Homogeneous Semantics Preserving Deployments of Heterogeneous Networks of Embedded Systems

Aaron D. Ames, Alberto Sangiovanni-Vincentelli and Shankar Sastry

Center for Hybrid and Embedded Software Systems[†]
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley
{adames,alberto,sastry}@eecs.berkeley.edu

Summary. Tagged systems provide a denotational semantics for embedded systems. A heterogeneous network of embedded systems can be modeled mathematically by a network of tagged systems. Taking the heterogeneous composition of this network results in a single, homogeneous, tagged system. The question this paper addresses is: when is semantics (behavior) preserved by composition? To answer this question, we use the framework of category theory to reason about heterogeneous system composition and derive results that are as general as possible. In particular, we define the category of tagged systems, demonstrate that a network of tagged systems corresponds to a diagram in this category and prove that taking the composition of a network of tagged systems is equivalent to taking the limit of this diagram—thus composition is endowed with a universal property. Using this universality, we are able to derive verifiable necessary and sufficient conditions on when composition preserves semantics.

1 Introduction

In an embedded system, different components of the system evolve according to processes local to the specific components. Across the entire system, these typically heterogeneous processes may not be compatible, i.e., answering questions regarding the concurrency, timing and causality of the entire system—all of which are vital in the actual physical implementation of the system—can be challenging even if these questions can be answered for specific components. Denotational semantics provide a mathematical framework in which to study the behavior (signals, flows, executions, traces) of embedded systems or networks thereof. This framework is naturally applicable to the study of heterogeneous networks of embedded systems since signals always

[†] This research is supported by the National Science Foundation (NSF award number CCR-0225610).

can be compared, regardless of the specific model of computation from which they were produced.

Tagged systems provide a denotational semantics for heterogeneous models of computation; they consist of a set of tags (a tag structure), variables and maps (behaviors) from the set of variables to the set of tags—hence, tagged systems are a specific case of the tagged signal model (cf. [12]). A heterogeneous network of embedded systems, e.g., a network consisting of both synchronous and asynchronous systems, can be modeled by a network of tagged systems with heterogeneous tag structures communicating through *mediator* tagged systems. Benveniste et al. [3], [4] and [5], introduced the notion of tagged systems and dealt with the issues we set forth in this paper; this work extends and generalizes the ideas introduced in these papers. Of course, there is a wealth of literature on semantics preservation in heterogeneous networks, cf. [6], [7], [13], [14], and [18], the last of which approaches the problem from a categorical prospective.

A network of tagged systems can be implemented, or deployed, through heterogeneous parallel composition—obtained by taking the *conjunction* (intersection) of the behaviors that agree on the mediator tagged systems—which results in a single, homogeneous, tagged system. Thus, heterogeneous networks of tagged systems can be homogenized through the operation of composition. This paper addresses the question:

When is semantics preserved by composition?

That is, when is the homogeneous tagged system obtained by composing a heterogeneous network of tagged systems semantically identical to the original network? Understanding this question is essential to understanding when networks of (possibly synchronous) embedded systems can be implemented asynchronously while preserving the semantics of the original system. Since implementing asynchronous systems is often more efficient (less overhead) when compared to the implementation of synchronous systems, deriving conditions on when this can be done while simultaneously preserving semantics would have many important implications.

In this paper, taking a similar approach to Benveniste et al., we address the issue of semantics preservation. However, we use the formalism of category theory, i.e., we introduce the category of tagged systems: TagSys to obtain more general conditions for semantics preservation. We begin by considering a network of two tagged systems $\mathscr{P}_1$ and $\mathscr{P}_2$ communicating through a mediator tagged system $\mathscr{M}$ as described by the diagram: $\mathscr{P}_1 \rightarrow \mathscr{M} \leftarrow \mathscr{P}_2$. The first contribution of this paper is that we are able to show that the (classical notion of) heterogeneous composition of $\mathscr{P}_1$ and $\mathscr{P}_2$ over $\mathscr{M}$, $\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2$, is given by the *pullback* (or *fibered product*) of this diagram:

$$\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2 = \mathscr{P}_1 \times_{\mathscr{M}} \mathscr{P}_2.$$

