PEDECIBA Informática
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay

Reporte Técnico RT 11-02

An institution for UML 2.0 state machines

Daniel Calegari **Nora Szasz**

2011

An institution for UML 2.0 state machines
Calegari, Daniel; Szasz, Nora
ISSN 0797-6410
Reporte Técnico RT 11-02
PEDECIBA
Instituto de Computación – Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay, abril de 2011

Una Institución para Máquinas de Estado de UML 2.0

Daniel Calegari[1] and Nora Szasz[2]

[1] Facultad de Ingeniería, Universidad de la República
11300 Montevideo, Uruguay
dcalegar@fing.edu.uy

[2] Facultad de Ingeniería, Universidad ORT Uruguay
11100 Montevideo, Uruguay
szasz@ort.edu.uy

Abstract. La teoría de instituciones provee un marco robusto y elegante para la programación de alto nivel y en particular para la composicionalidad. Puede ser utilizado para definir un ambiente heterogéneo para la especificación semántica de UML, el cual consiste de una familia de formalismos que capturan varios sublenguajes UML, y morfismos que representan las relaciones semánticas esperadas entre ellos. En este artículo se presenta una institución para el lenguaje de Máquinas de Estado de UML 2.0, ideada para colaborar con la definición del ambiente heterogéneo. La semántica detrás de la institución está basada en trabajos previos. Dicha semántica considera el procesamiento de un evento de entrada en el contexto de una transición de un paso. Adicionalmente extendemos la semántica para manejar secuencias de eventos, y además para considerar corridas a través de la máquina de estado.

Keywords: UML 2.0, Máquinas de Estado, Instituciones

An Institution for UML 2.0 State Machines *

Daniel Calegari[1] and Nora Szasz[2]

[1] Facultad de Ingeniería, Universidad de la República
11300 Montevideo, Uruguay
dcalegar@fing.edu.uy
[2] Facultad de Ingeniería, Universidad ORT Uruguay
11100 Montevideo, Uruguay
szasz@ort.edu.uy

Abstract. The theory of institutions provides an elegant and robust framework for programming in the large and in particular for compositionality. It can be used to define an heterogeneous environment for the semantic definition of UML, consisting of a family of formalisms which capture various UML sublanguages, and morphisms that represent the expected semantic relationships between them. In this article we present an institution for UML 2.0 State Machines devised for collaborating with the definition of such environment. The semantics behind the institution is based on a previous work which deals with processing simple input events within a transition step. We also extend this semantics for handling sequences of events, and then for considering runs through the state machine.

Keywords: UML 2.0, State Machines, Institutions

1 Introduction

Quality in model-intensive approaches for software development relies on a precise definition of the models used for describing the software system to be developed. Since the Unified Modeling Language (UML [1]) is the most widely adopted software modeling notation in use today, there are many efforts, often uncoordinated, to define its formal semantics. In addition to the variety of sublanguages, perhaps the greatest complexity lies in the fact that these languages are naturally described using heterogeneous semantic domains that cannot be easily integrated. Indeed handling heterogeneity seems to be the key challenge. In [2], an heterogeneous approach to the semantics of UML is proposed which deals with the integration of the different formalisms. The proposal is based on the definition of an *heterogeneous institution environment* consisting of a family of institutions capturing various UML sublanguages, and morphisms that represent the expected semantic relationships between them. This allows each sublanguage to be described using its own semantic domains.

The concept of Institution [3] was originally introduced to formalise the notion of logical system. Informally, an institution consists of a collection of signatures (vocabularies for use in constructing sentences in a logical system), signature morphisms

* Technical Report / InCo-PEDECIBA / TR11-02 / ISSN 0797-6410 / April 2011
http://www.fing.edu.uy/inco/pedeciba/bibpm/field.php/Main/ReportesT%e9cnicos

(allowing many different vocabularies at once), a collection of sentences and models for a given signature, and a satisfaction relation of sentences by models, such that when signatures are changed (by a signature morphism), satisfaction of sentences by models changes consistently. The theory provides many interesting results, for example, it states that any institution whose signatures can be glued together will also allow gluing together theories (collections of sentences) to form larger specifications. This gives a very rich and flexible framework that can be used in program specification [4].

To the best of our knowledge, the heterogeneous environment devised so far lacks an institution for State Machines [1], which constitute a valuable notation for describing behavioral aspects of a system. The contribution of this paper is the definition of an institution for UML 2.0 State Machines.

There is a plethora of formal semantics for UML State Machines [5]. Most of them either consider previous versions of the standard, or just consider a small subset of the language. In [6] Fecher and Schönborn present a complete formal semantics of UML 2.0 State Machines. They give a sublanguage with fewer design features for which a precise syntax and a formal semantics are defined. Then the semantics of a state machine is given via a transformation into the sublanguage. We base our work on this proposal since it is complete enough and it is presented within an algebraic approach that is specially useful for our purpose. We also extend these semantic definitions for dealing with sequences of events, inspired by [7]. In this work, von der Beeck proposes a two-phase formal semantics for UML 1.4 State Machines introducing the idea of using the output of one step as part of the input of the next step. Moreover, we study how the semantics can be modified to define an institution which considers runs through the state machine instead of a simple transition step.

The rest of the paper is organized as follows: Section 2 presents a few elementary concepts about state machines and the main components of the semantics defined in [6]. In Section 3 we use those components to define the building blocks of our institution. Then, in Section 4, we extend the semantics in [6] for defining the satisfaction relation and finally constructing the institution. As a complement, we study in Section 5 how to define an institution which considers runs through the state machine. Finally, in Section 6 we conclude and point out some issues for further work.

2 UML 2.0 State Machines

UML 2.0 state machines basically consist of states and transitions between them. The main feature of state machines is that states may contain regions (containing other states), defining a state hierarchy. A state may have *entry* and *exit* actions (executed when the state is entered/left), and *do* actions ((partly) executed as long as the state is active). The environment may send events to the state machine which are stored in the so called input queue of events. State machines follow the *run-to-completion* assumption, i.e., “an event occurrence can only be taken from the pool and dispatched if the processing of the previous current occurrence is fully completed” [1]. A dispatched event from the input queue that does not trigger transitions is either discarded or deferred. A transition connects a source state to a target state. It includes (optionally) a guard, a sequence of output actions, and it is triggered by an event. A transition not exiting any state is called an internal transition. There are also some special kind of states called pseudostates. Join and fork pseudostates are used to collect different transitions into a compound transition. Exit and entry pseudostates are used to change the order of the action execution. Junction pseudostates are a shorthand notation for the set of transitions obtained by combining any incoming transition with any outgoing one. Finally, history pseudostates activate those substates of the region that were active when the region was the last time active.

In what follows we present the main components of the semantics defined in [6]. For convenience, we adapt some definitions. Changes will be pointed out along the document. For more details the reader is referred to the source.

The core language for state machines consists of composite and final states, regions, choice, entry and exit pseudostates, internal/external transitions, do actions, and event deferral. No interlevel transitions, i.e., transitions crossing a state border, are allowed, instead additional exit, entry pseudostates are used. Default exit and entry must be explicitly modeled. There are three different kinds of exit pseudostates: a normal one, a priority relevant one, and a completion relevant one, which is only 'enabled' if the do actions of the corresponding state have terminated (but not necessarily its corresponding regions). States are only allowed as sources and targets of internal transitions. The construction of a core state machine is done by aplying a transformation from a UML 2.0 State Machines such that a precise semantics coincident with the UML standard is obtained [6].

$\mathcal{S}$ denotes a set of states, partitioned into composite, final, exit, entry, and choice states, denoted by $\mathcal{S}_{\text{com}}$, $\mathcal{S}_{\text{fin}}$, $\mathcal{S}_{\text{exit}}$, $\mathcal{S}_{\text{entry}}$, and $\mathcal{S}_{\text{choice}}$, respectively. Furthermore, $\mathcal{S}_{\text{exit}}$ is partitioned into priority, non-priority, and completion exit states, denoted by $\mathcal{S}^{\text{pr}}_{\text{exit}}$, $\mathcal{S}^{\text{npr}}_{\text{exit}}$, and $\mathcal{S}^{\text{cp}}_{\text{exit}}$, respectively. Exit states belong to final states and not only to composite states. The set $\mathcal{S}$ is defined together a set of regions $\mathcal{R}$ (a region is a set of states) and a function `parent`. Function `parent` maps composite, final, and choice states to regions ($\texttt{parent} : (\mathcal{S}_{\text{com}} \cup \mathcal{S}_{\text{fin}} \cup \mathcal{S}_{\text{choice}}) \to \mathcal{R}$); maps regions (different from the outermost region) and entry states to composite states ($\texttt{parent} : \mathcal{R} \to \mathcal{S}_{\text{com}}$ and $\texttt{parent} : \mathcal{S}_{\text{entry}} \to \mathcal{S}_{\text{com}}$); and maps exit states to composite or final states ($\texttt{parent} : \mathcal{S}_{\text{exit}} \to (\mathcal{S}_{\text{com}} \cup \mathcal{S}_{\text{fin}})$). Furthemore, $\succ$ denotes the containing relation derived from `parent`, and $\succeq$ denotes the reflexive closure of $\succ$. Moreover, there are functions $\texttt{stateOf} : \mathcal{S} \to (\mathcal{S}_{\text{com}} \cup \mathcal{S}_{\text{fin}} \cup \mathcal{S}_{\text{choice}})$ and $\texttt{regOf} : \mathcal{S} \to \mathcal{R}$ that yield the deepest composite, final, or choice state (respectively region) that contains the argument. The direct subregions of a composite state $s \in \mathcal{S}_{\text{com}}$ are given by the function $\texttt{dsr} : \mathcal{S}_{\text{com}} \to \mathcal{R}$.

$\mathcal{E}$ denotes the set of all events. The silent event is denoted by τ. $\mathcal{G}$ denotes a set of boolean expressions, which depend on global information such as the attribute values of the objects. Furthermore, the atomic predicates `wla` and `nab` are members of $\mathcal{G}$, and are used to model the history mechanism, such that `wla` indicates that the target state of the transition having `wla` as guard was last active, and `nab` indicates that the region of the target state of the transition having `nab` as guard was not active before.

An action is a sequence of atomic actions. $\mathcal{A}$ is the set of all actions and $\texttt{skip} \in \mathcal{A}$ is used to denote the termination of a sequence of actions. $\mathcal{B}$ is the type of all sets of action sequences. More precisely, $B \in \mathcal{B}$ encodes a set of sequences of actions; between every action of such a sequence an interleaving point exists, i.e., another action (of a transition fired in parallel) can be executed before the sequence continues. The transitions of the state machines will be labeled with elements of $\mathcal{B}$ instead of $\mathcal{A}^*$, however, each transition contains only one sequence of actions (a singleton set).

Definition 1 (Transition). A transition is a tuple $(s_1, e, g, B, s2)$ such that:

- $s_1 \in \mathcal{S} \backslash \mathcal{S}_{\text{fin}}$ is called its source state,
- $s_2 \in \mathcal{S}$ is called its target state,
- $e \in (\mathcal{E} \cup \{\tau\})$ is called its trigger event
- $g \in \mathcal{G}$ is called its guard, constraining the necessary condition for the enabling of the transition, and
- $B \in \mathcal{B}$ is called its action encoding.

The projections of transitions to the corresponding components are denoted by π_{sor}, π_{tar}, π_{ev}, π_{gua}, and π_{act}, respectively.

Given these definitions a core state machine can be defined as follows.

Definition 2 (Core State Machine). Given domains $\mathcal{E}$, $\mathcal{A}$ and $\mathcal{G}$, a core state machine is a tuple $((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}, \texttt{defer}, \mathcal{T}, s_{\text{start}})$ where:

- $\mathcal{S}$ is a set of states defined together a set of regions $\mathcal{R}$ and function $\texttt{parent}$,
- $\texttt{doAct} : \mathcal{S}_{\text{com}} \to \mathcal{A}$ assigns to each state the do action that can be executed when the state is active,
- $\texttt{defer} : \mathcal{E} \to 2^{\mathcal{S}_{\text{com}}}$ assigns to each event those states in which it will be deferred,
- $\mathcal{T}$ is a set of transitions, where $\mathcal{T}_{\text{int}}$ denotes the set of the singleton sets of internal transitions, and
- $s_{\text{start}} \in \mathcal{S}_{\text{com}}$ is the initial state, belonging to the uppermost region and having no subregions.

The set of compound transitions of a core state machine (denoted $\texttt{CoTr}$) is either a set consisting of one internal transition ($\in \mathcal{T}_{\text{int}}$) or a collection of (non-internal) transitions such that a single outermost state is exited. Note that here, contrary to UML state machines, only transitions outgoing exit pseudostates are collected in a compound transition.

Definition 3 (Compound Transitions). Given a core state machine $K = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$, the set of compound transitions is defined as follows:

$$\mathtt{CoTr}_K = \{\{t\} \cup T \mid t \in \mathcal{T} \wedge \pi_{\text{sor}}(t) \in \mathcal{S}_{\text{exit}} \wedge \pi_{\text{tar}}(t) \notin \mathcal{S}_{\text{exit}} \wedge T \in \Upsilon_K(\pi_{\text{sor}}(t))\} \cup \mathcal{T}_{\text{int}}$$

Function $\Upsilon_K : \mathcal{S}_{\text{exit}} \to 2^{2^{\mathcal{T}}}$ collects all sets of transitions 'below' s that can belong to a compound transition that 'includes' s. Formally:

$$\Upsilon_K(s) = \{ \bigcup_{r \in \mathtt{dsr}(s)} (\{f(r)\} \cup F(r)) \mid f : \mathtt{dsr}(s) \to \mathcal{T} \wedge F : \mathtt{dsr}(s) \to 2^{\mathcal{T}} \wedge$$
$$\forall r \in \mathtt{dsr}(s) : \pi_{\text{tar}}(f(r)) = s \wedge \mathtt{regOf}(\pi_{\text{sor}}(f(r))) = r \wedge \Upsilon_K(\pi_{\text{sor}}(t))\}$$

A history is a function $H : \mathcal{R} \to \mathcal{S}_{\text{com}} \cup \mathcal{S}_{\text{fin}} \cup \{\bot\}$ mapping a region r to its direct substate that was active the last time r was left. If r was not active before (or a final state was last active), r is mapped to the default value $\bot$. The set of histories of a core state machine is denoted $\mathcal{H}$.

Definition 4 (History). The set of histories of a core state machine $((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$ is defined as follows:

$$\mathcal{H} = \{H : \mathcal{R} \to \mathcal{S}_{\text{com}} \cup \mathcal{S}_{\text{fin}} \cup \{\bot\} \mid \forall r \in \mathcal{R} : H(r) \neq \bot \Rightarrow r = \mathtt{regOf}(H(r))\}$$

A csm-configuration (or configuration) is a snapshot of a state machine execution. For the sake of simplicity, the definition below allows configurations that cannot occur during execution.

Definition 5 (CSM-Configuration). Given domains $\mathcal{E}$, $\mathcal{A}$ and $\mathcal{G}$, and a core state machine $K = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$, a configuration is a tuple $(S, \mathtt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$ where

- $S \subseteq \mathcal{S}_{\text{com}} \cup \mathcal{S}_{\text{fin}}$, denotes which states are active,
- $\mathtt{do} : \mathcal{S}_{\text{com}} \to \mathcal{A}$, denotes the corresponding do action that remains to be executed,
- $H \in \mathcal{H}$, denotes its current history information,
- $\alpha \in \mathcal{A}$, denotes the action that has to be executed next w.r.t. transition execution
- $\ddot{s} \in \{\emptyset\} \cup \mathcal{S}_{\text{fin}} \cup \mathcal{S}_{\text{exit}} \cup \mathcal{S}_{\text{entry}} \cup \mathcal{S}_{\text{choice}}$, denotes the state that has to be activated after α is completed,
- $\beta \subseteq \mathcal{B} \times (\{\emptyset\} \cup \mathcal{S}_{\text{fin}} \cup \mathcal{S}_{\text{exit}} \cup \mathcal{S}_{\text{entry}} \cup \mathcal{S}_{\text{choice}})$, denotes the currently executing transitions (i.e. remaining actions and target states),
- $T \in \mathtt{CoTr}_K \cup \{\emptyset\}$, denotes the currently executing compound transition, and
- $\ddot{T} \subseteq \mathtt{CoTr}_K$, denotes the transitions that are left to be executed in order to complete the step.

Unlike [6], we omit the variable environment in both core state machines and configuration definitions. We will explain the reason for this in the next section.

3 The Ingredients of the SM Institution

The formal definition of an *institution* relies on Category Theory [8]. As defined in [9], an institution $\mathcal{I}$ [3] consists of:

1. a category $\mathrm{Sign}_{\mathcal{I}}$ of *signatures*;
2. a functor $\mathrm{Sen}_{\mathcal{I}} : \mathrm{Sign}_{\mathcal{I}} \to \mathrm{Set}$, giving a set $\mathrm{Sen}(\Sigma)$ of Σ-*sentences* for each signature $\Sigma \in |\mathrm{Sign}_{\mathcal{I}}|$ [4] and a function $\mathrm{Sen}_{\mathcal{I}}(\sigma)$:$\mathrm{Sen}_{\mathcal{I}}(\Sigma_1) \to \mathrm{Sen}_{\mathcal{I}}(\Sigma_2)$ translating Σ_1-sentences to Σ_2-sentences for each signature morphism $\sigma : \Sigma_1 \to \Sigma_2$;
3. a functor $\mathrm{Mod}_{\mathcal{I}} : \mathrm{Sign}_{\mathcal{I}}^{op} \to \mathbf{Cat}$ [5], giving a category $\mathrm{Mod}(\Sigma)$ of Σ-*models* for each signature $\Sigma \in |\mathrm{Sign}_{\mathcal{I}}|$ and a functor $\mathrm{Mod}_{\mathcal{I}}(\sigma)$:$\mathrm{Mod}_{\mathcal{I}}(\Sigma_2) \to \mathrm{Mod}_{\mathcal{I}}(\Sigma_1)$ translating Σ_2-models to Σ_1-models (and Σ_2-morphisms to Σ_1-morphisms) for each signature morphism $\sigma : \Sigma_1 \to \Sigma_2$;
4. for each signature $\Sigma \in |\mathrm{Sign}_{\mathcal{I}}|$, a *satisfaction relation* $\models_{\mathcal{I},\Sigma} \subseteq |\mathrm{Mod}_{\mathcal{I}}(\Sigma)| \times \mathrm{Sen}_{\mathcal{I}}(\Sigma)$;

such that for any signature morphism $\sigma : \Sigma_1 \to \Sigma_2$ the translation $\mathrm{Mod}_{\mathcal{I}}(\sigma)$ of models and $\mathrm{Sen}_{\mathcal{I}}(\sigma)$ of sentences preserve the satisfaction relation, that is, for any $\varphi \in \mathrm{Sen}_{\mathcal{I}}(\Sigma)$ and $M_2 \in |\mathrm{Mod}_{\mathcal{I}}(\Sigma_2)|$:

$$M_2 \models_{\mathcal{I},\Sigma_2} \mathrm{Sen}_{\mathcal{I}}(\sigma)(\varphi) \text{ iff } \mathrm{Mod}_{\mathcal{I}}(\sigma)(M_2) \models_{\mathcal{I},\Sigma_1} \varphi$$

We now introduce the basic definitions and properties to define the institution SM of UML 2.0 state machines. As mentioned before, we use the syntax and semantics defined in [6].

As in algebraic specifications, a signature defines the syntax of an algebra by characterising the ways in which its components may legally be combined [9]. In our case, a signature defines a core state machine along with the set of events, actions and guards.

Definition 6 (Signature). A state machine signature, or signature for short, is a tuple $(\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$ where:

- $\mathcal{E}$ is a set of events,
- $\mathcal{A}$ a set of actions, where $\mathtt{skip} \in \mathcal{A}$,
- $\mathcal{G}$ a set of guards, where $\mathtt{wla} \in \mathcal{G}$ and $\mathtt{nab} \in \mathcal{G}$, and
- K is a core state machine, w.r.t. the domains $\mathcal{E}$, $\mathcal{A}$ and $\mathcal{G}$.

A signature morphism allows modifications in the sets of events, actions and guards, while keeping states unmodified.

[3] We often omit the subscript $\mathcal{I}$

[4] $|C|$ is the collection of objects of a category C

[5] Sign^{op} is the opposite category of the category Sign

Definition 7 (Signature Morphism). Given signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1, 2), a signature morphism $\sigma{:}\Sigma_1 \to \Sigma_2$ is a tuple $(\sigma_\mathcal{E}, \sigma_\mathcal{A}, \sigma_\mathcal{G})$ of functions, such that

- $\sigma_\mathcal{E} : \mathcal{E}_1 \to \mathcal{E}_2$ is a bijection
- $\sigma_\mathcal{A} : \mathcal{A}_1 \to \mathcal{A}_2$ is an injective function, and also $\sigma_\mathcal{A}(\texttt{skip}) = \texttt{skip} \in \mathcal{A}_2$
- $\sigma_\mathcal{G} : \mathcal{G}_1 \to \mathcal{G}_2$, and also $\sigma_\mathcal{G}(\texttt{wla}) = \texttt{wla} \in \mathcal{G}_2$, $\sigma_\mathcal{G}(\texttt{nab}) = \texttt{nab} \in \mathcal{G}_2$
- $\sigma_\mathcal{A}(\texttt{doAct}_1(s)) = \texttt{doAct}_2(s)$ for each $s \in \mathcal{S}_{com}$
- $\texttt{defer}_1(e) = \texttt{defer}_2(\sigma_\mathcal{E}(e))$ for each $e \in \mathcal{E}_1$
- $t_1 = (s_1, e, g, B, s_2) \in \mathcal{T}_1$ iff $t_2 = (s_1, \sigma_\mathcal{E}(e), \sigma_\mathcal{G}(g), \sigma_\mathcal{A}(B), s_2) \in \mathcal{T}_2$.
  Also, $t_1 \in \mathcal{T}_{\text{int}1}$ iff $t_2 \in \mathcal{T}_{\text{int}2}$.

$\sigma_\mathcal{A}(B)$ is the extension of $\sigma_\mathcal{A}$ for processing sets of sequences of actions, defined as follow: $\sigma_\mathcal{A}(B) = \{\widehat{\sigma_\mathcal{A}}(b) \mid b \in B\}$. $\widehat{\sigma_\mathcal{A}}(b)$ is the extension of $\sigma_\mathcal{A}$ for processing sequences of actions, defined as follows: $\widehat{\sigma_\mathcal{A}}(\texttt{skip}) = \sigma_\mathcal{A}(\texttt{skip})$, and $\widehat{\sigma_\mathcal{A}}(ab) = \sigma_\mathcal{A}(a)\widehat{\sigma_\mathcal{A}}(b)$ with $a \in \mathcal{A}_1$, and $b \in {\mathcal{A}_1}^*$. Moreover, the extension of $\sigma_\mathcal{A}$ for processing set of actions is defined as follows: $\sigma_\mathcal{A}(A) = \{\sigma_\mathcal{A}(a) \mid a \in A\}$.

Lemma 1. Signatures and signature morphisms define a category Sign_{SM}. The points of the category are the signatures, and its arrows are the signature morphisms.

Proof. Let $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ be signatures with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1..4), and let $\sigma_i : \Sigma_i \to \Sigma_{i+1}$ (i=1..3) be signature morphisms, then:

- signature morphisms can be composed. We define the composition $\sigma_2 \circ \sigma_1$ as a tuple $(\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}}, \sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}}, \sigma_{2\mathcal{G}} \circ \sigma_{1\mathcal{G}})$ such that $\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}}(e) = \sigma_{2\mathcal{E}}(\sigma_{1\mathcal{E}}(e))$ for each $e \in \mathcal{E}_1$, $\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}}(a) = \sigma_{2\mathcal{A}}(\sigma_{1\mathcal{A}}(a))$ for each $a \in \mathcal{A}_1$, and $\sigma_{2\mathcal{G}} \circ \sigma_{1\mathcal{G}}(g) = \sigma_{2\mathcal{G}}(\sigma_{1\mathcal{G}}(g))$ for each $g \in \mathcal{G}_1$. We have to show that $\sigma_2 \circ \sigma_1$ is a signature morphism:
  - $e \in \mathcal{E}_1$ implies $(\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(e) \in \mathcal{E}_3$ since $b = \sigma_{1\mathcal{E}}(e) \in \mathcal{E}_2$ and $\sigma_{2\mathcal{E}}(b) \in \mathcal{E}_3$ by definition $\sigma_{1\mathcal{E}}$ and $\sigma_{2\mathcal{E}}$, respectively. Also, $\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}}$ is bijective, since the composition of bijective functions is a bijective function.
  - $a \in \mathcal{A}_1$ implies $(\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(a) \in \mathcal{A}_3$ since $b = \sigma_{1\mathcal{A}}(a) \in \mathcal{A}_2$ and $\sigma_{2\mathcal{A}}(b) \in \mathcal{A}_3$ by definition $\sigma_{1\mathcal{A}}$ and $\sigma_{2\mathcal{A}}$, respectively. Also, $\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}}$ is injective, since the composition of injective functions is an injective function. Moreover, $(\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(\texttt{skip})$ with $\texttt{skip} \in \mathcal{A}_1$ is equal to $\sigma_{2\mathcal{A}}(\sigma_{1\mathcal{A}}(\texttt{skip}))$ that is equal to $\sigma_{2\mathcal{A}}(\texttt{skip})$ with $\texttt{skip} \in \mathcal{A}_2$, that is equal to $\texttt{skip} \in \mathcal{A}_3$.
  - $g \in \mathcal{G}_1$ implies $(\sigma_{2\mathcal{G}} \circ \sigma_{1\mathcal{G}})(g) \in \mathcal{G}_3$ since $b = \sigma_{1\mathcal{G}}(g) \in \mathcal{G}_2$ and $\sigma_{2\mathcal{G}}(b) \in \mathcal{G}_3$ by definition $\sigma_{1\mathcal{G}}$ and $\sigma_{2\mathcal{G}}$, respectively. Also, $(\sigma_{2\mathcal{G}} \circ \sigma_{1\mathcal{G}})(\texttt{wla})$ (resp. $\texttt{nab}$) with $\texttt{wla} \in \mathcal{G}_1$ is equal to $\sigma_{2\mathcal{G}}(\sigma_{1\mathcal{G}}(\texttt{wla}))$ that is equal to $\sigma_{2\mathcal{G}}(\texttt{wla})$ with $\texttt{wla} \in \mathcal{G}_2$, by definition of $\sigma_{1\mathcal{G}}$, and that is equal to $\texttt{wla} \in \mathcal{G}_3$ by definition of $\sigma_{2\mathcal{G}}$.
  - $(\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(\texttt{doAct}_1(s)) = \sigma_{2\mathcal{A}}(\sigma_{1\mathcal{A}}(\texttt{doAct}_1(s)))$ for each $s \in \mathcal{S}_{com}$, by definition of $\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}}$. Then $\sigma_{2\mathcal{A}}(\sigma_{1\mathcal{A}}(\texttt{doAct}_1(s))) = \sigma_{2\mathcal{A}}(\texttt{doAct}_2(s))$ by definition of σ_1. Moreover, we have that $\sigma_{2\mathcal{A}}(\texttt{doAct}_2(s)) = \texttt{doAct}_3(s)$ by definition of σ_2. Finally, we conclude that $(\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(\texttt{doAct}_1(s)) = \texttt{doAct}_3(s)$ as expected.

