref}\,\tau \qquad \Sigma; \Gamma \vdash N : \tau}{\Sigma; \Gamma \vdash M := N : \mathrm{Unit}}$$

$$\frac{\Sigma; \Gamma \vdash M_1 : \mathrm{Bool} \qquad \Sigma; \Gamma \vdash M_2 : \tau \qquad \Sigma; \Gamma \vdash M_2 : \tau}{\Sigma; \Gamma \vdash \texttt{if } M_1 \texttt{ then } M_2 \texttt{ else } M_3 : \tau}$$

$$\frac{\Sigma; \Gamma \vdash M_1 : \mathrm{Int} \qquad \Sigma; \Gamma \vdash M_2 : \mathrm{Int}}{\Sigma; \Gamma \vdash M_1 + M_2 : \mathrm{Int}} \qquad \frac{\Sigma; \Gamma \vdash M_1 : \mathrm{Int} \qquad \Sigma; \Gamma \vdash M_2 : \mathrm{Int}}{\Sigma; \Gamma \vdash M_1 == M_2 : \mathrm{Bool}}$$

$$\frac{\Sigma; \Gamma \vdash M_1 : \mathrm{ref}\,\tau \qquad \Sigma; \Gamma \vdash M_2 : \mathrm{ref}\,\tau}{\Sigma; \Gamma \vdash M_1 == M_2 : \mathrm{Bool}}$$

B Trace semantics

Lemma 2. *Let us consider C_1, C_2 two Opponent configuration compatible for some spans $\Phi, \mathcal{D}$, two actions a_1, a_2 such that there exists two spans $\Phi' \sqsupseteq \Phi, \mathcal{D}' \sqsupseteq \mathcal{D}$ with*

$a_1 \sim^{\mathcal{D}'}_{\Phi'} a_2$. *Then there exists two configurations* C'_1, C'_2 *compatible for* $\Phi', \mathcal{D}'$ *such that* $C_1 \stackrel{a_1}{\Rightarrow} C'_1$ *iff* $C_2 \stackrel{a_2}{\Rightarrow} C'_2$.

Proof: Suppose that there exists a configuration C'_1 such that $C_1 \stackrel{a_1}{\Rightarrow} C'_1$ with $C'_1 = \langle \mathcal{S}'_1, \gamma_1, \Phi'_1, h'_1, \mathcal{D}'_1 \rangle$.

If a_1 is an Opponent question $(f_1 \langle v_1 \rangle, h'_1)$, then from the fact that C_1, C_2 are compatible, we get that there exists a functional name f_2 such that $(f_1, f_2, \sigma \to \sigma') \in \Phi$ and $f_2 \in \gamma_2$. One can decompose Φ' into $\Phi \cdot \Phi_v \cdot \Phi_h$ such that $(v_1, \Phi_{v,1}) \in [\![\sigma]\!]$ and $(h'_1, \Phi_{h,1}) \in [\![\mathcal{D}'_2]\!]$. Then, we consider any v_2, h_2 such that $(v_2, \Phi_{v,2}) \in [\![\sigma]\!]$ and $(h_2, \Phi_{h,2}) \in [\![\mathcal{D}'_1]\!]$. Defining a_2 as $(f_2 \langle v_2 \rangle, h'_2)$, we get that $C_2 \stackrel{a_1}{\Rightarrow} C'_2$ with $C'_2 = \langle \mathcal{S}'_2, \gamma_2, \Phi'_2, h'_2, \mathcal{D}'_2 \rangle$, so that C'_1, C'_2 are compatible for $\Phi', \mathcal{D}'$.

The same reasoning applies for Opponent answers, and when there exists a configuration C'_2 such that $C_2 \stackrel{a_2}{\Rightarrow} C'_2$. □

Lemma 3. *Taking* $C = \langle \mathcal{S}, \gamma, \phi, h, D \rangle$ *an Opponent configuration, we have* $\mathbf{comp}(\mathrm{Tr}(C)) = \varnothing$ *iff* $C \in \mathcal{O}^{\natural i}$.

Proof: Suppose that $\mathbf{comp}(\mathrm{Tr}(C)) = \varnothing$. Let us consider any action a and Player configuration C' such that $C \stackrel{a}{\Rightarrow} C'$.We must prove that $(C', C'^{\natural 2}) \in \mathcal{P}_{\Phi', \mathcal{D}'}$. If $C' \uparrow$, it is indeed the case. Otherwise, there exists an action a' and an Opponent configuration C'' such that $C' \stackrel{a'}{\Rightarrow} C''$. Then from $\mathbf{comp}(\mathrm{Tr}(C)) = \varnothing$ we get that $\mathbf{comp}(\mathrm{Tr}(C'')) = \varnothing$, so we can apply the coinduction hypothesis to deduce that $C'' \in \mathcal{O}^{\natural i}$. Thus, $(C', C'^{\natural 2}) \in \mathcal{P}_{\Phi', \mathcal{D}'}$. The same reasoning applies to prove that $(C'^{\natural 1}, C') \in \mathcal{P}_{\Phi', \mathcal{D}'}$.

Now, suppose that $C \in \mathcal{O}^{\natural i}$, necessary C cannot be final so that $\varepsilon \notin \mathrm{Tr}(C)$. Suppose that there exists a trace $(a \cdot a' \cdot T) \in \mathbf{comp}(\mathrm{Tr}(C))$. Then, there exists two configurations C', C'' such that $C \stackrel{a}{\Rightarrow} C' \stackrel{a'}{\Rightarrow} C''$ and $T \in \mathbf{comp}(\mathrm{Tr}(C''))$. We get that $C'' \in \mathcal{O}^{\natural i}$, so the coinduction hypothesis gives us that $\mathbf{comp}(\mathrm{Tr}(C''))$, which is absurd since $T \in \mathbf{comp}(\mathrm{Tr}(C''))$. □

Theorem 7. *Taking* C_1, C_2 *be two configurations of polarity* $X \in \{O, P\}$, *we have* $C_1 \simeq^{\mathcal{D}}_{\Phi} C_2$ *iff* $(C_1, C_2) \in X_{\Phi, \mathcal{D}}$.

Proof:

From left to right Suppose that $C_1 \simeq^{\mathcal{D}}_{\Phi} C_2$. Then if both C_i are Player configurations, there is five possibilities:

1) Both $\mathrm{Tr}(C_i)$ are empty, in which case both $C_i \uparrow$, so that indeed $(C_1, C_2) \in \mathcal{P}_{\Phi, \mathcal{D}}$.
2) There exists two action a_1, a_2 and two configuration C'_1, C'_2 such that both $C_i \stackrel{a_i}{\Rightarrow} C'_i$. Then, up to nominal equivalence, $\mathbf{comp}(\mathrm{Tr}(C_i)) \sim a_i \cdot \mathbf{comp}(\mathrm{Tr}(C'_i))$.
   2a) If both $\mathbf{comp}(\mathrm{Tr}(C'_i))$ are non empty, from the initial hypothesis, we get the existence of two spans $\Phi' \sqsupseteq \Phi$, $\mathcal{D}' \sqsupseteq \mathcal{D}$ such that $a_1 \sim^{\mathcal{D}'}_{\Phi'} a_2$ and $C'_1 \simeq^{\mathcal{D}'}_{\Phi'} C'_2$, so by the coinduction hypothesis $(C'_1, C'_2) \in \mathcal{O}_{\Phi', \mathcal{D}'}$, and thus $(C_1, C_2) \in \mathcal{P}_{\Phi, \mathcal{D}}$.

2b) Otherwise, from the initial hypothesis both $\mathbf{comp}(\mathrm{Tr}(C_i'))$ are empty. From Lemma 3, we get that both $C_i' \in \mathcal{O}^{\not\downarrow i}$, so that $(C_1, C_2) \in \mathcal{P}_{\Phi,\mathcal{D}}$.

3) There exists an action a_2 and a configuration C_2' such that $C_2 \stackrel{a_2}{\Longrightarrow} C_2'$, and $C_1 \uparrow$. Then $\mathbf{comp}(\mathrm{Tr}(C_2)) \sim a_2 \cdot \mathbf{comp}(\mathrm{Tr}(C_2'))$ and from the initial hypothesis, $\mathbf{comp}(\mathrm{Tr}(C_2))$ is empty, so that $\mathbf{comp}(\mathrm{Tr}(C_2'))$ is also empty. From Lemma 3 $C_2' \in \mathcal{O}^{\not\downarrow 2}$. Thus $(C_1, C_2) \in \mathcal{P}_{\Phi,\mathcal{D}}$.

4) There exists an action a_1 and a configuration C_1' such that $C_1 \stackrel{a_2}{\Longrightarrow} C_1'$, and $C_2 \uparrow$. Then the same reasoning as in 3) applied to prove that $(C_1, C_2) \in \mathcal{P}_{\Phi,\mathcal{D}}$.

Otherwise, both C_i are Opponent configurations. Let $\Phi' \sqsupseteq \Phi$, $\mathcal{D}' \sqsupseteq \mathcal{D}$ and two actions a_1, a_2 such that $a_1 \sim_{\Phi'}^{\mathcal{D}'} a_2$. From Lemma 2, there exists two Player configuration C_1', C_2' compatible with $\Phi', \mathcal{D}'$ such that $C_1 \stackrel{a_1}{\Longrightarrow} C_1'$ iff $C_2 \stackrel{a_2}{\Longrightarrow} C_2'$. From $(a_i \cdot \mathbf{comp}(\mathrm{Tr}(C_i'))) \subseteq \mathbf{comp}(\mathrm{Tr}(C_i))$, the fact that the LTS is deterministic, and $C_1 \simeq_{\Phi}^{\mathcal{D}} C_2$, we get that $C_1' \simeq_{\Phi'}^{\mathcal{D}'} C_2'$. Applying the induction hypothesis, we get that $(C_1', C_2') \in \mathcal{P}_{\Phi',\mathcal{D}'}$, so that $(C_1, C_2) \in \mathcal{O}_{\Phi,\mathcal{D}}$.

From right to left Suppose that $(C_1, C_2) \in X_{\Phi,\mathcal{D}}$. Let us first suppose that both are Player configuration (i.e. $X = P$). There exists two actions a_1, a_2 and two configurations C_1', C_2' s.t.:

- both $C_i \stackrel{a_i}{\Longrightarrow} C_i'$ and there exists $\Phi' \sqsupseteq \Phi$ and $\mathcal{D}' \sqsupseteq \mathcal{D}$ such that $a_1 \sim_{\Phi'}^{\mathcal{D}'} a_2$ and $(C_1', C_2') \in \mathcal{O}_{\Phi',\mathcal{D}'}$. Then the coinduction hypothesis gives us that $C_1' \simeq_{\Phi'}^{\mathcal{D}'} C_2'$. Since all the actions performed by C_i are nominally equivalent, this gives us that $C_1 \simeq_{\Phi}^{\mathcal{D}} C_2$.
- Or both $C_i \uparrow$, in which case both $\mathbf{comp}(\mathrm{Tr}(C_i)) = \varnothing$.
- Or both $C_i \stackrel{a_i}{\longrightarrow} C_i'$ and both $C_i' \in \mathcal{O}^{\not\downarrow i}$, and the coinduction hypothesis gives us that both $\mathbf{comp}(\mathrm{Tr}(C_i)) = \varnothing$,
- or $C_1 \stackrel{a_1}{\longrightarrow} C_1'$, $C_2 \uparrow$, and $C_1', \in \mathcal{O}^{\not\downarrow 1}$ so that $\mathbf{comp}(\mathrm{Tr}(C_2)) = \varnothing$, and Lemma 3 coinduction hypothesis gives us that $\mathbf{comp}(\mathrm{Tr}(C_1)) = \varnothing$
- or $C_2 \stackrel{a_2}{\longrightarrow} C_2'$, $C_1 \uparrow$, and $C_2' \in \mathcal{O}^{\not\downarrow 2}$ so that $\mathbf{comp}(\mathrm{Tr}(C_1)) = \varnothing$, and Lemma 3 gives us that $\mathbf{comp}(\mathrm{Tr}(C_2)) = \varnothing$.

Otherwise, both C_1, C_2 are Opponent configurations. Suppose first that $\mathbf{comp}(\mathrm{Tr}(C_1))$ is empty, and suppose there exists $T_2 \in \mathbf{comp}(\mathrm{Tr}(C_2))$. If T_2 is the empty trace ε, then C_2 is a final configuration, so since C_1, C_2 are compatible, C_1 is also a final configuration, which is absurd because otherwise the empty trace would also be in $\mathbf{comp}(\mathrm{Tr}(C_1))$. Then there exists an action a_2 and a configuration C_2' such that $C_2 \stackrel{C_2}{\longrightarrow}{}'$, $T_2 = a_2 \cdot T_2'$ with $T_2' \in \mathbf{comp}(\mathrm{Tr}(C_2'))$. But then, there exists two spans $\Phi' \sqsupseteq \Phi$ and $\mathcal{D}' \sqsupseteq \mathcal{D}$ and an action a_1 such that $a_1 \sim_{\Phi'}^{\mathcal{D}'} a_2$. From $(C_1, C_2) \in \mathcal{O}_{\Phi,\mathcal{D}}$, we get the existence of C_1' such that $(C_1', C_2') \in \mathcal{P}_{\Phi',\mathcal{D}'}$ and $C_1 \stackrel{a_1}{\longrightarrow} C_1'$. But from $\mathbf{comp}(\mathrm{Tr}(C_1')) = \varnothing$ and $(C_1', C_2') \in \mathcal{P}_{\Phi',\mathcal{D}'}$, we get from the coinduction hypothesis that $\mathbf{comp}(\mathrm{Tr}(C_2')) = \varnothing$, which is absurd since $T_2' \in \mathbf{comp}(\mathrm{Tr}(C_2'))$. So $\mathbf{comp}(\mathrm{Tr}(C_2))$ is empty.