The importance of this result is that it implies that composition is endowed with a universal property; this universal property is fundamental in understanding when semantics is preserved. Consider the case when $\mathscr{P}_1$ and $\mathscr{P}_2$ have the same semantics, i.e., the same tag structure. Therefore, they always can communicate through the *identity mediator* tagged system, $\mathscr{I}$, and the homogeneous composition of $\mathscr{P}_1$ and $\mathscr{P}_2$, $\mathscr{P}_1 \| \mathscr{P}_2$, is given by the pullback $\mathscr{P}_1 \times_{\mathscr{I}} \mathscr{P}_2$ of the diagram: $\mathscr{P}_1 \to \mathscr{I} \leftarrow \mathscr{P}_2$. It is possible through this framework to give a precise statement of what it means to preserve semantics by composition over the mediator tagged system $\mathscr{M}$:

Semantics is preserved by composition if $\mathscr{P}_1 \| \mathscr{P}_2 \equiv \mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2$.

Through the universality of the pullback, we are able to give verifiable *necessary and sufficient conditions* on semantics preservation. A corollary of our result is the sufficient conditions on semantics preservation established by Benveniste et al..

A network of tagged systems is given by an oriented graph $\Gamma = (Q, E)$ together with a set of tagged systems $\mathscr{P} = \{\mathscr{P}_q\}_{q \in Q}$ communicating through a set of mediator tagged systems $\mathscr{M} = \{\mathscr{M}_e\}_{e \in E}$; that is, for every $e \in E$, there is a diagram in TagSys of the form:

$$\mathscr{P}_{\mathrm{source}(e)} \xrightarrow{\alpha_e} \mathscr{M}_e \xleftarrow{\alpha'_e} \mathscr{P}_{\mathrm{target}(e)}.$$

Equivalently, a network of tagged systems is given by a functor

$$\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)} : \mathsf{H}_\Gamma \to \mathsf{TagSys},$$

where H_Γ is a small category of a special form, termed an *H-category*, and obtained from Γ. As in the case of a network of two tagged systems, the heterogeneous composition of a network of tagged systems is given by the *limit* (a generalization of the pullback) of this diagram,

$$\|_{\mathscr{M}} \mathscr{P} = \varprojlim^{\mathsf{H}_\Gamma} (\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)}),$$

and so composition again is defined by a universal property. If all of the $\mathscr{P}_i' s$ have the same semantics, then we can again consider the identity mediator $\mathscr{I}$ (which in this case is a set), and the homogeneous composition of this network is given by $\|\mathscr{P} = \varprojlim^{\mathsf{H}_\Gamma}(\mathbf{P}_{(\mathscr{P},\mathscr{I},\mathrm{id})})$. This formulation allows us to give a precise statement of when semantics is preserved:

Semantics is preserved by composition if $\|_{\mathscr{M}} \mathscr{P} \equiv \|\mathscr{P}$.

The universality of composition allows us to derive concrete necessary and sufficient conditions on when semantics is preserved, indicating that this framework can produce results on semantics preservation that are both practical and verifiable.

Although applications are not specifically discussed in this work, it is important to note that there are many practical implications of the results derived in this paper. Tagged systems were originally developed in order to better understand time-trigged architectures (cf. [10]). Therefore, this theory is naturally applicable to such a framework; in fact, Benveniste et al. address locally time-trigged architectures and globally asynchronous locally synchronous architectures in [3], [4] and [5]. Addressing compositionality issues in specific architectures utilizing the results presented in this paper is a promising area of future research.

This paper is structured as follows:

Section 2: Tag structures and tagged systems are introduced, along with the corresponding categories, Tag and TagSys, respectively.

Section 3: Classical heterogeneous composition is reviewed, and it is shown that composing two tagged systems corresponds to taking the pullback of a specific diagram in TagSys.