- $\texttt{defer}(e) = \texttt{defer}_2(\sigma_{1\mathcal{E}}(e))$ for each $e \in \mathcal{E}_1$ by definition of $\sigma_{1\mathcal{E}}$ and $\texttt{defer}_2(\sigma_{1\mathcal{E}}(e)) = \texttt{defer}_3(\sigma_{2\mathcal{E}}(\sigma_{1\mathcal{E}}(e)))$ by definition of $\sigma_{2\mathcal{E}}$. Finally, $\texttt{defer}(e) = \texttt{defer}_3((\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(e))$ by definition of $\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}}$.
- $(s_1, e, g, B, s_2) \in \mathcal{T}_1$ iff $(s_1, \sigma_{1\mathcal{E}}(e), \sigma_{1\mathcal{G}}(g), \sigma_{1\mathcal{A}}(B), s_2) \in \mathcal{T}_2$ by definition of σ_1, iff $(s_1, \sigma_{2\mathcal{E}}(\sigma_{1\mathcal{E}}(e)), \sigma_{2\mathcal{E}}(\sigma_{1\mathcal{G}}(g)), \sigma_{2\mathcal{E}}(\sigma_{1\mathcal{A}}(B)), s_2) \in \mathcal{T}_3$ by definition of σ_2. Finally, that is equal to $(s_1, (\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(e), (\sigma_{2\mathcal{G}} \circ \sigma_{1\mathcal{G}})(g), (\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(B), s_2)$ by definition of $\sigma_2 \circ \sigma_1$. Also, $t \in \mathcal{T}_{\text{int}1}$ iff $(\sigma_2 \circ \sigma_1)(t) \in \mathcal{T}_{\text{int}3}$ by definition of σ_1, σ_2, and $\sigma_2 \circ \sigma_1$

– Composition of signature morphisms is associative, i.e. $(\sigma_3 \circ \sigma_2) \circ \sigma_1 = \sigma_3 \circ (\sigma_2 \circ \sigma_1)$:
  - For each $e \in \mathcal{E}_1$, $((\sigma_{3\mathcal{E}} \circ \sigma_{2\mathcal{E}}) \circ \sigma_{1\mathcal{E}})(e) = (\sigma_{3\mathcal{E}} \circ \sigma_{2\mathcal{E}})(\sigma_{1\mathcal{E}}(e)) = \sigma_{3\mathcal{E}}(\sigma_{2\mathcal{E}}(\sigma_{1\mathcal{E}}(e)))$ $= \sigma_{3\mathcal{E}}((\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(e)) = (\sigma_{3\mathcal{E}} \circ (\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}}))(e)$.
  - The proof is the same in the case of $\sigma_{\mathcal{A}}$ and $\sigma_{\mathcal{G}}$.

– There exists an identity signature morphism $id_{\Sigma_1} : \Sigma_1 \to \Sigma_1$ defined as a tuple $(id_{\mathcal{E}}, id_{\mathcal{A}}, id_{\mathcal{G}})$ such that $id_{\mathcal{E}}(e) = e$ for all $e \in \mathcal{E}_1$, $id_{\mathcal{A}}(a) = a$ for all $a \in \mathcal{A}_1$, and $id_{\mathcal{G}}(g) = g$ for all $g \in \mathcal{G}_1$. This morphism satisfies the signature morphism conditions:
  - $e \in \mathcal{E}_1$ iff $id_{\mathcal{E}}(e) \in \mathcal{E}_1$
  - $a \in \mathcal{A}_1$ implies $id_{\mathcal{A}}(a) \in \mathcal{A}_1$ (and it is inyective), and also $id_{\mathcal{A}}(\texttt{skip}) = \texttt{skip} \in \mathcal{A}_1$
  - $g \in \mathcal{G}_1$ implies $id_{\mathcal{G}}(g) \in \mathcal{G}_1$, and also $id_{\mathcal{G}}(\texttt{wla}) = \texttt{wla} \in \mathcal{G}_1$ and $id_{\mathcal{G}}(\texttt{nab}) = \texttt{nab} \in \mathcal{G}_1$
  - $id_{\mathcal{A}}(\texttt{doAct}_1(s)) = \texttt{doAct}_1(s)$ for each $s \in \mathcal{S}_{com}$
  - $\texttt{defer}(e) = \texttt{defer}(id_{\mathcal{E}}(e))$ for each $e \in \mathcal{E}_1$
  - $(s_1, e, g, B, s_2) \in \mathcal{T}_1$ iff $(s_1, id_{\mathcal{E}}(e), id_{\mathcal{G}}(g), id_{\mathcal{A}}(B), s_2) \in \mathcal{T}_1$ (there is a bijection)

$\square$

Sentences are syntactic components within a given signature. A sentence represents possible adjacent configurations of the core state machine w.r.t. a transition, together with the input queue of events associated to each configuration.

Definition 8 (Σ-sentence). Given a signature $\Sigma = (\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$, a Σ-sentence is a tuple $((\mathcal{C}_1, E_1), (\mathcal{C}_2, E_2))$ where $\mathcal{C}_1$ and $\mathcal{C}_2$ are configurations w.r.t. domains $\mathcal{E}$, $\mathcal{A}$, and $\mathcal{G}$, and core state machine K, and $E_1, E_2 \in \mathcal{E}^*$ are sets of events (event queues).

Signature morphisms induce translations of sentences. In our case, a sentence morphism allows modifications in the sets of events, actions and guards, while keeping states unmodified.

Definition 9 (Sentence Morphism). The extension of a signature morphism to Σ-sentences is defined as follows. Given a Σ_1-sentence ψ_1 = $((\mathcal{C}_1, E_1), (\mathcal{C}_2, E_2))$ with $\mathcal{C}_i = (S_i, \mathtt{do}_i, H_i, \alpha_i, \ddot{s}_i, \beta_i, T_i, \ddot{T}_i)$ $(i{=}1, 2)$, $\sigma(\psi_1)$ is a Σ_2-sentence $((\sigma(\mathcal{C}_1), \sigma_{\mathcal{E}}(E_1)), (\sigma(\mathcal{C}_2), \sigma_{\mathcal{E}}(E_2)))$ where $\sigma_{\mathcal{E}}(E_i) \stackrel{\text{def}}{=} \{\sigma_{\mathcal{E}}(e) \mid e \in E_i\}$, and $\sigma(\mathcal{C}_i)$ = $(S_i, \sigma_{\mathtt{do}}(\mathtt{do}_i), H_i, \sigma_{\mathcal{A}}(\alpha_i), \ddot{s}_i, \sigma_\beta(\beta_i), \sigma_T(T_i), \sigma_T(\ddot{T}_i))$ $(i{=}1, 2)$ such that:

- $\sigma_{\mathtt{do}}(\mathtt{do}_i)(s) \stackrel{\text{def}}{=} \sigma_{\mathcal{A}}(\mathtt{do}_i(s))$ for each $s \in \mathcal{S}_{\text{com}}$
- $\sigma_\beta(\beta_i) \stackrel{\text{def}}{=} \{(\sigma_{\mathcal{A}}(B), s) \mid (B, s) \in \beta_i\}$
- $\sigma_T(T_i) \stackrel{\text{def}}{=} (s_1, \sigma_{\mathcal{E}}(e), \sigma_{\mathcal{G}}(g), \sigma_{\mathcal{A}}(B), s_2)$ for each $T_i = (s_1, e, g, B, s_2) \in \mathcal{T}$
- $\sigma_T(\ddot{T}_i) \stackrel{\text{def}}{=} \{\sigma_T(t) \mid t \in \ddot{T}_i\}$

Signatures related by a signature morphism must respect the set of states and transitions, while events, actions and guards can be modified. Additionally, bijective (and sometimes inyective) functions are needed since the morphism is a renaming of some elements. From a semantical point of view, this is needed to ensure that the satisfaction of sentences by models changes consistently with changes in signatures.

Lemma 2. There is a functor Sen_{SM} giving a set of sentences ψ (object in the category Set) for each signature Σ (object in the category Sign_{SM}), as shown in Definition 8, and a function $\sigma : \text{Sen}_{\text{SM}}(\Sigma_1) \to \text{Sen}_{\text{SM}}(\Sigma_2)$ (arrow in the category Set) translating sentences for each signature morphism $\sigma : \Sigma_1 \to \Sigma_2$ (arrow in the category Sign_{SM}), as shown in Definition 9.

Proof. We have to prove that Sen_{SM} is indeed a functor, i.e.: (a) domain and codomain of the image of an arrow are the images of domain and codomain, respectively, of the arrow, (b) composition is preserved, and (c) identities are preserved.

(a) By Definition 9, the image of an arrow $\sigma : \text{Sen}_{\text{SM}}(\Sigma_1) \to \text{Sen}_{\text{SM}}(\Sigma_2)$ in the category Set is the arrow $\sigma : \Sigma_1 \to \Sigma_2$ in the category Sign_{SM}. Also, by Definition 8, the image of any object $\text{Sen}_{\text{SM}}(\Sigma)$ in the category Set is a signature Σ in the category Sign_{SM}. Thus, domain and codomain of the image of an arrow are the images of domain and codomain, respectively, of the arrow.

(b) We have to prove that $\text{Sen}_{\text{SM}}(\sigma_2 \circ \sigma_1)$ = $\text{Sen}_{\text{SM}}(\sigma_2) \circ \text{Sen}_{\text{SM}}(\sigma_1)$.
Let $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ be signatures with K_i = $((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}_i, \mathtt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ $(i{=}1..4)$, and let $\sigma_i{:}\Sigma_i \to \Sigma_{i+1}$ $(i{=}1, 2)$ be signature morphisms. By Definition 9, $\text{Sen}_{\text{SM}}(\sigma_2 \circ \sigma_1)$ is a function $\sigma_{\sigma_2 \circ\, \sigma_1}$ such that for any Σ_1-sentence ψ_1 = $((C_1, E_1), (C_2, E_2))$ with $\mathcal{C}_i = (S_i, \mathtt{do}_i, H_i, \alpha_i, \ddot{s}_i, \beta_i, T_i, \ddot{T}_i)$ $(i{=}1, 2)$, $\sigma_{\sigma_2 \circ\, \sigma_1}(\psi_1)$ is a Σ_3-sentence $((\sigma_{\sigma_2 \circ\, \sigma_1}(C_1), \sigma_{\mathcal{E}_{\sigma_2 \circ\, \sigma_1}}(E_1)), (\sigma_{\sigma_2 \circ\, \sigma_1}(C_2), \sigma_{\mathcal{E}_{\sigma_2 \circ\, \sigma_1}}(E_2)))$
with $\sigma_{\mathcal{E}_{\sigma_2 \circ\, \sigma_1}}(E_i) = \{\sigma_{\mathcal{E}_{\sigma_2 \circ\, \sigma_1}}(e) | \ e \in E_i\} = \{(\sigma_{2\mathcal{E}} \circ\ \sigma_{1\mathcal{E}})(e) | \ e \in E_i\}$ = $(\sigma_{2\mathcal{E}} \circ\ \sigma_{1\mathcal{E}})(E_i)$, and $\sigma_{\sigma_2 \circ\, \sigma_1}(C_i) =$
$(S_i, \sigma_{\sigma_2 \circ\, \sigma_1 \mathtt{do}}(\mathtt{do}_i), H_i, \sigma_{\sigma_2 \circ\, \sigma_1 \mathcal{A}}(\alpha_i), \ddot{s}_i, \sigma_{\sigma_2 \circ\, \sigma_1 \beta}(\beta_i), \sigma_{\sigma_2 \circ\, \sigma_1 T}(T_i), \sigma_{\sigma_2 \circ\, \sigma_1 T}(\ddot{T}_i))$ $(i{=}1, 2)$ such that:

- $\sigma_{\sigma_2 \circ\, \sigma_1 \mathtt{do}}(\mathtt{do}_i)(s) = (\sigma_{2\mathcal{A}} \circ\ \sigma_{1\mathcal{A}})(\mathtt{do}_i(s)) = (\sigma_{2\mathtt{do}} \circ\ \sigma_{1\mathtt{do}})(\mathtt{do}_i)(s)$ for each $s \in \mathcal{S}_{i\text{com}}$.

- $\sigma_{\sigma_2 \circ\, \sigma_1 \mathcal{A}}(\alpha_i) \;=\; (\sigma_{2\mathcal{A}} \circ\; \sigma_{1\mathcal{A}})(\alpha_i)$ for each $\alpha_i \in \mathcal{A}_i$
- $\sigma_{\sigma_2 \circ\, \sigma_1 \beta}(\beta_i) \;=\; \{((\sigma_{2\mathcal{A}} \circ\; \sigma_{1\mathcal{A}})(B), s) \mid (B, s) \in \beta_i\} \;=\; (\sigma_{2\beta} \circ\; \sigma_{1\beta})(\beta_i)$
- $\sigma_{\sigma_2 \circ\, \sigma_1 T}(T_i) \;=\; (s_1, (\sigma_{2\mathcal{E}} \circ\; \sigma_{1\mathcal{E}})(e), (\sigma_{2\mathcal{G}} \circ\; \sigma_{1\mathcal{G}})(g), (\sigma_{2\mathcal{A}} \circ\; \sigma_{1\mathcal{A}})(B), s_2) \;=\; (\sigma_{2T} \circ\; \sigma_{1T})(T_i)$ for each $T_i = (s_1, e, g, B, s_2) \in \mathcal{T}_i$
- $\sigma_{\sigma_2 \circ\, \sigma_1 T}(\ddot{T}_i) \;=\; \{(\sigma_{2T} \circ\; \sigma_{1T})(t) \mid t \in \ddot{T}_i\} \;=\; (\sigma_{2T} \circ\; \sigma_{1T})(\ddot{T}_i)$

Finally, this means that $\sigma_{\sigma_2 \circ\, \sigma_1} = \sigma_{\sigma_2} \circ \sigma_{\sigma_1}$ and thus $\text{Sen}_{\text{SM}}(\sigma_2 \circ \sigma_1) = \text{Sen}_{\text{SM}}(\sigma_2) \circ \text{Sen}_{\text{SM}}(\sigma_1)$.

(c) Let $id_{\Sigma_1} : \Sigma_1 \to \Sigma_1$ be an identity signature morphism (defined in Lemma 1). By Definition 9, for any Σ_1-sentence $\psi_1 = ((C_1, E_1), (C_2, E_2))$ with $\mathcal{C}_i \;=\; (S_i, \mathtt{do}_i, H_i, \alpha_i, \ddot{s}_i, \beta_i, T_i, \ddot{T}_i)$ $(i{=}1,2)$, $id_{\Sigma_1}(\psi_1)$ is a Σ_1-sentence $((id_{\Sigma_1}(C_1), id_{\Sigma_1}(E_1)), (id_{\Sigma_1}(C_2), id_{\Sigma_1}(E_2)))$ with $id_{\Sigma_1}(E_i) = \{id_{\mathcal{E}}(e) \mid e \in E_i\} = E_i$, and $id_{\Sigma_1}(C_i) = (S_i, id_{\mathtt{do}}(\mathtt{do}_i), H_i, id_{\mathcal{A}}(\alpha_i), \ddot{s}_i, id_{\beta}(\beta_i), id_T(T_i), id_T(\ddot{T}_i))$ $(i{=}1,2)$ such that:

- $id_{\mathtt{do}}(\mathtt{do}_i)(s) \stackrel{\text{def}}{=} id_{\mathcal{A}}(\mathtt{do}_i(s)) \;=\; \mathtt{do}_i(s)$ for each $s \in \mathcal{S}_{i\text{com}}$.
- $id_{\beta}(\beta_i) \stackrel{\text{def}}{=} \{(id_{\mathcal{A}}(B), s) \mid (B, s) \in \beta_i\} \;=\; \{(B, s) \mid (B, s) \in \beta_i\} \;=\; \beta_i$
- $id_T(T_i) \stackrel{\text{def}}{=} (s_1, id_{\mathcal{E}}(e), id_{\mathcal{G}}(g), id_{\mathcal{A}}(B), s_2) \;=\; (s_1, e, g, B, s_2) \;=\; T_i$ for each $T_i \in \mathcal{T}_i$
- $id_T(\ddot{T}_i) \stackrel{\text{def}}{=} \{id_T(t) \mid t \in \ddot{T}_i\} \;=\; \{t \mid t \in \ddot{T}_i\} \;=\; \ddot{T}_i$

Finally, we conclude that identities are preserved. □

As in algebraic specifications, a model assigns an interpretation to the elements in the signature. In order to simplify the definitions and without loss of potential, the states in $\mathcal{S}$ are not interpreted within a specific semantic domain. State are just used as progress marks of the execution of the state machine.

Definition 10 (**Σ-model).** Given a signature $\Sigma = (\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$ with $K = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$, a Σ-model is a tuple $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \surd)$ where:

- $\mathbb{A}$ and $\mathbb{E}$ are domains for actual actions and actual events, respectively
- $\mu : \mathcal{A} \to \mathbb{A}$ and $\eta : \mathcal{E} \to \mathbb{E}$ are bijective interpretation functions for actions and events, respectively
- $\mathtt{calc} : \mathbb{A} \to \mathbb{A} \times \mathbb{E}^*$ calculates the effect of action execution and returns the remaining atomic actions to be executed and a set of output events
- $\mathtt{eval} : \mathcal{H} \times \mathcal{S} \to 2^{\mathcal{G}}$ evaluates guards returning those which are true
- $\mathtt{sel} : \mathbb{E}^* \to \mathbb{E} \times \mathbb{E}^*$ separates the input event queue into an event and the remaining event queue
- $\mathtt{join} : \mathbb{E}^* \times \mathbb{E}^* \to \mathbb{E}^*$ composes a set of output events and a set of events into a new set of events
- $\rightsquigarrow : \mathcal{B} \times \mathbb{A} \times \mathcal{B}$ is a relation determining when an action can be executed, i.e. if $(B_1, a, B_2) \in \rightsquigarrow$, written $B_1 \stackrel{\alpha}{\rightsquigarrow} B_2$, then B_1 is not empty and B_2 is the result of removing (i.e. executing) a from B_1

– $\surd \subseteq \mathcal{B} \times \mathbb{A}$ is a termination predicate, where $(B, a) \in \surd$ indicates that an execution of a which leads to termination is possible

μ and η are also denoting the extension of μ and η to sets, respectively: $\mu(A) = \{\mu(a) \mid a \in A\}$, and $\eta(E) = \{\eta(e) \mid e \in E\}$. We assume that there is an action $\mathtt{skip} \in \mathbb{A}$ such that $\mu(\mathtt{skip}) = \mathtt{skip}$.

We assume that `wla` holds in $\mathtt{eval}(H, s)$ if s was the last active state of the corresponding region, and `nab` holds in $\mathtt{eval}(H, s)$ if the region of s was not visited before, or a final state was last active there.

We include here the functions `sel` and `join` from [7]. These functions must be defined accordingly for a concrete scheduling strategy of the input event queue. As pointed out in the previous section, we omit the variable environment from both core state machines and configuration definitions, and also from the function `calc`. The variable environment is omitted since we think of a model as the classifier for which the state machine is defined (or as the system as a whole), containing not only global variables but also its attributes and links to other classifiers. In this sense, the functions defined in the model are the interface for "manipulating" the system including the implicit variable environment.

In [6] `calc` returns a label representing the observable communications (e.g. the sending of an output event). Since further observable communications other than the generated events do not have impact in the semantics, we decide to modify the definition and consider `calc` to return only those events generated. These events can be processed by the state machine in a future step. In fact, an action execution could generate output events which are not consumed by the state machine but by another one which can also generate new events to be consumed by the first state machine. This processing is made by the function `join`.

The relation $\rightsquigarrow$ and the predicate $\surd$ were formerly presented as part of a transition system of actions. Since in the former semantics there is no difference between syntactic and semantic actions, the transition system was a standalone component of the semantics. In our case, we locate them within the model, since is the model the one who allows the interpretation of the syntactic elements and determines the state machine execution.

A set (a multiset indeed) of sequences (with possibly multiple occurrences of the same action) of actions is needed in order to represent parallel action sequences. Consider the example in Figure 1. Suppose the states `State2` and `State3` have the same entry action `act1` whereas the entry action of `State4` is `act4` and the entry action of `State5` is `act5`. When the first transition is taken, the only action to be performed is $\{(\mathtt{skip})\}$ and the target state is the choice state (this information is the field β of the csm-configuration). After that, the state machine performs two transitions in parallel giving (through the relation $\rightsquigarrow$) the following set of sequences $\{(\mathtt{act1},\mathtt{act4}), (\mathtt{act1},\mathtt{act5})\}$ to be executed in any order, possibly interleaving actions from different sequences, and targeting two different states (`State4` and `State5`). The next step requires to execute an action `act1` from one of the sequences, leading to the set $\{(\mathtt{act4}), (\mathtt{act1},\mathtt{act5})\}$, but also to the set $\{(\mathtt{act1},\mathtt{act4}),(\mathtt{act5})\}$.

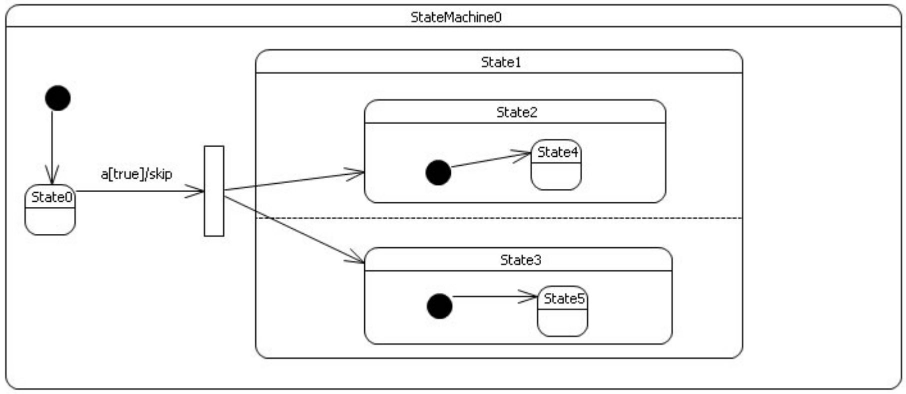

Fig. 1. Sequences of Actions

The definition of a model allows interpreting syntactic components, as for example the configurations, in the semantic domains.

Definition 11 **($\mathcal{M}$-interpretations).** Lets $\mathcal{M} = (\mathbb{A}, \mu, \texttt{eval}, \texttt{calc}, \mathbb{E}, \eta, \texttt{sel}, \texttt{join}, \rightsquigarrow, \surd)$ be a model. The semantic interpretation of a configuration $\mathcal{C} = (S, \texttt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$ within the model $\mathcal{M}$, denoted by $[\![\mathcal{C}]\!]_{\mathcal{M}}$, is defined as follows.
$[\![(S, \texttt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})]\!]_{\mathcal{M}} = (S, \texttt{do}^{\mathcal{M}}, H, \alpha^{\mathcal{M}}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})$, where:

- $\texttt{do}^{\mathcal{M}}(s) = \mu(\texttt{do}(s))$
- $\alpha^{\mathcal{M}} = \mu(\alpha)$
- $\beta^{\mathcal{M}} = \{(\mu(B), s) | \ (B, s) \in \beta\}$, with $\mu(B) = \{\mu(A) \mid A \in B\}$
- $T^{\mathcal{M}} = (s_1, \eta(e), g, B^{\mathcal{M}}, s2)$ with $T = (s_1, e, g, B, s2)$
- $\ddot{T}^{\mathcal{M}} = \{T^{\mathcal{M}} \mid T \in \ddot{T}\}$

The semantic interpretation of a set of events E within the model $\mathcal{M}$, denoted by $[\![E]\!]_{\mathcal{M}}$, is defined as follows: $[\![E]\!]_{\mathcal{M}} = \eta(E)$. Moreover, $\mathcal{T}^{\mathcal{M}}$ and $\mathcal{T}^{\mathcal{M}}_{\text{int}}$ represents the interpretation of the set of transition and internal transitions of a state machine, respectively. Finally, the semantic interpretation could be trivially extended to actions, events, and transitions.