Otherwise, let us take a trace $T_1 \in \mathbf{comp}(\mathrm{Tr}(C_1))$, one must build $T_2 \in \mathbf{comp}(\mathrm{Tr}(C_2))$ such that $T_1 \simeq_{\Phi}^{\mathcal{D}} T_2$. If T_1 is the empty trace, this is straightforward since C_1 is thus a final configuration, so does C_2 since C_1, C_2 are compatible. Otherwise there exists an opponent action a_1 and a Player configuration C_1' such that $C_1 \stackrel{a_1}{\longrightarrow} C_1'$ and

$T_1 = a_1 \cdot T_1'$. Then, there exists of an action a_2 and two spans $\Phi' \sqsupseteq \Phi$ and $\mathcal{D}' \sqsupseteq \mathcal{D}$ such that $a_1 \sim_{\Phi'}^{\mathcal{D}'} a_2$. From $(C_1, C_2) \in \mathcal{O}_{\Phi,\mathcal{D}}$, we get the existence of C_2' such that $(C_1', C_2') \in \mathcal{P}_{\Phi',\mathcal{D}'}$ and $C_2 \xrightarrow{a_2} C_2'$. Then, the coinduction hypothesis gives us that $C_1' \simeq_{\Phi'}^{\mathcal{D}'} C_2'$ so that there exists a trace $T_2' \in \mathbf{comp}(\mathrm{Tr}(C_2'))$ such that $T_1' \simeq_{\Phi'}^{\mathcal{D}'} T_2'$. Finally, writing T_2 for the trace $a_2 \cdot T_2$, we get that $T_1 \simeq_{\Phi}^{\mathcal{D}} T_2$ and $T_2 \in \mathbf{comp}(\mathrm{Tr}(C_1))$, which is what we wanted to prove. Conversely, taking a trace $T_2 \in \mathbf{comp}(\mathrm{Tr}(C_2))$, one can build in the same way a trace $T_1 \in \mathbf{comp}(\mathrm{Tr}(C_1))$ such that $T_1 \simeq_{\Phi}^{\mathcal{D}} T_2$.

□

C Basic Properties on Kripke Open Bisimulations

We first state monotonicity properties of KOB.

Lemma 4. *Taking $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi} w$ with $\Phi' \sqsupseteq \Phi$ satisfying $\Phi' \# \Phi_P$ and $w' \sqsupseteq^{\mathcal{F}*} w$, we get that $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi'} w'$.*

Lemma 5. *Taking $(K_1, K_2) \in \mathcal{K}_{\mathcal{A}} [\![\tau, \sigma]\!]_{\Phi} (w, w_0)$ with $\Phi' \sqsupseteq \Phi$ and $w' \sqsupseteq^{\mathcal{F}*} w$, we get that $(K_1, K_2) \in \mathcal{K}_{\mathcal{A}} [\![\tau, \sigma]\!]_{\Phi'} (w', w_0)$.*

Lemma 6. *Taking $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi} (w, w_0)$ with $\Phi' \sqsupseteq \Phi$, we get that $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi'} (w, w_0)$.*

Proof: The three lemmas are proven by a mutual coinduction.

Functional environments Let $(f_1, f_2, \sigma \to \sigma') \in \Phi_P$ with $\gamma_i(f_i) = u_i$, we must prove that $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\sigma \to \sigma']\!]_{\Phi'} w'$. From $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\sigma \to \sigma']\!]_{\Phi} w$, we get that for all $\Phi'' \# \Phi'$ and $(v_1, \Phi_1''), (v_2, \Phi_2'') \in [\![\tau]\!]$ with $v_1 \sim_{\Phi'', w'.\mathcal{D}} v_2$, $(u_1\ v_1, u_2\ v_2) \in \mathcal{E}_{\mathcal{A}} [\![\sigma]\!]_{\Phi.\Phi''} (w', w')$. But from the coinduction hypothesis on Terms, we get that $(u_1\ v_1, u_2\ v_2) \in \mathcal{E}_{\mathcal{A}} [\![\sigma]\!]_{\Phi.\Phi''} (w', w')$.

Contexts The same proof applies as for functional environments.

Terms Let $(h_1, h_2, \mathcal{D}, \Phi'') \in \mathbf{P}_{\Phi'}(w)$, then writing $\widetilde{\Phi}$ for $\Phi' \backslash \Phi$, we get that $(h_1, h_2, \mathcal{D}, \Phi'' \backslash \widetilde{\Phi}) \in \mathbf{P}_{\Phi}(w)$. From $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi} (w, w_0)$, we get the existence of $M_1, M_2, w' \sqsupseteq w$ and $(h_1', h_2', \mathcal{D}') \in \mathbf{Q}_{\Phi'' \backslash \widetilde{\Phi}}(w')$. such that both $(M_i, h_i) \mapsto^* (M_i', h_i')$. But we have $(h_1', h_2', \mathcal{D}') \in \mathbf{Q}_{\Phi''}(w')$ from the coinduction hypothesis on values, and we conclude the proof using the coinduction hypothesis on values and contexts.

□

Then, we prove a property of the predicative reasoning of KOB.

Lemma 7. *Taking M a term, w a world such that for all $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi}(w)$, we have $(M, h_1, \mathcal{D}_1) \in \mathcal{E}_{\mathcal{A}}^1 [\![\tau]\!]_{\Phi_1'} (w, w_0)$ (resp. $(M, h_2, \mathcal{D}_2) \in \mathcal{E}_{\mathcal{A}}^2 [\![\tau]\!]_{\Phi_2} (w, w_0)$), then we get that $(M, \bot_\tau) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi} (w, w_0)$ (resp. $(\bot_\tau, M) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi} (w, w_0)$).*

Proof: Let us prove that $(M, \bot_\tau) \in \mathcal{E}_\mathcal{A}[\![\tau]\!]_\Phi(w, w_0)$. Taking $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$, since $(\bot_\tau, h_2) \Uparrow$, one must prove that $(M, h_1, \mathcal{D}_1) \in \mathcal{E}^1_\mathcal{A}[\![\tau]\!]_{\Phi'_1}(w, w_0)$ which is exactly the hypothesis of the lemma.

The same reasoning applies to prove that $(\bot_\tau, M) \in \mathcal{E}_\mathcal{A}[\![\tau]\!]_\Phi(w, w_0)$. □

Finally, we prove properties of KOB at inconsistent states.

Lemma 8. *Taking* $(v_1, v_2) \in \mathcal{V}_\mathcal{A}[\![\tau]\!]_\Phi w$ *with* $\mathbf{incons}(w)$*, we get that both* $v_i \in \mathcal{V}^i_\mathcal{A}[\![\tau]\!]_{\Phi_i} w$.

Lemma 9. *Taking* $(K_1, K_2) \in \mathcal{K}_\mathcal{A}[\![\sigma, \tau]\!]_\Phi(w, w_0)$ *with* $\mathbf{incons}(w)$*, we get that both* $K_i \in \mathcal{K}_\mathcal{A}[\![i, \sigma]\!]_\tau \Phi_i(w, w_0)$.

Lemma 10. *Taking* $(M_1, M_2) \in \mathcal{E}_\mathcal{A}[\![\tau]\!]_\Phi(w, w_0)$ *with* $\mathbf{incons}(w)$*, we get that for all* $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$*, both* $(M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_\mathcal{A}[\![\tau]\!]_{\Phi'_i}(w, w_0)$.

Proof:

Values If τ is a ground type, then it is straightforward that $v_i \in \mathcal{V}^i_\mathcal{A}[\![\tau]\!]_{\Phi_i} w$.

Suppose that $\tau = \sigma \to \sigma'$. Then taking $w' \sqsupseteq^{\mathcal{F}*} w$, and $(u_1, \phi_1), (u_2, \phi_2) \in [\![\sigma]\!]$ such that there exists Φ' with $\Phi'_i = \phi'_i$, $\Phi' \# \Phi$ and $u_1 \sim^{w'.\mathcal{D}}_{\Phi'} $, we get that $(v_1\ u_1, v_2\ u_2) \in \mathcal{E}_\mathcal{A}[\![\sigma']\!]_{\Phi.\Phi'} w'$.

Then, the coinduction hypothesis on terms gives us that for all $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$, both $(v_i\ u_i, h_i, \mathcal{D}_i, \Phi'_i) \in \mathcal{E}^i_\mathcal{A}[\![\sigma]\!]_{\Phi'_i}(w', w_0)$. So from Lemma 7, we have that $(v_1\ u_1, \bot_\sigma) \in \mathcal{E}_\mathcal{A}[\![\sigma]\!]_{\Phi_i}(w', w_0)$ and $(\bot_\sigma, v_2\ u_2) \in \mathcal{E}_\mathcal{A}[\![\sigma]\!]_{\Phi_i}(w', w_0)$, i.e. both $v_i \in \mathcal{V}^i_\mathcal{A}[\![\tau]\!]_{\Phi_i} w$.

Contexts The same proof as for values applies.

Terms Suppose first that both $(M_i, h_i) \mapsto^* (M'_i, h'_i)$ with M'_i irreducible. If both M'_i are equal to some values v_i, and there exists a world $w' \sqsupseteq^{\mathcal{F}*} w$ such that $(h'_1, h'_2, \mathcal{D}) \in \mathbf{Q}_{\Phi'}(w')$ and $(v_1, v_2) \in \mathcal{V}_\mathcal{A}[\![\tau]\!]_{\Phi'} w'$. Then the coinduction hypothesis on values gives us that both $v_i \in \mathcal{V}^i_\mathcal{A}[\![\tau]\!]_{\Phi'_i} w'$, since $\mathbf{incons}(w')$. Moreover, both $(h'_i, \mathcal{D}_i) \in \mathbf{Q}^i_{\Phi'_i}(w')$, so we conclude that both $(M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_\mathcal{A}[\![\tau]\!]_{\Phi'_i}(w, w_0)$.

Otherwise, suppose that both M'_i are equal to some callbacks $K_i[f_i\ v_i]$ with $(f_1, f_2, \sigma \to \sigma') \in \Phi'$, and there exists a world $w' \sqsupseteq^{\mathcal{F}*} w$ such that $(h'_1, h'_2, \mathcal{D}) \in \mathbf{Q}_{\Phi'}(w')$, $(K_1, K_2) \in \mathcal{K}_\mathcal{A}[\![\sigma, \tau]\!]_{\Phi'} w'$ and $(v_1, v_2) \in \mathcal{V}_\mathcal{A}[\![\tau]\!]_{\Phi'} w'$. Then the coinduction hypothesis on values and contexts gives us that both $K_i \in \mathcal{K}^i_\mathcal{A}[\![\sigma, \tau]\!]_{\Phi'_i} w'$ and both $v_i \in \mathcal{V}^i_\mathcal{A}[\![\sigma']\!]_{\Phi'_i} w'$, since $\mathbf{incons}(w')$. Moreover, both $(h'_i, \mathcal{D}_i) \in \mathbf{Q}^i_{\Phi'_i}(w')$, so we conclude that both $(M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_\mathcal{A}[\![\tau]\!]_{\Phi_i}(w, w_0)$.

Otherwise, we have directly that both $(M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_\mathcal{A}[\![\tau]\!]_{\Phi_i}(w, w_0)$.

□

C.1 Compositionality

Let us consider w_1, w_2 two worlds such that $w_j = (s^j, h_1^j, h_2^j, \mathcal{D}^j, b^j)$ with $\mathrm{dom}(h_i^1) \cap \mathrm{dom}(h_i^2) = \varnothing$ for $i \in \{1,2\}$ and $\mathrm{dom}(\mathcal{D}^1) \cap \mathrm{dom}(\mathcal{D}^2) = \varnothing$. Then we define the product of w_1, w_2, written $w_1 \otimes w_2$, as the world $((s^1, s^2), h_1^1 \cdot h_2^1, h_2^1 \cdot h_2^2, b^1 \vee b^2)$.

Definition 7. *Let consider $\mathcal{A}_1, \mathcal{A}_2$ two WTS such that $\mathcal{A}_i = (\delta^i_{\mathbf{priv}}, \delta^i_{\mathbf{pub}})$. We define the product of $\mathcal{A}_1, \mathcal{A}_2$, written $\mathcal{A}_1 \otimes \mathcal{A}_2$, as the WTS $(\delta_{\mathbf{priv}}, \delta_{\mathbf{pub}})$ where $\delta_{\mathbf{X}} \stackrel{def}{=} \{(w, w') \mid \exists w_1 w_2, w_1', w_2'. w = w_1 \otimes w_2 \wedge w' = w_1' \otimes w_2' \wedge \delta^1_{\mathbf{X}}(w_1, w_1') \wedge \delta^2_{\mathbf{X}}(w_2, w_2')\}$ with $X \in \{\mathbf{priv}, \mathbf{pub}\}$.*

Lemma 11. *Let us consider $\mathcal{A}_1, \mathcal{A}_2$ two WTS and w_1, w_2 two worlds such that such that and $w_1 \otimes w_2$ is well defined. Suppose that $(v_1, v_2) \in \mathcal{V}_{\mathcal{A}_1} [\![\tau]\!]_\Phi w_1$, then $(v_1, v_2) \in \mathcal{V}_{(\mathcal{A}_1 \otimes \mathcal{A}_2)} [\![\tau]\!]_\Phi (w_1 \otimes w_2)$.*

Lemma 12. *Let us consider $\mathcal{A}_1, \mathcal{A}_2$ two WTS and w_0^1, w_0^2, w^1, w^2 four worlds such that $w_0^1 \otimes w_0^2$ and $w^1 \otimes w^2$ are well defined. Suppose that $(K_1, K_2) \in \mathcal{E}_{\mathcal{A}_1} [\![\tau]\!]_\Phi (w^1, w_0^1)$, then $(M_1, M_2) \in \mathcal{K}_{(\mathcal{A}_1 \otimes \mathcal{A}_2)} [\![\tau, \sigma]\!]_\Phi ((w_1 \otimes w_2), (w_0^1 \otimes w_0^2))$.*

Lemma 13. *Let us consider $\mathcal{A}_1, \mathcal{A}_2$ two WTS and w_0^1, w_0^2, w^1, w^2 four worlds such that $w_0^1 \otimes w_0^2$ and $w^1 \otimes w^2$ are well defined. Suppose that for all $\widetilde{w_0^1} \in \mathcal{F}(w_0^1)$ and $\widetilde{w^1} \in \mathcal{F}(w^1)$ with $\widetilde{w^1} \sqsupseteq^{\mathcal{F}*} \widetilde{w_0^1}$, we have $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}_1} [\![\tau]\!]_\Phi (\widetilde{w^1}, \widetilde{w_0^1})$, then $(M_1, M_2) \in \mathcal{E}_{(\mathcal{A}_1 \otimes \mathcal{A}_2)} [\![\tau]\!]_\Phi ((w_1 \otimes w_2), (w_0^1 \otimes w_0^2))$.*

Theorem 8. *Let us consider $\mathcal{A}_1, \mathcal{A}_2$ two WTS and w_1, w_2 two worlds such that such that $\mathcal{A}_1 \otimes \mathcal{A}_2$ and $w_1 \otimes w_2$ are well defined. Taking $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}_1} [\![\tau \to \sigma]\!]_\Phi w_1$ and $(N_1, N_2) \in \mathcal{E}_{\mathcal{A}_2} [\![\tau]\!]_\Phi w_2$, we get that $(M_1\ N_1, M_2\ N_2) \in \mathcal{E}_{(\mathcal{A}_1 \otimes \mathcal{A}_2)} [\![\tau]\!]_\Phi (w_1 \otimes w_2)$.*

C.2 Dealing with η-equivalence.