Section 4: The general composite of two tagged systems, communicating through a mediator tagged system, is defined. The notion of semantics preservation for simple networks of this form is introduced, and necessary and sufficient conditions on semantics preservation are derived.

Section 5: Networks of tag structures, tagged systems and behaviors are defined. It is shown how to associated to these networks a functor and a small category, i.e., how these networks correspond to diagrams in the categories Tag, TagSys and Set, respectively.

Section 6: A general method for universally composing networks of tagged systems is introduced. This is related to the composite of corresponding networks of tag structures and behaviors.

Section 7: The notion of a semantics preserving deployment of a network of tagged systems is introduced. Necessary and sufficient conditions on semantics preservation are derived.

Section 8: Concluding remarks are given.

2 A Categorical Formulation of Tagged Systems

In this section, we begin by defining the category of tag structures. This definition is used to understand how to associate a common tag structure to a pair of tag structures which can communicate through a mediator tag structure. Later, it will be seen how composing tagged systems mirrors this construction on tag structures. Before discussing composition, we first must introduce the category of these systems. This category will be instrumental later in understanding how to form the heterogeneous composition of a network of tagged systems.

Tag structures and the corresponding category. Fundamental to the notion of tagged systems is the notion of timing. This timing is encoded in a set of tags; these “tag” the occurrences of events, i.e., they index the events such that they are (partially) ordered. Hence, a *set of tags* or a *tag structure* is a partially ordered set $\mathcal{T}$, with the partial order denoted by $\leq$. The category of tags, Tag, can be defined as follows:

Objects: Partially ordered sets, i.e., tag structures.
Morphisms: Nondecreasing maps between sets $\rho : \mathcal{T} \to \mathcal{T}'$, i.e., if $t \leq t' \in \mathcal{T}$ then $\rho(t) \leq \rho(t') \in \mathcal{T}'$.
Composition: The standard composition of maps between sets.

Clearly, two objects in the category Tag are isomorphic if $\rho : \mathcal{T} \to \mathcal{T}'$ is a bijection: there exists a $\rho' : \mathcal{T}' \to \mathcal{T}$ such that $\rho \circ \rho' = \mathrm{id}_{\mathcal{T}'}$ and $\rho' \circ \rho = \mathrm{id}_{\mathcal{T}}$. Note also that the *terminal objects* in the category Tag are just one point sets $\mathcal{T}_{\mathrm{triv}} := \{*\}$ (called asynchronous tag structures), i.e., for all tag structures $\mathcal{T}$ there exists a unique morphism $\rho : \mathcal{T} \to \mathcal{T}_{\mathrm{triv}}$ defined by $\rho(t) \equiv *$ which desynchronizes the tag structure. The synchronous tag structure is given by $\mathcal{T}_{\mathrm{sync}} = \mathbb{N}$.

Common tag structures. Tags are fundamental in understanding tagged systems in that morphisms of tag structures will induce morphisms of tagged systems. To better understand this, we will discuss an important operation on tag structures: the pullback (the pullback of elements in a category will be used extensively in the paper—see [11] for a formal definition). Consider two tag structures $\mathcal{T}_1$ and $\mathcal{T}_2$. We would like to find a tag structure that is more general than $\mathcal{T}_1$ and $\mathcal{T}_2$ and has morphisms to both of these tag structures, i.e., we would like to find a *common tag structure* for these two tag structures. To do this, first consider the diagram in Tag:

$$\mathcal{T}_1 \xrightarrow{\rho_1} \mathcal{T} \xleftarrow{\rho_2} \mathcal{T}_2, \tag{1}$$

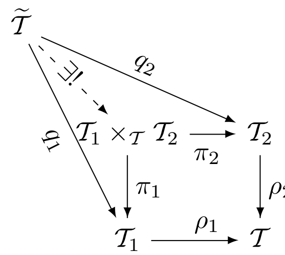
where $\mathcal{T}$ is the *mediator tag structure*. We know that such a tag structure always exists since it always can be taken to be $\mathcal{T}_{\mathrm{triv}}$ (although this rarely is the wisest choice). We define the common tag structure to be the pullback of this diagram:

$$\mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2 = \{(t_1, t_2) \in \mathcal{T}_1 \times \mathcal{T}_2 : \rho_1(t_1) = \rho(t_2)\}. \tag{2}$$