When there is no risk of confusion we will omit the subscript $\mathcal{M}$ in the semantic evaluation function $[\![]\!]_{\mathcal{M}}$.

There are some interesting results about the invariance of interpretations with respect to signature morphisms, which will be used later.

Lemma 3. Given signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ (i=1, 2), with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1, 2), a signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_1-configuration $\mathcal{C} = (S, \texttt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$, and a Σ_2-model $\mathcal{M} = (\mathbb{A}, \mu, \texttt{eval}, \texttt{calc}, \mathbb{E}, \eta, \texttt{sel}, \texttt{join}, \rightsquigarrow, \surd)$, the following properties hold:

1. $\alpha^{\mathcal{M}|_\sigma} = \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}$ for any $\alpha \in \mathcal{A}_1$. This result is trivially extended to sets of actions ($\alpha \in \mathcal{A}^*{}_1$), and sets of sequences of actions ($\alpha \in \mathcal{B}_1$).

Proof.

$$\begin{aligned}
\alpha^{\mathcal{M}|_\sigma} &= \mu|_\sigma\ (\alpha) && \text{by definition of } \mathcal{M}\text{-interpretation} \\
&= \mu(\sigma_{\mathcal{A}}(\alpha)) && \text{by definition of } \mu|_\sigma \\
&= \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}} && \text{by definition of } \mathcal{M}\text{-interpretation}
\end{aligned}$$

2. $t^{\mathcal{M}|_\sigma} = \sigma_T(t)^{\mathcal{M}}$ for any $t \in \mathcal{T}_1$. This result is trivially extended to sets of transitions ($t \in \mathcal{T}^*{}_1$).

   Proof. Given $t = (s_1, e, g, B, s_2)$,

$$\begin{aligned}
t^{\mathcal{M}|_\sigma} &= (s_1, \eta|_\sigma\ (e), g, B^{\mathcal{M}|_\sigma}, s_2) && \text{by definition of } \mathcal{M}\text{-interpretation} \\
&= (s_1, \eta(\sigma_{\mathcal{E}}(e)), g, \sigma_{\mathcal{A}}(B)^{\mathcal{M}}, s_2) && \text{by definition of } \mu|_\sigma \text{ and last proof} \\
&= \sigma_T(t)^{\mathcal{M}} && \text{by definition of } \mathcal{M}\text{-interpretation}
\end{aligned}$$

3. $\mathtt{do}^{\mathcal{M}|_\sigma} = \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}$

   Proof.

$$\begin{aligned}
\mathtt{do}^{\mathcal{M}|_\sigma} &= \mu|_\sigma\ (\mathtt{do}(s)) && \text{by definition of } \mathcal{M}\text{-interpretation} \\
&= \mu(\sigma_{\mathcal{A}}(\mathtt{do}(s))) && \text{by definition of } \mu|_\sigma \\
&= \mu(\sigma_{\mathtt{do}}(\mathtt{do}(s))) && \text{by definition of } \sigma_{\mathtt{do}} \\
&= \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}} && \text{by definition of } \mathcal{M}\text{-interpretation}
\end{aligned}$$

4. $[\![\mathcal{C}]\!]_{\mathcal{M}|_\sigma} = [\![\sigma(\mathcal{C})]\!]_{\mathcal{M}}$.

   Proof.

$$\begin{aligned}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} &= (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \alpha^{\mathcal{M}|_\sigma}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) && \text{by } \mathcal{M}\text{-int.} \\
&= (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) && \text{by 1, 2 and 3} \\
&= [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} && \text{by } \mathcal{M}\text{-int.}
\end{aligned}$$

□

The model allows determining when a transition is enable and fireable. As defined in [6], a compound transition is enabled for an event if the sources of its transitions are active, if the event is the trigger of all of its transitions, if the guards of its transitions evaluate to true, and if the do actions are terminated for its transitions having elements from $\mathcal{S}^{\mathrm{cp}}_{\mathrm{exit}}$ as its sources. Also, two compound transitions are in conflict if their source states coincide. Moreover, a compound transition has priority over another one if every priority relevant source state of the first transition is a substate of a priority relevant source state of the second one. Finally, a set of compound transitions is fireable if it is a non-empty maximal set of enabled and conflict free compound transitions such that no enabled compound transition with higher priority exists.

Definition 12 (Enable and Fireable Transitions). Given a signature $\Sigma = (\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$ with $K = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$, a configuration $\mathcal{C} = (S, \mathtt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$, and a Σ-model $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \surd)$, the set of enable transitions for trigger $e \in \mathbb{E} \cup \{\tau\}$ is defined as follows:

$$
\begin{aligned}
\mathtt{Enable}_{K,\mathcal{C},\mathcal{M},e} = \{T' \in \mathtt{CoTr}_K \mid \forall t \in T' : &\ \mathtt{stateOf}(\pi_{\text{sor}}(t)) \in S \wedge \\
&\eta(\pi_{\text{ev}}(t)) = e \ \wedge \pi_{\text{gua}}(t) \in \mathtt{eval}(H, \pi_{\text{tar}}(t)) \wedge \\
&(\pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}} \Rightarrow \mathtt{do}(\mathtt{stateOf}(\pi_{\text{sor}}(t))) = \mathtt{skip})\}
\end{aligned}
$$

Two set of pairs of conflict transitions of a given core state machines is:

$$
\begin{aligned}
\mathtt{Conflict}_K = \{(T_1, T_2) \in \mathtt{CoTr}_K \times \mathtt{CoTr}_K \mid &\bigcup_{t_1 \in T_1} \mathtt{stateOf}(\pi_{\text{sor}}(t_1)) \cap \\
&\bigcup_{t_2 \in T_2} \mathtt{stateOf}(\pi_{\text{sor}}(t_2)) \neq \emptyset\}
\end{aligned}
$$

The set of priorities between compound transitions is:

$$
\begin{aligned}
\mathtt{Priority}_K = \{(T_1, T_2) \in \mathtt{CoTr}_K \times \mathtt{CoTr}_K \mid &\ \mathtt{PrBelow}_K(T_1, T_2) \wedge \\
&\ \mathtt{PrStrBelow}_K(T_1, T_2)\}
\end{aligned}
$$

with

$$
\begin{aligned}
\mathtt{PrBelow}_K(T_1, T_2) \Leftrightarrow (\forall t_1 \in T_1 \ : &\ \pi_{\text{sor}}(t_1) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \Rightarrow \exists t_2 \in T_2 \ : \\
&\ \pi_{\text{sor}}(t_2) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \wedge \\
&\ \mathtt{stateOf}(\pi_{\text{sor}}(t_2)) \succeq \mathtt{stateOf}(\pi_{\text{sor}}(t_1)))
\end{aligned}
$$

$$
\begin{aligned}
\mathtt{PrStrBelow}_K(T_1, T_2) \Leftrightarrow (&\exists t_2 \in T_2 \ : \ \pi_{\text{sor}}(t_2) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \wedge \forall t_1 \in T_1 \ : \\
&\pi_{\text{sor}}(t_1) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \Rightarrow \\
&\neg(\mathtt{stateOf}(\pi_{\text{sor}}(t_1)) \succeq \mathtt{stateOf}(\pi_{\text{sor}}(t_2))))
\end{aligned}
$$

Finally, a set of fireable compound transitions for trigger $e \in \mathbb{E} \cup \{\tau\}$ is:

$$
\begin{aligned}
\mathtt{Fireable}_{K,[\![\mathcal{C}]\!],\mathcal{M},e} = \{T''^{\mathcal{M}} \mid &\ T'' \subseteq \mathtt{Enable}_{K,\mathcal{C},\mathcal{M},e} \ \wedge \ T'' \neq \emptyset \wedge \\
&(\forall T' \in \mathtt{Enable}_{K,\mathcal{C},\mathcal{M},e} \setminus T'' : \\
&(\forall T \in T'' : (T', T) \notin \mathtt{Priority}_K) \wedge \\
&(\exists T \in T'' : (T, T') \in \mathtt{Conflict}_K)) \wedge \\
&\forall T_1, T_2 \in T'' : (T_1, T_2) \in \mathtt{Conflict}_K \Rightarrow T_1 = T_2\}
\end{aligned}
$$

A homomorphism between models is a function between the domains $\mathbb{A}$ and $\mathbb{E}$ which preserves the operations.

Definition 13 (Σ-homomorphism). Given a signature $\Sigma = (\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$ with $K = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$, and Σ-models $\mathcal{M}_i = (\mathbb{A}_i, \mu_i, \mathtt{eval}_i, \mathtt{calc}_i, \mathbb{E}_i, \eta_i, \mathtt{sel}_i, \mathtt{join}_i, \rightsquigarrow_i, \surd_i)$ $(i{=}1, 2)$, a Σ-homomorphism h from $\mathcal{M}_1$ to $\mathcal{M}_2$ consists of a mapping $h_{\mathbb{A}} : \mathbb{A}_1 \to \mathbb{A}_2$ and a mapping $h_{\mathbb{E}} : \mathbb{E}_1 \to \mathbb{E}_2$, such that:

- $h_{\mathbb{A}}(\mu_1(a)) \stackrel{\text{def}}{=} \mu_2(a)$ for each $a \in \mathcal{A}$
- $h_{\mathbb{E}}(\eta_1(e)) \stackrel{\text{def}}{=} \eta_2(e)$ for each $e \in \mathcal{E}$
- $h_{\mathbb{A}}(\mathtt{calc}_1(\mu_1(a))) \stackrel{\text{def}}{=} \mathtt{calc}_2(\mu_2(a))$ for each $a, a' \in \mathcal{A}$ and $E \in \mathcal{E}^*$, such that $\mathtt{calc}_1(\mu_1(a)) = (\mu_1(a'), \eta_1(E))$ and $\mathtt{calc}_2(\mu_2(a)) = (\mu_2(a'), \eta_2(E))$
- $h_{\mathbb{E}}(\mathtt{sel}_1(\eta_1(E))) \stackrel{\text{def}}{=} \mathtt{sel}_2(\eta_2(E))$ for each $E \in \mathcal{E}^*$, such that $\mathtt{sel}_1(\eta_1(E)) = (\eta_1(e), \eta_1(E_2))$ and $\mathtt{sel}_2(\eta_2(E)) = (\eta_2(e), \eta_2(E_2))$
- $h_{\mathbb{E}}(\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2))) \stackrel{\text{def}}{=} \mathtt{join}_2(\eta_2(E_1), \eta_2(E_2))$ for each $E_i \in \mathcal{E}^*$.
- $h_{\mathbb{A}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2))) \in \rightsquigarrow_1$ iff $(\mu_2(B_1), \mu_2(a), \mu_2(B_2)) \in \rightsquigarrow_2$ for each $a \in \mathcal{A}$, $B_i \in 2^{\mathcal{A}}$
- $h_{\mathbb{A}}((\mu_1(B), \mu_1(a))) \in \surd_1$ iff $(\mu_2(B), \mu_2(a)) \in \surd_2$ for each $a \in \mathcal{A}$, $B \in 2^{\mathcal{A}}$

Lemma 4. For any signatures, the Σ-models and Σ-homomorphisms define a category Mod(Σ). The points of the category are the Σ-models, its arrows are the Σ-homomorphisms.

Proof. Let $\Sigma = (\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$ with $K = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}, \mathtt{defer}, \mathcal{T}, s_{\text{start}})$ be a signature, let $\mathcal{M}_i = (\mathbb{A}_i, \mu_i, \mathtt{eval}_i, \mathtt{calc}_i, \mathbb{E}_i, \eta_i, \mathtt{sel}_i, \mathtt{join}_i, \rightsquigarrow_i, \surd_i)$ $(i{=}1..4)$ be Σ-models, and let $h_i : \mathcal{M}_i \to \mathcal{M}_{i+1}$ $(i{=}1..3)$ be Σ-homomorphisms, then:

- Σ-homomorphisms can be composed. We define the composition $h_2 \circ h_1$ as a mapping $h'_{\mathbb{A}} : \mathbb{A}_1 \to \mathbb{A}_3$ such that for each $a \in \mathcal{A}$ and $B_i \in 2^{\mathcal{A}}$:
  - $h'_{\mathbb{A}}(\mu_1(a)) = h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mu_1(a)))$
  - $h'_{\mathbb{A}}(\mathtt{calc}_1(a)) = h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mathtt{calc}_1(a)))$
  - $h'_{\mathbb{A}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2))) \in \rightsquigarrow_1$ iff $h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2)))) \in \rightsquigarrow_1$
  - $h'_{\mathbb{A}}((\mu_1(B), \mu_1(a))) \in \surd_1$ iff $h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}((\mu_1(B), \mu_1(a)))) \in \surd_1$

  and a mapping $h'_{\mathbb{E}} : \mathbb{E}_1 \to \mathbb{E}_3$ such that for each $e \in \mathcal{E}$ and $E_i \in \mathcal{E}^*$
  - $h'_{\mathbb{E}}(\eta_1(e)) = h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\eta_1(e)))$
  - $h'_{\mathbb{E}}(\mathtt{sel}_1(E)) = h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\mathtt{sel}_1(E)))$
  - $h'_{\mathbb{E}}(\mathtt{join}_1(E_1, E_2)) = h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\mathtt{join}_1(E_1, E_2)))$

  We have to prove that $h_2 \circ h_1$ is a Σ-homomorphism, i.e.
  - $h'_{\mathbb{A}}(\mu_1(a)) = \mu_3(a)$
  - $h'_{\mathbb{A}}(\mathtt{calc}_1(\mu_1(a))) = \mathtt{calc}_3(h'_{\mathbb{A}}(\mu_1(a)))$
  - $h'_{\mathbb{E}}(\eta_1(e)) = \eta_3(e)$
  - $h'_{\mathbb{E}}(\mathtt{sel}_1(\eta_1(E))) = \mathtt{sel}_3(h'_{\mathbb{E}}(\eta_1(E)))$
  - $h'_{\mathbb{E}}(\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2))) = \mathtt{join}_3(h'_{\mathbb{E}}(\eta_1(E_1)), h'_{\mathbb{E}}(\eta_1(E_2)))$
  - $h'_{\mathbb{A}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2))) \in \rightsquigarrow_1$ iff $(\mu_3(B_1), \mu_3(a), \mu_3(B_2)) \in \rightsquigarrow_3$
  - $h'_{\mathbb{A}}((\mu_1(B), \mu_1(a))) \in \surd_1$ iff $(\mu_3(B), \mu_3(a)) \in \surd_3$

$$\begin{aligned} h'_{\mathbb{A}}(\mu_1(a)) &= h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mu_1(a))) && \text{by definition of } h'_{\mathbb{A}} \\ &= h_{2_{\mathbb{A}}}(\mu_2(a)) && \text{by definition of } h_{1_{\mathbb{A}}} \\ &= \mu_3(a) && \text{by definition of } h_{2_{\mathbb{A}}} \end{aligned}$$

$$
\begin{aligned}
h'_{\mathbb{A}}(\texttt{calc}_1(\mu_1(a))) &= h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\texttt{calc}_1(\mu_1(a)))) && \text{by definition of } h'_{\mathbb{A}}\\
&= h_{2_{\mathbb{A}}}(\texttt{calc}_2(h_{1_{\mathbb{A}}}(\mu_1(a)))) && \text{by definition of } h_{1_{\mathbb{A}}}\\
&= \texttt{calc}_3(h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mu_1(a)))) && \text{by definition of } h_{2_{\mathbb{A}}}\\
&= \texttt{calc}_3(h'_{\mathbb{A}}(\mu_1(a))) && \text{by definition of } h'_{\mathbb{A}}
\end{aligned}
$$

$$
\begin{aligned}
h'_{\mathbb{E}}(\eta_1(e)) &= h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\eta_1(e))) && \text{by definition of } h'_{\mathbb{E}}\\
&= h_{2_{\mathbb{E}}}(\eta_2(e)) && \text{by definition of } h_{1_{\mathbb{E}}}\\
&= \eta_3(e) && \text{by definition of } h_{2_{\mathbb{E}}}
\end{aligned}
$$

$$
\begin{aligned}
h'_{\mathbb{E}}(\texttt{sel}_1(\eta_1(E))) &= h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\texttt{sel}_1(\eta_1(E)))) && \text{by definition of } h'_{\mathbb{E}}\\
&= h_{2_{\mathbb{E}}}(\texttt{sel}_2(\eta_2(E))) && \text{by definition of } h_{1_{\mathbb{E}}}\\
&= \texttt{sel}_3(\eta_3(E)) && \text{by definition of } h_{2_{\mathbb{E}}}\\
&= \texttt{sel}_3(h'_{\mathbb{E}}(\eta_1(E))) && \text{by definition of } h'_{\mathbb{E}}
\end{aligned}
$$

$$h'_{\mathbb{E}}(\texttt{join}_1(\eta_1(E_1),\eta_1(E_2)))$$

$$
\begin{aligned}
&= h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\texttt{join}_1(\eta_1(E_1),\eta_1(E_2))) && \text{by definition of } h'_{\mathbb{E}}\\
&= h_{2_{\mathbb{E}}}(\texttt{join}_2(\eta_2(E_1),\eta_2(E_2))) && \text{by definition of } h_{1_{\mathbb{E}}}\\
&= \texttt{join}_3(\eta_3(E_1),\eta_3(E_2)) && \text{by definition of } h_{2_{\mathbb{E}}}\\
&= \texttt{join}_3(h'_{\mathbb{E}}(\eta_1(E_1)),h'_{\mathbb{E}}(\eta_1(E_2))) && \text{by definition of } h'_{\mathbb{E}}
\end{aligned}
$$

$$h'_{\mathbb{A}}((\mu_1(B_1),\mu_1(a),\mu_1(B_2))) \in \rightsquigarrow_1$$

$$
\begin{aligned}
&\text{iff } h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}((\mu_1(B_1),\mu_1(a),\mu_1(B_2)))) \in \rightsquigarrow_1 && \text{by definition of } h'_{\mathbb{A}}\\
&\text{iff } h_{2_{\mathbb{A}}}((\mu_2(B_1),\mu_2(a),\mu_2(B_2))) \in \rightsquigarrow_2 && \text{by definition of } h_{1_{\mathbb{A}}}\\
&\text{iff } (\mu_3(B_1),\mu_3(a),\mu_3(B_2)) \in \rightsquigarrow_3 && \text{by definition of } h_{2_{\mathbb{A}}}
\end{aligned}
$$

$$h'_{\mathbb{A}}((\mu_1(B),\mu_1(a))) \in \surd_1$$

$$
\begin{aligned}
&\text{iff } h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}((\mu_1(B),\mu_1(a)))) \in \surd_1 && \text{by definition of } h'_{\mathbb{A}}\\
&\text{iff } h_{2_{\mathbb{A}}}((\mu_2(B),\mu_2(a))) \in \surd_2 && \text{by definition of } h_{1_{\mathbb{A}}}\\
&\text{iff } (\mu_3(B),\mu_3(a)) \in \surd_3 && \text{by definition of } h_{2_{\mathbb{A}}}
\end{aligned}
$$

– Composition of Σ-homomorphisms is associative, i.e., $(h_3 \circ h_2) \circ h_1 = h_3 \circ (h_2 \circ h_1)$.
  Let $h' \stackrel{\text{def}}{=} h_3 \circ h_2$ such that:
  - $h'_{\mathbb{A}}(\mu_2(a)) = \mu_4(a)$ for each $a \in \mathcal{A}$
  - $h'_{\mathbb{A}}(\texttt{calc}_2(\mu_2(a))) = \texttt{calc}_4(h'_{\mathbb{A}}(\mu_2(a)))$ for each $a \in \mathcal{A}$
  - $h'_{\mathbb{E}}(\eta_2(E)) = \eta_4(e)$ for each $e \in \mathcal{E}$
  - $h'_{\mathbb{E}}(\texttt{sel}_2(\eta_2(E))) = \texttt{sel}_4(\eta_4(E))$ for each $E \in \mathcal{E}^*$
  - $h'_{\mathbb{E}}(\texttt{join}_2(\eta_2(E_1),\eta_2(E_2))) = \texttt{join}_4(\eta_4(E_1),\eta_4(E_2))$ for each $E_i \in \mathcal{E}^*$.
  - $h'_{\mathbb{A}}((\mu_2(B_1),\mu_2(a),\mu_2(B_2))) \in \rightsquigarrow_4$ iff $(\mu_4(B_1),\mu_4(a),\mu_4(B_2)) \in \rightsquigarrow_4$
  - $h'_{\mathbb{A}}((\mu_2(B),\mu_2(a))) \in \surd_4$ iff $(\mu_4(B),\mu_4(a)) \in \surd_4$

  Also, let $h'' \stackrel{\text{def}}{=} h_2 \circ h_1$ such that:
  - $h''_{\mathbb{A}}(\mu_1(a)) = \mu_3(a)$ for each $a \in \mathcal{A}$

- $h''_{\mathbb{A}}(\texttt{calc}_1(\mu_1(a))) = \texttt{calc}_3(h''_{\mathbb{A}}(\mu_1(a)))$ for each $a \in \mathcal{A}$
- $h''_{\mathbb{E}}(\eta_1(E)) = \eta_3(e)$ for each $e \in \mathcal{E}$
- $h''_{\mathbb{E}}(\texttt{sel}_1(\eta_1(E))) = \texttt{sel}_3(\eta_3(E))$ for each $E \in \mathcal{E}^*$
- $h''_{\mathbb{E}}(\texttt{join}_1(\eta_1(E_1), \eta_1(E_2))) = \texttt{join}_3(\eta_3(E_1), \eta_3(E_2))$ for each $E_i \in \mathcal{E}^*$
- $h''_{\mathbb{A}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2))) \in \rightsquigarrow_3$ iff $(\mu_3(B_1), \mu_3(a), \mu_3(B_2)) \in \rightsquigarrow_3$ for each $a \in \mathcal{A}$, $B_i \in 2^{\mathcal{A}}$
- $h''_{\mathbb{A}}((\mu_1(B), \mu_1(a))) \in \surd_3$ iff $(\mu_3(B), \mu_3(a)) \in \surd_3$ for each $a \in \mathcal{A}$, $B \in 2^{\mathcal{A}}$

Then,

$$
\begin{aligned}
h'_{\mathbb{A}}(h_{1_{\mathbb{A}}}(\mu_1(a))) &= h'_{\mathbb{A}}(\mu_2(a)) && \text{by definition of } h_{1_{\mathbb{A}}} \\
&= \mu_4(a) && \text{by definition of } h'_{\mathbb{A}} \\
&= h_{3_{\mathbb{A}}}(\mu_3(a)) && \text{by definition of } h_{3_{\mathbb{A}}} \\
&= h_{3_{\mathbb{A}}}(h''_{\mathbb{A}}(\mu_1(a))) && \text{by definition of } h''_{\mathbb{A}}
\end{aligned}
$$

$$
\begin{aligned}
h'_{\mathbb{A}}(h_{1_{\mathbb{A}}}(\texttt{calc}_1(\mu_1(a)))) &= h'_{\mathbb{A}}(\texttt{calc}_2(h_{1_{\mathbb{A}}}(\mu_1(a)))) && \text{by definition of } h_{1_{\mathbb{A}}} \\
&= \texttt{calc}_4(h'_{\mathbb{A}}(h_{1_{\mathbb{A}}}(\mu_1(a)))) && \text{by definition of } h'_{\mathbb{A}} \\
&= \texttt{calc}_4(h_{3_{\mathbb{A}}}(h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mu_1(a))))) && \text{by definition of } h'_{\mathbb{A}} \\
&= h_{3_{\mathbb{A}}}(\texttt{calc}_3(h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mu_1(a))))) && \text{by definition of } h_{3_{\mathbb{A}}} \\
&= h_{3_{\mathbb{A}}}(\texttt{calc}_3(h''_{\mathbb{A}}(\mu_1(a)))) && \text{by definition of } h''_{\mathbb{A}} \\
&= h_{3_{\mathbb{A}}}(h''_{\mathbb{A}}(\texttt{calc}_1(\mu_1(a)))) && \text{by definition of } h''_{\mathbb{A}}
\end{aligned}
$$

$$
\begin{aligned}
h'_{\mathbb{E}}(h_{1_{\mathbb{E}}}(\eta_1(e))) &= h'_{\mathbb{E}}(\eta_2(e)) && \text{by definition of } h_{1_{\mathbb{E}}} \\
&= \eta_4(e) && \text{by definition of } h'_{\mathbb{E}} \\
&= h_{3_{\mathbb{E}}}(\eta_3(e)) && \text{by definition of } h_{3_{\mathbb{E}}} \\
&= h_{3_{\mathbb{E}}}(h''_{\mathbb{E}}(\eta_1(e))) && \text{by definition of } h''_{\mathbb{E}}
\end{aligned}
$$

$$
\begin{aligned}
h'_{\mathbb{E}}(h_{1_{\mathbb{E}}}(\texttt{sel}_1(\eta_1(E)))) &= h'_{\mathbb{E}}(\texttt{sel}_2(\eta_2(E))) && \text{by definition of } h_{1_{\mathbb{E}}} \\
&= \texttt{sel}_4(\eta_4(E)) && \text{by definition of } h'_{\mathbb{E}} \\
&= h_{3_{\mathbb{E}}}(\texttt{sel}_3(\eta_3(E))) && \text{by definition of } h_{3_{\mathbb{E}}} \\
&= h_{3_{\mathbb{E}}}(h''_{\mathbb{E}}(\texttt{sel}_1(\eta_1(E)))) && \text{by definition of } h''_{\mathbb{E}}
\end{aligned}
$$

$$
\begin{aligned}
&h'_{\mathbb{E}}(h_{1_{\mathbb{E}}}(\texttt{join}_1(\eta_1(E_1), \eta_1(E_2)))) \\
&\quad = h'_{\mathbb{E}}(\texttt{join}_2(\eta_2(E_1), \eta_2(E_2))) && \text{by definition of } h_{1_{\mathbb{E}}} \\
&\quad = \texttt{join}_4(\eta_4(E_1), \eta_4(E_2)) && \text{by definition of } h'_{\mathbb{E}} \\
&\quad = h_{3_{\mathbb{E}}}(\texttt{join}_3(\eta_3(E_1), \eta_3(E_2))) && \text{by definition of } h_{3_{\mathbb{E}}} \\
&\quad = h_{3_{\mathbb{E}}}(h''_{\mathbb{E}}(\texttt{join}_1(\eta_1(E_1), \eta_1(E_2)))) && \text{by definition of } h''_{\mathbb{E}}
\end{aligned}
$$

$$
\begin{aligned}
&h'_{\mathbb{A}}(h_{1_{\mathbb{A}}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2)))) \in \rightsquigarrow_4 \\
&\quad \text{iff } h'_{\mathbb{E}}((\mu_2(B_1), \mu_2(a), \mu_2(B_2))) \in \rightsquigarrow_4 && \text{by definition of } h_{1_{\mathbb{A}}} \\
&\quad \text{iff } (\mu_4(B_1), \mu_4(a), \mu_4(B_2)) \in \rightsquigarrow_4 && \text{by definition of } h'_{\mathbb{A}} \\
&\quad \text{iff } h_{3_{\mathbb{A}}}((\mu_3(B_1), \mu_3(a), \mu_3(B_2))) \in \rightsquigarrow_4 && \text{by definition of } h_{3_{\mathbb{A}}} \\
&\quad \text{iff } h_{3_{\mathbb{A}}}(h''_{\mathbb{A}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2)))) \in \rightsquigarrow_4 && \text{by definition of } h''_{\mathbb{A}}
\end{aligned}
$$