We treat λ-abstractions and variables of functional types in a uniform way. This allows Kripke open bisimulations to be compatible with η-equivalence, which is not the case of RTS defined in [3]. Let us prove that $(f, \lambda x.fx) \in \mathcal{V}_{\mathcal{A}} [\![\sigma \to \tau]\!]_\Phi w$ with $\Phi = (f, f, \sigma \to \tau)$. To do so, we take $\forall w' \sqsupseteq^{\mathcal{F}*} w, \Phi' \# \Phi$, and v_1, v_2 two abstract values such that $(v_1, \Phi_1'), (v_2, \Phi_2') \in [\![\sigma]\!]$ and $v_1 \sim_{\Phi', w'.\mathcal{D}} v_2$. We then have to prove that $(fv_1, (\lambda x.fx)v_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi.\Phi'} w'$, which amounts to prove that $(v_1, v_2) \in \mathcal{V}_{\mathcal{A}} [\![\sigma]\!]_{\Phi.\Phi'} w'$, which is proven in the following lemma.

Lemma 14. *Let us consider Φ a span on functional names, w a world and v_1, v_2 two abstract values such that $(v_1, \Phi_1), (v_2, \Phi_2) \in [\![\tau]\!]$ and $v_1 \sim_{\Phi, w.\mathcal{D}} v_2$. Then $(v_1, v_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_\Phi w$,*

Proof: By induction on τ:

- If τ is of ground type, this is straightforward.
- If τ is some functional type $\sigma \to \sigma'$, we take $w' \sqsupseteq^{\mathcal{F}*} w, \Phi' \# \Phi$, and v_1', v_2' two abstract values such that $(v_1', \Phi_1'), (v_2', \Phi_2') \in [\![\sigma]\!]$ and $v_1' \sim_{\Phi', w'.\mathcal{D}} v_2'$. Then, we must prove that $(v_1\ v_1', v_2\ v_2') \in \mathcal{E}_{\mathcal{A}} [\![\sigma']\!]_{\Phi.\Phi'} w'$. To do so, we just have to prove that $(v_1', v_2') \in \mathcal{V}_{\mathcal{A}} [\![\sigma]\!]_{\Phi.\Phi'} w'$, which comes from the induction hypothesis.

– If τ is some product type $\sigma \times \sigma'$, then both $v_i = \langle v_i^1, v_i^2 \rangle$, and from $v_1 \sim_{\Phi, w.\mathcal{D}} v_2$ we get the existence of two disjoint spans Φ^1, Φ^2 such that:
  - $\Phi = \Phi^1 \cdot \Phi^2$,
  - for all $j \in \{1, 2\}$, $(v_1^j, \Phi_1^j), (v_2^j, \Phi_2^2) \in [\![\sigma_j]\!]$,
  - for all $j \in \{1, 2\}$, $v_1^j \sim_{\Phi^j, w.\mathcal{D}} v_2^j$.

  Then, the coinduction hypothesis gives us that both $(v_1^j, v_2^j) \in \mathcal{V}_{\mathcal{A}} [\![\sigma_j]\!]_{\Phi^j} w$, so that $(\langle v_1^1, v_1^2 \rangle, \langle v_2^1, v_2^2 \rangle) \in \mathcal{V}_{\mathcal{A}} [\![\sigma \times \sigma']\!]_{\Phi} w$.

□

D Soundness

Lemma 15. *Let us consider* $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} w$. *Then taking* $(v_1, \phi_1, \gamma_1) \in \mathbf{AVal}_{u_1}(\tau)$ *and* $(v_2, \phi_2, \gamma_2) \in \mathbf{AVal}_{u_2}(\tau)$ *such that* $\mathrm{dom}(\phi_i) \cap \mathrm{dom}(\Phi_{O,i}) = \varnothing$, *there exists a span* Φ_P *satisfying* $\Phi_{P,i} = \phi_i$ *such that* $v_1 \sim_{\Phi_P}^{w.\mathcal{D}} v_2$ *and* $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$.

Proof: By induction on τ.

– If τ is a ground type, then it is straightforward since both γ_i are empty, and both $u_i = v_i$.
– Suppose that $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\sigma \to \tau]\!]_{\Phi_O} w$. Then we have both $v_i = f_i$ with $f_i \notin \mathrm{dom}(\Phi_{O,i})$, $\phi = [f_i \mapsto \sigma \to \tau]$ and $\gamma_i = [f_i \mapsto u_i]$. So we define Φ_P as the span $\{(f_1, f_2, \sigma \to \tau)\}$, and indeed $v_1 \sim_{\Phi_P \cdot \Phi_O}^{w.\mathcal{D}} v_2$ and $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$.
– Suppose that $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\sigma_1 \times \sigma_2]\!]_{\Phi_O} w$, then both u_i are equal to some pairs $\langle u_i^1, u_i^2 \rangle$. Then for all $i \in \{1, 2\}$ we have $v_i = \langle v_i^1, v_i^2 \rangle$ with $(v_i^1, \phi_i^1, \gamma_i^1) \in \mathbf{AVal}_{u_i^1}(\sigma_1)$ and $(v_i^2, \phi_i^2, \gamma_i^2) \in \mathbf{AVal}_{u_i^2}(\sigma_2)$ such that $\mathrm{dom}(\phi_i^1), \mathrm{dom}(\phi_i^2)$ and $\mathrm{dom}(\Phi_{O,i})$ are two by two disjoint. Then, the induction hypothesis gives us the existence of two spans Φ_P^1, Φ_P^2 satisfying for all $j \in \{1, 2\}$, $\Phi_{P,i}^j = \phi_i^j$ and $v_1^j \sim_{\Phi_P^j \cdot \Phi_O}^{w.\mathcal{D}} v_2^j$ and $(\gamma_1^j, \gamma_2^j) \in \mathcal{G}_{\mathcal{A}} \left[\!\!\left[\Phi_P^j\right]\!\!\right]_{\Phi_O} w$. From the fact that $\mathrm{dom}(\phi_i^1) \cap \mathrm{dom}(\phi_i^2) = \varnothing$, we deduce that the span $\Phi_P^1 \cup \Phi_P^2$ is well formed, and we can conclude easily.

□

Lemma 16. *Let* w *a world and* $(h_1, h_2, \mathcal{D}) \in \mathbf{Q}_{\Phi_O}(w)$. *Then taking* $(h_1', \gamma_1, \phi_1) \in \mathbf{AHeap}_{\mathcal{D}_1}(h_{1|\mathcal{D}_1})$ *and* $(h_2', \gamma_2, \phi_2) \in \mathbf{AHeap}_{\mathcal{D}_2}(h_{2|\mathcal{D}_2})$ *such that* $\mathrm{dom}(\phi_i) \cap \Phi_{O,i} = \varnothing$, *there exists a span* Φ_P *satisfying* $\Phi_{P,i} = \phi_i$ *such that* $(h_1[h_1'], h_2[h_2'], \mathcal{D}, \Phi_P \cdot \Phi_O) \in \mathbf{P}_{\Phi_O}(w)$ *and* $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$.

Proof: By induction on the size of $\mathcal{D}$. If it is empty it is straightforward. Otherwise, let suppose that $\mathcal{D} = \{(l_1, l_2, \tau)\} \cup \mathcal{D}'$. Then, we have both:

– $h_i = \widetilde{h_i} \cdot [l_i \mapsto v_i]$,
– $h_i' = h_i'' \cdot [l_i \mapsto v_i]$,
– $\gamma_i = \gamma_i' \cdot \gamma_i''$ and
– $\phi_i = \phi_i' \cdot \phi_i''$ with

– $(h_1'', \gamma_1'', \phi_1'') \in \mathbf{AHeap}_{\mathcal{D}_1}(\widetilde{h_{i|\mathcal{D}_1}})$,
– $(v_i, \phi_1', \gamma_i') \in \mathbf{AVal}_{h_i(l_1)}(\tau)$.

From Lemma 1, we get that there exists a span Φ'_P satisfying $\Phi'_{P,i} = \phi_i'$ such that $v_1 \sim^{w,\mathcal{D}}_{\Phi'_P} v_2$ and $(\gamma_1', \gamma_2') \in \mathcal{G}_{\mathcal{A}} [\![\Phi'_P]\!]_{\Phi_O} w$. Moreover, the induction hypothesis gives us that there exists a span Φ''_P satisfying $\Phi''_{P,i} = \phi_i''$ such that $(\widetilde{h_1}[h_1''], \widetilde{h_2}[h_2''], \mathcal{D}, \Phi''_P \cdot \Phi_O) \in \mathbf{P}_{\Phi_O}(w)$ and $(\gamma_1'', \gamma_2'') \in \mathcal{G}_{\mathcal{A}} [\![\Phi''_P]\!]_{\Phi_O} w$. Since both $h_i[h_i'] = \widetilde{h_i}[h_i''] \cdot [l_i \mapsto v_i]$, we can conclude easily. □

Lemma 17. *Let us consider $i \in \{1, 2\}$, w a world and $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_\phi w$. Then taking $(v, \gamma, \phi') \in \mathbf{AVal}_u(\tau)$ such that $\mathrm{dom}(\phi) \cap \mathrm{dom}(\phi') = \varnothing$, we get that $\gamma \in \mathcal{G}_{\mathcal{A}} [\![\phi']\!]_\phi w$.*

Proof: By a straightforward induction on τ. □

Lemma 18. *Let us consider $i \in \{1, 2\}$, w a world and $(h, D) \in \mathbf{Q}^i_\phi(w)$. Then taking $(h', \gamma, \phi') \in \mathbf{AHeap}_{\mathcal{D}_i}(h_{|\mathcal{D}_i})$ such that $\mathrm{dom}(\phi) \cap \mathrm{dom}(\phi') = \varnothing$, we get that $(h[h'], \mathcal{D}_i, \phi \cdot \phi') \in \mathbf{P}_\phi(w)$ and $\gamma \in \mathcal{G}_{\mathcal{A}} [\![\phi']\!]_\phi w$.*

Proof: By a straightforward induction on the size of D, using Lemma 17 to conclude. □

Then, we first prove a correspondence between the predicative reasoning with the Kripke open bisimulations and with the bisimulations on traces.

Lemma 19. *Let $i \in \{1, 2\}$, $n \in \mathbb{N}$ and $n + 1$ world $w_n \sqsupseteq^{\mathcal{F}*} \ldots \sqsupseteq^{\mathcal{F}*} w_0$ such that*
– $w \sqsupseteq^*_{\mathbf{pub}} w_0.\mathbf{cons}(w)$,
– *for all $j \in \{1, \ldots, n\}$, $K^j \in \mathcal{K}_{\mathcal{A}} [\![\sigma_j, \tau_j]\!]_{\phi_O} (w_n, w_{j-1})$,*
– $\gamma \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_P} \phi_O$.

Then, for all $(h, D, \phi') \in \mathbf{P}^i_{\phi_P \cdot \phi_O}(w_n)$, writing $\mathcal{S}$ for the stack $(K^n[\bullet_{\sigma_n}], \tau_n) :: \ldots :: (K^1[\bullet_{\sigma_1}], \tau_1)$, we get that $\langle \mathcal{S}, \gamma, \phi', h, \mathcal{D} \rangle \in \mathcal{O}^{\natural i}$.

Lemma 20. *Let $i \in \{1, 2\}$, $n \in \mathbb{N}$ and $n + 2$ world $w_{n+1} \sqsupseteq^{\mathcal{F}*} \ldots \sqsupseteq^{\mathcal{F}*} w_0$ such that*
– $w \sqsupseteq^*_{\mathbf{pub}} w_0.\mathbf{cons}(w)$,
– *for all $j \in \{1, \ldots, n\}$, $K^j \in \mathcal{K}^i_{\mathcal{A}} [\![\sigma_j, \tau_j]\!]_{\phi_O} (w_n, w_{j-1})$,*
– $\gamma \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_P} \phi_O$.

Writing $\mathcal{S}$ for the stack $(K^n[\bullet_{\sigma_n}], \tau_n) :: \ldots :: (K^1[\bullet_{\sigma_1}], \tau_1)$, we get that for all $(h, D, \phi') \in \mathbf{P}^i_{\phi_P \cdot \phi_O}(w_n)$, if $(M, h, D) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} (w_{n+1}, w_n)$ then $\langle (M, \tau)\mathcal{S}, \gamma, \phi', h, D \rangle \in \mathcal{P}^{\natural i}$.

Proof: The two lemmas are proven by a mutual coinduction.

Soundness on Diverging Contexts Taking $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi_P \cdot \Phi_O}(w_n)$, we write C for the configuration $\langle \mathcal{S}, \gamma, \phi', h, D \rangle$. Let us consider any action a and a configuration $C' = \langle \mathcal{S}', \gamma, \phi'', h[h'], D' \rangle$ such that $C \stackrel{a}{\Rightarrow} C'$.

– If a is an answer $(\langle v\rangle, h')$, we get that $\mathcal{S}' = (K^n[v], \tau_n) :: (K^{n-1}[\bullet_{\sigma_{n-1}}], \tau_{n-1}) :: \ldots :: (K^1[\bullet_{\sigma_1}], \tau_1)$. Let us consider $\mathcal{D}'$ such that $\mathcal{D}'_i = D' \backslash D$, so that we define $w' = (w.s, w.h_1, w.h_2, \mathcal{D} \cup \mathcal{D}', w.b)$. Then, from $K^n \in \mathcal{K}^i_{\mathcal{A}} [\![\sigma_n, \tau_n]\!]_{\phi'} (w_n, w_{n-1})$ and $(h[h'], D', \phi'') \in \mathbf{P}^i_{\phi''}(w')$, we get that $(K^n[v], h', D') \in \mathcal{E}^i_{\mathcal{A}} [\![\tau_n]\!]_{\phi''} (w', w_n)$. Finally, the coinduction hypothesis on diverging terms gives us that $C' \in \mathcal{P}^{\natural\, i}$.
– If a is a question $(f\, \langle v\rangle, h'')$ with $\gamma(f) = u$ and $\phi_P(f) = \sigma \to \tau$, we get that $\mathcal{S}' = (u\ v, \tau) :: \mathcal{S}$. Let us consider $\mathcal{D}'$ such that $\mathcal{D}'_i = D' \backslash D$, so that we define $w' = (w.s, w.h_1, w.h_2, \mathcal{D} \cup \mathcal{D}', w.b)$. Then, from $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} w_n$ and $(h[h'], D', \phi'') \in \mathbf{P}^i_{\phi''}(w')$, we get that $(u\ v, h[h'], D') \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi''} (w', w_n)$. Finally, the coinduction hypothesis on diverging terms gives us that $C' \in \mathcal{P}^{\natural\, i}$.