The pullback is the desired common tag structure since it sits in a commutative diagram of the form:

$$\begin{array}{ccc} \mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2 & \xrightarrow{\pi_2} & \mathcal{T}_2 \\ {\scriptstyle \pi_1}\downarrow & & \downarrow{\scriptstyle \rho_2} \\ \mathcal{T}_1 & \xrightarrow{\rho_1} & \mathcal{T} \end{array} \tag{3}$$

Moreover, that fact that the common tag structure is the pullback implies that for any other tag structure that displays the properties of a common tag structure, there exists a unique morphism from this tag structure to *the* common tag structure. More precisely, for any tag structure $\widetilde{\mathcal{T}}$ such that the following diagram commutes:



(4)

there exists a unique morphism from $\widetilde{\mathcal{T}}$ to $\mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2$ also making the diagram commute. This construction on tag structures both motivates and mirrors constructions that will be performed throughout this paper on tagged systems. To demonstrate this we must, as with tag structures, define tagged systems and the associated category.

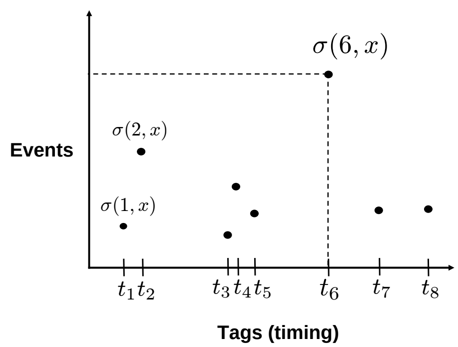


Fig. 1. A graphical representation of a behavior of a tagged system.

Tagged systems. Following from [3], [4] and [5] (although our notation slightly deviates from theirs), we define a tagged system. We then proceed to introduce the category of tagged systems.

Let $\mathcal{V}$ be an underlying set of variables and D be the set of values that these variables can take, i.e., the domain of the variables. A tagged system is a tuple

$$\mathscr{P} = (V, \mathcal{T}, \Sigma),$$

where V is a finite subset of the underlying set of variables $\mathcal{V}$, $\mathcal{T}$ is a tag structure, i.e., an object of Tag, and Σ is a set of maps:

$$\sigma : \mathbb{N} \times V \to \mathcal{T} \times D.$$

Each of the elements of Σ, i.e., each of the maps σ, are referred to as V-*behaviors* (or just *behaviors* when the variable set is understood). It is required that for each $v \in V$, the map $\delta_v(n) := \pi_1(\sigma(n, v)) : \mathbb{N} \to \mathcal{T}$ is a morphism in Tag, that is, nondecreasing (and called a *clock* in [5]).

Remark 1. In defining the set of behaviors of a tagged system, we made an explicit choice for the domain of the behaviors: $\mathbb{N} \times V$. This choice is motivated by the fact that the behaviors of a tagged system are signals generated by a computer, and hence discrete in nature. It is possible to consider other domains for the behaviors, e.g., $\mathbb{R} \times V$, without any significate change to the theory introduced here. This indicates an interesting extension of this work to behavioral dynamical system theory (cf. [15, 16, 17]).

The category of tagged systems. We can use the formulation of tagged systems above in order to define the category of tagged systems, TagSys, as follows:

Objects: Tagged systems $\mathscr{P} = (V, \mathcal{T}, \Sigma)$.

Morphisms: A morphism of tagged systems $\alpha : \mathscr{P} = (V, \mathcal{T}, \Sigma) \to \mathscr{P}' = (V', \mathcal{T}', \Sigma')$ is a morphism (in the category of sets) of behaviors $\alpha : \Sigma \to \Sigma'$.