$h'_{\mathbb{A}}(h_{1_{\mathbb{A}}}((\mu_1(B), \mu_1(a)))) \in \sqrt{}_4$

iff $h'_{\mathbb{A}}((\mu_2(B), \mu_2(a))) \in \sqrt{}_4$	by definition of $h_{1_{\mathbb{A}}}$
iff $(\mu_4(B), \mu_4(a)) \in \sqrt{}_4$	by definition of $h'_{\mathbb{A}}$
iff $h_{3_{\mathbb{A}}}((\mu_3(B_1), \mu_3(a))) \in \sqrt{}_4$	by definition of $h_{3_{\mathbb{A}}}$
iff $h_{3_{\mathbb{A}}}(h''_{\mathbb{A}}((\mu_1(B_1), \mu_1(a)))) \in \sqrt{}_4$	by definition of $h''_{\mathbb{A}}$

- There exist an identity Σ-homomorphism $id_{\mathcal{M}_1} : \mathcal{M}_1 \to \mathcal{M}_1$ consisting of a mapping $h_{id_{\mathbb{A}}} : \mathbb{A}_1 \to \mathbb{A}_1$ and a mapping $h_{id_{\mathbb{E}}} : \mathbb{E}_1 \to \mathbb{E}_1$, such that:
  - $h_{id_{\mathbb{A}}}(\mu_1(a)) = \mu_1(a)$ for each $a \in \mathcal{A}$
  - $h_{id_{\mathbb{A}}}(\mathtt{calc}_1(\mu_1(a))) = \mathtt{calc}_1(\mu_1(a))$ for each $a \in \mathcal{A}$
  - $h_{id_{\mathbb{E}}}(\eta_1(e)) = \eta_1(e)$ for each $e \in \mathcal{E}$
  - $h_{id_{\mathbb{E}}}(\mathtt{sel}_1(\eta_1(E))) = \mathtt{sel}_1(\eta_1(E))$ for each $E \in \mathcal{E}^*$
  - $h_{id_{\mathbb{E}}}(\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2))) = \mathtt{join}_1(\eta_1(E_1), \eta_1(E_2))$ for each $E_i \in \mathcal{E}^*$
  - $h_{id_{\mathbb{A}}}((\mu_1(B_1), \mu_1(a), \mu_1(B_2))) \in \rightsquigarrow_1$ iff $(\mu_1(B_1), \mu_1(a), \mu_1(B_2)) \in \rightsquigarrow_1$ for each $a \in \mathcal{A}$, $B_i \in 2^{\mathcal{A}}$
  - $h_{id_{\mathbb{A}}}((\mu_1(B), \mu_1(a))) \in \sqrt{}_1$ iff $(\mu_1(B), \mu_1(a)) \in \sqrt{}_1$ for each $a \in \mathcal{A}$, $B \in 2^{\mathcal{A}}$

  It trivially holds that $id_{\mathcal{M}_1}$ is a Σ-homomorphism.

□

As the last ingredient, we define the notion of reduct. Any signature morphism $\sigma : \Sigma_1 \to \Sigma_2$ induces a mapping called the σ-reduct which allows using a Σ_2-model (and a Σ_2-homomorphism) for assigning an interpretation to the elements in Σ_1.

Definition 14 (σ-reduct). Let $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ $(i{=}1, 2)$ be signatures, let $\sigma : \Sigma_1 \to \Sigma_2$ be a signature morphism, and let $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \sqrt{})$ be a Σ_2-model.

(a) The reduct of $\mathcal{M}$ along σ, written $\mathcal{M}|_\sigma$, is the Σ_1-model $(\mathbb{A}, \mu|_\sigma, \mathtt{eval}|_\sigma, \mathtt{calc}|_\sigma, \mathbb{E}, \eta|_\sigma, \mathtt{sel}|_\sigma, \mathtt{join}|_\sigma, \rightsquigarrow|_\sigma, \sqrt{}|_\sigma)$ such that:
  - $\mu|_\sigma\ (a) \stackrel{\text{def}}{=} \mu(\sigma_{\mathcal{A}}(a))$ for each $a \in \mathcal{A}_1$
  - $\mathtt{eval}|_\sigma\ (H, s) \stackrel{\text{def}}{=} \{g \mid \sigma_{\mathcal{G}}(g) \in \mathtt{eval}(H, s)\}$ for each $H \in \mathcal{H}$ and $s \in S$
  - $\mathtt{calc}|_\sigma\ (\mu|_\sigma\ (a)) \stackrel{\text{def}}{=} \mathtt{calc}(\mu(\sigma_{\mathcal{A}}(a)))$ for each $a \in \mathcal{A}_1$
  - $\eta|_\sigma\ (e) \stackrel{\text{def}}{=} \eta(\sigma_{\mathcal{E}}(e))$ for each $e \in \mathcal{E}_1$
  - $\mathtt{sel}|_\sigma\ (\eta|_\sigma\ (E)) \stackrel{\text{def}}{=} \mathtt{sel}(\eta(\sigma_{\mathcal{E}}(E)))$ for each $E \in \mathcal{E}_1^*$
  - $\mathtt{join}|_\sigma\ (\eta|_\sigma\ (E_1), \eta|_\sigma\ (E_2)) \stackrel{\text{def}}{=}$
    $\mathtt{join}(\eta(\sigma_{\mathcal{E}}(E_1)), \eta(\sigma_{\mathcal{E}}(E_2)))$ for each $E_i \in \mathcal{E}_1^*$
  - $(\mu|_\sigma\ (B_1), \mu|_\sigma\ (a), \mu|_\sigma\ (B_2)) \in \rightsquigarrow|_\sigma$
    iff $(\mu(\sigma_{\mathcal{A}}(B_1), \mu(\sigma_{\mathcal{A}}(a)), \mu(\sigma_{\mathcal{A}}(B_2))) \in \rightsquigarrow$ for each $a \in \mathcal{A}_1$, $B_i \in 2^{\mathcal{A}_1}$
  - $(\mu|_\sigma\ (B), \mu|_\sigma\ (a)) \in \sqrt{}|_\sigma$ iff $(\mu(\sigma_{\mathcal{A}}(B), \mu(\sigma_{\mathcal{A}}(a))) \in \sqrt{}$
    for each $a \in \mathcal{A}_1$, $B \in 2^{\mathcal{A}_1}$

(b) Let $\mathcal{M}_i$ $(i{=}1, 2)$ be Σ_2-models, and $h : \mathcal{M}_1 \to \mathcal{M}_2$ be a Σ_2-homomorphism, the reduct $h|_\sigma$ of h along σ is the Σ_1-homomorphism from $\mathcal{M}_1|_\sigma$ to $\mathcal{M}_2|_\sigma$ defined by $h_{\mathbb{A}}|_\sigma = h_{\mathbb{A}}$ and $h_{\mathbb{E}}|_\sigma = h_{\mathbb{E}}$.

(c) The reduct of the functions μ and η for sets is defined as follows:

$$\mu|_\sigma\ (A) \stackrel{\text{def}}{=} \mu(\sigma_{\mathcal{A}}(A)) \text{ for each } A \in \mathcal{A}_1^*$$
$$\eta|_\sigma\ (E) \stackrel{\text{def}}{=} \eta(\sigma_{\mathcal{E}}(E)) \text{ for each } E \in \mathcal{E}_1^*$$

Lemma 5. The reduct of Σ-models and Σ-homomorphisms is a functor $\mathrm{Mod}_{\mathrm{SM}}(\sigma)$ from Σ_2-models to Σ_1-models (and Σ_2-homomorphisms to Σ_1-homomorphisms) for each signature morphism $\sigma : \Sigma_1 \to \Sigma_2$.

Proof. By definition, domain and codomain of the reduct of an Σ-homomorphism are the reduct of domain and codomain, respectively, of the Σ-homomorphism. We have now to prove that: (a) the reduct of a composition of two Σ-homomorphisms is the composition of the reducts of those Σ-homomorphisms, and (b) that the reduct of an identity Σ-homomorphisms is likewise an identity.

Let $\Sigma = (\mathcal{E}, \mathcal{A}, \mathcal{G}, K)$ with $K = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}, \texttt{defer}, \mathcal{T}, s_{\text{start}})$ be a signature, let $\mathcal{M}_i = (\mathbb{A}_i, \mu_i, \texttt{eval}_i, \texttt{calc}_i, \mathbb{E}_i, \eta_i, \texttt{sel}_i, \texttt{join}_i, \rightsquigarrow_i, \surd_i)$ $(i{=}1..3)$ be Σ-models, and let $h_i : \mathcal{M}_i \to \mathcal{M}_{i+1}$ $(i{=}1, 2)$ be Σ-homomorphisms.

(a) $(h_2 \circ h_1)|_\sigma = h_2|_\sigma \circ h_1|_\sigma$

$$\begin{aligned}
(h_2 \circ h_1)_{\mathbb{A}}|_\sigma\ (\mu_1(a)) &= (h_2 \circ h_1)_{\mathbb{A}}(\mu_1(a)) && \text{by definition of } (h_2 \circ h_1)|_\sigma \\
&= h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\mu_1(a))) && \text{by definition of } (h_2 \circ h_1) \\
&= h_{2\mathbb{A}}|_\sigma\ (h_{1\mathbb{A}}|_\sigma\ (\mu_1(a))) && \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\
&= (h_{2\mathbb{A}}|_\sigma \circ h_{1\mathbb{A}}|_\sigma)(\mu_1(a)) && \text{by definition of } h_2|_\sigma \circ h_1|_\sigma
\end{aligned}$$

$(h_2 \circ h_1)_{\mathbb{A}}|_\sigma\ (\texttt{calc}_1(\mu_1(a)))$

$$\begin{aligned}
&= (h_2 \circ h_1)_{\mathbb{A}}(\texttt{calc}_1(\mu_1(a))) && \text{by definition of } (h_2 \circ h_1)|_\sigma \\
&= h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}(\texttt{calc}_1(\mu_1(a)))) && \text{by definition of } (h_2 \circ h_1) \\
&= h_{2\mathbb{A}}|_\sigma\ (h_{1\mathbb{A}}|_\sigma\ (\texttt{calc}_1(\mu_1(a)))) && \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\
&= (h_{2\mathbb{A}}|_\sigma \circ h_{1\mathbb{A}}|_\sigma)(\texttt{calc}_1(\mu_1(a))) && \text{by definition of } h_2|_\sigma \circ h_1|_\sigma
\end{aligned}$$

$$\begin{aligned}
(h_2 \circ h_1)_{\mathbb{E}}|_\sigma\ (\eta_1(e)) &= (h_2 \circ h_1)_{\mathbb{E}}(\eta_1(e)) && \text{by definition of } (h_2 \circ h_1)|_\sigma \\
&= h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\eta_1(e))) && \text{by definition of } (h_2 \circ h_1) \\
&= h_{2\mathbb{E}}|_\sigma\ (h_{1\mathbb{E}}|_\sigma\ (\eta_1(e))) && \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\
&= (h_{2\mathbb{E}}|_\sigma \circ h_{1\mathbb{E}}|_\sigma)(\eta_1(e)) && \text{by definition of } h_2|_\sigma \circ h_1|_\sigma
\end{aligned}$$

$(h_2 \circ h_1)_{\mathbb{E}}|_\sigma\ (\texttt{sel}_1(\eta_1(E)))$

$$\begin{aligned}
&= (h_2 \circ h_1)_{\mathbb{E}}(\texttt{sel}_1(\eta_1(E))) && \text{by definition of } (h_2 \circ h_1)|_\sigma \\
&= h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\texttt{sel}_1(\eta_1(e)))) && \text{by definition of } (h_2 \circ h_1) \\
&= h_{2\mathbb{E}}|_\sigma\ (h_{1\mathbb{E}}|_\sigma\ (\texttt{sel}_1(\eta_1(e)))) && \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\
&= (h_{2\mathbb{E}}|_\sigma \circ h_{1\mathbb{E}}|_\sigma)(\texttt{sel}_1(\eta_1(E))) && \text{by definition of } h_2|_\sigma \circ h_1|_\sigma
\end{aligned}$$

$(h_2 \circ h_1)_{\mathbb{E}}|_\sigma\ (\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2)))$

$$\begin{array}{ll} = (h_2 \circ h_1)_{\mathbb{E}}(\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2))) & \text{by definition of } (h_2 \circ h_1)|_\sigma \\ = h_{2_{\mathbb{E}}}(h_{1_{\mathbb{E}}}(\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2)))) & \text{by definition of } (h_2 \circ h_1) \\ = h_{2\mathbb{E}}|_\sigma\ (h_{1\mathbb{E}}|_\sigma\ (\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2)))) & \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\ = (h_{2\mathbb{E}}|_\sigma \circ h_{1\mathbb{E}}|_\sigma)(\mathtt{join}_1(\eta_1(E_1), \eta_1(E_2))) & \text{by definition of } h_2|_\sigma \circ h_1|_\sigma \end{array}$$

$(h_2 \circ h_1)_{\mathbb{A}}|_\sigma\ ((B_1, a, B_2))\ \in \rightsquigarrow_3$

$$\begin{array}{ll} \text{iff } (h_2 \circ h_1)_{\mathbb{A}}((B_1, a, B_2))\ \in \rightsquigarrow_3 & \text{by definition of } (h_2 \circ h_1)|_\sigma \\ \text{iff } h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}((B_1, a, B_2)))\ \in \rightsquigarrow_3 & \text{by definition of } (h_2 \circ h_1) \\ \text{iff } h_{2\mathbb{A}}|_\sigma\ (h_{1\mathbb{A}}|_\sigma\ ((B_1, a, B_2)))\ \in \rightsquigarrow_3 & \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\ \text{iff } (h_{2\mathbb{A}}|_\sigma \circ h_{1\mathbb{A}}|_\sigma)((B_1, a, B_2))\ \in \rightsquigarrow_3 & \text{by definition of } h_2|_\sigma \circ h_1|_\sigma \end{array}$$

$(h_2 \circ h_1)_{\mathbb{A}}|_\sigma\ ((B, a))\ \in \surd_3$

$$\begin{array}{ll} \text{iff } (h_2 \circ h_1)_{\mathbb{A}}((B, a))\ \in \surd_3 & \text{by definition of } (h_2 \circ h_1)|_\sigma \\ \text{iff } h_{2_{\mathbb{A}}}(h_{1_{\mathbb{A}}}((B, a)))\ \in \surd_3 & \text{by definition of } (h_2 \circ h_1) \\ \text{iff } h_{2\mathbb{A}}|_\sigma\ (h_{1\mathbb{A}}|_\sigma\ ((B, a)))\ \in \surd_3 & \text{by definition of } h_2|_\sigma \text{ and } h_1|_\sigma \\ \text{iff } (h_{2\mathbb{A}}|_\sigma \circ h_{1\mathbb{A}}|_\sigma)((B, a))\ \in \surd_3 & \text{by definition of } h_2|_\sigma \circ h_1|_\sigma \end{array}$$

(b) Let $id_{\mathcal{M}_2}$ be an identity Σ_2-homomorphism, then $id_{\mathcal{M}_2}|_\sigma$ is an identity Σ_1-homomorphism since $id_{\mathcal{M}_2}|_\sigma = id_{\mathcal{M}_2}$ by definition of reduct of a homomorphism. $\square$

Lemma 6. There is a functor $\mathrm{Mod}_{\mathrm{SM}}$ giving a category Mod(Σ) of Σ-models (object in the category **Cat**) for each signature Σ (object in the category $\mathrm{Sign}_{\mathrm{SM}}$), as shown in Lemma 4, and a functor $\mathrm{Mod}_{\mathrm{SM}}(\sigma)$ (arrow in the category **Cat**) from Σ_2-models to Σ_1-models (and Σ_2-homomorphisms to Σ_1-homomorphisms) for each signature morphism $\sigma : \Sigma_1 \to \Sigma_2$ (arrow in the category $\mathrm{Sign}_{\mathrm{SM}}$), as shown in Lemma 5.

Proof. We have to prove that $\mathrm{Mod}_{\mathrm{SM}}$ is indeed a functor, i.e.: (a) domain and codomain of the image of an arrow are the images of domain and codomain, respectively, of the arrow, (b) composition is preserved, and (c) identities are preserved.

(a) By Lemma 5, the image of an arrow $\mathrm{Mod}_{\mathrm{SM}}(\sigma) : \mathrm{Mod}_{\mathrm{SM}}(\Sigma_2) \to \mathrm{Mod}_{\mathrm{SM}}(\Sigma_1)$ in the category **Cat** is the arrow $\sigma : \Sigma_2 \to \Sigma_1$ in the category $\mathrm{Sign}_{\mathrm{SM}}^{op}$. Also, by Lemma 4, the image of any object Mod(Σ) in the category **Cat** is a signature Σ in the category $\mathrm{Sign}_{\mathrm{SM}}$. Thus, domain and codomain of the image of an arrow are the images of domain and codomain, respectively, of the arrow.

(b) We have to prove that $\mathrm{Mod}_{\mathrm{SM}}(\sigma_2 \circ \sigma_1) = \mathrm{Mod}_{\mathrm{SM}}(\sigma_2) \circ \mathrm{Mod}_{\mathrm{SM}}(\sigma_1)$ for both, models and homomorphisms. Let $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ be signatures with $K_i = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}_i, \mathtt{defer}_i, \mathcal{T}_i, s_{\mathrm{start}})$ (i=1..3), let $\sigma_i : \Sigma_i \to \Sigma_{i+1}$ (i=1, 2) be signature morphisms, let $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \surd)$ be a Σ_3-model, and let h be a Σ_3-homomorphism. Then, we have to prove:

- $\mathcal{M}|_{\sigma_2 \circ \sigma_1} = (\mathcal{M}|_{\sigma_2})|_{\sigma_1}$.
  By Definition 14, $\mathcal{M}|_{\sigma_2}$ is the Σ_2-model
  $(\mathbb{A}, \mu|_{\sigma_2}, \mathtt{eval}|_{\sigma_2}, \mathtt{calc}|_{\sigma_2}, \mathbb{E}, \eta|_{\sigma_2}, \mathtt{sel}|_{\sigma_2}, \mathtt{join}|_{\sigma_2}, \rightsquigarrow|_{\sigma_2}, \surd|_{\sigma_2})$ such that:

- $\mu|_{\sigma_2}(a) = \mu(\sigma_2(a))$ for each $a \in \mathcal{A}_2$
- $\mathtt{eval}|_{\sigma_2}(H,s) = \{g \mid \sigma_{2\mathcal{G}}(g) \in \mathtt{eval}(H,s)\}$
- $\mathtt{calc}|_{\sigma_2}(\mu|_{\sigma_2}(a)) = \mathtt{calc}(\mu(\sigma_{2\mathcal{A}}(a)))$ for each $a \in \mathcal{A}_2$
- $\eta|_{\sigma_2}(e) = \eta(\sigma_2(e))$ for each $e \in \mathcal{E}_2$
- $\mathtt{sel}|_{\sigma_2}(\eta|_{\sigma_2}(E)) = \mathtt{sel}(\eta(\sigma_{2\mathcal{E}}(E)))$ for each $E \in \mathcal{E}_2^*$
- $\mathtt{join}|_{\sigma_2}(\eta|_{\sigma_2}(E_1), \eta|_{\sigma_2}(E_2)) = \mathtt{join}(\eta(\sigma_{2\mathcal{E}}(E_1)), \eta(\sigma_{2\mathcal{E}}(E_2)))$ for each $E_i \in \mathcal{E}_2^*$
- $(\mu|_{\sigma_2}(B_1), \mu|_{\sigma_2}(a), \mu|_{\sigma_2}(B_2)) \in \rightsquigarrow\!|_{\sigma_2}$
  iff $(\mu(\sigma_{2\mathcal{A}}(B_1)), \mu(\sigma_{2\mathcal{A}}(a)), \mu(\sigma_{2\mathcal{A}}(B_2))) \in \rightsquigarrow$ for each $a \in \mathcal{A}_2$, $B_i \in 2^{\mathcal{A}_2}$
- $(\mu|_{\sigma_2}(B), \mu|_{\sigma_2}(a)) \in \surd|_{\sigma_2}$ iff $(\mu(\sigma_{2\mathcal{A}}(B), \mu(\sigma_{2\mathcal{A}}(a))) \in \surd$
  for each $a \in \mathcal{A}_2$, $B \in 2^{\mathcal{A}_2}$

Then $(\mathcal{M}|_{\sigma_2})|_{\sigma_1}$ is the Σ_1-model $(\mathbb{A}, (\mu|_{\sigma_2})|_{\sigma_1}, (\mathtt{eval}|_{\sigma_2})|_{\sigma_1}, (\mathtt{calc}|_{\sigma_2})|_{\sigma_1}$ $, \mathbb{E}, (\eta|_{\sigma_2})|_{\sigma_1}, (\mathtt{sel}|_{\sigma_2})|_{\sigma_1}, (\mathtt{join}|_{\sigma_2})|_{\sigma_1}, (\rightsquigarrow\!|_{\sigma_2})|_{\sigma_1}, (\surd|_{\sigma_2})|_{\sigma_1})$ such that:

- $(\mu|_{\sigma_2})|_{\sigma_1}(a) = \mu((\sigma_2 \circ \sigma_1)(a))$ for each $a \in \mathcal{A}_1$
- $(\mathtt{eval}|_{\sigma_2})|_{\sigma_1}(H,s) = \{g \mid (\sigma_{2\mathcal{G}} \circ \sigma_{1\mathcal{G}})(g) \in \mathtt{eval}(H,s)\}$
- $(\mathtt{calc}|_{\sigma_2})|_{\sigma_1}((\mu|_{\sigma_2})|_{\sigma_1}(a)) = \mathtt{calc}(\mu((\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(a)))$ for each $a \in \mathcal{A}_1$
- $(\eta|_{\sigma_2})|_{\sigma_1}(e) = \eta((\sigma_2 \circ \sigma_1)(e))$ for each $e \in \mathcal{E}_1$
- $(\mathtt{sel}|_{\sigma_2})|_{\sigma_1}((\eta|_{\sigma_2})|_{\sigma_1}(E)) = \mathtt{sel}(\eta((\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(E)))$ for each $E \in \mathcal{E}_1^*$
- $(\mathtt{join}|_{\sigma_2})|_{\sigma_1}((\eta|_{\sigma_2})|_{\sigma_1}(E_1), (\eta|_{\sigma_2})|_{\sigma_1}(E_2)) =$
  $\mathtt{join}(\eta((\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(E_1)), \eta((\sigma_{2\mathcal{E}} \circ \sigma_{1\mathcal{E}})(E_2)))$ for each $E_i \in \mathcal{E}_1^*$
- $((\mu|_{\sigma_2})|_{\sigma_1}(B_1), (\mu|_{\sigma_2})|_{\sigma_1}(a), (\mu|_{\sigma_2})|_{\sigma_1}(B_2)) \in (\rightsquigarrow\!|_{\sigma_2})|_{\sigma_1}$
  iff $(\mu((\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(B_1)), \mu((\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(a)), \mu((\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(B_2))) \in \rightsquigarrow$ for each $a \in \mathcal{A}_1$, $B_i \in 2^{\mathcal{A}_1}$
- $((\mu|_{\sigma_2})|_{\sigma_1}(B), (\mu|_{\sigma_2})|_{\sigma_1}(a)) \in (\surd|_{\sigma_2})|_{\sigma_1}$ iff $(\mu((\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(B)), \mu((\sigma_{2\mathcal{A}} \circ \sigma_{1\mathcal{A}})(a))) \in \surd$
  for each $a \in \mathcal{A}_1$, $B \in 2^{\mathcal{A}_1}$

and this is equal to $\mathcal{M}|_{\sigma_2 \circ \sigma_1}$.