Soundness on Diverging Terms Taking $(h, D, \phi') \in \mathbf{P}^i_{\phi_P \cdot \phi_O}(w_n)$, such that $(M, h, D) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} (w_{n+1}, w_n)$, we write C for the configuration $\langle (M, \tau)\mathcal{S}, \gamma, \phi', h, D\rangle$.

Let us consider any action a such that $C \to C' \xrightarrow{a} C''$ with $C' = \langle (M', \tau) :: \mathcal{S}, \gamma, \Phi'_i, h', \mathcal{D}_i\rangle$ and $C'' = \langle \mathcal{S}', \gamma', \phi'', h'[h''], D'\rangle$, where $\gamma' = \gamma \cdot \gamma_v \cdot \gamma_h$ and $\phi'' = \Phi'_i \cdot \phi_v \cdot \phi_h$ and $(h'', \gamma_h, \phi_h) \in \mathbf{AHeap}_{\mathcal{D}'}(h')$.

From $(M, h, D) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} (w_{n+1}, w_n)$ we get the existence of a future world $w' \sqsupseteq w_{n+1}$ with $\mathbf{incons}(w')$ such that $(h', D') \in \mathbf{Q}^i_{\phi'}(w')$. Thus, Lemma 18 gives us that $\gamma_h \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_h} \Phi'_i$ and $(h[h''], D', \phi'') \in \mathbf{P}^i_{\phi''}(w')$.

– If a is an answer $(\langle \bar{v}\rangle, h'')$, we get that $\mathcal{S}' = (K^{n-1}[\bullet_{\sigma_{n-1}}], \tau_{n-1}) :: \ldots :: (K^1[\bullet_{\sigma_1}], \tau_1)$, $M' = u$ with $(v, \gamma_v, \phi_v) \in \mathbf{AVal}_u(\tau)$. Crucially, w_n is inconsistent since $w' \sqsupseteq_{\mathbf{pub}} w_n$ (because a is an answer), and w' is inconsistent. This means that w_n is distinct from w_0, so that $n \neq 0$.
From $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_{\Phi'_i} w$, we get from Lemma 17 that $\gamma_v \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_v} \Phi'_i$. From Lemma 4, we finally get that $\gamma' \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\Phi_{P,i} \cdot \phi_v \cdot \phi_h} \Phi'_i$, so the coinduction hypothesis on diverging contexts (since $n \neq 0$) gives us that $C'' \in \mathcal{O}^{\natural\, \phi''} D'$.
– If a is a question $(\bar{f}\, \langle v\rangle, h'')$ we get that $\mathcal{S}' = \mathcal{S}$ and $M' = K[f\ u]$ with $\Phi'_i(f) = \sigma \to \sigma'$. From $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_{\Phi'_i} w$, we get from Lemma 17 that $\gamma_v \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_v} \Phi'_i$, so using Lemma 4, we finally get that $\gamma' \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\Phi_{P,i} \cdot \phi_v \cdot \phi_h} \Phi'_i$. Using the fact that $K \in \mathcal{K}^i_{\mathcal{A}} [\![\sigma', \tau]\!]_{\Phi'_i} w$, the coinduction hypothesis on diverging contexts gives us that $C' \in \mathcal{P}^{\natural\, i}$.

□

Finally, we can state the wanted soundness theorems on Player and Opponent stacks.

Theorem 9. *Let $n \in \mathbb{N}$ and $n+2$ world $w_{n+1} \sqsupseteq^{\mathcal{F}*} w_n \sqsupseteq^{\mathcal{F}*} \ldots \sqsupseteq^{\mathcal{F}*} w_0$ such that*
– $w \sqsupseteq^*_{\mathbf{pub}} w_0.\mathbf{cons}(w)$,
– $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} (w_{n+1}, w_n)$,
– *for all* $j \in \{1, \ldots, n\}$, $(K^j_1, K^j_2) \in \mathcal{K}_{\mathcal{A}} [\![\sigma_j, \tau_j]\!]_{\Phi_O} (w_{n+1}, w_{j-1})$,
– $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w_{n+1}$.
Then for all $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi_P \cdot \Phi_O}(w_{n+1})$, writing $\mathcal{S}_i$ for $(K^n_i[\bullet_{\sigma_n}], \tau_n) :: \ldots :: (K^1_i[\bullet_{\sigma_1}], \tau_1)$, $(\langle (M_1, \tau) :: \mathcal{S}_1, \gamma_1, \Phi'_1, h_1, \mathcal{D}_1\rangle, \langle (M_2, \tau) :: \mathcal{S}_2, \gamma_2, \Phi'_2, h_2, \mathcal{D}_2\rangle) \in \mathcal{P}_{\Phi', \mathcal{D}}$.

Theorem 10. *Let $n \in \mathbb{N}$ and n world $w_n \sqsupseteq^{\mathcal{F}*} \ldots \sqsupseteq^{\mathcal{F}*} w_0$ such that*

– $\forall w'_0 \sqsupseteq^*_{\mathbf{pub}} w_0.\mathbf{cons}(w'_0)$,
– *for all* $j \in \{1,\dots,n\}, (K_1^j, K_2^j) \in \mathcal{K}_{\mathcal{A}}\,[\![\sigma_j,\tau_j]\!]_{\Phi_O}\,(w_n, w_{j-1})$,
– $(\gamma_1,\gamma_2) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_P]\!]_{\Phi_O}\,w_n$.

Then,for all $(h_1,h_2,\mathcal{D},\Phi') \in \mathbf{P}_{\Phi_P\cdot\Phi_O}(w_n)$*, writing* $\mathcal{S}_i$ *for the stack* $(K_i^n[\bullet_{\sigma_n}],\tau_n) :: \dots :: (K_i^1[\bullet_{\sigma_1}],\tau_1)$, $(\langle \mathcal{S}_1,\gamma_1,\Phi'_1,h_1,\mathcal{D}_1\rangle, \langle \mathcal{S}_2,\gamma_2,\Phi'_2,\mathcal{D}_2,\rangle) \in \mathcal{O}_{\Phi',\mathcal{D}}$.

Proof:

Soundness on Terms Let us consider $(h_1,h_2,\mathcal{D},\Phi') \in \mathbf{P}_{\Phi_O\cdot\Phi'}(w)$, so that we write C_i for the Player configuration $\langle (M_i,\theta_i),\gamma_i,\Phi'_i,h_i,\mathcal{D}_i\rangle$.

Suppose first that both $(M_i,h_i,D_i) \in \mathcal{E}^i_{\mathcal{A}}\,[\![\tau]\!]_{\Phi'_{O,i}}\,(w,w_0)$, we prove that both $C_i \in \mathcal{P}^{\natural\, i}$. So let $i \in \{1,2\}$. First, if $(M_i,h_i)\uparrow$, then indeed $C_i \in \mathcal{P}^{\natural\, i}$. Otherwise, there exists (M',h') irreducible such that $(M_i,h_i) \mapsto^* (M',h')$. Then there exists a worlds $w' \sqsupseteq w_{n+1}$ with $\mathbf{incons}(w')$ and $(M',h',D') \in \mathcal{E}^i_{\mathcal{A}}\,[\![\tau]\!]_{\Phi'_{O,i}}\,(w',w_n)$. Moreover, Lemma 9 and Lemma 5 gives us that for all $j \in \{1,\dots,n\}, K_i^j \in \mathcal{K}^i_{\mathcal{A}}\,[\![\sigma_j,\tau_j]\!]_{\Phi_O}\,(w',w_{j-1})$, so Lemma 20 gives us that $\langle \mathcal{S}_i,\gamma'_i,\Phi'_i,h'_i[h''_i],\mathcal{D}'_i\rangle \in \mathcal{O}^{\natural\, i}$.

Otherwise, we get that both

$$\begin{aligned} C_i \to\ & \langle (M'_i,\theta_i) :: \mathcal{S}_i,\gamma_i,\Phi'_i,h'_i,\mathcal{D}_i\rangle \\ \xrightarrow{a_i}\ & \langle \mathcal{S}_i,\gamma'_i,\phi''_i,h'_i[h''_i],\mathcal{D}'_i\rangle \end{aligned}$$

i.e. $(M_i,h_i) \mapsto (M'_i,h'_i)$ with M'_1, M'_2 irreducible. From $(M_1,M_2) \in \mathcal{E}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi_O}\,(w_{n+1},w_n)$, we get that here exists a span $\mathcal{D}'$ with $\mathcal{D}'_i = D'_i$, and a world $w'_{n+1} \sqsupseteq w_{n+1}$ such that $(h'_1,h'_2,\mathcal{D}') \in \mathbf{Q}_{\Phi'}(w'_{n+1})$.

– If both M'_i are equal to values u_i, then both $a_i = (\langle \bar{v_i}\rangle, h''_i)$ where $(v_i,\gamma_{v,i},\phi_{v,i}) \in \mathbf{AVal}_{u_i}(\theta_i)$ and $(h''_i,\gamma_{h,i},\phi_{h,i}) \in \mathbf{AHeap}_{\mathcal{D}'_i}(h'_{i|\mathcal{D}'_i})$ with $\phi''_i = \Phi'_i \cdot \phi_{v,i}\cdot\phi_{h,i}$. From $(u_1,u_2) \in \mathcal{V}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi'}\,w'_{n+1}$, Lemma 1 gives us the existence of a span Φ_v satisfying $\Phi_{v,i} = \phi_{v,i}$ such that $v_1 \sim^{\mathcal{D}'}_{\Phi'\cdot\Phi_v} v_2$ and $(\gamma_{v,1},\gamma_{v,2}) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_v]\!]_{\Phi'}\,w'_{n+1}$. Moreover, Lemma 16 gives us the existence of a span Φ_v satisfying $\Phi_{v,i} = \phi_{v,i}$ such that $(h'_1[h''_1],h'_2[h''_2],\mathcal{D}',\Phi_v\cdot\Phi') \in \mathbf{P}_{\Phi'}(w'_{n+1})$ and $(\gamma_{h,1},\gamma_{h,2}) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_h]\!]_{\Phi'}\,w'_{n+1}$. Then $\gamma'_i = \gamma'_i\cdot\gamma_{v,i}\cdot\gamma_{h,i}$, so that from $(\gamma_1,\gamma_2) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_P]\!]_{\Phi_O}\,w_n$ we get from Lemma 4 that $(\gamma_1,\gamma_2) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_P]\!]_{\Phi'}\,w'_{n+1}$, since $w'_{n+1} \sqsupseteq^* w_{n+1} \sqsupseteq^{\mathcal{F}*} w_n$. So $(\gamma'_1,\gamma'_2) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_P\cdot\Phi_v\cdot\Phi_h]\!]_{\Phi'}\,w'_{n+1}$. Finally, from the fact that for all $j \in \{1,\dots,n\}$, $(K_1^j,K_2^j) \in \mathcal{K}_{\mathcal{A}}\,[\![\sigma_j,\tau_j]\!]_{\Phi_O}\,(w_{n+1},w_{j-1})$ and $w'_{n+1} \sqsupseteq_{\mathbf{pub}} w_j$, we get using Lemma 5 that $(K_1^j,K_2^j) \in \mathcal{K}_{\mathcal{A}}\,[\![\sigma_j,\tau_j]\!]_{\Phi'}\,(w'_{n+1},w_{j-1}$, and applying the coinduction hypothesis on Opponent evaluation stacks, we get that

$$(\langle \mathcal{S}_1,\gamma'_1,\phi''_1,h'_1[h''_1],\mathcal{D}_1\rangle, \langle \mathcal{S}_2,\gamma'_2,\phi''_2,h_2[h''_2],\mathcal{D}_2\rangle) \in \mathcal{O}_{\Phi'\cdot\Phi_v\cdot\Phi_h,\mathcal{D}'}$$

– Otherwise both M'_i are equal to callbacks $K_i[f_i\ u_i]$, then both $a_i = (\bar{f_i}\,\langle v_i\rangle, h''_i)$ where $(v_i,\gamma_{v,i},\phi_{v,i}) \in \mathbf{AVal}_{u_i}(\theta_i)$ and $(h''_i,\gamma_{h,i},\phi_{h,i}) \in \mathbf{AHeap}_{\mathcal{D}'_i}(h'_{i|\mathcal{D}'_i})$ with $\mathrm{dom}(\phi_{v,i}), \mathrm{dom}(\phi_{v,i})$ and $\mathrm{dom}(\Phi'_{O,i})$ two by two disjoint. From $(u_1,u_2) \in \mathcal{V}_{\mathcal{A}}\,[\![\tau]\!]_{\Phi'}\,w'$, Lemma 1 gives us the existence of a span Φ_v satisfying $\Phi_{v,i} = \phi_{v,i}$ such that $v_1 \sim^{\mathcal{D}'}_{\Phi\cdot\Phi_v} v_2$ and $(\gamma_{v,1},\gamma_{v,2}) \in \mathcal{G}_{\mathcal{A}}\,[\![\Phi_v]\!]_{\Phi'}\,w'_{n+1}$. Moreover, Lemma 16 gives us the existence of a span Φ_h satisfying $\Phi_{h,i} = \phi_{h,i}$ such that $(h'_1[h''_1],h'_2[h''_2],\mathcal{D}',\Phi_h\cdot$

$\Phi') \in \mathbf{P}_{\Phi'}(w'_{n+1})$ and $(\gamma_{h,1}, \gamma_{h,2}) \in \mathcal{G}_{\mathcal{A}}[\![\Phi_h]\!]_{\Phi'} w'_{n+1}$. Then $\gamma'_i = \gamma'_i \cdot \gamma_{v,i} \cdot \gamma_{h,i}$, so that from $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}}[\![\Phi_P]\!]_{\Phi_O} w_n$ we get that $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}}[\![\Phi_P]\!]_{\Phi'} w'_{n+1}$, since $w'_{n+1} \sqsupseteq^* w_{n+1} \sqsupseteq^{\mathcal{F}*} w_n$. So $(\gamma'_1, \gamma'_2) \in \mathcal{G}_{\mathcal{A}}[\![\Phi_P \cdot \Phi_v \cdot \Phi_h]\!]_{\Phi'} w'_{n+1}$. Then, from $(K_1, K_2) \in \mathcal{K}_{\mathcal{A}}[\![\sigma, \tau]\!]_{\Phi'} w'_{n+1}$, the coinduction hypothesis gives us that $(\langle (K_1[\bullet_\sigma], \tau) :: \mathcal{S}_1, \gamma'_1, \phi''_1, h'_1[h''_1], D'_1\rangle, \langle (K_2[\bullet_\sigma], \tau) :: \mathcal{S}_2, \gamma'_2, \phi''_2, h'_2[h''_2], D'_2\rangle) \in \mathcal{O}_{\Phi', \mathcal{D}'}$.