Composition: The standard composition of maps between sets. In other words, for $\alpha : \mathscr{P} \to \mathscr{P}'$ and $\alpha' : \mathscr{P}' \to \mathscr{P}''$ the composition of $\alpha : \Sigma \to \Sigma'$ and $\alpha' : \Sigma' \to \Sigma''$ is given by $\alpha' \circ \alpha : \Sigma \to \Sigma''$.

From the definition of morphisms in the category TagSys, it follows that two tagged systems, $\mathscr{P}$ and $\mathscr{P}'$, are isomorphic, $\mathscr{P} \cong \mathscr{P}'$, if and only if, to use the terminology from the literature, the two tagged systems are in bijective correspondence.

A "forgetfull" functor. By the definition of the category TagSys, there is a *fully faithful* functor (cf. [11]):

$$\mathbf{F} : \mathsf{TagSys} \to \mathsf{Set},$$

where Set is the category of sets. This functor is defined on objects and morphisms in TagSys as follows: for every diagram in TagSys of the form:

$$\mathscr{P} = (V, \mathcal{T}, \Sigma) \xrightarrow{\alpha} \mathscr{P}' = (V', \mathcal{T}', \Sigma'),$$

the functor $\mathbf{F}$ is given by:

$$\mathbf{F}\left(\mathscr{P} = (V, \mathcal{T}, \Sigma) \xrightarrow{\alpha} \mathscr{P}' = (V', \mathcal{T}', \Sigma')\right) = \Sigma \xrightarrow{\alpha} \Sigma'.$$

When discussing composition, we often will blur the distinction between the categories TagSys and Set, i.e., we often will define the composition of a diagram of tagged systems by the behaviors of the composite system, and hence implicitly view it as an object of Set. In this case, we always will construct an object in TagSys with behaviors isomorphic to the corresponding object in Set.

Induced morphisms of tag systems. Suppose that there is a morphism of tag structures $\rho : \mathcal{T} \to \mathcal{T}'$. Then there exists a tagged system $\mathscr{P}^\rho$ together with an induced morphism of tagged systems $(\,\cdot\,)^\rho : \mathscr{P} \to \mathscr{P}^\rho$. First, if $\mathscr{P} = (V, \mathcal{T}, \Sigma)$ we define $\mathscr{P}^\rho = (V, \mathcal{T}', \Sigma^\rho)$ where

$$\Sigma^\rho := \{\sigma^\rho : \mathbb{N} \times V \to \mathcal{T}' \times D : \sigma^\rho(n,v) = (\rho(t), d) \text{ iff } (t,d) = \sigma(n,v) \text{ for some } \sigma \in \Sigma\}.$$

That is, Σ^ρ is defined by replacing t with $\rho(t)$ in the codomain of σ. With this definition of $\mathscr{P}^\rho$, we obtain a morphism $(\,\cdot\,)^\rho : \mathscr{P} \to \mathscr{P}^\rho$, called the *desynchronization morphism* and defined by, for each $\sigma \in \Sigma$,

$$\sigma^\rho(n,v) = (\rho(t), d) \quad \overset{\text{def}}{\Leftrightarrow} \quad \sigma(n,v) = (t,d).$$

Note that $(\,\cdot\,)^\rho$ is always surjective.

Example 1. Consider the following synchronous tagged systems, $\mathscr{P}_1$ and $\mathscr{P}_2$, defined as follows:

$$\mathscr{P}_1 := (V_1 = \{x\}, \mathcal{T}_{\text{sync}} = \mathbb{N}, \Sigma_1 = \{\sigma_1\}),$$
$$\mathscr{P}_2 := (V_2 = \{x, y\}, \mathcal{T}_{\text{sync}} = \mathbb{N}, \Sigma_2 = \{\sigma_2, \tilde{\sigma}_2\}),$$

where

$$\sigma_1(n,x) := (m(n), \star),$$
$$\sigma_2(n,v) := \begin{cases} (m(n), \star) & \text{if } v = x \in V_2 \\ (k(n), \star) & \text{if } v = y \in V_2 \end{cases},$$
$$\tilde{\sigma}_2(n,v) := \begin{cases} (m(n), \star) & \text{if } v = x \in V_2 \\ (l(n), \star) & \text{if } v = y \in V_2 \end{cases}.$$

Here $m(n)$, $k(n)$ and $l(n)$ are any strictly increasing functions with $k(n) \neq l(n)$, and $\star$ is a (single) arbitrary value in D.