- $h|_{\sigma_2 \circ \sigma_1} = (h|_{\sigma_2})|_{\sigma_1}$.
  By Definition 14, $(h|_{\sigma_2})|_{\sigma_1} = h|_{\sigma_2} = h(a) = h|_{\sigma_2 \circ \sigma_1}$.

(c) Let $id_\sigma : \Sigma \rightarrow \Sigma$ be an identity signature morphism (defined in Lemma 1). We have to prove that $\mathrm{Mod}_{\mathrm{SM}}(id_\sigma)$ is an identity functor, i.e., if it is composed by the identity reduct of Σ-models and the identity reduct of Σ-homomorphisms.

- By Definition 14, for any Σ-model $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}_1, \eta_1, \mathtt{sel}_1, \mathtt{join}_1, \rightsquigarrow_1$ $, \surd_1)$, $\mathcal{M}|_{id_\sigma}$ is the Σ-model $(\mathbb{A}, \mu|_{id_\sigma}, \mathtt{eval}|_{id_\sigma}, \mathtt{calc}|_{id_\sigma}, \mathbb{E}_1, \eta|_{id_\sigma}, \mathtt{sel}|_{id_\sigma}$ $, \mathtt{join}|_{id_\sigma}, \rightsquigarrow\!|_{id_\sigma}, \surd|_{id_\sigma})$ such that:
  - $\mu|_{id_\sigma}(a) = \mu(id_\sigma(a)) \in \mathbb{A}$ for each $a \in \mathcal{A}$
  - $\mathtt{eval}|_{id_\sigma}(H,s) = \{g \mid id_{\sigma\mathcal{G}}(g) \in \mathtt{eval}(H,s)\}$
  - $\mathtt{calc}|_{id_\sigma}(\mu|_{id_\sigma}(a)) = \mathtt{calc}(\mu(id_{\sigma\mathcal{A}}(\alpha)))$ for each $a \in \mathcal{A}$
  - $\eta|_{id_\sigma}(e) = \eta(id_\sigma(e)) \in \mathbb{E}$ for each $e \in \mathcal{E}$
  - $\mathtt{sel}|_{id_\sigma}(\eta|_{id_\sigma}(E)) = \mathtt{sel}(\eta(id_{\sigma\mathcal{E}}(E)))$ for each $E \in \mathcal{E}^*$
  - $\mathtt{join}|_{id_\sigma}(\eta|_{id_\sigma}(E_1), \eta|_{id_\sigma}(E_2)) = \mathtt{join}(\eta(id_{\sigma\mathcal{E}}(E_1), \eta(id_{\sigma\mathcal{E}}(E_2)))$ for each $E_i \in \mathcal{E}^*$
  - $(\mu|_{id_\sigma}(B_1), \mu|_{id_\sigma}(a), \mu|_{id_\sigma}(B_2)) \in \rightsquigarrow\!|_{id_\sigma}$
    iff $(\mu(id_{\sigma\mathcal{A}}(B_1)), \mu(id_{\sigma\mathcal{A}}(a)), \mu(id_{\sigma\mathcal{A}}(B_2))) \in \rightsquigarrow$ for each $a \in \mathcal{A}$, $B_i \in 2^{\mathcal{A}}$

- $(\mu|_{id_\sigma}\ (B), \mu|_{id_\sigma}\ (a)) \in \surd|_{id_\sigma}$ iff $(\mu(id_{\sigma\mathcal{A}}(B), \mu(id_{\sigma\mathcal{A}}(a))) \in \surd$
  for each $a \in \mathcal{A}$, $B \in 2^{\mathcal{A}}$

  Finally, by the definition of id_σ, $\mathcal{M}|_{id_\sigma} = \mathcal{M}$, thus $\_|_{id_\sigma}$ is the identity reduct of Σ-models.
- By Definition 14, given a Σ-model $\mathcal{M}_1 = (\mathbb{A}_1, \mu_1, \texttt{eval}_1, \texttt{calc}_1, \mathbb{E}_1, \eta_1, \texttt{sel}_1, \texttt{join}_1, \rightsquigarrow_1, \surd_1)$, for any Σ-homomorphism $h : \mathcal{M}_1 \to \mathcal{M}_2$, the reduct $h|_{id_\sigma}$ is defined by $h|_{id_\sigma} = h$. Now, since $\mathcal{M}|_{id_\sigma} = \mathcal{M}$, we have that $\_|_{id_\sigma}$ is the identity reduct of Σ-homomorphisms.

$\square$

At this point we have almost every component of our institution SM. The satisfaction condition will be introduced in the next section.

4 An Institution for UML 2.0 State Machine

In order to define the satisfaction relation we proceed in the same way as in [7]. First, we define an auxiliary satisfaction condition which deals with processing simple input events. For this purpose we use the semantics of transitions between configurations defined in [6].

Definition 15 (Auxiliary Semantics). Given a signature Σ, and a Σ-model $\mathcal{M}$, the semantics of transitions between configurations is given in terms of an event-labelled transition system $(\llbracket C \rrbracket \, \mathcal{L}, \rightarrowtail_{\Sigma}, s)$ where

- $\llbracket C \rrbracket$ is the set of states representing the possible interpreted configurations
- $\mathcal{L} = (\mathbb{E} \cup \{\tau\} \cup \{\mathtt{defer}(e) \mid e \in \mathbb{E}\}) \times \mathbb{A}^*$ is the set of labels
- $\rightarrowtail_{\Sigma,\mathcal{M}} \; \subseteq \llbracket C \rrbracket \; \times \mathcal{L} \times \llbracket C \rrbracket$ is the transition relation
- s is the start state, i.e., an initial interpreted configuration

The transition relation is obtained by the derivation rules given in Table 1. In the table, $f[x \mapsto v]$ denotes the function that is everywhere equal to f except on x (if it is in its range) where it is equal to v. This notation is straightforwardly extended to sequences $(x_i \mapsto v_i)^{i \in I}$. We write $\llbracket \mathcal{C}_1 \rrbracket \overset{l}{\rightarrowtail} \llbracket \mathcal{C}_2 \rrbracket$ instead of $(\llbracket \mathcal{C}_1 \rrbracket, l, \llbracket \mathcal{C}_2 \rrbracket) \in \rightarrowtail_{\Sigma,\mathcal{M}}$.

As explained in [6]: Rule `do-act` describes an atomic action execution of a do action of an active composite state. Rule `cur-act` describes the next atomic action execution of the atomic action sequence currently being executed. Rules `next-tr-1` and `next-tr-2` selects from the transitions currently being fired a new atomic action sequence that will be executed next. In `next-tr-2`, contrary to `next-tr-1`, the target of the transition has to be activated after the execution of the action sequence, since this completes the firing of the transition. Rule `next-com` determines the next non-internal compound transition which will be fired, that is only possible if the previous fired compound transition is completed. Rule `next-int` determines the next internal transition which will be fired, that is only possible if the previously fired compound transition is finished. Rule `next-completion` determines the next trigger-free compound transition that will be fired. This is only possible if the previous set of compound transitions is completely executed. Rule `next-trigger` determines the next compound transition triggered by an event that will be fired. Rule `defer` describes the deferral of events. Rule `a-fin` activates a final state directly contained in a region and resets the history information of all its subregions. Rule `a-ch` activates a choice pseudostate, where it is immediately determined which of its currently enabled outgoing transitions is fired. Rule `a-en` activates an entry pseudostate. Finally, Rule `acti-ex-1` and `acti-ex-2` deal with the activation of exit pseudostates, that will not happen if there is a transition with a source below the exit pseudostate, w.r.t. state hierarchy, which has not yet been completely executed.

We can now define and prove the auxiliary satisfaction condition relating configurations and signature morphisms.

Table 1. Auxiliary Semantics

$$\text{do-act}\ \frac{s \in S \cap \mathcal{S}_{\mathrm{com}} \qquad \texttt{do}^{\mathcal{M}}(s) \neq \texttt{skip} \\ \texttt{calc}(\texttt{do}^{\mathcal{M}}(s)) = (\alpha'^{\mathcal{M}}, E) \qquad \alpha^{\mathcal{M}} = \texttt{skip} \Rightarrow \ddot{s} = \emptyset}{(S, \texttt{do}^{\mathcal{M}}, H, \alpha^{\mathcal{M}}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/E}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}[s \mapsto \alpha'^{\mathcal{M}}], H, \alpha^{\mathcal{M}}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{cur-act}\ \frac{\alpha^{\mathcal{M}} \neq \texttt{skip} \qquad \texttt{calc}(\alpha^{\mathcal{M}}) = (\alpha'^{\mathcal{M}}, E)}{(S, \texttt{do}^{\mathcal{M}}, H, \alpha^{\mathcal{M}}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/E}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \alpha'^{\mathcal{M}}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{next-tr-1}\ \frac{(B, \ddot{s}) \in \beta^{\mathcal{M}} \qquad B \overset{\alpha^{\mathcal{M}}}{\rightsquigarrow} B' \qquad \beta'^{\mathcal{M}} = \{(B', \ddot{s})\} \cup \beta^{\mathcal{M}} \setminus \{(B, \ddot{s})\}}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \alpha^{\mathcal{M}}, \emptyset, \beta'^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{next-tr-2}\ \frac{(B, \ddot{s}) \in \beta^{\mathcal{M}} \qquad (B, \alpha^{\mathcal{M}}) \in \surd \qquad \beta'^{\mathcal{M}} = \beta^{\mathcal{M}} \setminus \{(B, \ddot{s})\}}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \alpha^{\mathcal{M}}, \ddot{s}, \beta'^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{next-com}\ \frac{T'^{\mathcal{M}} \in \ddot{T}^{\mathcal{M}} \setminus \mathcal{T}^{\mathcal{M}}_{\mathrm{int}} \qquad \beta^{\mathcal{M}} = \{(\texttt{skip}, \pi_{\mathrm{sor}}(t) \mid t \in T'^{\mathcal{M}} \wedge \forall t' \in T'^{\mathcal{M}} : \\ \neg(\texttt{stateOf}(\pi_{\mathrm{sor}}(t)) \succ \texttt{stateOf}(\pi_{\mathrm{sor}}(t')))\}}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta^{\mathcal{M}}, T'^{\mathcal{M}}, \ddot{T}^{\mathcal{M}} \setminus \{T'^{\mathcal{M}}\})}$$

$$\text{next-int}\ \frac{\{t\} \in \ddot{T}^{\mathcal{M}} \cap \mathcal{T}^{\mathcal{M}}_{\mathrm{int}} \qquad \beta^{\mathcal{M}} = \{(\pi_{\mathrm{act}}(t), \emptyset)\}}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta^{\mathcal{M}}, \{t\}, \ddot{T}^{\mathcal{M}} \setminus \{\{t\}\})}$$

$$\text{next-completion}\ \frac{\ddot{T}^{\mathcal{M}} \in \texttt{Fireable}_{K, [\![\mathcal{C}_1]\!], \mathcal{M}, \tau} \qquad T'^{\mathcal{M}} \in \ddot{T}^{\mathcal{M}}}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \emptyset) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \{T'^{\mathcal{M}}\})}$$

$$\text{next-trigger}\ \frac{\texttt{Fireable}_{K, [\![\mathcal{C}_1]\!], \mathcal{M}, \tau} = \emptyset \qquad \ddot{T}^{\mathcal{M}} \in \texttt{Fireable}_{K, [\![\mathcal{C}_1]\!], \mathcal{M}, e}}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \emptyset) \overset{e/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{defer}\ \frac{\eta(e_1) = e \qquad \texttt{defer}(e_1) \cap S \neq \emptyset \qquad \texttt{Fireable}_{K, [\![\mathcal{C}_1]\!], \mathcal{M}, e} = \emptyset}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \emptyset) \overset{\texttt{defer}(e)/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \emptyset, T^{\mathcal{M}}, \emptyset)}$$

$$\text{a-fin}\ \frac{\ddot{s} \in \mathcal{S}_{\mathrm{fin}} \qquad H' = (H[(r \mapsto \bot)^{r \in \mathcal{R} \cap \uparrow\{\texttt{regOf}(\ddot{s})\}}])}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S \cup \{\ddot{s}\}, \texttt{do}^{\mathcal{M}}, H', \texttt{skip}, \emptyset, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{a-ch}\ \frac{\ddot{s} \in \mathcal{S}_{\mathrm{choice}} \qquad t \in \mathcal{T}^{\mathcal{M}} \qquad \pi_{\mathrm{sor}}(t) = \ddot{s} \qquad \pi_{\mathrm{gua}}(t) \in \texttt{eval}(H, \pi_{\mathrm{tar}}(t))}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta^{\mathcal{M}} \cup \{(\pi_{\mathrm{act}}(t), \pi_{\mathrm{tar}}(t))\}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{a-en}\ \frac{\ddot{s} \in \mathcal{S}_{\mathrm{entry}} \qquad \texttt{do}'^{\mathcal{M}} = \texttt{do}^{\mathcal{M}}[\texttt{stateOf}(\ddot{s}) \mapsto \mu(\texttt{doAct}(\texttt{stateOf}(\ddot{s})))] \\ f : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T} \qquad \beta' = \beta \cup \bigcup_{r \in \texttt{dsr}(\texttt{stateOf}(\ddot{s}))} \{(\pi_{\mathrm{act}}(f(r)^{\mathcal{M}}), \pi_{\mathrm{tar}}(f(r)^{\mathcal{M}}))\} \\ \forall r \in \texttt{dsr}(\texttt{stateOf}(\ddot{s})) : \pi_{\mathrm{sor}}(f(r)^{\mathcal{M}}) = \ddot{s} \wedge \pi_{\mathrm{gua}}(f(r)^{\mathcal{M}}) \in \texttt{eval}(H, \pi_{\mathrm{tar}}(f(r)^{\mathcal{M}}))}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S \cup \{\texttt{stateOf}(\ddot{s})\}, \texttt{do}'^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta'^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{a-ex-1}\ \frac{\ddot{s} \in \mathcal{S}_{\mathrm{exit}} \qquad \forall (B, \ddot{s}') \in \beta^{\mathcal{M}} : \ddot{s}' \neq \ddot{s} \qquad \forall s \in S : \neg(\texttt{stateOf}(\ddot{s}) \succ s) \qquad t \in T^{\mathcal{M}} \\ \pi_{\mathrm{sor}}(t) = \ddot{s} \qquad \beta'^{\mathcal{M}} = \beta^{\mathcal{M}} \cup \{(\pi_{\mathrm{act}}(t), \pi_{\mathrm{tar}}(t))\} \qquad H' = H[\texttt{regOf}(\ddot{s}) \mapsto \texttt{stateOf}(\ddot{s})]}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S \setminus \{\texttt{stateOf}(\ddot{s})\}, \texttt{do}^{\mathcal{M}}, H', \texttt{skip}, \emptyset, \beta'^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

$$\text{a-ex-2}\ \frac{\ddot{s} \in \mathcal{S}_{\mathrm{exit}} \qquad \exists B : (B, \ddot{s}) \in \beta^{\mathcal{M}} \vee \exists s \in S : \texttt{stateOf}(\ddot{s}) \succ s}{(S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \ddot{s}, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}}) \overset{\tau/\emptyset}{\rightarrowtail} (S, \texttt{do}^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \beta^{\mathcal{M}}, T^{\mathcal{M}}, \ddot{T}^{\mathcal{M}})}$$

Theorem 1 (Auxiliary Satisfaction Condition). Given signatures Σ_1 and Σ_2, a signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M}$, a pair of Σ_1-configurations $(\mathcal{C}_1, \mathcal{C}_2)$, an event $e \in \mathbb{E}_1 \cup \{\tau\}$ and a set of events $E \in \mathbb{E}_1^*$, the following satisfaction condition holds.

$$[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{e/E}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} \text{ iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{e/E}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}}$$

Proof. Lets prove the satisfaction condition by cases on the derivation rules of Table 1. First, let:

$$\begin{aligned}
\Sigma_i &= (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)\ (i{=}1,2) \\
K_i &= ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}_i, \mathtt{defer}_i, \mathcal{T}_i, s_{\text{start}})\ (i{=}1,2) \\
\mathcal{M} &= (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \surd)
\end{aligned}$$

Case `do-act`.

$$\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/E}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \alpha^{\mathcal{M}|_\sigma}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}[s \mapsto \alpha'^{\mathcal{M}|_\sigma}], H, \alpha^{\mathcal{M}|_\sigma}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } s \in S \cap \mathcal{S}_{\text{com}} & \\
\quad\quad \text{and } \mathtt{do}^{\mathcal{M}|_\sigma}(s) \neq \mathtt{skip} & \\
\quad\quad \text{and } \mathtt{calc}|_\sigma\ (\mathtt{do}^{\mathcal{M}|_\sigma}(s)) = (\alpha'^{\mathcal{M}|_\sigma}, E) & \\
\quad\quad \text{and } \alpha^{\mathcal{M}|_\sigma} = \mathtt{skip} \Rightarrow \ddot{s} = \emptyset & \text{by Table 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = & \\
\quad\quad (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = & \\
\quad\quad (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}[s \mapsto \sigma_{\mathcal{A}}(\alpha')^{\mathcal{M}}], H, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } s \in S \cap \mathcal{S}_{\text{com}} & \\
\quad\quad \text{and } \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}(s) \neq \mathtt{skip} & \text{by lem. 3.3} \\
\quad\quad \text{and } \mathtt{calc}(\sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}(s)) = (\sigma_{\mathcal{A}}(\alpha')^{\mathcal{M}}, E) & \text{by } \mathbf{(1)} \\
\quad\quad \text{and } \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}} = \mathtt{skip} \Rightarrow \ddot{s} = \emptyset & \text{by lem. 3.1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/E}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}$$

(1) First we have that $\mathtt{calc}|_\sigma\ (\mu|_\sigma\ (\mathtt{do}(s))) = \mathtt{calc}(\mu(\sigma_{\mathtt{do}}(\mathtt{do})(s)))$ by definition of σ-reduct and definition of $\sigma_{\mathtt{do}}$. Then, we have that $\mu|_\sigma\ (\alpha') = \mu(\sigma_{\mathcal{A}}(\alpha'))$ by definition of σ-reduct. Thus, $\mathtt{calc}|_\sigma\ (\mu|_\sigma\ (\mathtt{do}(s))) = (\mu|_\sigma\ (\alpha'), E)$ iff $\mathtt{calc}(\mu(\sigma_{\mathtt{do}}(\mathtt{do})(s))) = (\mu(\sigma_{\mathcal{A}}(\alpha'), E)$. Finally, we conclude that $\mathtt{calc}|_\sigma\ (\mathtt{do}^{\mathcal{M}|_\sigma}(s)) = (\alpha'^{\mathcal{M}|_\sigma}, E)$ iff $\mathtt{calc}(\sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}(s)) = (\sigma_{\mathcal{A}}(\alpha')^{\mathcal{M}}, E)$.

Case `cur-act`.

$[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/E}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma}$

iff $[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \alpha^{\mathcal{M}|_\sigma}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma})$
and $[\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \alpha'^{\mathcal{M}|_\sigma}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma})$
and $\alpha^{\mathcal{M}|_\sigma} \neq \mathtt{skip}$
and $\mathtt{calc}|_\sigma\, (\alpha^{\mathcal{M}|_\sigma}) = (\alpha'^{\mathcal{M}|_\sigma}, E)$ — by Table 1

iff $[\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}})$ — by lem. 3.4
and $[\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} =$
$(S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\alpha')^{\mathcal{M}}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}})$ — by lem. 3.4
and $\sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}} \neq \mathtt{skip}$ — by lem. 3.1
and $\mathtt{calc}(\sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}) = (\sigma_{\mathcal{A}}(\alpha')^{\mathcal{M}}, E)$ — by **(1)**

iff $[\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/E}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}}$ — by Table 1

(1) As proved in the case `do-act`.**(2)**, it holds that $\mathtt{calc}|_\sigma\, (\alpha^{\mathcal{M}|_\sigma}) = (\alpha'^{\mathcal{M}|_\sigma}, E)$ iff $\mathtt{calc}(\sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}) = (\sigma_{\mathcal{A}}(\alpha')^{\mathcal{M}}, E)$.

Case `next-tr-1`.

$[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma}$

iff $[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma})$
and $[\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \alpha^{\mathcal{M}|_\sigma}, \emptyset, \beta'^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma})$
and $(B^{\mathcal{M}|_\sigma}, \ddot{s}) \in \beta^{\mathcal{M}|_\sigma}$
and $B^{\mathcal{M}|_\sigma} \overset{\alpha^{\mathcal{M}|_\sigma}}{\rightsquigarrow}|_\sigma\; B'^{\mathcal{M}|_\sigma}$
and $\beta'^{\mathcal{M}|_\sigma} = \{(B'^{\mathcal{M}|_\sigma}, \ddot{s})\} \cup \beta^{\mathcal{M}|_\sigma} \setminus \{(B^{\mathcal{M}|_\sigma}, \ddot{s})\}$ — by Table 1

iff $[\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\mathtt{skip})^{\mathcal{M}}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}})$ — by lem. 3.4
and $[\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}, \emptyset, \sigma_\beta(\beta')^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}})$ — by lem. 3.4
and $(\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \ddot{s}) \in \sigma_\beta(\beta)^{\mathcal{M}}$ — by lem. 3.1
and $\sigma_{\mathcal{A}}(B)^{\mathcal{M}} \overset{\sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}}{\rightsquigarrow} \sigma_{\mathcal{A}}(B')^{\mathcal{M}}$ — by def. of $\rightsquigarrow$
and $\sigma_\beta(\beta')^{\mathcal{M}} = \{(\sigma_{\mathcal{A}}(B')^{\mathcal{M}}, \ddot{s})\} \cup \sigma_\beta(\beta)^{\mathcal{M}} \setminus \{(\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \ddot{s})\}$ — by lem. 3.1

iff $[\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}}$ — by Table 1

Case `next-tr-2`.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad\text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad\text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \alpha^{\mathcal{M}|_\sigma}, \ddot{s}, \beta'^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad\text{and } (B^{\mathcal{M}|_\sigma}, \ddot{s}) \in \beta^{\mathcal{M}|_\sigma} & \\
\quad\quad\text{and } (B^{\mathcal{M}|_\sigma}, \alpha^{\mathcal{M}|_\sigma}) \in \surd|_\sigma & \\
\quad\quad\text{and } \beta'^{\mathcal{M}|_\sigma} = \beta^{\mathcal{M}|_\sigma} \setminus \{(B^{\mathcal{M}|_\sigma}, \ddot{s})\} & \text{by Table 1} \\
\quad\text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad\text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}, \ddot{s}, \sigma_\beta(\beta')^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad\text{and } (\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \ddot{s}) \in \sigma_\beta(\beta)^{\mathcal{M}} & \text{by lem. 3.1} \\
\quad\quad\text{and } (\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \sigma_{\mathcal{A}}(\alpha)^{\mathcal{M}}) \in \surd & \text{by def. of } \surd_\sigma \\
\quad\quad\text{and } \sigma_\beta(\beta')^{\mathcal{M}} = \sigma_\beta(\beta)^{\mathcal{M}} \setminus \{(\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \ddot{s})\} & \text{by lem. 3.1} \\
\quad\text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

Case `next-com`.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad\text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad\text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma}, T'^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma} \setminus \{T'^{\mathcal{M}|_\sigma}\}) & \\
\quad\quad\text{and } T'^{\mathcal{M}|_\sigma} \in \ddot{T}^{\mathcal{M}|_\sigma} \setminus \mathcal{T}_{\text{int}_1}^{\mathcal{M}|_\sigma} & \\
\quad\quad\text{and } \beta^{\mathcal{M}|_\sigma} = \{(\mathtt{skip}, \pi_{\text{sor}}(t^{\mathcal{M}|_\sigma}) \mid t^{\mathcal{M}|_\sigma} \in T'^{\mathcal{M}|_\sigma} \wedge \forall t'^{\mathcal{M}|_\sigma} \in T'^{\mathcal{M}|_\sigma} : & \\
\quad\quad\quad\quad \neg(\mathtt{stateOf}(\pi_{\text{sor}}(t^{\mathcal{M}|_\sigma})) \succ \mathtt{stateOf}(\pi_{\text{sor}}(t'^{\mathcal{M}|_\sigma})))\} & \text{by Table 1} \\
\quad\text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad\text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = & \\
\quad\quad (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T')^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}} \setminus \{\sigma_T(T')^{\mathcal{M}}\}) & \text{by lem. 3.4} \\
\quad\quad\text{and } \sigma_T(T')^{\mathcal{M}} \in \sigma_T(\ddot{T})^{\mathcal{M}} \setminus \mathcal{T}_{\text{int}_2}^{\mathcal{M}} & \text{by lem. 3.2} \\
\quad\quad\text{and } \sigma_\beta(\beta)^{\mathcal{M}} = \{(\mathtt{skip}, \pi_{\text{sor}}(\sigma_T(t)^{\mathcal{M}})) \mid & \\
\quad\quad\quad\quad \sigma_T(t)^{\mathcal{M}} \in \sigma_T(T')^{\mathcal{M}} \wedge \forall \sigma_T(t')^{\mathcal{M}} \in \sigma_T(T')^{\mathcal{M}} : & \\
\quad\quad\quad\quad \neg(\mathtt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)^{\mathcal{M}})) \succ \mathtt{stateOf}(\pi_{\text{sor}}(\sigma_T(t')^{\mathcal{M}})))\} & \text{by } \textbf{(1)} \\
\quad\text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

(1) As proved before, $T^{\mathcal{M}|_\sigma} = \sigma_T(T)^{\mathcal{M}}$ for any set of transitions T. It also holds $\beta^{\mathcal{M}|_\sigma} = \sigma_\beta(\beta)^{\mathcal{M}}$. Finally, since states do not change after signature morphisms, we conclude that the premise holds.