We conclude that $(\langle (M_1, \tau) :: \mathcal{S}_1, \gamma_1)\rangle, \langle (M_2, \tau) :: \mathcal{S}_2, \gamma_2)\rangle) \in \mathcal{P}_{\mathcal{A}}(\Phi_P \cdot \Phi_O, w_{n+1})$.

Soundness on Contexts Let us consider $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi_P \cdot \Phi_O}(w)$, $\Phi'' \sqsupseteq \Phi', \mathcal{D}' \sqsupseteq \mathcal{D}$ and a_1, a_2 two actions such that $a_1 \sim^{\mathcal{D}'}_{\Phi'(} a_2$.

- Suppose that both $a_i = (\langle v_i \rangle, h'_i)$, so that $n > 0$. From $(K_1^n, K_2^n) \in \mathcal{K}_{\mathcal{A}}[\![\sigma_n, \tau_n]\!]_{\Phi_O}(w_n, w_{n-1})$, we get that, working in the world $w' = (w.s, w.h_1, w.h_2, \mathcal{D}')$, since $w' \in \mathcal{F}_w()$, $(K_1^n[v_1], K_2^n[v_n]) \in \mathcal{E}_{\mathcal{A}}[\![\tau_i]\!]_{\Phi''} w'_n, w_{n-1}$. Then, from the fact that $(h_1[h'_1], h_2[h'_2], \mathcal{D}', \Phi'') \in \mathbf{P}_{\Phi''}(w')$, the coinduction hypothesis gives us that

  $$(\langle (K_1^n[v_1], \tau_n) :: \mathcal{S}'_1, \gamma_1), \Phi''_1, h_1[h'_1], \mathcal{D}'_1\rangle, \langle (K_2^n[v_2], \tau_n) :: \mathcal{S}'_1, \gamma_2), \Phi''_2, h_2[h'_2], \mathcal{D}'_2\rangle) \in \mathcal{O}_{\Phi'', \mathcal{D}'}$$

  with both $\mathcal{S}_i = (K_i[\bullet_\sigma], \tau) :: \mathcal{S}'_i$.
- Otherwise, both $a_i = (f_i \langle u_i \rangle, h'_i)$ with $f_i \in \mathrm{dom}(\gamma_i)$, so that there exists a functional type $\sigma \to \tau$ such that $(f_1, f_2, \sigma \to \tau) \in \Phi_P$. Then, we have that both $C_i \xrightarrow{a_i} \langle (u_i v_i, \sigma) :: \mathcal{S}_i), \gamma_i, \Phi''_i, h_i[h'_i], \mathcal{D}'_i\rangle$. with $\gamma_i(f_i) = u_i$. From $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}}[\![\Phi_P]\!]_{\Phi_O} w_n$, we get that $(u_1\ v_1, u_2\ v_2) \in \mathcal{E}_{\mathcal{A}}[\![\tau_n]\!]_{\Phi'_O} w'$, and the coinduction hypothesis gives us that

  $$(\langle (u_1\ v_1, \tau) :: \mathcal{S}_1, \gamma_1), \Phi''_1, h_1[h'_1], \mathcal{D}'_1\rangle, \langle (u_2\ v_2, \tau) :: \mathcal{S}_2, \gamma_2), \Phi''_2, h_2[h'_2], \mathcal{D}'_2\rangle) \in \mathcal{O}_{\Phi'', \mathcal{D}'}$$

□

E Completeness

E.1 Exhaustive WTS

The exhaustive WTS $\mathcal{A} = (\delta_{\mathbf{priv}}, \delta_{\mathbf{pub}})$ has a tree-structure, i.e., there does not exist two states s, s' s.t. $s' \geq_{\mathcal{A}} s$ and $s \geq_{\mathcal{A}} s'$, where we define $s' \geq_{\mathcal{A}} s$ as $\delta^*_{\mathbf{priv}}(s, s')$ non empty. Thus, for any state s of $\mathcal{A}$ we can define the sub-WTS $\mathcal{A}_{\geq s}$ whose transition functions are restricted to states $s' \geq_{\mathcal{A}} s$. Then, one can relate the properties of two WTS $\mathcal{A}, \mathcal{A}'$ about worlds w, when $\mathcal{A}_{\geq s} = \mathcal{A}'_{\geq s}$ and the state of w is greater than s.

Lemma 21. *Let L a stack of world $w_n :: \ldots w_0$ such that $(\langle \mathcal{S}_1, \gamma_1\rangle \langle \mathcal{S}_2, \gamma_2\rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi, L)$. Let $\mathcal{A}'$ an LTS s.t. $\mathcal{A}_{\geq w_n.s} = \mathcal{A}'_{\geq w_n.s}$ and for all $i \in \{0, \ldots, n-1\}$, $\{s' \mid \mathcal{A}.\delta_{\mathbf{pub}}(w_i.s, s') \wedge s' \geq_{\mathcal{A}} w_n.s\} = \{s' \mid \mathcal{A}'.\delta_{\mathbf{pub}}(w_i.s, s') \wedge s' \geq_{\mathcal{A}'} w_n.s\}$. Then $(\langle \mathcal{S}_1, \gamma_1\rangle \langle \mathcal{S}_2, \gamma_2\rangle) \in \overline{\mathcal{P}}_{\mathcal{A}'}(\Phi, (w :: L))$.*

Theorem 11. *Taking $\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle$ two Player reduced configurations such that both $\mathcal{S}_1, \mathcal{S}_2$ have the same size n, Φ a spans on functional names, L a list of n world whose top element is written w, and $(h_1, h_2, \mathcal{D}, \Phi) \in \mathbf{P}_{\Phi}(w)$. Writing C_i for $\langle \mathcal{S}_i, \gamma_i, \Phi_i, h_i, \mathcal{D}_i\rangle$, if $(C_1, C_2) \in \mathcal{P}_{\Phi, \mathcal{D}}$ then $(\langle \mathcal{S}_1, \gamma_1\rangle \langle \mathcal{S}_2, \gamma_2\rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi, L)$, where $\mathcal{A} = \mathbf{SE}^L_{\Phi}(\langle \mathcal{S}_1, \gamma_1\rangle, \langle \mathcal{S}_2, \gamma_2\rangle)$.*

Theorem 12. *Taking $\langle \mathcal{S}_1, \gamma_1 \rangle, \langle \mathcal{S}_2, \gamma_2 \rangle$ two Opponent reduced configurations such that both $\mathcal{S}_1, \mathcal{S}_2$ have the same size n, Φ a spans on functional names, L a list of $n+2$ world whose top element is written w, and $(h_1, h_2, \mathcal{D}, \Phi) \in \mathbf{P}_\Phi(w)$. Writing C_i for $\langle \mathcal{S}_i, \gamma_i, \Phi_i, h_i, \mathcal{D}_i \rangle$, if $(C_1, C_2) \in \mathcal{O}_{\Phi,\mathcal{D}}$ then $(\langle \mathcal{S}_1, \gamma_1 \rangle \langle \mathcal{S}_2, \gamma_2 \rangle) \in \overline{\mathcal{P}}_\mathcal{A}(\Phi, L)$, where $\mathcal{A} = \mathbf{SK}^L_\Phi(\langle \mathcal{S}_1, \gamma_1 \rangle, \langle \mathcal{S}_2, \gamma_2 \rangle)$.*

Proof:

Terms Let us consider $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$, and suppose that there exists $\Phi'' \sqsupseteq \Phi', \mathcal{D}' \sqsupseteq \mathcal{D}$ and $a_1 \sim^{\mathcal{D}'}_{\Phi''} a_2$ such that, writing C_i for $\langle \mathcal{S}_i, \gamma_i, \Phi'_i, h_i, \mathcal{D}_i \rangle$, one has that both $C_i \stackrel{a_i}{\Longrightarrow} \langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}' \rangle$. Then, defining w' as $(s, h'_{1|\overline{\mathcal{D}'_1}}, h'_{2|\overline{\mathcal{D}'_1}}, \mathcal{D}'')$, one get that $w'\mathcal{A}. \sqsupseteq w$.

Writing $\mathcal{A}'$ for $\mathbf{SK}^{L'}_{\Phi'}(\langle \mathcal{S}'_1, \gamma'_1 \rangle, \langle \mathcal{S}'_2, \gamma'_2 \rangle)$, the coinduction hypothesis gives us that $(\langle \mathcal{S}'_1, \gamma'_1 \rangle \langle \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{O}}_\mathcal{A}(\mathcal{A}', \Phi'')L'$. From Lemma 21, we finally get that $(\langle \mathcal{S}'_1, \gamma'_1 \rangle \langle \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{O}}_\mathcal{A}(\Phi'', L')$.

Contexts Let us consider $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$, a span $\Phi'' \sqsupseteq \Phi$, any world $w' \in \mathcal{F}(w)$ and two actions $a_1 \sim^{w', \mathcal{D}}_{\Phi''} a_2$. From the fact that $\mathcal{A}$ exists, we get that there exists $\mathcal{S}'_1, \mathcal{S}'_2$ such that $(C_1 \stackrel{a_1}{\Longrightarrow} \langle \mathcal{S}'_1, \gamma_1, \Phi''_1, h'_1, \mathcal{D}'_1 \rangle)$, iff $(C_1 \stackrel{a_1}{\Longrightarrow} \langle \mathcal{S}'_1, \gamma_1, \Phi''_1, h'_1, \mathcal{D}'_1 \rangle)$.

Writing $\mathcal{A}'$ for $\mathbf{SE}^{L'}_{\Phi'}(\langle \mathcal{S}'_1, \gamma_1 \rangle, \langle \mathcal{S}'_2, \gamma_2 \rangle)$, the coinduction hypothesis gives us that $(\langle \mathcal{S}'_1, \gamma_1 \rangle \langle \mathcal{S}'_2, \gamma_2 \rangle) \in \overline{\mathcal{P}}_{\mathcal{A}'}(\Phi'', L')$. From Lemma 21, we finally get that $(\langle \mathcal{S}'_1, \gamma_1 \rangle \langle \mathcal{S}'_2, \gamma_2 \rangle) \in \oplus''_\mathcal{A}(L',)$.

We now prove that $(\langle \mathcal{S}_1, \gamma_1 \rangle, \langle \mathcal{S}_2, \gamma_2 \rangle) \in \mathbf{Faithful}_\Phi(w)$. Let us consider $w' \sqsupseteq^{\mathcal{F}*} w$, so that there exists $n \in \mathbb{N}$ and $2 * n$ worlds

- $w_1 \in \mathcal{F}(w)$,
- for all $i \in \{1, \ldots, n\}, w'_i \sqsupseteq w_i$,
- for all $i \in \{1, \ldots, n-1\} w_{i+1} \in \mathcal{F}(w'_i)$,
- $w' \sqsupseteq w'_n$.

We then reason by induction on n. If $n = 0$ it is straightforward since we can take T_i the empty trace. Otherwise, suppose that $n > 0$ and let $(h'_1, h'_2, \mathcal{D}', \Phi'') \in \mathbf{P}_\Phi(w')$, one must find $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$ and build two traces T_1, T_2 such that $\langle \mathcal{S}_i, \gamma_i, \Phi'_i, h_i, \mathcal{D}_i \rangle \stackrel{T_i}{\Longrightarrow} \langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}'_i \rangle$ with $(\langle \mathcal{S}'_1, \gamma'_1 \rangle, \langle \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{O}}_\mathcal{A}(\Phi'', (w' :: L'))$.

From the fact that $w_1 \in \mathcal{F}(w)$ and $w'_1 \sqsupseteq w_1$, the fact that $\mathcal{A} = \mathbf{SE}^L_\Phi(\langle \mathcal{S}_1, \gamma_1 \rangle, \langle \mathcal{S}_2, \gamma_2 \rangle)$ gives us the existence of:

- $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_\Phi(w)$,
- $(h^1_1, h^1_2, \mathcal{D}^1, \Phi^1) \in \mathbf{P}_{\Phi'}(w_1)$,
- $(h^2_1, h^2_2, \mathcal{D}^2, \Phi^2) \in \mathbf{P}_{\Phi^1}(w_2)$,
- $a^1_1 \sim^{\mathcal{D}^1}_{\Phi^1} a^1_2$,
- $a^2_1 \sim^{\mathcal{D}^2}_{\Phi^2} a^2_2$,
- $(\langle \mathcal{S}^1_1, \gamma_1 \rangle, \langle \mathcal{S}^1_2, \gamma_2 \rangle) \in \overline{\mathcal{O}}_\mathcal{A}(\Phi^1, L^1)$,
- $(\langle \mathcal{S}^2_1, \gamma^2_1 \rangle, \langle \mathcal{S}^2_2, \gamma^2_2 \rangle) \in \overline{\mathcal{O}}_\mathcal{A}(\Phi^2, L^2)$

such that, writing C^1_i for the configuration $\langle \mathcal{S}^1_i, \gamma_i, \Phi^1_i, h^1_i, \mathcal{D}^1_i \rangle$ and C^1_i for the configuration $\langle \mathcal{S}^2_i, \gamma^2_i, \Phi^2_i, h^2_i, \mathcal{D}^2_i \rangle$, we get that $C_i \stackrel{a^1_i}{\Longrightarrow} C^1_i \stackrel{a^2_i}{\Longrightarrow} C^2_i$.

From $(\langle \mathcal{S}_1^2, \gamma_1^2 \rangle, \langle \mathcal{S}_2^2, \gamma_2^2 \rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi^2, L^2)$, we get that $(\langle \mathcal{S}_1^2, \gamma_1^2 \rangle, \langle \mathcal{S}_2^2, \gamma_2^2 \rangle) \in \mathbf{Faitful}_{\Phi''}(w_1')$. The induction hypothesis thus gives us the existence of two traces T_1', T_2' and two partial configurations $(\langle \mathcal{S}_1', \gamma_1' \rangle, \langle \mathcal{S}_2', \gamma_2' \rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', (w' :: L'))$ such that, writing C_i' for the configuration $\langle \mathcal{S}_i', \gamma_i', \Phi_i'', h_i', \mathcal{D}_i' \rangle$, we have that both $C_i^2 \overset{T_i'}{\Longrightarrow} C_i'$. Thus, we have that both $C_i \overset{a_1 \cdot a_2 \cdot T_i'}{\Longrightarrow} C_i'$.