For $\rho : \mathcal{T}_{\text{sync}} \to \mathcal{T}_{\text{triv}}$ the desyncronization morphism, $\mathscr{P}_1 \cong \mathscr{P}_1^\rho$ because $\mathscr{P}_1$ consists of a single behavior. Since $\Sigma_2 = \{\sigma_2, \tilde{\sigma}_2\}$, $\Sigma_2^\rho = \{\sigma_2^\rho = \tilde{\sigma}_2^\rho\}$, i.e., Σ_2^ρ consists of a single behavior. Therefore, $\mathscr{P}_2$ is not in bijective correspondence with $\mathscr{P}_2^\rho$

3 Universal Heterogeneous Composition

In this section, we discuss how to take the composition of a "simple" network of embedded systems. Given two tagged systems, $\mathscr{P}_1$ and $\mathscr{P}_2$, we would like to form their composition, i.e., a single tagged system obtainable from these two tagged systems. We begin by reviewing the "standard" definition of composition, followed by a categorical reformulation of composition. We demonstrate that the composition of two tagged systems corresponds to the pullback of a specific diagram in the category of tagged systems. This will allow us later to generalize the notion of composition.

Heterogeneous composition. Let $\mathscr{P}_1 = (V_1, \mathcal{T}_1, \Sigma_1)$ and $\mathscr{P}_2 = (V_2, \mathcal{T}_2, \Sigma_2)$ be two tagged systems. Consider a mediator tag structure $\mathcal{T}$ between the tag structures $\mathcal{T}_1$ and $\mathcal{T}_2$, i.e., there exists a diagram in Tag:

$$\mathcal{T}_1 \xrightarrow{\rho_1} \mathcal{T} \xleftarrow{\rho_2} \mathcal{T}_2$$

Recall that the common tag structure to $\mathcal{T}_1$ and $\mathcal{T}_2$ (relative to $\mathcal{T}$) is given by the pullback of the above diagram in Tag (the fibered product as defined in (2)): $\mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2$.

We define the parallel composition of $\mathscr{P}_1$ and $\mathscr{P}_2$ over the mediator tag structure $\mathcal{T}$ by

$$\mathscr{P}_1 \parallel_{\mathcal{T}} \mathscr{P}_2 := (V_1 \cup V_2, \mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2, \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2).$$

This notation for parallel composition is taken from [3]; the morphisms ρ_1 and ρ_2 are implicit in this notation. In the above definition, $\Sigma_1 \wedge_{\mathcal{T}} \Sigma_2$ is given by the set of behaviors

$$\sigma : \mathbb{N} \times (V_1 \cup V_2) \to (\mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2) \times D$$

such that the following condition holds: for all $(n, v) \in \mathbb{N} \times (V_1 \cup V_2)$, there exist unique $\sigma_1 \in \Sigma_1$ and $\sigma_2 \in \Sigma_2$ such that[3]

$$\sigma(n,v) = ((t_1,t_2),d) \quad \Leftrightarrow \quad \begin{array}{lc} \text{(i)} & \sigma_1(n,v) = (t_1,d) \text{ if } v \in V_1 \\ & \text{and} \\ \text{(ii)} & \sigma_2(n,v) = (t_2,d) \text{ if } v \in V_2 \\ & \text{and} \\ \text{(iii)} & \sigma_1^{\rho_1}(n,v) = \sigma_2^{\rho_2}(n,v) \text{ if } v \in V_1 \cap V_2. \end{array} \tag{5}$$

Since σ is uniquely determined by σ_1 and σ_2, and vise-versa, we write $\sigma = \sigma_1 \sqcup_{\mathcal{T}} \sigma_2$. We pick this notation so as to be consistent with the literature, cf. [3], where a pair $(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2$ is called *unifiable* when it satisfies condition (iii), and $\sigma_1 \sqcup_{\mathcal{T}} \sigma_2$ is called the *unification* of σ_1 and σ_2. We will always assume that such a pair exists; in this case composition is well-defined ($\Sigma_1 \wedge_{\mathcal{T}} \Sigma_2$ is not the empty set).