Case `next-int`.

$$
\begin{array}{ll}
\llbracket \mathcal{C}_1 \rrbracket_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1, \mathcal{M}|_\sigma} \llbracket \mathcal{C}_2 \rrbracket_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } \llbracket \mathcal{C}_1 \rrbracket_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } \llbracket \mathcal{C}_2 \rrbracket_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma}, \{t^{\mathcal{M}|_\sigma}\}, \ddot{T}^{\mathcal{M}|_\sigma} \setminus \{\{t^{\mathcal{M}|_\sigma}\}\}) & \\
\quad\quad \text{and } \{t^{\mathcal{M}|_\sigma}\} \in \ddot{T}^{\mathcal{M}|_\sigma} \cap \mathcal{T}_{\mathrm{int}_1}^{\mathcal{M}|_\sigma} & \\
\quad\quad \text{and } \beta^{\mathcal{M}|_\sigma} = \{(\pi_{\mathrm{act}}(t^{\mathcal{M}|_\sigma}), \emptyset)\} & \text{by Table 1} \\
\quad \text{iff } \llbracket \sigma(\mathcal{C}_1) \rrbracket_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } \llbracket \sigma(\mathcal{C}_2) \rrbracket_{\mathcal{M}} = & \\
\quad\quad (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}}, \{\sigma_T(t)^{\mathcal{M}}\}, \sigma_T(\ddot{T})^{\mathcal{M}} \setminus \{\{\sigma_T(t)^{\mathcal{M}}\}\}) & \text{by lem. 3.4} \\
\quad\quad \text{and } \{\sigma_T(t)^{\mathcal{M}}\} \in \sigma_T(\ddot{T})^{\mathcal{M}} \cap \mathcal{T}_{\mathrm{int}_2}^{\mathcal{M}} & \text{by lem. 3.2} \\
\quad\quad \text{and } \sigma_\beta(\beta)^{\mathcal{M}} = \{(\pi_{\mathrm{act}}(\sigma_T(t)^{\mathcal{M}}), \emptyset)\} & \text{by lem. 3.1 and 3.2} \\
\quad \text{iff } \llbracket \sigma(\mathcal{C}_1) \rrbracket_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2, \mathcal{M}} \llbracket \sigma(\mathcal{C}_2) \rrbracket_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

Case `next-completion`.

$$
\begin{array}{ll}
\llbracket \mathcal{C}_1 \rrbracket_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1, \mathcal{M}|_\sigma} \llbracket \mathcal{C}_2 \rrbracket_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } \llbracket \mathcal{C}_1 \rrbracket_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \emptyset) & \\
\quad\quad \text{and } \llbracket \mathcal{C}_2 \rrbracket_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \{T'^{\mathcal{M}|_\sigma}\}) & \\
\quad\quad \text{and } \ddot{T}^{\mathcal{M}|_\sigma} \in \mathtt{Fireable}_{K_1, \llbracket \mathcal{C}_1 \rrbracket, \mathcal{M}|_\sigma, \tau} & \\
\quad\quad \text{and } T'^{\mathcal{M}|_\sigma} \in \ddot{T}^{\mathcal{M}|_\sigma} & \text{by Table 1} \\
\quad \text{iff } \llbracket \sigma(\mathcal{C}_1) \rrbracket_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \emptyset) & \text{by lem. 3.4} \\
\quad\quad \text{and } \llbracket \sigma(\mathcal{C}_2) \rrbracket_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \{\sigma_T(T')^{\mathcal{M}}\}) & \text{by lem. 3.4} \\
\quad\quad \text{and } \sigma_T(\ddot{T})^{\mathcal{M}} \in \mathtt{Fireable}_{K_2, \llbracket \sigma(\mathcal{C}_1) \rrbracket, \mathcal{M}, \tau} & \text{by Prop. 1} \\
\quad\quad \text{and } \sigma_T(T')^{\mathcal{M}} \in \sigma_T(\ddot{T})^{\mathcal{M}} & \text{by lem. 3.2} \\
\quad \text{iff } \llbracket \sigma(\mathcal{C}_1) \rrbracket_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2, \mathcal{M}} \llbracket \sigma(\mathcal{C}_2) \rrbracket_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

Case `next-trigger`.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{e/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \emptyset) & \\
\quad\quad \text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } \mathtt{Fireable}_{K_1,[\![\mathcal{C}_1]\!],\mathcal{M}|_\sigma,\tau} = \emptyset & \\
\quad\quad \text{and } \ddot{T}^{\mathcal{M}|_\sigma} \in \mathtt{Fireable}_{K_1,[\![\mathcal{C}_1]\!],\mathcal{M}|_\sigma,e} & \text{by Table 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \emptyset) & \text{by lem. 3.4} \\
\quad\quad \text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C}_1)]\!],\mathcal{M},\tau} = \emptyset & \text{by Prop. 2} \\
\quad\quad \text{and } \sigma_T(\ddot{T})^{\mathcal{M}} \in \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C}_1)]\!],\mathcal{M},e} & \text{by Prop. 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{e/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

Case `defer`.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\mathtt{defer}(e)/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } [\![\mathcal{C}_i]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \emptyset, T^{\mathcal{M}|_\sigma}, \emptyset)\ \{1 \le i \le 2\} & \\
\quad\quad \text{and } \eta|_\sigma\ (e_1) = e & \\
\quad\quad \text{and } \mathtt{defer}_1(e_1) \cap S \neq \emptyset & \\
\quad\quad \text{and } \mathtt{Fireable}_{K_1,[\![\mathcal{C}_1]\!],\mathcal{M}|_\sigma,e} = \emptyset & \text{by Table 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_i)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \emptyset, \sigma_T(T)^{\mathcal{M}}, \emptyset)\ \{1 \le i \le 2\} & \text{by lem. 3.4} \\
\quad\quad \text{and } \eta(\sigma_{\mathcal{E}}(e_1)) = e & \text{by def. of } \eta|_\sigma \\
\quad\quad \text{and } \mathtt{defer}_2(\sigma_{\mathcal{E}}(e_1)) \cap S \neq \emptyset & \text{by def. of } \sigma \\
\quad\quad \text{and } \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C}_1)]\!],\mathcal{M},e} = \emptyset & \text{by Prop. 2} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\mathtt{defer}(e)/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

Case `a-fin`.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S \cup \{\ddot{s}\}, \mathtt{do}^{\mathcal{M}|_\sigma}, H', \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } \ddot{s} \in \mathcal{S}_{\text{fin}} & \\
\quad\quad \text{and } H' = (H[(r \mapsto \bot)^{r \in \mathcal{R} \cap \uparrow\{\mathtt{regOf}(\ddot{s})\}}]) & \text{by Table 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = (S \cup \ddot{s}, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H', \mathtt{skip}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } \ddot{s} \in \mathcal{S}_{\text{fin}} & \\
\quad\quad \text{and } H' = (H[(r \mapsto \bot)^{r \in \mathcal{R} \cap \uparrow\{\mathtt{regOf}(\ddot{s})\}}]) & \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

Case a-ch.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma} \cup & \\
\quad\quad\quad \{(\pi_{\text{act}}(t^{\mathcal{M}|_\sigma}), \pi_{\text{tar}}(t^{\mathcal{M}|_\sigma}))\}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\quad\quad \text{and } \ddot{s} \in \mathcal{S}_{\text{choice}} & \\
\quad\quad \text{and } t^{\mathcal{M}|_\sigma} \in \mathcal{T}_1^{\mathcal{M}|_\sigma} & \\
\quad\quad \text{and } \pi_{\text{sor}}(t^{\mathcal{M}|_\sigma}) = \ddot{s} & \\
\quad\quad \text{and } \pi_{\text{gua}}(t^{\mathcal{M}|_\sigma}) \in \mathtt{eval}|_\sigma \, (H, \pi_{\text{tar}}(t^{\mathcal{M}|_\sigma})) & \text{by Table 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do}), H, \mathtt{skip}, \ddot{s}, \sigma_\beta(\beta), \sigma_T(T), \sigma_T(\ddot{T})) & \text{by lem. 3.4} \\
\quad\quad \text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}} \cup & \\
\quad\quad\quad \{(\pi_{\text{act}}(\sigma_T(t)^{\mathcal{M}}), \pi_{\text{tar}}(\sigma_T(t)^{\mathcal{M}}))\}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\quad\quad \text{and } \ddot{s} \in \mathcal{S}_{\text{choice}} & \\
\quad\quad \text{and } \sigma_T(t)^{\mathcal{M}} \in \mathcal{T}_2^{\mathcal{M}} & \text{by lem. 3.2} \\
\quad\quad \text{and } \pi_{\text{sor}}(\sigma_T(t)^{\mathcal{M}}) = \ddot{s} & \text{by lem. 3.2} \\
\quad\quad \text{and } \pi_{\text{gua}}(\sigma_T(t)^{\mathcal{M}}) \in \mathtt{eval}(H, \pi_{\text{tar}}(\sigma_T(t)^{\mathcal{M}})) & \text{by } \textbf{(1)} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

(1) First, $\pi_{\text{tar}}(t^{\mathcal{M}|_\sigma}) = \pi_{\text{tar}}(\sigma_T(t)^{\mathcal{M}})$ by lem. 3.2. Then, we have that $g \in \mathtt{eval}|_\sigma$ (H, s) iff $\sigma_{\mathcal{G}}(g) \in \mathtt{eval}(H, s)$ by definition of σ-reduct. Finally, since $\pi_{\text{gua}}(\sigma_T(t)^{\mathcal{M}}) = \sigma_{\mathcal{G}}(\pi_{\text{gua}}(t^{\mathcal{M}|_\sigma}))$ by definition of σ_T, we conclude that $\pi_{\text{gua}}(t^{\mathcal{M}|_\sigma}) \in \mathtt{eval}|_\sigma$ $(H, \pi_{\text{tar}}(t^{\mathcal{M}|_\sigma}))$ iff $\pi_{\text{gua}}(\sigma_T(t)^{\mathcal{M}}) \in \mathtt{eval}(H, \pi_{\text{tar}}(\sigma_T(t)^{\mathcal{M}}))$.

Case $\texttt{a-en}$.

$$
\begin{array}{ll}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad\text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \texttt{do}^{\mathcal{M}|_\sigma}, H, \texttt{skip}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\qquad\text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S \cup \{\texttt{stateOf}(\ddot{s})\}, \texttt{do}'^{\mathcal{M}|_\sigma}, H, \texttt{skip}, \emptyset, \beta'^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\qquad\text{and } \ddot{s} \in \mathcal{S}_{\text{entry}} & \\
\qquad\text{and } \texttt{do}'^{\mathcal{M}|_\sigma} = \texttt{do}^{\mathcal{M}|_\sigma}[\texttt{stateOf}(\ddot{s}) \mapsto \mu|_\sigma\ (\texttt{doAct}_1(\texttt{stateOf}(\ddot{s})))] & \\
\qquad\text{and } f : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T}_1 & \\
\qquad\text{and } \beta'^{\mathcal{M}|_\sigma} = \beta^{\mathcal{M}|_\sigma} \cup \bigcup_{r \in \texttt{dsr}(\texttt{stateOf}(\ddot{s}))} \{(\pi_{\text{act}}(f(r)^{\mathcal{M}|_\sigma}), \pi_{\text{tar}}(f(r)^{\mathcal{M}|_\sigma}))\} & \\
\qquad\text{and } \forall\, r \in \texttt{dsr}(\texttt{stateOf}(\ddot{s})) : \pi_{\text{sor}}(f(r)^{\mathcal{M}|_\sigma}) = \ddot{s} & \\
\qquad\quad \wedge\, \pi_{\text{gua}}(f(r)^{\mathcal{M}|_\sigma}) \in \texttt{eval}|_\sigma\ (H, \pi_{\text{tar}}(f(r)^{\mathcal{M}|_\sigma})) & \text{by Table 1} \\
\quad\text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\texttt{do}}(\texttt{do})^{\mathcal{M}}, H, \texttt{skip}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\qquad\text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = & \\
\qquad (S \cup \{\texttt{stateOf}(\ddot{s})\}, \sigma_{\texttt{do}}(\texttt{do}')^{\mathcal{M}}, H, \texttt{skip}, \emptyset, \sigma_\beta(\beta')^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\qquad\text{and } \ddot{s} \in \mathcal{S}_{\text{entry}} & \\
\qquad\text{and } \sigma_{\texttt{do}}(\texttt{do}')^{\mathcal{M}} = \sigma_{\texttt{do}}(\texttt{do})^{\mathcal{M}}[\texttt{stateOf}(\ddot{s}) \mapsto \mu(\texttt{doAct}_2(\texttt{stateOf}(\ddot{s})))] & \text{by lem. 3.3} \\
\qquad\text{and } \sigma_T(f) : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T}_2 & \text{by } \mathbf{(1)} \\
\qquad\text{and } \sigma_\beta(\beta')^{\mathcal{M}} = \sigma_\beta(\beta)^{\mathcal{M}} \cup & \\
\qquad\qquad \bigcup_{r \in \texttt{dsr}(\texttt{stateOf}(\ddot{s}))} \{(\pi_{\text{act}}(\sigma_T(f)(r)^{\mathcal{M}}), \pi_{\text{tar}}(\sigma_T(f)(r)^{\mathcal{M}}))\} & \text{by lem. 3.1 and 3.2} \\
\qquad\text{and } \forall\, r \in \texttt{dsr}(\texttt{stateOf}(\ddot{s})) : \pi_{\text{sor}}(\sigma_T(f)(r)^{\mathcal{M}}) = \ddot{s} & \\
\qquad\quad \wedge\, \pi_{\text{gua}}(\sigma_T(f)(r)^{\mathcal{M}}) \in \texttt{eval}(H, \pi_{\text{tar}}(\sigma_T(f)(r)^{\mathcal{M}})) & \text{by } \mathbf{(2)} \\
\quad\text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

(1) If there is a function $f : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T}_1$ which returns a transition from a region, the morphism $\sigma_T(f)$ induced by the signature morphism, is the same function where the morphism σ_T is applied to the outgoin transitions, i.e. $\sigma_T(f) : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T}_2$.

(2) By lemma 3.2 we have that $\pi_{\text{sor}}(f(r)^{\mathcal{M}|_\sigma}) = \pi_{\text{sor}}(\sigma_T(f)(r)^{\mathcal{M}})$, and the same for π_{tar}. Also, we have that $\pi_{\text{gua}}(f(r)^{\mathcal{M}|_\sigma}) \in \texttt{eval}|_\sigma\ (H, \pi_{\text{tar}}(f(r)^{\mathcal{M}|_\sigma}))$ iff $\pi_{\text{gua}}(\sigma_T(f)(r)^{\mathcal{M}}) \in \texttt{eval}(H, \pi_{\text{tar}}(\sigma_T(f)(r)^{\mathcal{M}}))$ as proved in the case $\texttt{a-ch}$.(**1**). Finally, we conclude that the premise holds.

Case `a-ex-1`.

$$
\begin{aligned}
&[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} && \\
&\quad \text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) && \\
&\qquad \text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S \setminus \{\mathtt{stateOf}(\ddot{s})\}, \mathtt{do}^{\mathcal{M}|_\sigma}, H', \mathtt{skip}, \emptyset, \beta'^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) && \\
&\qquad \text{and } \ddot{s} \in \mathcal{S}_{\text{exit}} && \\
&\qquad \text{and } \forall\, (B^{\mathcal{M}|_\sigma}, \ddot{s}') \in \beta^{\mathcal{M}|_\sigma} : \ddot{s}' \neq \ddot{s} && \\
&\qquad \text{and } \forall\, s \in S : \neg(\mathtt{stateOf}(\ddot{s}) \succ s) && \\
&\qquad \text{and } t^{\mathcal{M}|_\sigma} \in T^{\mathcal{M}|_\sigma} && \\
&\qquad \text{and } \pi_{\text{sor}}(t^{\mathcal{M}|_\sigma}) = \ddot{s} && \\
&\qquad \text{and } \beta'^{\mathcal{M}|_\sigma} = \beta^{\mathcal{M}|_\sigma} \cup \{(\pi_{\text{act}}(t^{\mathcal{M}|_\sigma}), \pi_{\text{tar}}(t^{\mathcal{M}|_\sigma}))\} && \\
&\qquad \text{and } H' = H[\mathtt{regOf}(\ddot{s}) \mapsto \mathtt{stateOf}(\ddot{s})] && \text{by Table 1} \\
&\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) && \text{by lem. 3.4} \\
&\qquad \text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = && \\
&\qquad (S \setminus \{\mathtt{stateOf}(\ddot{s})\}, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H', \mathtt{skip}, \emptyset, \sigma_\beta(\beta')^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) && \text{by lem. 3.4} \\
&\qquad \text{and } \ddot{s} \in \mathcal{S}_{\text{exit}} && \\
&\qquad \text{and } \forall\, (\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \ddot{s}') \in \sigma_\beta(\beta)^{\mathcal{M}} : \ddot{s}' \neq \ddot{s} && \text{by lem. 3.1} \\
&\qquad \text{and } \forall\, s \in S : \neg(\mathtt{stateOf}(\ddot{s}) \succ s) && \\
&\qquad \text{and } \sigma_T(t)^{\mathcal{M}} \in \sigma_T(T)^{\mathcal{M}} && \text{by lem. 3.2} \\
&\qquad \text{and } \pi_{\text{sor}}(\sigma_T(t)^{\mathcal{M}}) = \ddot{s} && \text{by lem. 3.2} \\
&\qquad \text{and } \sigma_\beta(\beta')^{\mathcal{M}} = \sigma_\beta(\beta)^{\mathcal{M}} \cup \{(\pi_{\text{act}}(\sigma_T(t)^{\mathcal{M}}), \pi_{\text{tar}}(\sigma_T(t)^{\mathcal{M}}))\} && \text{by lem. 3.1 and 3.2} \\
&\qquad \text{and } H' = H[\mathtt{regOf}(\ddot{s}) \mapsto \mathtt{stateOf}(\ddot{s})] && \\
&\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} && \text{by Table 1}
\end{aligned}
$$

Case a-ex-2.

$$
\begin{array}{lr}
[\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_1} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \\
\quad \text{iff } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \ddot{s}, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\qquad \text{and } [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} = (S, \mathtt{do}^{\mathcal{M}|_\sigma}, H, \mathtt{skip}, \emptyset, \beta^{\mathcal{M}|_\sigma}, T^{\mathcal{M}|_\sigma}, \ddot{T}^{\mathcal{M}|_\sigma}) & \\
\qquad \text{and } \ddot{s} \in \mathcal{S}_{\text{exit}} & \\
\qquad \text{and } \exists\, B^{\mathcal{M}|_\sigma} : (B^{\mathcal{M}|_\sigma}, \ddot{s}) \in \beta^{\mathcal{M}|_\sigma} \vee \exists\, s \in S : \mathtt{stateOf}(\ddot{s}) \succ s & \text{by Table 1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \ddot{s}, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\qquad \text{and } [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} = (S, \sigma_{\mathtt{do}}(\mathtt{do})^{\mathcal{M}}, H, \mathtt{skip}, \emptyset, \sigma_\beta(\beta)^{\mathcal{M}}, \sigma_T(T)^{\mathcal{M}}, \sigma_T(\ddot{T})^{\mathcal{M}}) & \text{by lem. 3.4} \\
\qquad \text{and } \ddot{s} \in \mathcal{S}_{\text{exit}} & \\
\qquad \text{and } \exists\, \sigma_{\mathcal{A}}(B)^{\mathcal{M}} : (\sigma_{\mathcal{A}}(B)^{\mathcal{M}}, \ddot{s}) \in \sigma_\beta(\beta)^{\mathcal{M}} \vee \exists\, s \in S : \mathtt{stateOf}(\ddot{s}) \succ s & \text{by lem. 3.1} \\
\quad \text{iff } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/\emptyset}{\rightarrowtail}_{\Sigma_2} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Table 1}
\end{array}
$$

□

As a second step, we use the auxiliary satisfaction condition to define a small step semantics dealing with processing sequences of input events. As stated in [7], the semantics considers that in each step one event of the current sequence of input events is consumed and therefore deleted from this sequence, and a sequence of events is generated which is added to the shortened sequence of input events resulting in a new sequence of input events to be used in the following step.

Definition 16 (Small Step Semantics). Given a signature Σ, and a Σ-model $\mathcal{M}$, the small step semantics of transitions between configurations is given in terms of the relation $\twoheadrightarrow_{\Sigma,\mathcal{M}}$. Given a Σ-sentence $\psi = ((\mathcal{C}_1, E_1), (\mathcal{C}_2, E_2))$, the relation $(\mathcal{C}_1, E_1) \twoheadrightarrow_{\Sigma,\mathcal{M}} (\mathcal{C}_2, E_2)$ holds if it holds any of the following cases:

$$
\begin{array}{l}
(\mathtt{eve})\ [\![E_1]\!] \neq \emptyset \ \wedge\ \mathtt{sel}([\![E_1]\!]) = (e, [\![E_2]\!]) \ \wedge\ [\![\mathcal{C}_1]\!] \overset{e/\emptyset}{\rightarrowtail}_{\Sigma,\mathcal{M}} [\![\mathcal{C}_2]\!] \\
(\mathtt{def})\ [\![E_1]\!] \neq \emptyset \ \wedge\ \mathtt{sel}([\![E_1]\!]) = (e, [\![E_2]\!]) \ \wedge\ [\![\mathcal{C}_1]\!] \overset{\mathtt{defer}(e)/\emptyset}{\rightarrowtail}_{\Sigma,\mathcal{M}} [\![\mathcal{C}_2]\!] \\
(\mathtt{int})\ \mathtt{join}(E, [\![E_1]\!]) = [\![E_2]\!] \ \wedge\ [\![\mathcal{C}_1]\!] \overset{\tau/E}{\rightarrowtail}_{\Sigma,\mathcal{M}} [\![\mathcal{C}_2]\!]
\end{array}
$$

Rule eve corresponds to the consumption of an event where an event is selected from the sequence of input events but no output events are generated. Rule def describes the deferral of events where, as in the last rule, an event is selected but no output events are generated (in fact no transition is triggered). Finally, Rule int corresponds to the execution of an internal transition where no event is consumed but output events are possibly generated (as in rule do-act of Table 1).

The satisfaction relation states the fact that there is a transition between two configurations with the corresponding event queues.

Definition 17 (Satisfaction Relation). Given a signature Σ, a Σ-model $\mathcal{M}$, and a Σ-sentence $\psi = ((\mathcal{C}_1, E_1), (\mathcal{C}_2, E_2))$, the *satisfaction relation* is expressed as follows:

$$
\mathcal{M} \models_\Sigma \psi \text{ iff } (\mathcal{C}_1, E_1) \twoheadrightarrow_{\Sigma,\mathcal{M}} (\mathcal{C}_2, E_2)
$$

Finally, we state the satisfaction condition for the institution.

Theorem 2 (Satisfaction Condition). Given signatures Σ_1 and Σ_2, a signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M}$, and a Σ_1-sentence ψ, the following satisfaction condition holds.

$$\mathcal{M}|_\sigma \models_{\Sigma_1} \psi \text{ iff } \mathcal{M} \models_{\Sigma_2} \sigma(\psi)$$

Proof. By cases on the satisfaction rules of the relation $\twoheadrightarrow_{\Sigma,\mathcal{M}}$.

Case `eve`.

$$\begin{array}{ll}
\mathcal{M}|_\sigma \models_{\Sigma_1} \psi & \\
\quad\text{iff } (\mathcal{C}_1, E_1) \twoheadrightarrow_{\Sigma_1,\mathcal{M}|_\sigma} (\mathcal{C}_2, E_2) & \text{by def. of} \models \\
\quad\text{iff } [\![E_1]\!]_{\mathcal{M}|_\sigma} \neq \emptyset & \\
\quad\quad\text{and } \mathtt{sel}|_\sigma ([\![E_1]\!]_{\mathcal{M}|_\sigma}) = (e, [\![E_2]\!]_{\mathcal{M}|_\sigma}) & \\
\quad\quad\text{and } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{e/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \text{by satisfaction rule } \mathtt{eve} \\
\quad\text{iff } [\![E_1]\!]_{\mathcal{M}} \neq \emptyset & \text{by } \textbf{(1)} \\
\quad\quad\text{and } \mathtt{sel}([\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}}) = (e, [\![\sigma_{\mathcal{E}}(E_2)]\!]_{\mathcal{M}}) & \text{by } \textbf{(2)} \\
\quad\quad\text{and } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{e/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Theorem 1} \\
\quad\text{iff } (\sigma(\mathcal{C}_1), \sigma_{\mathcal{E}}(E_1)) \twoheadrightarrow_{\Sigma_2,\mathcal{M}} (\sigma(\mathcal{C}_2), \sigma_{\mathcal{E}}(E_2)) & \text{by satisfaction rule } \mathtt{eve} \\
\quad\text{iff } \mathcal{M} \models_{\Sigma_2} \sigma(\psi) & \text{by def. of} \models
\end{array}$$

(1) By definition of $[\![]\!]$, we have that $[\![E_1]\!] \neq \emptyset$ iff $E_1 \neq \emptyset$. Thus, it holds $[\![E_1]\!]_{\mathcal{M}} \neq \emptyset$ iff $[\![E_1]\!]_{\mathcal{M}|_\sigma} \neq \emptyset$.