Finally, we prove that $\mathbf{Faitful}_{\mathbf{pub},\Phi}(w)\langle \mathcal{S}_1, \gamma_1 \rangle \langle \mathcal{S}_1, \gamma_1 \rangle$ in the same way. □

E.2 Proof of Completeness

Lemma 22. *Let u_1, u_2 two values, $(v_1, \phi_1, \gamma_1) \in \mathbf{AVal}_{u_1}(\tau)$, $(v_2, \phi_2, \gamma_2) \in \mathbf{AVal}_{u_2}(\tau)$ such that there exists two disjoint span on functional names Φ_P, Φ_O and a world w such that $\Phi_{P,i} = \phi_i$, $v_1 \sim_{\Phi_P}^{w.\mathcal{D}} v_2$ and $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$. Then $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} w$.*

Proof: By induction on τ:

- If τ is ground, it is straightforward since both $u_i = v_i$.
- If $\tau = \sigma \to \sigma'$, then both $v_i = f_i$ with $f_i \notin \mathrm{dom}(\Phi_{O,i})$, $\phi_i = [f_i \mapsto \tau]$ and $\gamma_i = [f_i \mapsto u_i]$. So from $v_1 \sim_{\Phi}^{w.\mathcal{D}} v_2$, we get that $\Phi_P = (f_1, f_2, \tau)$, and from $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$ we get that $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} w$.
- If $\tau = \sigma_1 \times \sigma_2$ then both u_i are equal to some pairs $\langle u_i^1, u_i^2 \rangle$. Then for all $i \in \{1, 2\}$ we have $v_i = \langle v_i^1, v_i^2 \rangle$ with $(v_i^1, \phi_i^1, \gamma_i^1) \in \mathbf{AVal}_{u_i^1}(\sigma_1)$ and $(v_i^1, \phi_i^2, \gamma_i^2) \in \mathbf{AVal}_{u_i^2}(\sigma_1)$. From $v_1 \sim_{\Phi}^{w.\mathcal{D}} v_2$, we get that $\Phi_P = \Phi_P^1 \cup \Phi_P^2$ with $\Phi_{P,i}^j = \phi_i^j$ for all $i, j \in \{1, 2\}$. We can conclude easily using the induction hypothesis.

□

Lemma 23. *Let h_1, h_2 two heaps, $\mathcal{D}$ a span on part of their domain, $(h_1', \gamma_1, \phi_1) \in \mathbf{AHeap}_{\mathcal{D}_1}(h_{1|\mathcal{D}_1})$ and $(h_2', \gamma_2, \phi_2) \in \mathbf{AHeap}_{\mathcal{D}_2}(h_{2|\mathcal{D}_2})$ such that there exists two disjoint span on functional names Φ_P, Φ_O and a world w such that $\Phi_{P,i} = \phi_i$, $(h_1[h_1'], h_2[h_2'], \mathcal{D}, \Phi_P) \in \mathbf{P}_{\Phi_P}(w)$ and $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$. Then $(h_1, h_2, \mathcal{D}) \in \mathbf{Q}_{\Phi_O}(w)$.*

Proof: By induction on the size of $\mathcal{D}$. If it is empty it is straightforward. Otherwise, let suppose that $\mathcal{D} = \{(l_1, l_2, \tau)\} \cup \mathcal{D}'$. Then, we have both:

- $h_i = \widetilde{h_i} \cdot [l_i \mapsto u_i]$,
- $h_i' = h_i'' \cdot [l_i \mapsto v_i]$,
- $\gamma_i = \gamma_i' \cdot \gamma_i''$ and
- $\phi_i = \phi_i' \cdot \phi_i''$ with
- $(h_1'', \gamma_1'', \phi_1'') \in \mathbf{AHeap}_{\mathcal{D}_1}(\widetilde{h_{i|\mathcal{D}_1}})$,
- $(v_i, \phi_1', \gamma_i') \in \mathbf{AVal}_{u_i}(\tau)$.

From $(h_1[h_1'], h_2[h_2'], \mathcal{D}, \Phi_P) \in \mathbf{P}_{\Phi_P}(w)$, we get that there exists two spans Φ_P', Φ_P'' such that $\Phi_P = \Phi_P' \cdot \Phi_P''$ and for all $i \in \{1, 2\}, \Phi_{P,i}' = \phi_i'$ and $\Phi_{P,i}'' = \phi_i''$. From Lemma 22, we get that $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} w$, while the induction gives us that $(\widetilde{h_1}, \widetilde{h_2}, \mathcal{D}') \in \mathbf{Q}_{\Phi_O}(w)$, so we can conclude easily. □

Lemma 24. *Let u a value, $(v, \gamma, \phi) \in \mathbf{AVal}_v(\tau)$, w a world and ϕ_O a typing function such that $\mathrm{dom}(\phi) \cap \mathrm{dim}\, \phi_O = \varnothing$. If $\gamma \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_\phi \phi_O$, then $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_{\phi_O} w$.*

Proof: By a straightforward induction on τ. □

Lemma 25. *Let h a heap, $D \subseteq \mathrm{dom}(h)$, $(h', \gamma, \phi) \in \mathbf{AHeap}_D(h_{|D})$, w a world and ϕ_O a typing function such that $\mathrm{dom}(\phi) \cap \mathrm{dim}\, \phi_O = \varnothing$. If $(h[h'], D, \phi) \in \mathbf{P}^i_{\phi_O}(w)$ and $\gamma \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_\phi \phi_O$, then $(h, D) \in \mathbf{Q}^i_{\phi_O}(w)$.*

Proof: By a straightforward induction on the size of D, using Lemma 24 to conclude. □

Lemma 26. *Let $\langle (K[\bullet_\sigma], \tau) :: \mathcal{S}, \gamma \rangle \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi, L)$, then for all $w' \sqsupseteq^{\mathcal{F}*} w$ and $(h', D', \phi'') \in \mathbf{P}_\phi(w')$, there exists $(h, D, \phi') \in \mathbf{P}_\phi(w)$, a trace T and a reduced configuration $(\langle \mathcal{S}', \gamma' \rangle) \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi'', (w' :: L'))$ such that $\langle \mathcal{S}, \gamma, \phi, h, \mathcal{D} \rangle \overset{T}{\Rightarrow} \langle \mathcal{S}', \gamma', \Phi'', h', \mathcal{D}' \rangle$.*

Lemma 27. *Let $\langle (K[\bullet_\sigma], \tau) :: \mathcal{S}, \gamma \rangle \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi, L)$, then for all $w' \sqsupseteq^{\mathcal{F}*}_{\mathbf{pub}} w$ and $(h', D', \phi'') \in \mathbf{P}_\phi(w')$, there exists $(h, D, \phi') \in \mathbf{P}_\phi(w)$, a trace T and a reduced configuration $(\langle \mathcal{S}', \gamma' \rangle) \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi'', (w' :: L'))$ such that $\langle \mathcal{S}, \gamma, \Phi', h, \mathcal{D} \rangle \overset{T}{\Rightarrow} \langle \mathcal{S}, \gamma', \Phi'', h', \mathcal{D}' \rangle$.*

Lemma 28. *Let K a ground-closed evaluation context, w and w_0 two worlds, $\phi = \phi_P \cdot \phi_O$ a typing function such that $(w_j.\mathcal{D})_i; \phi_O \vdash K^j : \tau_j \rightsquigarrow \sigma_j$ and γ a functional environments with $\mathrm{dom}(\gamma) = \mathrm{dom}(\phi_p)$. Taking an Opponent evaluation stack $\mathcal{S}$ and a list of worlds L, of $(\langle (K[\bullet_\sigma], \tau) :: \mathcal{S}, \gamma \rangle) \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi, L)$, then $K \in \mathcal{K}^i_{\mathcal{A}} [\![\sigma, \tau]\!]_{\phi_O} (w, w_0)$ and $\gamma \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_P} \phi_O$.*

Lemma 29. *Let M a ground closed term, w and w_0 two worlds, $\phi = \phi_P \cdot \phi_O$ a typing function such that $(w\mathcal{D})_i; \phi_O \vdash M : \tau$, and γ a functional environments with $\mathrm{dom}(\gamma) = \mathrm{dom}(\phi_p)$. Taking an Opponent evaluation stack $\mathcal{S}$ and a list of worlds L, for all $(h, D, \phi') \in \mathbf{P}^i_{\phi_O}(w)$, if $\langle (M, \tau) :: \mathcal{S}, \gamma, \phi', h, D \rangle \in \overline{\mathcal{P}}^{\natural\, i}_{\mathcal{A}}(\phi, (w :: w_0 :: L))$, then $(M, h, D) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} (w, w_0)$.*

Proof: The two lemmas are proven by a mutual coinduction.

Completeness for Diverging Terms Taking $(h, D, \phi') \in \mathbf{P}^i_{\phi_O}(w)$, we suppose that $C \in \overline{\mathcal{P}}^{\natural\, i}_{\mathcal{A}}(\phi, (w :: w_0 :: L))$ where $C = \langle (M, \tau) :: \mathcal{S}, \gamma, \phi', h, D \rangle$.

If $C \uparrow$ then $(M, h) \uparrow$, so that $(M, h, D) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} (w, w_0)$. Otherwise, there exists an action a and two configurations C', C'' such that $C \xrightarrow{C}{}' \xrightarrow{a} C''$. Let us write C' as $\langle (M', \tau) :: \mathcal{S}, \gamma, \phi', h', D \rangle$, and C'' as $\langle \mathcal{S}', \gamma', \phi'', h'[h''], D' \rangle$ such that $(M, h) \mapsto^* (M', h')$ with (M', h') irreducible. We then get the existence of a world $w' \sqsupseteq w$ such that $(h'[h''], D', \phi') \in \mathbf{P}^i_{\phi'}(w')$ and $\langle \mathcal{S}', \gamma' \rangle \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi, w' :: L')$. We then reason by case analysis on a:

- if a is an answer $(\langle \bar{v} \rangle, h''$, so that M' is equal to a value u, then
  - $w' \sqsupseteq_{\mathbf{pub}} w_0$

- $\mathcal{S}' = \mathcal{S}$,
- $\phi'' = \phi' \cdot \phi_v \cdot \phi_h$
- $\gamma' = \gamma \cdot \gamma_v \cdot \gamma_h$,
- $(v, \gamma_v, \phi_v) \in \mathbf{AVal}_u(\tau)$,
- $(h'', \gamma_h, \phi_h) \in \mathbf{AHeap}_{D'}(h'')$.

From $\langle \mathcal{S}', \gamma' \rangle \in \overline{\mathcal{O}}_{\mathcal{A}}(i, \phi'')(w', w_0)$, the coinduction hypothesis gives us that $\gamma' \in \mathcal{G}_{\mathcal{A}} [\![\phi_P \cdot \phi_v \cdot \phi_h]\!]_{\phi'} w'$ so from Lemma 24 and 25, we get that so that $(h'', D') \in \mathbf{Q}^i_{\phi'}(w')$ and $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} w'$. Thus $(M, h'', D') \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi_O} (w, w_0)$.

– Otherwise a is a question $a = (\bar{f} \langle v \rangle, h'')$, so M' is equal to a callback $K_{[}f\ u]$. Then, there exists a functional type $\sigma \to \sigma'$ such that:
  - $(f, \sigma \to \sigma') \in \phi_P$
  - $\phi'' = \phi' \cdot \phi_v \cdot \phi_h$,
  - $\mathcal{S}' = (u\ v, \sigma') :: \mathcal{S}$ with $\gamma(f) = u$,
  - $\gamma' = \gamma \cdot \gamma_v \cdot \gamma_{hi}$,
  - $(v, \gamma_v, \phi_v) \in \mathbf{AVal}_u(\tau)$,
  - $(h'', \gamma_{h,i}, \phi_{h,i}) \in \mathbf{AHeap}_{D'}(h'')$.

  From $\langle \mathcal{S}', \gamma' \rangle \in \overline{\mathcal{O}}_{\mathcal{A}}(i, \phi'')(w', w_0)$, the coinduction hypothesis gives us that $K \in \mathcal{K}^i_{\mathcal{A}} [\![\tau, \sigma]\!]_{\phi_O} (w', w_0)$ and $\gamma' \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_P \cdot \phi_v \cdot \phi_h} \phi'$ so from Lemma 24 and 25, we get that $(h'', D') \in \mathbf{Q}^i_{\phi'}(w')$ and $u \in \mathcal{V}^i_{\mathcal{A}} [\![\tau]\!]_{\phi'} w'$. Thus $M \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi_O} (w, w_0)$.

Completeness for Diverging Contexts We first prove that for all $j \in \{1, \ldots, n\}, K^j \in \mathcal{K}^i_{\mathcal{A}} [\![\tau_j, \sigma_j]\!]_{\phi_O} (w_j, w_{j_1})$. If $n = 0$, this is straightforward. Otherwise we consider $w' \sqsupseteq^{\mathcal{F}*}_{\mathbf{pub}} w_n$ and $(v, \phi_v) \in [\![\tau]\!]$ such that ϕ_v is disjoint of $\phi_0 \cdot \phi_P$. Let $(h', D', \phi'') \in \mathbf{P}^i_{\phi_0 \cdot \phi_P \cdot \phi_v}(w')$, from Lemma 27, we get the existence of $(h, D, \phi') \in \mathbf{P}_{\phi}(w)$, a trace T and a functional environment γ such that $(\langle \mathcal{S}, \gamma' \rangle) \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi'', (w' :: L'))$ and $\langle \mathcal{S}, \gamma, \phi', h, \mathcal{D} \rangle \overset{T}{\Rightarrow} \langle \mathcal{S}, \gamma', \phi'', h', D' \rangle$. Then, one has

$$\langle \mathcal{S}, \gamma', \phi'', h', D' \rangle \xrightarrow{\langle u_i \rangle, h''} \langle (K^n[v], \sigma) :: \mathcal{S}', \gamma', \phi''', h'[h''], D'' \rangle$$

with $\mathcal{S} = (K^n[\bullet_{\tau_n}], \sigma_n) :: \mathcal{S}'$. From $(\langle \mathcal{S}', \gamma' \rangle) \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi'', (w' :: L'))$, one get that $\langle (K^n[v], \sigma) :: \mathcal{S}', \gamma' \rangle \in \overline{\mathcal{P}}^{\natural\, i}_{\mathcal{A}}(\phi''', (w' :: L))$, so the coinduction hypothesis then gives us that $(K^n[v], h', D') \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi''} (w', w_0)$.

Then, we prove that $\gamma \in \mathcal{G}^i_{\mathcal{A}} [\![i]\!]_{\phi_P} \phi_O$. Let $(f, \sigma \to \sigma') \in \phi_P$ with $\gamma(f) = u$, one must prove that $u \in \mathcal{V}_{\mathcal{A}} [\![\sigma \to \sigma']\!]_{\phi_O} w$. To do so, let us consider $w' \sqsupseteq^* w$, ϕ' a typing function disjoint from ϕ_O, v an abstract value such that $(v, \phi_v) \in [\![\tau]\!]$ and $(h', D', \phi'') \in \mathbf{P}^i_{\phi_P \cdot \phi_O \cdot \phi_v}(w')$. One must then prove that $(u\ v, h', D') \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi''} (w', w_0)$.