Universal heterogeneous composition. The common tag structure for the composition of two tagged systems is given by the pullback of a certain diagram. The natural question to ask is: can the composition of two tagged systems be realized as the pullback of a diagram of tagged systems of the form:

[3] Note that conditions (i) and (ii) imply condition (iii); this follows from the fact that $(t_1, t_2) \in \mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2$, so $\rho_1(t_1) = \rho_2(t_2)$ in condition (i) and (ii). Condition (iii) is stated for the sake of clarity.

$$\mathscr{P}_1 \xrightarrow{\alpha_1} \mathscr{M} \xleftarrow{\alpha_2} \mathscr{P}_2?$$

The importance of this question is that if the answer is yes, then the composition between two heterogeneous tagged systems is universal, i.e., defined by a universal property. We then can ask when the composition of two tagged systems is the same as the composition of these tagged systems with different tag structures, i.e., when semantics is preserved. It is possible to show that composition is in fact given by a universal property.

In order to define composition universally, we must define the tagged system $\mathscr{M}$ in the above diagram. In this vein, and using the same notation as the above paragraph, define

$$\mathscr{I}_{\mathcal{T}} := (V_1 \cap V_2, \mathcal{T}, \Sigma_1 \vee_{\mathcal{T}} \Sigma_2), \tag{6}$$

where $\mathcal{T}$ is a mediator tag structure (between $\mathcal{T}_1$ and $\mathcal{T}_2$) and

$$\Sigma_1 \vee_{\mathcal{T}} \Sigma_2 := \{\sigma : \mathbb{N} \times (V_1 \cap V_2) \to \mathcal{T} \times D : \\ \sigma = \sigma_i^{\rho_i}|_{V_1 \cap V_2}, \text{ for } \sigma_i \in \Sigma_i, \ i = 1 \text{ or } 2\}. \tag{7}$$

Now for the tagged systems $\mathscr{P}_1$ and $\mathscr{P}_2$ there exist morphisms

$$\mathscr{P}_1 \xrightarrow{\mathrm{Res}_1^{\rho_1}} \mathscr{I}_{\mathcal{T}} \xleftarrow{\mathrm{Res}_2^{\rho_2}} \mathscr{P}_2 \tag{8}$$

defined as follows:

$$\mathrm{Res}_i^{\rho_i}(\sigma_i) = \sigma_i^{\rho_i}|_{V_1 \cap V_2} : \mathbb{N} \times (V_1 \cap V_2) \to \mathcal{T} \times D$$

for $\sigma_i \in \Sigma_i$, $i = 1, 2$. Clearly, such a morphism always exists.

Note that $\mathscr{I}_{\mathcal{T}}$ is a *mediator tagged system* or *channel* between the tagged systems $\mathscr{P}_1$ and $\mathscr{P}_2$; $\mathscr{I}_{\mathcal{T}}$ "communicates" between $\mathscr{P}_1$ and $\mathscr{P}_2$. In the case when $\mathcal{T} = \mathcal{T}_1 = \mathcal{T}_2$, $\mathscr{I} := \mathscr{I}_{\mathcal{T}}$ is exactly the *identity mediator tagged system* or *identity channel* introduced in [3].