(2) We have that $\mathtt{sel}\,|_\sigma\ ([\![E_1]\!]_{\mathcal{M}|_\sigma}) = \mathtt{sel}\,|_\sigma\ (\eta\,|_\sigma\ (E_1)) = \mathtt{sel}(\eta(\sigma_{\mathcal{E}}(E_1))) = \mathtt{sel}([\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}})$, by definition of $[\![]\!]$ and $\mathtt{sel}|_\sigma$. We also have that $[\![\sigma_{\mathcal{E}}(E)]\!]_{\mathcal{M}} = [\![E]\!]_{\mathcal{M}|_\sigma}$, by definition of $[\![]\!]$ and $\eta|_\sigma$. Finally, we have that $\mathtt{sel}|_\sigma ([\![E_1]\!]_{\mathcal{M}|_\sigma}) = (e, [\![E_2]\!]_{\mathcal{M}|_\sigma})$ iff $\mathtt{sel}([\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}}) = (e, [\![\sigma_{\mathcal{E}}(E_2)]\!]_{\mathcal{M}})$.

Case `def`.

$$\begin{array}{ll}
\mathcal{M}|_\sigma \models_{\Sigma_1} \psi & \\
\quad\text{iff } (\mathcal{C}_1, E_1) \twoheadrightarrow_{\Sigma_1,\mathcal{M}|_\sigma} (\mathcal{C}_2, E_2) & \text{by def. of} \models \\
\quad\text{iff } [\![E_1]\!]_{\mathcal{M}|_\sigma} \neq \emptyset & \\
\quad\quad\text{and } \mathtt{sel}|_\sigma ([\![E_1]\!]_{\mathcal{M}|_\sigma}) = (e, [\![E_2]\!]_{\mathcal{M}|_\sigma}) & \\
\quad\quad\text{and } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\mathtt{defer}(e)/\emptyset}{\rightarrowtail}_{\Sigma_1,\mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & \text{by satisfaction rule } \mathtt{def} \\
\quad\text{iff } [\![E_1]\!]_{\mathcal{M}} \neq \emptyset & \text{by } \textbf{(1)} \text{ in } \mathtt{eve} \\
\quad\quad\text{and } \mathtt{sel}([\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}}) = (e, [\![\sigma_{\mathcal{E}}(E_2)]\!]_{\mathcal{M}}) & \text{by } \textbf{(2)} \text{ in } \mathtt{eve} \\
\quad\quad\text{and } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\mathtt{defer}(e)/\emptyset}{\rightarrowtail}_{\Sigma_2,\mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & \text{by Theorem 1} \\
\quad\text{iff } (\sigma(\mathcal{C}_1), \sigma_{\mathcal{E}}(E_1)) \twoheadrightarrow_{\Sigma_2,\mathcal{M}} (\sigma(\mathcal{C}_2), \sigma_{\mathcal{E}}(E_2)) & \text{by satisfaction rule } \mathtt{def} \\
\quad\text{iff } \mathcal{M} \models_{\Sigma_2} \sigma(\psi) & \text{by def. of} \models
\end{array}$$

Case `int`.

$$
\begin{array}{lll}
\mathcal{M}|_\sigma \models_{\Sigma_1} \psi & & \\
\quad \text{iff } (\mathcal{C}_1, E_1) \twoheadrightarrow_{\Sigma_1, \mathcal{M}|_\sigma} (\mathcal{C}_2, E_2) & & \text{by def. of } \models \\
\quad \text{iff } \texttt{join}|_\sigma\ (E, [\![E_1]\!]_{\mathcal{M}|_\sigma}) = [\![E_2]\!]_{\mathcal{M}|_\sigma} & & \\
\qquad \text{and } [\![\mathcal{C}_1]\!]_{\mathcal{M}|_\sigma} \overset{\tau/E}{\rightarrowtail}_{\Sigma_1, \mathcal{M}|_\sigma} [\![\mathcal{C}_2]\!]_{\mathcal{M}|_\sigma} & & \text{by satisfaction rule } \texttt{int} \\
\quad \text{iff} & & \\
\qquad \text{and } \texttt{join}(E, [\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}}) = [\![\sigma_{\mathcal{E}}(E_2)]\!]_{\mathcal{M}} & & \text{by } \textbf{(1)} \\
\qquad \text{and } [\![\sigma(\mathcal{C}_1)]\!]_{\mathcal{M}} \overset{\tau/E}{\rightarrowtail}_{\Sigma_2, \mathcal{M}} [\![\sigma(\mathcal{C}_2)]\!]_{\mathcal{M}} & & \text{by Theorem 1} \\
\quad \text{iff } (\sigma(\mathcal{C}_1), \sigma_{\mathcal{E}}(E_1)) \twoheadrightarrow_{\Sigma_2, \mathcal{M}} (\sigma(\mathcal{C}_2), \sigma_{\mathcal{E}}(E_2)) & & \text{by satisfaction rule } \texttt{int} \\
\quad \text{iff } \mathcal{M} \models_{\Sigma_2} \sigma(\psi) & & \text{by def. of } \models
\end{array}
$$

(1) We have that $[\![\sigma_{\mathcal{E}}(E)]\!]_{\mathcal{M}} = [\![E]\!]_{\mathcal{M}|_\sigma}$, by definition of $[\![]\!]$ and $\eta|_\sigma$. Thus, we have that $\texttt{join}(E, [\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}}) = \texttt{join}|_\sigma\ (E, [\![E_1]\!]_{\mathcal{M}|_\sigma})$, and $[\![E_2]\!]_{\mathcal{M}|_\sigma} = [\![\sigma_{\mathcal{E}}(E_2)]\!]_{\mathcal{M}}$. Finally, we have that $\texttt{join}|_\sigma\ (E, [\![E_1]\!]_{\mathcal{M}|_\sigma}) = [\![E_2]\!]_{\mathcal{M}|_\sigma}$ iff $\texttt{join}(E, [\![\sigma_{\mathcal{E}}(E_1)]\!]_{\mathcal{M}}) = [\![\sigma_{\mathcal{E}}(E_2)]\!]_{\mathcal{M}}$.

$\square$

Given this result, signatures, sentences, models, reducts together with the satisfaction relation, define an institution.

5 Runs as Sentences

In the last section we defined Σ-sentences to represent possible adjacent configurations of a core state machine w.r.t. a transition. Now we change their meaning to represent the initial and final configurations of a core state machine w.r.t. a run. Therefore, runs are considered instead of transition steps, and in consequence we define a new institution SM_ρ.

Lets denote a pair of a configuration and a set of events $(\mathcal{C}_1, E_1)$ as $\widehat{\mathcal{C}_1}$, and $(\sigma(\mathcal{C}_1), \sigma(E_1))$ as $\sigma(\widehat{\mathcal{C}_1})$.

Definition 18 (Run). Given the relation $\twoheadrightarrow_{\Sigma,\mathcal{M}}$ used for the definition of the small step semantics of transitions, we define a *run* through the state machine as a finite path $\widehat{\mathcal{C}_1} \twoheadrightarrow_{\Sigma,\mathcal{M}} \widehat{\mathcal{C}_2} \twoheadrightarrow_{\Sigma,\mathcal{M}} \cdots \twoheadrightarrow_{\Sigma,\mathcal{M}} \widehat{\mathcal{C}_n}$. We denote a run between $\widehat{\mathcal{C}_1}$ and $\widehat{\mathcal{C}_n}$ as $\rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_n})$. The length of a run ρ, denoted by $|\rho|$, is its number of transitions. The shortest run consists of one transition between two configurations (the minimum length of a run is 1).

Now we modify the definition of the satifaction relation for this new interpretation of a Σ-sentence.

Definition 19 (Satisfaction Relation for Runs). Given a signature Σ, a Σ-model $\mathcal{M}$, and a Σ-sentence $\psi = ((\mathcal{C}_1, E_1), (\mathcal{C}_2, E_2))$, the *satisfaction relation for runs* is expressed as follows:

$$\mathcal{M} \models_{\rho,\Sigma} \psi \text{ iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_2})$$

Now we have to adapt the satisfaction condition and prove that it still holds, which gives rise to the following theorem.

Theorem 3 (Satisfaction Condition for Runs). Given signatures Σ_1 and Σ_2, a signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M}$, and a Σ_1-sentence $\psi = ((\mathcal{C}_1, E_1), (\mathcal{C}_2, E_2))$, the following satisfaction condition holds:

$$\mathcal{M}|_\sigma \models_{\rho,\Sigma_1} \psi \text{ iff } \mathcal{M} \models_{\rho,\Sigma_2} \sigma(\psi)$$

Proof. By induction on the length of the run $\rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_2})$.

Case $\left|\rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_2})\right| = 1$.

$$\begin{array}{lll}
\mathcal{M}|_\sigma \models_{\rho,\Sigma_1} \psi & & \\
 & \text{iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_2}) & \text{by definition of } \models_\rho \\
 & \text{iff } \widehat{\mathcal{C}_1} \twoheadrightarrow_{\Sigma_1,\mathcal{M}|_\sigma} \widehat{\mathcal{C}_2} & \text{by definition of a run} \\
 & \text{iff } \mathcal{M}|_\sigma \models_{\Sigma_1} \psi & \text{by definition of } \models \\
 & \text{iff } \mathcal{M} \models_{\Sigma_2} \sigma(\psi) & \text{by Theorem 2} \\
 & \text{iff } \sigma(\widehat{\mathcal{C}_1}) \twoheadrightarrow_{\Sigma_2,\mathcal{M}} \sigma(\widehat{\mathcal{C}_2}) & \text{by definition of } \models \\
 & \text{iff } \exists\, \rho(\sigma(\widehat{\mathcal{C}_1}), \sigma(\widehat{\mathcal{C}_2})) & \text{by definition of a run} \\
 & \text{iff } \mathcal{M} \models_{\rho,\Sigma_2} \sigma(\psi) & \text{by definition of } \models_\rho
\end{array}$$

Case $\left|\rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_{n-1}})\right| = n > 1$.

$$
\begin{array}{ll}
\mathcal{M}|_\sigma \models_{\rho,\Sigma_1} \psi & \\
\quad \text{iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_{n-1}}) & \text{by definition of } \models_\rho \\
\quad \text{iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_{n-2}}) \wedge \exists\, \rho(\widehat{\mathcal{C}_{n-2}}, \widehat{\mathcal{C}_{n-1}}) & \text{by definition of a run} \\
\quad \text{iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_{n-2}}) \wedge \widehat{\mathcal{C}_{n-2}} \twoheadrightarrow_{\Sigma_1,\mathcal{M}|_\sigma} \widehat{\mathcal{C}_{n-1}} & \text{by definition of a run} \\
\quad \text{iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_{n-2}}) \wedge \mathcal{M}|_\sigma \models_{\Sigma_1} (\widehat{\mathcal{C}_{n-2}}, \widehat{\mathcal{C}_{n-1}}) & \text{by Definition 17} \\
\quad \text{iff } \exists\, \rho(\widehat{\mathcal{C}_1}, \widehat{\mathcal{C}_{n-2}}) \wedge \mathcal{M} \models_{\Sigma_2} (\sigma(\widehat{\mathcal{C}_{n-2}}), \sigma(\widehat{\mathcal{C}_{n-1}})) & \text{by Theorem 2} \\
\quad \text{iff } \exists\, \rho(\sigma(\widehat{\mathcal{C}_1}), \sigma(\widehat{\mathcal{C}_{n-2}})) \wedge \mathcal{M} \models_{\Sigma_2} (\sigma(\widehat{\mathcal{C}_{n-2}}), \sigma(\widehat{\mathcal{C}_{n-1}})) & \text{by inductive hypothesis} \\
\quad \text{iff } \exists\, \rho(\sigma(\widehat{\mathcal{C}_1}), \sigma(\widehat{\mathcal{C}_{n-2}})) \wedge \sigma(\widehat{\mathcal{C}_{n-2}}) \twoheadrightarrow_{\Sigma_2,\mathcal{M}} \sigma(\widehat{\mathcal{C}_{n-1}}) & \text{by Definition 17} \\
\quad \text{iff } \exists\, \rho(\sigma(\widehat{\mathcal{C}_1}), \sigma(\widehat{\mathcal{C}_{n-2}})) \wedge \exists\, \rho(\sigma(\widehat{\mathcal{C}_{n-2}}), \sigma(\widehat{\mathcal{C}_{n-1}})) & \text{by definition of a run} \\
\quad \text{iff } \exists\, \rho(\sigma(\widehat{\mathcal{C}_1}), \sigma(\widehat{\mathcal{C}_{n-1}})) & \text{by definition of a run} \\
\quad \text{iff } \mathcal{M} \models_{\rho,\Sigma_2} \sigma(\psi) & \text{by definition of } \models_\rho
\end{array}
$$

$\square$

Given that the satisfaction condition holds, and that the former institution constructs remain unmodified, signatures, sentences, models, reducts together with the new satisfaction relation define an institution SM_ρ.

6 Conclusions and Further Work

In this work we defined an institution for UML 2.0 State Machines. The institution allows to describe the language using its own semantic domain. We based our work on existing semantics. The semantics deals with processing sequences of input events such that in each step one of these events is consumed and a sequence of events is generated which can be used in the following step. As a complement we extended the semantics for considering runs through the state machine instead of a simple transition step, and defined a new institution based on this.

With this work we expect to contribute to the development of the heterogeneous institution environment for the semantic definition of UML in [2], which provides a rich and flexible framework for program specification. The choice of UML State Machines was not random. This decision was taken considering the impact of the inclusion of the language within the heterogeneous environment since it constitutes a valuable notation for describing behavioral aspects of a system. For the contribution to be complete we need to relate this intitution with the others within the environment by means of institution morphisms. In particular, there is a close relationship between state machines and interactions [10,11,12], which are already defined as an institution [13], which may help in the definition of an institution morphism in a natural way. This is subject of further work.

Finally, this work is part of a broad research agenda. We are concerned with the relation between institution morphisms and model transformations in the context of Model-Driven Engineering [14]. Model transformations are functions taking input models and producing output models such that both models conform to given metamodels (possibly the same). Transformations allow defining not only syntactical relations between models but also complex semantical ones. Intuitively, there is some relationship between model transformations and institution morphisms, which are transformations preserving truth from one logical system to another. In this context we want to study the mathematical properties of model transformations from the perspective given by institution morphisms.

References

1. OMG: Unified Modeling Language: Superstructure. formal/2005-07-04 v2.0, Object Management Group (2005)
2. Cengarle, M.V., Knapp, A., Tarlecki, A., Wirsing, M.: A Heterogeneous Approach to UML Semantics. In Degano, P., Nicola, R.D., Meseguer, J., eds.: Concurrency, Graphs and Models. Volume 5065 of Lecture Notes in Computer Science., Springer (2008) 383–402
3. Goguen, J.A., Burstall, R.M.: Introducing Institutions. In Clarke, E.M., Kozen, D., eds.: Logic of Programs. Volume 164 of Lecture Notes in Computer Science., Springer (1983) 221–256
4. Goguen, J.A., Burstall, R.M.: Institutions: Abstract Model Theory for Specification and Programming. J. ACM **39** (1992) 95–146
5. Crane, M.L., Dingel, J.: On the Semantics of UML State Machines: Categorization and Comparison. Technical Report 2005-501, School of Computing, Queen's University, Kingston, Ontario, Canada (2005)

6. Fecher, H., Schönborn, J.: UML 2.0 State Machines: Complete Formal Semantics via Core State Machines. In: Proceedings of the 11th international workshop, FMICS 2006 and 5th international workshop, PDMC conference on Formal methods: Applications and technology. FMICS'06/PDMC'06, Berlin, Heidelberg, Springer-Verlag (2007) 244–260
7. von der Beeck, M.: A Structured Operational Semantics for UML-Statecharts. Software and Systems Modeling **1** (2002) 130–141
8. Lane, S.M.: Categories for the Working Mathematician. 2nd edn. Graduate Texts in Mathematics. Springer (1998)
9. Sannella, D., Tarlecki, A.: Foundations of Algebraic Specification and Formal Software Development. 1st edn. Monographs in Theoretical Computer Science. Springer (2011)
10. Grønmo, R., Møller-Pedersen, B.: From Sequence Diagrams to State Machines by Graph Transformation. In: Theory and Practice of Model Transformations. Volume 6142 of LNCS. Springer (2010) 93–107
11. Krüger, I.H.: Distributed System Design with Message Sequence Charts. PhD. Thesis, Technischen Universität München (2000)
12. Whittle, J., Schumann, J.: Generating Statechart Designs from Scenarios. In: Proc. of the 22nd Intl. Conf. on Software Engineering, ACM (2000) 314–323
13. Cengarle, M.V., Knapp, A.: An Institution for UML 2.0 Interactions. Technical Report TUM-I0808, Institut für Informatik, Technische Universität München (2008)
14. Kent, S.: Model-Driven Engineering. In: Proc. 3rd , Third Intl. Conf. on Integrated Formal Methods. Volume 2335 of LNCS., Springer (2002) 286–298

A Auxiliary Properties

Proposition 1 (Invariance of Fireable Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ $(i{=}1,2)$ with $K_i = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}_i, \mathtt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ $(i{=}1,2)$, a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \surd)$, a configuration $\mathcal{C} = (S, \mathtt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$ for K_i, and an event $e \in \mathbb{E} \cup \{\tau\}$,

$$T^{\mathcal{M}|_\sigma} \in \mathtt{Fireable}_{K_1,[\![\mathcal{C}]\!],\mathcal{M}|_\sigma,e} \text{ iff } \sigma_T(T)^{\mathcal{M}} \in \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C})]\!],\mathcal{M},e}$$

Proof.

$$\begin{array}{ll}
T^{\mathcal{M}|_\sigma} \in \mathtt{Fireable}_{K_1,[\![\mathcal{C}]\!],\mathcal{M}|_\sigma,e} & \\
\quad \text{iff } T \subseteq \mathtt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} & \\
\quad\quad \text{and } \forall T' \in \mathtt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} \setminus T : & \\
\quad\quad\quad \forall T_1 \in T : (T', T_1) \notin \mathtt{Priority}_{K_1} & \\
\quad\quad \text{and } \forall T' \in \mathtt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} \setminus T : & \\
\quad\quad\quad \exists T_1 \in T : (T_1, T') \in \mathtt{Conflict}_{K_1} & \\
\quad\quad \text{and } \forall T_1, T_2 \in T : (T_1, T_2) \in \mathtt{Conflict}_{K_1} \Rightarrow T_1 = T_2 & \text{by Def. 12} \\
\quad \text{iff } \sigma_T(T) \in \mathtt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} & \text{by Prop. 3} \\
\quad\quad \text{and } \forall \sigma_T(T') \in \mathtt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} \setminus \sigma_T(T) : & \\
\quad\quad \forall \sigma_T(T_1) \in \sigma_T(T) : (\sigma_T(T'), \sigma_T(T_1)) \notin \mathtt{Priority}_{K_2} & \text{by } \mathbf{(1)} \\
\quad\quad \text{and } \forall \sigma_T(T') \in \mathtt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} \setminus \sigma_T(T) : & \\
\quad\quad \exists \sigma_T(T_1) \in \sigma_T(T) : (\sigma_T(T_1), \sigma_T(T')) \in \mathtt{Conflict}_{K_2} & \text{by Prop. 5} \\
\quad\quad \text{and } \forall \sigma_T(T_1), \sigma_T(T_2) \in \sigma_T(T) : & \\
\quad\quad\quad (\sigma_T(T_1), \sigma_T(T_2)) \in \mathtt{Conflict}_{K_2} \Rightarrow \sigma_T(T_1) = \sigma_T(T_2) & \text{by } \mathbf{(2)} \\
\quad \text{iff } \sigma_T(T)^{\mathcal{M}} \in \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C})]\!],\mathcal{M},e} & \text{by Def. 12}
\end{array}$$

(1) It is a direct implication of Proposition 6.

(2) The implication $(T_1, T_2) \in \mathtt{Conflict}_{K_1} \Rightarrow T_1 = T_2$ can be written as $(T_1, T_2) \notin \mathtt{Conflict}_{K_1} \vee T_1 = T_2$.[6] We also have that $(T_1, T_2) \notin \mathtt{Conflict}_{K_1}$ iff $(\sigma_T(T_1), \sigma_T(T')) \notin \mathtt{Conflict}_{K_2}$ as a direct result of Proposition 5. Moreover, we have that $T_1 = T_2$ iff $\sigma_T(T_1) = \sigma_T(T_2)$ by definition of σ_T. Finally, we conclude that $(T_1, T_2) \notin \mathtt{Conflict}_{K_1} \vee T_1 = T_2$ iff $(\sigma_T(T_1), \sigma_T(T_2)) \notin \mathtt{Conflict}_{K_2} \vee \sigma_T(T_1) = \sigma_T(T_2)$, by definition of $\vee$, and thus $(T_1, T_2) \in \mathtt{Conflict}_{K_1} \Rightarrow T_1 = T_2$ iff $(\sigma_T(T_1), \sigma_T(T_2)) \in \mathtt{Conflict}_{K_2} \Rightarrow \sigma_T(T_1) = \sigma_T(T_2)$.

$\square$

[6] The implication $A \Rightarrow B$ is a shortcut of $\neg A \vee B$. Also, it negation is $A \wedge \neg B$

Proposition 2 (Invariance of No Fireable Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ (i=1, 2) with $K_i = ((\mathcal{S}, \mathcal{R}, \mathtt{parent}), \mathtt{doAct}_i, \mathtt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1, 2), a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M} = (\mathbb{A}, \mu, \mathtt{eval}, \mathtt{calc}, \mathbb{E}, \eta, \mathtt{sel}, \mathtt{join}, \rightsquigarrow, \surd)$, a configuration $\mathcal{C} = (S, \mathtt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$ for K_i, and an event $e \in \mathbb{E} \cup \{\tau\}$,

$$\mathtt{Fireable}_{K_1,[\![\mathcal{C}]\!],\mathcal{M}|_\sigma,e} = \emptyset \text{ iff } \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C})]\!],\mathcal{M},e} = \emptyset$$

Proof.

$$\begin{array}{lr}
\mathtt{Fireable}_{K_1,[\![\mathcal{C}]\!],\mathcal{M}|_\sigma,e} = \emptyset & \\
\quad \text{iff } \mathtt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} = \emptyset & \\
\qquad \text{or } \forall\, T' \in \mathtt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} \setminus T : & \\
\qquad\quad \forall\, T_1 \in T : (T', T_1) \in \mathtt{Priority}_{K_1} & \\
\qquad \text{or } \forall\, T' \in \mathtt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} \setminus T : & \\
\qquad\quad \exists\, T_1 \in T : (T_1, T') \notin \mathtt{Conflict}_{K_1} & \\
\qquad \text{or } \forall\, T_1, T_2 \in T : (T_1, T_2) \in \mathtt{Conflict}_{K_1} \wedge T_1 \neq T_2 & \text{by Def. 12} \\
\quad \text{iff } \mathtt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} = \emptyset & \text{by Prop. 4} \\
\qquad \text{or } \forall\, \sigma_T(T') \in \mathtt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} \setminus \sigma_T(T) : & \\
\qquad\quad \forall\, \sigma_T(T_1) \in \sigma_T(T) : (\sigma_T(T'), \sigma_T(T_1)) \in \mathtt{Priority}_{K_2} & \text{by Prop. 6} \\
\qquad \text{or } \forall\, \sigma_T(T') \in \mathtt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} \setminus \sigma_T(T) : & \\
\qquad\quad \exists\, \sigma_T(T_1) \in \sigma_T(T) : (\sigma_T(T_1), \sigma_T(T')) \notin \mathtt{Conflict}_{K_2} & \text{by } \mathbf{(1)} \\
\qquad \text{or } \forall\, \sigma_T(T_1), \sigma_T(T_2) \in \sigma_T(T) : & \\
\qquad\qquad\quad (\sigma_T(T_1), \sigma_T(T_2)) \in \mathtt{Conflict}_{K_2} & \\
\qquad\qquad\quad \wedge\, \sigma_T(T_1) \neq \sigma_T(T_2) & \text{by } \mathbf{(2)} \\
\quad \text{iff } \mathtt{Fireable}_{K_2,[\![\sigma(\mathcal{C})]\!],\mathcal{M},e} = \emptyset & \text{by Def. 12}
\end{array}$$

(1) It is a direct implication of Proposition 5

(2) We have that $(T_1, T_2) \in \mathtt{Conflict}_{K_1}$ iff $(\sigma_T(T_1), \sigma_T(T')) \in \mathtt{Conflict}_{K_2}$ by Proposition 5. Moreover, we have that $T_1 \neq T_2$ iff $\sigma_T(T_1) \neq \sigma_T(T_2)$ by definition of σ_T. Finally, we conclude that $(T_1, T_2) \in \mathtt{Conflict}_{K_1} \wedge T_1 \neq T_2$ iff $(\sigma_T(T_1), \sigma_T(T_2)) \in \mathtt{Conflict}_{K_2} \wedge \sigma_T(T_1) \neq \sigma_T(T_2)$, by definition of $\wedge$.