From Lemma 26, we get that there exists $(h, D, \phi') \in \mathbf{P}_{\phi}(w)$, a trace T and a reduced configuration $(\langle \mathcal{S}', \gamma' \rangle) \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\phi'', (w' :: L'))$ such that $\langle \mathcal{S}, \gamma, \phi', h, \mathcal{D} \rangle \overset{T}{\Rightarrow} \langle \mathcal{S}', \gamma', \phi'', h', D' \rangle$. Then, one has $\langle \mathcal{S}', \gamma', \phi'', h', D' \rangle \xrightarrow{f_i \langle v_i \rangle, h''} \langle (u\ v, \sigma') :: \mathcal{S}', \gamma', \phi''', h'[h''], D'' \rangle$, so from $\langle \mathcal{S}', \gamma' \rangle \in \overline{\mathcal{O}}^{\natural\, i}_{\mathcal{A}}(\Phi'', (w' :: L'))$ one get that $\langle (u\ v_1, \sigma') :: \mathcal{S}', \gamma' \rangle \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi'', (w' :: L'))$, so the coinduction hypothesis gives us that $(u\ v, h'[h'']) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\phi''} (w', w_0)$. $\square$

Theorem 13. *Let M_1, M_2 two ground-closed terms, w and w_0 two worlds, $\Phi = \Phi_P \cdot \Phi_O$ a span on functional names such that $(w.\mathcal{D})_i; \Phi_{O,i} \vdash M_i : \tau$, and γ_1, γ_2 two functional environments with $\mathrm{dom}(\gamma_i) = \Phi_{P,i}$. If $(\langle M_1, \gamma_1\rangle, \langle M_2, \gamma_2\rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi, (w, w_0))$, then $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} (w, w_0)$.*

Theorem 14. *Let (K_1, K_2) a pair of ground-closed evaluation contexts, w and w_0 two worlds, $\Phi = \Phi_P \cdot \Phi_O$ a span on functional names such that $(w.\mathcal{D})_i; \Phi_{O,i} \vdash K_i : \sigma \rightsquigarrow \tau$, and γ_1, γ_2 two functional environments with $\mathrm{dom}(\gamma_i) = \Phi_{P,i}$. Taking two evaluation stacks $\mathcal{S}_1, \mathcal{S}_2$ and a list of worlds L, if $(\langle (K_1[\bullet_\sigma], \tau) :: \mathcal{S}_1, \gamma_1\rangle, \langle (K_2[\bullet_\sigma], \tau) :: \mathcal{S}_2, \gamma_2\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi, (w :: w_0 :: L))$, then $(K_1, K_2) \in \mathcal{K}_{\mathcal{A}} [\![\tau, \sigma]\!]_{\Phi_O} (w, w_0)$ and $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w_n$.*

Proof:

Completeness for Terms Let us first take $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi_O \cdot \Phi_P}(w)$, so that we write C_i for the configuration $\langle (M_i, \tau_i) :: \mathcal{S}_i, \gamma_i, \Phi'_i, h_i, \mathcal{D}_i\rangle$.

Suppose that both $C_i \in \overline{\mathcal{P}}^{\natural\, i}_{\mathcal{A}}(\Phi_i, w)$. If $C_i \uparrow$, then $(M_i, h_i) \uparrow$ so that $(M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\Phi'_i} (w, w_0)$. Otherwise, there exists $w' \sqsupseteq w$ and $(h', D', \phi'') \in \mathbf{P}^i_{\phi'}(w')$ such that $\langle (M'_i, \tau) :: \mathcal{S}_i, \gamma_i, \phi'', h'_i, D'\rangle \in \overline{\mathcal{P}}^{\natural\, i}_{\mathcal{A}}(\phi', (w' :: L''))$. Thus, Lemma 29 gives us that $(M_i, h_i, \mathcal{D}_i) \in \mathcal{E}^i_{\mathcal{A}} [\![\tau]\!]_{\Phi'_i} (w, w_0)$.

Otherwise, there exists two spans $\mathcal{D}' \sqsupseteq \mathcal{D}$ and $\Phi'' \sqsupseteq \Phi'$, two Player actions a_1, a_2 such that $a_1 \sim^{\mathcal{D}'}_{\Phi''} a_2$, such that $C_i \to C'_i \xrightarrow{\overline{a_i}} C''_i$, where

- $C'_i = \langle (M_i, \tau_i) :: \mathcal{S}_i, \gamma_i, \Phi'_i, h_i, \mathcal{D}_i\rangle$ with $(M_i, h_i) \mapsto^* (M'_i, h'_i)$ and M'_i irreducible,
- $C''_i = \langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i[h''_i], \mathcal{D}'_i\rangle$.

Then there exists $w' \sqsupseteq w$ such that $(h''_1, h''_2, \mathcal{D}', \Phi'') \in \mathbf{P}^{\Phi''}_{w'}()$ and $(\langle \mathcal{S}'_2, \gamma'_2\rangle, \langle \mathcal{S}'_2, \gamma'_2\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', (w', w_0))$. We reason by case analysis on the a_i:

- If both a_i are equal to answers $(\langle \bar{v_i}\rangle, h''_i)$, so that both M'_i are equal to values u_i, then:
  - $w' \sqsupseteq_{\mathbf{pub}} w_0$,
  - both $\mathcal{S}'_i = \mathcal{S}_i$,
  - $\Phi'' = \Phi' \cdot \Phi_v \cdot \Phi_h$
  - both $\gamma'_i = \gamma_i \cdot \gamma_{v,i} \cdot \gamma_{h,i}$,
  - both $(v_i, \gamma_{v,i}, \Phi_{v,i}) \in \mathbf{AVal}_{u_i}(\tau)$,
  - both $(h''_i, \gamma_{h,i}, \Phi_{h,i}) \in \mathbf{AHeap}_{D'_i}(h''_i)$.

  From $(\langle \mathcal{S}'_2, \gamma'_2\rangle, \langle \mathcal{S}'_2, \gamma'_2\rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', (w', w_0))$, the coinduction hypothesis gives us that $(\gamma'_1, \gamma'_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P \cdot \Phi_v \cdot \Phi_h]\!]_{\Phi'} w'$ so from Lemma 22 and 23, we get that so that $(h''_1, h''_2, \mathcal{D}') \in \mathbf{Q}_{\Phi'}(w')$ and $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi'} w'$. Thus $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} (w, w_0)$.
- Otherwise both a_i are questions $(\bar{f_i}\, \langle v_i\rangle, h''_i)$ so that M'_i are equal to callbacks $K_i[f_i u_i]$, and from $a_1 \sim^{\mathcal{D}'}_{\Phi'_O \cdot \Phi_D} a_2$ there exists a functional type $\sigma \to \sigma'$ such that:
  - $(f_1, f_2, \sigma \to \sigma') \in \Phi_P$,
  - $\Phi'' = \Phi' \cdot \Phi_v \cdot \Phi_h$,
  - both $\mathcal{S}'_i = (u_i v_i, \sigma') :: \mathcal{S}_i$ with $\gamma_i(f_i) = u_i$,
  - both $\gamma'_i = \gamma_i \cdot \gamma_{v,i} \cdot \gamma_{h,i}$

- both $(v_i, \gamma_{v,i}, \Phi_{v,i}) \in \mathbf{AVal}_{u_i}(\tau)$,
- both $(h''_i, \gamma_{h,i}, \phi_{h,i}) \in \mathbf{AHeap}_{D'_i}(h''_i)$.

From $(\langle \mathcal{S}'_2, \gamma'_2 \rangle, \langle \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{O}}_{\mathcal{A}}(\Phi'', (w', w_0))$, the coinduction hypothesis gives us that $(K_1, K_2) \in \mathcal{K}_{\mathcal{A}} [\![\tau, \sigma]\!]_{\Phi_O} (w', w_0)$ and $(\gamma'_1, \gamma'_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P \cdot \Phi_v \cdot \Phi_h]\!]_{\Phi'} w'$ so from Lemma 22 and 23, we get that $(h''_1, h''_2, \mathcal{D}') \in \mathbf{Q}_{\Phi'}(w')$ and $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\tau]\!]_{\Phi'} w'$. Thus $(M_1, M_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O} (w, w_0)$.

Completeness for Contexts We first prove that for all $j \in \{1, \ldots, n\}, (K_1^j, K_2^j) \in \mathcal{K}_{\mathcal{A}} [\![\tau_j, \sigma_j]\!]_{\Phi_O} (w_j, w_{j_1})$. If $n = 0$, this is straightforward, otherwise we consider $w' \sqsupseteq^{\mathcal{F}*}_{\mathbf{pub}} w_n$, Φ' a span disjoint from Φ_O, and v_1, v_2 two abstract values such that $(v_i, \Phi'_i) \in [\![\tau]\!]$ and $v_1 \sim^{w'.\mathcal{D}'}_{\Phi'} v_2$.

From $\mathbf{Faitful}_{\mathbf{pub},\Phi}(w)\langle \mathcal{S}_1, \gamma_1 \rangle \langle \mathcal{S}_1, \gamma_1 \rangle$, we get that for all $(h'_1, h'_2, \mathcal{D}', \Phi'') \in \mathbf{P}_{\Phi}(w')$ there exists $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi}(w)$, two traces T_1, T_2 and two environments γ'_1, γ'_2 such that $\langle \mathcal{S}_1, \gamma'_1 \rangle, \langle \mathcal{S}_2, \gamma'_2 \rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi'', (w' :: L))$ and

$$\langle \mathcal{S}_i, \gamma_i, \Phi_i, h_i, \mathcal{D}_i \rangle \xrightarrow{T_i} \langle \mathcal{S}_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}'_i \rangle$$

Then, one has

$$\langle \mathcal{S}_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}'_i \rangle \xrightarrow{\langle u_i \rangle, h''_i} \langle (K^n[v_i], \sigma_i) :: \mathcal{S}'_i, \gamma'_i, \Phi'''_i, h'_i[h''_i], \mathcal{D}''_i \rangle$$

with $\mathcal{S}_i = (K_i^n[\bullet_{\tau_n}], \sigma_n) :: \mathcal{S}'_i$. From $\langle \mathcal{S}_1, \gamma'_1 \rangle, \langle \mathcal{S}_2, \gamma'_2 \rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi'', (w' :: L))$ and $v_1 \sim^{w'.\mathcal{D}'}_{\Phi'} v_2$, one get that

$$\langle (K_1^n[v_1], \sigma) :: \mathcal{S}'_1, \gamma'_1 \rangle, \langle (K_2^n[v_2], \sigma_n) :: \mathcal{S}'_2, \gamma'_2 \rangle \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi'', (w' :: L))$$

The coinduction hypothesis then gives us that $(K_1^n[v_1], K_2^n[v_2]) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O \cdot \Phi'} (w', w_0)$.

Then, we prove that $(\gamma_1, \gamma_2) \in \mathcal{G}_{\mathcal{A}} [\![\Phi_P]\!]_{\Phi_O} w$. Let $(f_1, f_2, \sigma \to \sigma') \in \Phi_P$ with $\gamma_i(f_i) = u_i$, one must prove that $(u_1, u_2) \in \mathcal{V}_{\mathcal{A}} [\![\sigma \to \sigma']\!]_{\Phi_O} w$. To do so, let us consider $w' \sqsupseteq^*_{\mathbf{pub}} w$, Φ' a span disjoint from Φ_O, and v_1, v_2 two abstract values such that $(v_i, \Phi'_i) \in [\![\tau]\!]$ and $v_1 \sim^{w'.\mathcal{D}'}_{\Phi'} v_2$ One must then prove that $(u_1 v_1, u_2 v_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O \cdot \Phi'} (w', w_0)$.

From $\mathbf{Faitful}_{\Phi}(w)\langle \mathcal{S}_1, \gamma_1 \rangle \langle \mathcal{S}_1, \gamma_1 \rangle$ one get that for all $(h'_1, h'_2, \mathcal{D}', \Phi'') \in \mathbf{P}_{\Phi}(w')$ there exists $(h_1, h_2, \mathcal{D}, \Phi') \in \mathbf{P}_{\Phi}(w)$ and two traces T_1, T_2 and two reduced configurations $(\langle \mathcal{S}'_1, \gamma'_1 \rangle, \langle \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi'', (w' :: L'))$ such that

$$\langle \mathcal{S}_i, \gamma_i, \Phi_i, h_i, \mathcal{D}_i \rangle \xrightarrow{T_i} \langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}'_i \rangle$$

Then, one has $\langle \mathcal{S}'_i, \gamma'_i, \Phi''_i, h'_i, \mathcal{D}'_i \rangle \xrightarrow{f_i \langle v_i \rangle, h''_i} \langle (u_i\ v_i, \sigma') :: \mathcal{S}'_i, \gamma'_i, \Phi'''_i, h'_i[h''_i], \mathcal{D}''_i \rangle$, so from $(\langle \mathcal{S}'_1, \gamma'_1 \rangle, \langle \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi'', (w' :: L'))$ one get that $(\langle (u_1\ v_1, \sigma') :: \mathcal{S}'_1, \gamma'_1 \rangle, \langle (u_2\ v_2, \sigma') :: \mathcal{S}'_2, \gamma'_2 \rangle) \in \overline{\mathcal{P}}_{\mathcal{A}}(\Phi''', (w'' :: L'))$, so the coinduction hypothesis gives us that $(u_1 v_1, u_2 v_2) \in \mathcal{E}_{\mathcal{A}} [\![\tau]\!]_{\Phi_O \cdot \Phi'} (w', w_0)$.

□

F Kripke Open Bisimulations at work

This section shows on well-known examples of the literature how to use direct-style reasoning, spans of names, LTSs and reasoning on divergence—which constitute the main concepts of KOBs.

In order to define transitions in a concise way, we use a representation with a pair of pre- and post-condition $\{P\} \Rrightarrow \{Q\}$ between two states. P and Q are predicates written in (Peano) arithmetic, together with heap predicates $l \mapsto_i u$, where the index i indicates whether we consider the left or the right heap, and u is either a closed value, or a logical variable. Such heap predicates are transformed to equality predicates using the *predicate transformer* $\mathbf{HT}_{h_1,h_2}(P)$ defined as $\mathbf{HT}_{h_1,h_2}(l \mapsto_i u) \stackrel{def}{=} h_i(l) = u$ and just propagating the definition on logical connectives. We also need to keep track of divergence transitions, indicated by the symbol $\lightning$ in the postcondition. This is done using the predicate $\mathbf{Div}_{b,b'}(P)$ defined as $b' = \mathbf{true}$ if $\lightning$ appears in P, and as $b = b'$ otherwise. Then, $\{P\} \Rrightarrow \{Q\}$ between two states s, s' corresponds to the transition

$$\left\{ \begin{array}{l|l} ((s, h_1, h_2, \mathcal{D}, b), & \exists \overrightarrow{z}. \mathbf{HT}_{h_1,h_2}(P) \wedge \\ \qquad (s', h_1', h_2', \mathcal{D}', b')) & \mathbf{HT}_{h_1',h_2'}(Q) \wedge \mathbf{Div}_{b,b'}(Q) \end{array} \right\}$$

where $\overrightarrow{z}$ ranges over logical variables of P, Q.