□

Proposition 3 (Invariance of Enable Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ (i=1, 2) with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1, 2), a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M} = (\mathbb{A}, \mu, \texttt{eval}, \texttt{calc}, \mathbb{E}, \eta, \texttt{sel}, \texttt{join}, \rightsquigarrow, \surd)$, a configuration $\mathcal{C} = (S, \texttt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$ for K_i, and an event $e \in \mathbb{E} \cup \{\tau\}$,

$$T \in \texttt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} \text{ iff } \sigma_T(T) \in \texttt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e}$$

Proof.

$$
\begin{array}{lll}
T \in \texttt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} & & \\
\quad \text{iff } T \in \texttt{CoTr}_{K_1} & & \\
\quad\quad \text{and } \forall t \in T : \texttt{stateOf}(\pi_{\text{sor}}(t)) \in S & & \\
\quad\quad \text{and } \forall t \in T : \eta|_\sigma\ (\pi_{\text{ev}}(t)) = e & & \\
\quad\quad \text{and } \forall t \in T : \pi_{\text{gua}}(t) \in \texttt{eval}|_\sigma\ (H, \pi_{\text{tar}}(t)) & & \\
\quad\quad \text{and } \forall t \in T : \pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}} & & \\
\quad\quad\quad\quad \Rightarrow \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) = \texttt{skip} & & \text{by Def. 12} \\
\quad \text{iff } \sigma_T(T) \in \texttt{CoTr}_{K_2} & & \text{by Prop. 7} \\
\quad\quad \text{and } \forall \sigma_T(t) \in \sigma_T(T) : \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t))) \in S & & \text{by } \textbf{(1)} \\
\quad\quad \text{and } \forall \sigma_T(t) \in \sigma_T(T) : \eta(\pi_{\text{ev}}(\sigma_T(t))) = e & & \text{by } \textbf{(2)} \\
\quad\quad \text{and } \forall \sigma_T(t) \in \sigma_T(T) : \pi_{\text{gua}}(\sigma_T(t)) \in \texttt{eval}(H, \pi_{\text{tar}}(\sigma_T(t))) & & \text{by } \textbf{(3)} \\
\quad\quad \text{and } \forall \sigma_T(t) \in \sigma_T(T) : \pi_{\text{sor}}(\sigma_T(t)) \in \mathcal{S}^{\text{cp}}_{\text{exit}} & & \\
\quad\quad\quad\quad \Rightarrow \sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))) = \sigma_{\mathcal{A}}(\texttt{skip}) & & \text{by } \textbf{(4)} \\
\quad \text{iff } \sigma_T(T) \in \texttt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} & & \text{by Def. 12}
\end{array}
$$

(1) Since the states of a transition t do not change in $\sigma_T(t)$, then $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, and thus $\texttt{stateOf}(\pi_{\text{sor}}(t)) \in S$ iff $\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t))) \in S$.

(2) By definition of $\eta|_\sigma$, we have that $\eta|_\sigma\ (\pi_{\text{ev}}(t)) = \eta(\sigma(\pi_{\text{ev}}(t)))$. Finally, by definition of σ we have that $\eta(\sigma(\pi_{\text{ev}}(t))) = \eta(\pi_{\text{ev}}(\sigma_T(t)))$.

(3) First, $\pi_{\text{tar}}(t) = \pi_{\text{tar}}(\sigma_T(t))$ as explained in **(1)**. Then, we have that $g \in \texttt{eval}|_\sigma$ (H, s) iff $\sigma_{\mathcal{G}}(g) \in \texttt{eval}(H, s)$ by definition of σ-reduct. Finally, since $\pi_{\text{gua}}(\sigma_T(t)) = \sigma_{\mathcal{G}}(\pi_{\text{gua}}(t))$ by definition of σ_T, we conclude that $\pi_{\text{gua}}(t) \in \texttt{eval}|_\sigma\ (H, \pi_{\text{tar}}(t))$ iff $\pi_{\text{gua}}(\sigma_T(t)) \in \texttt{eval}(H, \pi_{\text{tar}}(\sigma_T(t)))$.

(4) The implication $\pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}} \Rightarrow \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) = \texttt{skip}$ can be written as $\pi_{\text{sor}}(t) \notin \mathcal{S}^{\text{cp}}_{\text{exit}} \vee \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) = \texttt{skip}$. Then, we have that $\pi_{\text{sor}}(t) \notin \mathcal{S}^{\text{cp}}_{\text{exit}}$ iff $\pi_{\text{sor}}(\sigma_T(t)) \notin \mathcal{S}^{\text{cp}}_{\text{exit}}$, as explained in **(1)**. Moreover, since $\sigma_{\mathcal{A}}$ is inyective, we have that $\alpha = \texttt{skip}$ iff $\sigma_{\mathcal{A}}(\alpha) = \sigma_{\mathcal{A}}(\texttt{skip})$ for every $\alpha \in \mathcal{A}_1$. In this case, since $\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) \in \mathcal{A}_1$ then $\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) = \texttt{skip}$ iff $\sigma_{\mathcal{A}}(\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t)))) = \sigma_{\mathcal{A}}(\texttt{skip})$, where $\sigma_{\mathcal{A}}(\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t)))) = \sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(t)))$ by definition of $\sigma_{\texttt{do}}$. Then, we have that $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, as explained in **(1)**. Finally, we have that $\pi_{\text{sor}}(t) \notin \mathcal{S}^{\text{cp}}_{\text{exit}} \vee \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) = \texttt{skip}$ iff $\sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))) = \sigma_{\mathcal{A}}(\texttt{skip})$, by definition of $\vee$, and thus $\pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}} \Rightarrow \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) = \texttt{skip}$ iff $\pi_{\text{sor}}(\sigma_T(t)) \in \mathcal{S}^{\text{cp}}_{\text{exit}} \Rightarrow \sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))) = \sigma_{\mathcal{A}}(\texttt{skip})$

$\square$

Proposition 4 (Invariance of Disable Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ (i=1, 2) with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1, 2), a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$, a Σ_2-model $\mathcal{M} = (\mathbb{A}, \mu, \texttt{eval}, \texttt{calc}, \mathbb{E}, \eta, \texttt{sel}, \texttt{join}, \rightsquigarrow, \surd)$, a configuration $\mathcal{C} = (S, \texttt{do}, H, \alpha, \ddot{s}, \beta, T, \ddot{T})$ for K_i, and an event $e \in \mathbb{E} \cup \{\tau\}$,

$$\texttt{Enable}_{K_1, \mathcal{C}, \mathcal{M}|_\sigma, e} = \emptyset \text{ iff } \texttt{Enable}_{K_2, \sigma(\mathcal{C}), \mathcal{M}, e} = \emptyset$$

Proof.

$$
\begin{array}{ll}
\texttt{Enable}_{K_1,\mathcal{C},\mathcal{M}|_\sigma,e} = \emptyset & \\
\quad \text{iff } \texttt{CoTr}_{K_1} = \emptyset & \\
\quad\quad \text{or } T \in \texttt{CoTr}_{K_1} \wedge \forall t \in T : \texttt{stateOf}(\pi_{\text{sor}}(t)) \notin S & \\
\quad\quad \text{or } T \in \texttt{CoTr}_{K_1} \wedge \forall t \in T : \eta|_\sigma\ (\pi_{\text{ev}}(t)) \neq e & \\
\quad\quad \text{or } T \in \texttt{CoTr}_{K_1} \wedge \forall t \in T : \pi_{\text{gua}}(t) \notin \texttt{eval}|_\sigma\ (H, \pi_{\text{tar}}(t)) & \\
\quad\quad \text{or } T \in \texttt{CoTr}_{K_1} \wedge \forall t \in T : \pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}} & \\
\quad\quad\quad \wedge\ \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) \neq \texttt{skip} & \text{by Def. 12} \\
\quad \text{iff } \texttt{CoTr}_{K_2} = \emptyset & \text{by Prop. 8} \\
\quad\quad \text{or } \sigma_T(T) \in \texttt{CoTr}_{K_2} & \\
\quad\quad\quad \wedge\ \forall \sigma_T(t) \in \sigma_T(T) : \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t))) \notin S & \text{by } \mathbf{(1)} \\
\quad\quad \text{or } \sigma_T(T) \in \texttt{CoTr}_{K_2} & \\
\quad\quad\quad \wedge\ \forall \sigma_T(t) \in \sigma_T(T) : \eta(\pi_{\text{ev}}(\sigma_T(t))) \neq e & \text{by } \mathbf{(2)} \\
\quad\quad \text{or } \sigma_T(T) \in \texttt{CoTr}_{K_2} & \\
\quad\quad\quad \wedge\ \forall \sigma_T(t) \in \sigma_T(T) : \pi_{\text{gua}}(\sigma_T(t)) \notin \texttt{eval}(H, \pi_{\text{tar}}(\sigma_T(t))) & \text{by } \mathbf{(3)} \\
\quad\quad \text{or } \sigma_T(T) \in \texttt{CoTr}_{K_2} & \\
\quad\quad\quad \wedge\ \forall \sigma_T(t) \in \sigma_T(T) : \pi_{\text{sor}}(\sigma_T(t)) \in \mathcal{S}^{\text{cp}}_{\text{exit}} & \\
\quad\quad\quad \wedge\ \sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))) \neq \sigma_{\mathcal{A}}(\texttt{skip}) & \text{by } \mathbf{(4)} \\
\quad \text{iff } \texttt{Enable}_{K_2,\sigma(\mathcal{C}),\mathcal{M},e} = \emptyset & \text{by Def. 12}
\end{array}
$$

(1) Since the states of a transition t do not change in $\sigma_T(t)$, then $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, and thus $\texttt{stateOf}(\pi_{\text{sor}}(t)) \notin S$ iff $\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t))) \notin S$. Also, $T \in \texttt{CoTr}_{K_1}$ iff $\sigma_T(T) \in \texttt{CoTr}_{K_2}$, by Proposition 7.

(2) By definition of $\eta|_\sigma$, we have that $\eta|_\sigma\ (\pi_{\text{ev}}(t)) = \eta(\sigma(\pi_{\text{ev}}(t)))$. Finally, by definition of σ we have that $\eta(\sigma(\pi_{\text{ev}}(t))) = \eta(\pi_{\text{ev}}(\sigma_T(t)))$. Thus, $\eta|_\sigma\ (\pi_{\text{ev}}(t)) \neq e$ iff $\eta(\pi_{\text{ev}}(\sigma_T(t))) \neq e$. Also, $T \in \texttt{CoTr}_{K_1}$ iff $\sigma_T(T) \in \texttt{CoTr}_{K_2}$, by Proposition 7.

(3) First, $\pi_{\text{tar}}(t) = \pi_{\text{tar}}(\sigma_T(t))$ as explained in **(1)**. Then, we have that $g \notin \texttt{eval}|_\sigma$ (H, s) iff $\sigma_{\mathcal{G}}(g) \notin \texttt{eval}(H, s)$ by definition of σ-reduct. Finally, since $\pi_{\text{gua}}(\sigma_T(t)) = \sigma_{\mathcal{G}}(\pi_{\text{gua}}(t))$ by definition of σ_T, we conclude that $\pi_{\text{gua}}(t) \notin \texttt{eval}|_\sigma\ (H, \pi_{\text{tar}}(t))$ iff $\pi_{\text{gua}}(\sigma_T(t)) \notin \texttt{eval}(H, \pi_{\text{tar}}(\sigma_T(t)))$. Also, $T \in \texttt{CoTr}_{K_1}$ iff $\sigma_T(T) \in \texttt{CoTr}_{K_2}$, by Proposition 7.

(4) First, we have that $\pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}}$ iff $\pi_{\text{sor}}(\sigma_T(t)) \in \mathcal{S}^{\text{cp}}_{\text{exit}}$, as explained in **(1)**. Then, since $\sigma_{\mathcal{A}}$ is inyective, we have that $\alpha \neq \texttt{skip}$ iff $\sigma_{\mathcal{A}}(\alpha) \neq \sigma_{\mathcal{A}}(\texttt{skip})$ for every $\alpha \in \mathcal{A}_1$. In this case, since $\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) \in \mathcal{A}_1$ then $\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) \neq \texttt{skip}$ iff $\sigma_{\mathcal{A}}(\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t)))) \neq \sigma_{\mathcal{A}}(\texttt{skip})$, where $\sigma_{\mathcal{A}}(\texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t)))) = \sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(t)))$ by definition of $\sigma_{\texttt{do}}$. Moreover, we have that $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, as explained in **(1)**. Finally, we have that $\pi_{\text{sor}}(t) \in \mathcal{S}^{\text{cp}}_{\text{exit}} \wedge \texttt{do}(\texttt{stateOf}(\pi_{\text{sor}}(t))) \neq \texttt{skip}$ iff $\pi_{\text{sor}}(\sigma_T(t)) \in$

$\mathcal{S}^{\text{cp}}_{\text{exit}} \wedge \sigma_{\texttt{do}}(\texttt{do})(\texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))) \neq \sigma_{\mathcal{A}}(\texttt{skip})$, by definition of $\wedge$. Also, $T \in \texttt{CoTr}_{K_1}$ iff $\sigma_T(T) \in \texttt{CoTr}_{K_2}$, by Proposition 7.

□

Proposition 5 (Invariance of Conflict Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ $(i{=}1,2)$ with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ $(i{=}1,2)$, and a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$,

$$(T_1, T_2) \in \texttt{Conflict}_{K_1} \text{ iff } (\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{Conflict}_{K_2}$$

Proof.

$$\begin{aligned}
&(T_1, T_2) \in \texttt{Conflict}_{K_1} \\
&\qquad \text{iff } (T_1, T_2) \in \texttt{CoTr}_{K_1} \times \texttt{CoTr}_{K_1} \\
&\qquad\quad \text{and } \bigcup_{t_1 \in T_1} \texttt{stateOf}(\pi_{\text{sor}}(t_1)) \\
&\qquad\qquad \cap \bigcup_{t_2 \in T_2} \texttt{stateOf}(\pi_{\text{sor}}(t_2)) \neq \emptyset && \text{by Def. 12} \\
&\qquad \text{iff } (\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{CoTr}_{K_2} \times \texttt{CoTr}_{K_2} && \text{by } \textbf{(1)} \\
&\qquad\quad \text{and } \bigcup_{\sigma_T(t_1) \in \sigma_T(T_1)} \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t_1))) \\
&\qquad\qquad \cap \bigcup_{\sigma_T(t_2) \in \sigma_T(T_2)} \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t_2))) \neq \emptyset && \text{by } \textbf{(2)} \\
&\qquad \text{iff } (\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{Conflict}_{K_2} && \text{by Def. 12}
\end{aligned}$$

(1) We have that $T \in \texttt{CoTr}_{K_1}$ iff $\sigma_T(T) \in \texttt{CoTr}_{K_2}$, as proved in Proposition 7. Finally, we conclude that $(T_1, T_2) \in \texttt{CoTr}_{K_1} \times \texttt{CoTr}_{K_1}$ iff $(\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{CoTr}_{K_2} \times \texttt{CoTr}_{K_2}$.

(2) Since the states of a transition t do not change in $\sigma_T(t)$, then $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, and thus $\texttt{stateOf}(\pi_{\text{sor}}(t)) = \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))$.
Then, we have that $\bigcup_{t \in T} \texttt{stateOf}(\pi_{\text{sor}}(t)) = \bigcup_{\sigma_T(t) \in \sigma_T(T)} \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t)))$, since σ_T is a bijection. Finally, we conclude that $\bigcup_{t_1 \in T_1} \texttt{stateOf}(\pi_{\text{sor}}(t_1)) \cap \bigcup_{t_2 \in T_2} \texttt{stateOf}(\pi_{\text{sor}}(t_2)) \neq \emptyset$ iff $\bigcup_{\sigma_T(t_1) \in \sigma_T(T_1)} \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t_1))) \cap \bigcup_{\sigma_T(t_2) \in \sigma_T(T_2)} \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t_2))) \neq \emptyset$.

□

Proposition 6 (Invariance of Priority Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ $(i{=}1,2)$ with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ $(i{=}1,2)$, and a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$,

$$(T_1, T_2) \in \texttt{Priority}_{K_1} \text{ iff } (\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{Priority}_{K_2}$$

Proof.

$$
\begin{array}{ll}
(T_1, T_2) \in \texttt{Priority}_{K_1} & \\
\quad\quad \text{iff } (T_1, T_2) \in \texttt{CoTr}_{K_1} \times \texttt{CoTr}_{K_1} & \\
\quad\quad\quad \text{and } \texttt{PrBelow}_{K_1}(T_1, T_2) & \\
\quad\quad\quad \text{and } \texttt{PrStrBelow}_{K_1}(T_1, T_2) & \text{by Def. 12} \\
\quad\quad \text{iff } (\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{CoTr}_{K_2} \times \texttt{CoTr}_{K_2} & \text{by } \textbf{(1)} \\
\quad\quad\quad \text{and } \texttt{PrBelow}_{K_2}(\sigma_T(T_1), \sigma_T(T_2)) & \text{by } \textbf{(2)} \\
\quad\quad\quad \text{and } \texttt{PrStrBelow}_{K_2}(\sigma_T(T_1), \sigma_T(T_2)) & \text{by } \textbf{(3)} \\
\quad\quad \text{iff } (\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{Priority}_{K_2} & \text{by Def. 12}
\end{array}
$$

(1) We have that $T \in \texttt{CoTr}_{K_1}$ iff $\sigma_T(T) \in \texttt{CoTr}_{K_2}$, as proved in Proposition 7. Finally, we conclude that $(T_1, T_2) \in \texttt{CoTr}_{K_1} \times \texttt{CoTr}_{K_1}$ iff $(\sigma_T(T_1), \sigma_T(T_2)) \in \texttt{CoTr}_{K_2} \times \texttt{CoTr}_{K_2}$.

(2) We have that $\texttt{PrBelow}_{K_1}(T_1, T_2)$ iff $(\forall t_1 \in T_1 : \pi_{\text{sor}}(t_1) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \Rightarrow \exists t_2 \in T_2 : \pi_{\text{sor}}(t_2) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \wedge \texttt{stateOf}(\pi_{\text{sor}}(t_2)) \succeq \texttt{stateOf}(\pi_{\text{sor}}(t_1)))$. Then, since the states of a transition t do not change in $\sigma_T(t)$, we have that $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, and thus, the proposition above holds iff $(\forall \sigma_T(t_1) \in \sigma_T(T_1) : \pi_{\text{sor}}(\sigma_T(t_1)) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \Rightarrow \exists \sigma_T(t_2) \in \sigma_T(T_2) : \pi_{\text{sor}}(\sigma_T(t_2)) \in \mathcal{S}^{\text{pr}}_{\text{exit}} \cup \mathcal{S}_{\text{com}} \wedge \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t_2))) \succeq \texttt{stateOf}(\pi_{\text{sor}}(\sigma_T(t_1))))$. Finally, we conclude that $\texttt{PrBelow}_{K_1}(T_1, T_2)$ iff $\texttt{PrBelow}_{K_2}(\sigma_T(T_1), \sigma_T(T_2))$.

(3) In the same sense as **(2)**, since the states of a transition t do not change in $\sigma_T(t)$, we conclude that $\texttt{PrStrBelow}_{K_1}(T_1, T_2)$ iff $\texttt{PrStrBelow}_{K_2}(\sigma_T(T_1), \sigma_T(T_2))$.

□

Proposition 7 (Invariance of Compound Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ (i=1, 2) with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ (i=1, 2), and a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$,

$$T \in \texttt{CoTr}_{K_1} \text{ iff } \sigma_T(T) \in \texttt{CoTr}_{K_2}$$

Proof.

$$
\begin{array}{ll}
T \in \texttt{CoTr}_{K_1} & \\
\quad\quad \text{iff } T \in \mathcal{T}_{\text{int}\,1} & \\
\quad\quad\quad \text{or } T \in \{t\} \cup T & \\
\quad\quad\quad\quad \text{and } t \in \mathcal{T}_1 & \\
\quad\quad\quad\quad \text{and } \pi_{\text{sor}}(t) \in \mathcal{S}_{\text{exit}} & \\
\quad\quad\quad\quad \text{and } \pi_{\text{tar}}(t) \notin \mathcal{S}_{\text{exit}} & \\
\quad\quad\quad\quad \text{and } T \in \Upsilon_{K_1}(\pi_{\text{sor}}(t)) & \text{by Def. 12} \\
\quad\quad \text{iff } \sigma_T(T) \in \mathcal{T}_{\text{int}\,2} & \text{by Def. 7} \\
\quad\quad\quad \text{or } \sigma_T(T) \in \{\sigma_T(t)\} \cup \sigma_T(T) & \text{by Def. of } \sigma_T \\
\quad\quad\quad\quad \text{and } \sigma_T(t) \in \mathcal{T}_2 & \text{by Def. of } \sigma_T \\
\quad\quad\quad\quad \text{and } \pi_{\text{sor}}(\sigma_T(t)) \in \mathcal{S}_{\text{exit}} & \text{by } \mathbf{(1)} \\
\quad\quad\quad\quad \text{and } \pi_{\text{tar}}(\sigma_T(t)) \notin \mathcal{S}_{\text{exit}} & \text{by } \mathbf{(1)} \\
\quad\quad\quad\quad \text{and } \sigma_T(T) \in \Upsilon_{K_2}(\pi_{\text{sor}}(\sigma_T(t))) & \text{by } \mathbf{(2)} \\
\quad\quad \text{iff } \sigma_T(T) \in \texttt{CoTr}_{K_2} & \text{by Def. 12}
\end{array}
$$

(1) Since the states of a transition t do not change in $\sigma_T(t)$, then $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, and also that $\pi_{\text{tar}}(t) = \pi_{\text{tar}}(\sigma_T(t))$.

(2) If there is a function $f : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T}_1$ which returns a transition from a region, the morphism $\sigma_T(f)$ induced by the signature morphism, is the same function where the morphism σ_T is applied to the outgoin transitions, i.e. $\sigma_T(f) : \texttt{dsr}(\texttt{stateOf}(\ddot{s})) \to \mathcal{T}_2$. The same happens with the function $F : \texttt{dsr}(s) \to 2^{\mathcal{T}}$, it means, $\sigma_T(F) : \texttt{dsr}(s) \to 2^{\mathcal{T}_2}$. Then, it hols that $\forall T \in \bigcup_{r \in \texttt{dsr}(s)} (\{f(r)\} \cup F(r))$ iff $\forall \sigma_T(T) \in \bigcup_{r \in \texttt{dsr}(s)} (\{\sigma_T(f)(r)\} \cup \sigma_T(F)(r))$ with $T \in 2^{\mathcal{T}_1}$. Moreover, we have that $\forall r \in \texttt{dsr}(s) : \pi_{\text{tar}}(f(r)) = s \wedge \texttt{regOf}(\pi_{\text{sor}}(f(r))) = r$ iff $\forall r \in \texttt{dsr}(s) : \pi_{\text{tar}}(\sigma_T(f)(r)) = s \wedge \texttt{regOf}(\pi_{\text{sor}}(\sigma_T(f)(r))) = r$, since σ_T do not modify the states of a transition, as explained in **(1)**. Finally, by the same reason, we have that $\Upsilon_K(\pi_{\text{sor}}(t))$ iff $\Upsilon_K(\pi_{\text{sor}}(\sigma_T(t)))$. In conclusion, we have that $T \in \Upsilon_{K_1}(\pi_{\text{sor}}(t))$ iff $\sigma_T(T) \in \Upsilon_{K_2}(\pi_{\text{sor}}(\sigma_T(t)))$.

$\square$

Proposition 8 (Invariance of No Compound Transitions). Given SM-signatures $\Sigma_i = (\mathcal{E}_i, \mathcal{A}_i, \mathcal{G}_i, K_i)$ $(i{=}1,2)$ with $K_i = ((\mathcal{S}, \mathcal{R}, \texttt{parent}), \texttt{doAct}_i, \texttt{defer}_i, \mathcal{T}_i, s_{\text{start}})$ $(i{=}1,2)$, and a SM-signature morphism $\sigma : \Sigma_1 \to \Sigma_2$,

$$\texttt{CoTr}_{K_1} = \emptyset \text{ iff } \texttt{CoTr}_{K_2} = \emptyset$$

Proof.

$\texttt{CoTr}_{K_1} = \emptyset$

iff $\mathcal{T}_{\text{int}\,1} = \emptyset$

and $\forall\, t \in \mathcal{T}_1 : \pi_{\text{sor}}(t) \notin \mathcal{S}_{\text{exit}} \vee \pi_{\text{tar}}(t) \in \mathcal{S}_{\text{exit}}$ by Def. 12

iff $\mathcal{T}_{\text{int}\,2} = \emptyset$ by Def. 7

and $\forall\, \sigma_T(t) \in \mathcal{T}_2 : \pi_{\text{sor}}(\sigma_T(t)) \notin \mathcal{S}_{\text{exit}} \vee \pi_{\text{tar}}(\sigma_T(t)) \in \mathcal{S}_{\text{exit}}$ by **(1)**

iff $\texttt{CoTr}_{K_2} = \emptyset$ by Def. 12

(1) Since the states of a transition t do not change in $\sigma_T(t)$, then $\pi_{\text{sor}}(t) = \pi_{\text{sor}}(\sigma_T(t))$, and also that $\pi_{\text{tar}}(t) = \pi_{\text{tar}}(\sigma_T(t))$. Also, we have that $t \in \mathcal{T}_1$ iff $\sigma_T(t) \in \mathcal{T}_2$, by definition of σ_T. Finally, we conclude that $\forall\, t \in \mathcal{T}_1 : \pi_{\text{sor}}(t) \notin \mathcal{S}_{\text{exit}} \vee \pi_{\text{tar}}(t) \in \mathcal{S}_{\text{exit}}$ iff $\forall\, \sigma_T(t) \in \mathcal{T}_2 : \pi_{\text{sor}}(\sigma_T(t)) \notin \mathcal{S}_{\text{exit}} \vee \pi_{\text{tar}}(\sigma_T(t)) \in \mathcal{S}_{\text{exit}}$.

□