F.1 Disclosed Locations

We now present how to reason on equivalence of programs with an explicit span (see Section 2.3) on disclosed locations, contained in worlds. Let consider the following terms:

$$M_1 = \lambda \mathtt{f}.\, \mathtt{let\ x, y} = \mathrm{ref}\ \mathtt{0\ in\ f\ x;\ f\ y}$$
$$M_2 = \lambda \mathtt{f}.\, \mathtt{let\ x} \quad = \mathrm{ref}\ \mathtt{0\ in\ f\ x;\ f\ x}\,.$$

They are not contextually equivalent because the former discloses two different locations l_x, l_y during the two callbacks, while the later discloses only l_x. This is apparent in an attempt to prove they are related since we need to build a span $\mathcal{D} = (l_x, l_x)$ during the proof, and then we try to add (l_x, l_y) to it, which is not possible since it would not be a span anymore.

The two following terms:

$$M_1 = \lambda \mathtt{f}.\, \mathtt{let\ x} = \mathrm{ref}\ \mathtt{0\ in\ f\ x}$$
$$M_2 = \lambda \mathtt{f}.\, \mathtt{let\ x} = \mathrm{ref}\ \mathtt{0\ in\ f\ x;}\ x := 1$$

are also not equivalent, since l_x does not store the same value at the end of the execution, and this is observable by contexts since l_x has been disclosed. Trying to prove that these terms are related, we failed to conclude since we would have to show that there exists a world w s.t. (i) it contains the span $\mathcal{D} = (l_x, l_x)$, (ii) the two heaps $[l_x \mapsto 0]$ and $[l_x \mapsto 1]$ satisfies the constraint w, which is not possible since to satisfy the constraint w, one would need $0 = 1$ (whatever w is, as soon as its span contains $\mathcal{D}$).

F.2 Finite presentation of a infinite STS

We now consider the callback with lock example (taken from [1]) that compares two encoding of a counter object:

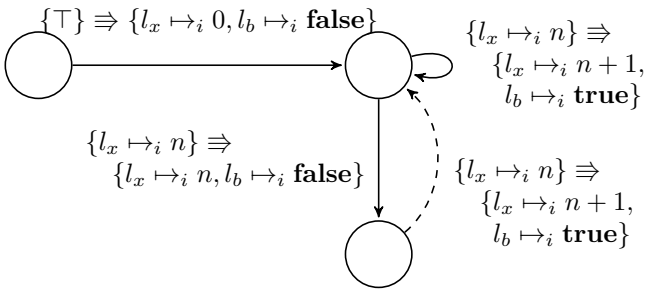


Fig. 9. State transition system for the callback with lock example.

$$M_1^{cbl} \stackrel{def}{=} C\,[\mathtt{f}();\mathtt{x} :=!\mathtt{x}+1]$$
$$M_2^{cbl} \stackrel{def}{=} C\,[\mathtt{let\ n} =!\mathtt{x}\ \mathtt{in}\ \mathtt{f}();\mathtt{x} := \mathtt{n}+1]$$

where $C \stackrel{def}{=} \mathtt{let\ b} = \mathrm{ref}\,\mathbf{true}\ \mathtt{in\ let\ x} = \mathrm{ref}\,0\ \mathtt{in}$
$(\lambda\mathtt{f.if}\ !\mathtt{b}\ \mathtt{then\ b} := \mathbf{false};\ \bullet\ ;\mathtt{b} := \mathbf{true}\ \mathtt{else}\ (), \lambda\_.!\mathtt{x})$

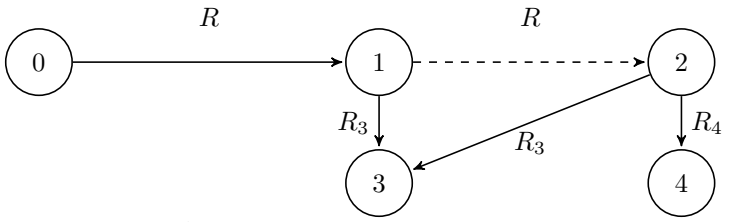
Using an STS approach, two states are needed for each natural number stores in the location x, to keep track of the increment of the counter [2]. To avoid using an infinite transition system, which can be problematic in order to automatize the reasoning, we rather introduce the WTS depicted in Figure 9, in order to use Hoare-style description of the heap—relating heaps before the transition to heaps after the transition. Such labels are written $\{P\} \Rrightarrow \{Q\}$, where P, Q are predicates on heaps. The increment of the counter is now modeled using the pre- and post-conditions $\{l_x \mapsto_i n\} \Rrightarrow \{l_x \mapsto_i n+1\}$ which allows to provide an WTS with only three states. Besides this technical improvement, the proof of [2] can be adapted directly in our framework, in the same way the well-bracketed state change example has been adapted in Section 4.3.

F.3 Dealing with Divergence

There is one loophole in the notion of synchronization of callbacks, when terms can diverge and thus do not return control to contexts. For example, the two following terms are contextually equivalent:

$$\lambda\mathtt{f}.\mathtt{f}();\bot_{\mathrm{Unit}} \simeq_{ctx} \lambda\mathtt{f}.\bot_{\mathrm{Unit}}$$

even if the second one do not perform any callback. This appears in game semantics via the fact that plays corresponding to interactions where Opponent (i.e., contexts) interrogates the λ-abstractions are never *complete*, that is Player (i.e., the term) do not answer to this first Opponent question. To prove such equivalences, one needs to tag specifically such paths in the control flow. This can be done using *inconsistent states*, as introduced in [2]. The idea is that with such states, we promise that the term is going to diverge at one point, before returning a value, thus we do not need to enforce the synchronization of callbacks.



$$R \stackrel{def}{=} \{\top\} \Rrightarrow \{l_x \mapsto_1 \mathbf{false}, l_y \mapsto_1 \mathbf{false}\}$$
$$R_3 \stackrel{def}{=} \{\top\} \Rrightarrow \{l_x \mapsto_1 \mathbf{true}, l_y \mapsto_1 \mathbf{false}\}$$
$$R_4 \stackrel{def}{=} \{\top\} \Rrightarrow \{l_x \mapsto_1 \mathbf{false}, l_y \mapsto_1 \mathbf{true}, \lightning\}$$

Fig. 10. State transition system for the deferred divergence example.

To exemplify this idea, we consider the following "deferred divergence" example, still taken from [2]:

$$\begin{aligned} M_1^{dd} = {} & \texttt{let x} = \mathrm{ref}\,\mathbf{false}\ \texttt{in let y} = \mathrm{ref}\,\mathbf{false}\ \texttt{in} \\ & \lambda\texttt{f.f}(\lambda\_.\texttt{if !x then} \perp_{\mathrm{Unit}} \texttt{else y} := \mathbf{true}); \\ & \texttt{if !y then} \perp_{\mathrm{Unit}} \texttt{else x} := \mathbf{true} \\ M_2^{dd} = {} & \lambda\texttt{f.f}(\lambda\_.\perp_{\mathrm{Unit}}) \end{aligned}$$

Figure 10 shows the WTS that allows us to prove that the two terms are equivalent by using the symbol $\lightning$ to indicate that the world becomes inconsistent while taking the transition to the right above state. It is similar to the one used in [2].

We now prove that $(M_1^{dd}, M_2^{dd}) \in \mathcal{E}_{\mathcal{A}}\,[\![\tau]\!]_{\varepsilon}\,(w_0, w_0)$ where $\tau = ((\mathrm{Unit} \to \mathrm{Unit}) \to \mathrm{Unit}) \to \mathrm{Unit}$ and $w_0 = (0, \varepsilon, \varepsilon, \varnothing, \mathbf{false})$. For sake of simplicity, we do not take into account any span on locations $\mathcal{D}$ here, since there is no disclosure of locations so it would be always empty. Taking $(h_1, h_2, e) \in \mathbf{P}_{\varepsilon}(w_0)$, we have $h_1 = h_2 = \varepsilon$, and $e = \varepsilon$, so $(M_1^{dd}, \varepsilon) \to^* (v_1, h_1')$ where

- $v_1 \stackrel{def}{=} \lambda\texttt{f.f}(\lambda\_.\texttt{if !l}_\texttt{x}\ \texttt{then} \perp_{\mathrm{Unit}} \texttt{else l}_\texttt{y} := \mathbf{true}); \texttt{if !l}_\texttt{y}\ \texttt{then} \perp_{\mathrm{Unit}} \texttt{else l}_\texttt{x} := \mathbf{true}$
- $h_1' \stackrel{def}{=} [l_x \mapsto \mathbf{false}, l_y \mapsto \mathbf{false}]$.

So defining w_1 as $(1, h_1', \varepsilon, \varnothing, \mathbf{false})$, we easily check that $w_1 \sqsupseteq_{\mathbf{pub}} w_0$, $\mathbf{cons}(w_1)$ and $(h_1', \varepsilon) \in \mathbf{Q}_{\varepsilon}(w_1)$. So we now have to prove that $(v_1, M_2^{dd}) \in \mathcal{V}_{\mathcal{A}}\,[\![\tau]\!]_{\varepsilon}\,w_1$.

Considering any world $w' \sqsupseteq^* w_1$ and Φ defined as $(f_1, f_2, (\mathrm{Unit} \to \mathrm{Unit}) \to \mathrm{Unit})$, this leads to prove that for all $(h_1, \_) \in \mathbf{P}_{\_}(w')$, there exists a world $w'' \sqsupseteq w'$ such that $(h_1, \_,) \in \mathbf{P}_{\_}(w'')$ $(M_1', M_2') \in \mathcal{E}_{\mathcal{A}}\,[\![\mathrm{Unit}]\!]_{\Phi}\,(w'', w'')$ where

- $M_1' \stackrel{def}{=} \texttt{f}_1(\lambda\_.\texttt{if !l}_\texttt{x}\ \texttt{then} \perp_{\mathrm{Unit}} \texttt{else l}_\texttt{y} := \mathbf{true}); \texttt{if !l}_\texttt{y}\ \texttt{then} \perp_{\mathrm{Unit}} \texttt{else l}_\texttt{x} := \mathbf{true}$
- $M_2' \stackrel{def}{=} \texttt{f}_2(\lambda\_.\perp_{\mathrm{Unit}})$.

We reason by case analysis on w', which can either be equal to w_1 or to:

- $w_2 \stackrel{def}{=} (2, [l_x \mapsto_1 \mathbf{false}, l_y \mapsto_1 \mathbf{false}], \varepsilon, \varnothing, \mathbf{false})$,
- $w_3 \stackrel{def}{=} (3, [l_x \mapsto_1 \mathbf{true}, l_y \mapsto_1 \mathbf{false}], \varepsilon, \varnothing, \mathbf{false})$,

– $w_4^{\lightning} \stackrel{def}{=} (4, [l_x \mapsto_1 \mathbf{false}, l_y \mapsto_1 \mathbf{true}], \varepsilon, \varnothing, \mathbf{true})$.

If w' is equal to w_1, we take w'' equal to w_2. Otherwise, we take w'' equal to w'. Then, we have to prove that:

– $(\lambda\_.\texttt{if !l}_\texttt{x}\texttt{ then } \bot_{\mathrm{Unit}} \texttt{ else l}_\texttt{y} := \mathbf{true}, \lambda\_.\bot_{\mathrm{Unit}}) \in \mathcal{V}_{\mathcal{A}} [\![\mathrm{Unit} \to \mathrm{Unit}]\!]_{\Phi} w''$

– $(\bullet; \texttt{if !l}_\texttt{y}\texttt{ then } \bot_{\mathrm{Unit}} \texttt{ else l}_\texttt{x} := \mathbf{true}, \bullet) \in \mathcal{K}_{\mathcal{A}} [\![\mathrm{Unit}, \mathrm{Unit}]\!]_{\Phi} (w'', w'')$

To prove the first point, we have to consider any world $w''' \sqsupseteq^* w''$ and prove that

$$(\underbrace{\texttt{if !l}_\texttt{x}\texttt{ then } \bot_{\mathrm{Unit}} \texttt{ else l}_\texttt{y} := \mathbf{true}}_{M_1''}, \bot_{\mathrm{Unit}})$$

is in $\mathcal{E}_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi} (w''', w''')$. Let us first consider the worlds w''' where $l_x \mapsto_1 \mathbf{false}$, namely w_2 and $w_4^{\lightning}$, then taking $(h_1, \_) \in \mathbf{P}_{\overline{7}}(w'')$ we have $(M_1'', h_1) \mapsto^* (\mathrm{Unit}, [l_x \mapsto \mathbf{false}, l_y \mapsto \mathbf{true}])$ and indeed we have $w_4^{\lightning} \sqsupseteq_{\mathbf{pub}} w'''$ such that $[l_x \mapsto \mathbf{false}, l_y \mapsto \mathbf{true}] \in \mathbf{Q}_{\varepsilon}(w_4^{\lightning})$. Moreover, $w_4^{\lightning}$ is inconsistent, so from $() \in \mathcal{V}^1_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi} w_4^{\lightning}$, we get that $(M_1'', h_1') \in \mathcal{E}^1_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi_1} (w''', w''')$ and $(\bot_{\mathrm{Unit}}, \varepsilon) \in \mathcal{E}^2_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi_2} (w''', w''')$. Considering now the world w_3 where $l_x \mapsto_1 \mathbf{true}$, it is straightforward to prove the first point since both terms are diverging.

Finally, we prove the second point. To do so, we must consider any world $w''' \sqsupseteq^*_{\mathbf{pub}} w''$ and prove that

$$(\texttt{if !l}_\texttt{y}\texttt{ then } \bot_{\mathrm{Unit}} \texttt{ else l}_\texttt{x} := \mathbf{true}, ()) \in \mathcal{E}_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi} (w''', w'')$$

We first consider the worlds w''' where $l_y \mapsto_1 \mathbf{true}$, namely $w_4^{\lightning}$. Since it is inconsistent, we have indeed that $() \in \mathcal{E}^2_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi} w_4^{\lightning}$. And considering worlds w''' where $l_y \mapsto_1 \mathbf{false}$, namely w_2, w_3, then we have indeed the existence of a world, namely w_3, such that $w_3 \sqsupseteq_{\mathbf{pub}} w'''$, $[l_x \mapsto \mathbf{true}, l_y \mapsto \mathbf{false}] \in \mathbf{Q}_{\varepsilon}(w_3)$ and $((), ()) \in \mathcal{V}_{\mathcal{A}} [\![\mathrm{Unit}]\!]_{\Phi} w_3$.