Theorem 1. *Consider two tagged systems $\mathscr{P}_1 = (V_1, \mathcal{T}_1, \Sigma_1)$ and $\mathscr{P}_2 = (V_2, \mathcal{T}_2, \Sigma_2)$ with mediator tag structure $\mathcal{T}$, i.e., suppose that there is a diagram in* Tag*:*

$$\mathcal{T}_1 \xrightarrow{\rho_1} \mathcal{T} \xleftarrow{\rho_2} \mathcal{T}_2.$$

The parallel composition of $\mathscr{P}_1$ and $\mathscr{P}_2$ over this tag structure, $\mathscr{P}_1 \parallel_{\mathcal{T}} \mathscr{P}_2$, is the pullback of the diagram:

$$\mathscr{P}_1 \xrightarrow{\mathrm{Res}_1^{\rho_1}} \mathscr{I}_{\mathcal{T}} \xleftarrow{\mathrm{Res}_2^{\rho_2}} \mathscr{P}_2$$

in the category of tagged systems, TagSys*.*

Implications of Theorem 1. Before proving Theorem 1, we discuss some of the implications of this theorem.

If we consider the following diagram in the category of sets, Set:

$$\Sigma_1 \xrightarrow{\mathrm{Res}_1^{\rho_1}} \Sigma_1 \vee_{\mathcal{T}} \Sigma_2 \xleftarrow{\mathrm{Res}_2^{\rho_2}} \Sigma_2 = \mathbf{F}\left(\mathscr{P}_1 \xrightarrow{\mathrm{Res}_1^{\rho_1}} \mathscr{I}_{\mathcal{T}} \xleftarrow{\mathrm{Res}_2^{\rho_2}} \mathscr{P}_2 \right),$$

the pullback of this diagram is given by:

$$\Sigma_1 \times_{\Sigma_1 \vee_{\mathcal{T}} \Sigma_2} \Sigma_2 = \{(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2 : \mathrm{Res}_1^{\rho_1}(\sigma_1) = \mathrm{Res}_2^{\rho_2}(\sigma_2)\}.$$

It is important to note that the pullback of the above diagram (which is an object in Set) is related to—in fact, isomorphic to—the behavior of the tagged system $\mathscr{P}_1 \parallel_{\mathcal{T}} \mathscr{P}_2$. More precisely, for the functor $\mathbf{F} : \mathsf{TagSys} \to \mathsf{Set}$, we have:

$$\Sigma_1 \times_{\Sigma_1 \vee_{\mathcal{T}} \Sigma_2} \Sigma_2 \cong \mathbf{F}(\mathscr{P}_1 \parallel_{\mathcal{T}} \mathscr{P}_2) = \Sigma_1 \vee_{\mathcal{T}} \Sigma_2. \tag{9}$$

This observation will allow us later to, justifiably, blur the distinction between pullbacks (and limits) in the categories TagSys and Set.

In order to produce the bijection given in (9), first note that there are projections defined by:

$$\pi_i : \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2 \to \Sigma_i \tag{10}$$

where for each $\sigma_1 \sqcup_{\mathcal{T}} \sigma_2 \in \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2$, $\pi_i(\sigma_1 \sqcup_{\mathcal{T}} \sigma_2) := \sigma_i$ for $i = 1, 2$. Because of (5), it follows that any element $\sigma \in \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2$ can be written as $\sigma = \pi_1(\sigma) \sqcup_{\mathcal{T}} \pi_2(\sigma)$.

Theorem 1 implies—by the universality of the pullback—that there is a bijection:

$$\begin{aligned} (\pi_1, \pi_2) : \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2 &\xrightarrow{\sim} \Sigma_1 \times_{\Sigma_1 \vee_{\mathcal{T}} \Sigma_2} \Sigma_2 \\ \sigma = \sigma_1 \sqcup_{\mathcal{T}} \sigma_2 &\mapsto (\sigma_1 = \pi_1(\sigma), \sigma_2 = \pi_2(\sigma)) \end{aligned} \tag{11}$$

where the inverse of this map is the unification operator:

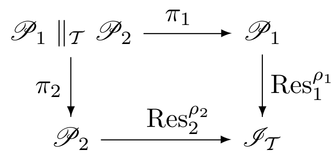
$$\begin{aligned} (\,\cdot\,) \sqcup_{\mathcal{T}} (\,\cdot\,) : \Sigma_1 \times_{\Sigma_1 \vee_{\mathcal{T}} \Sigma_2} \Sigma_2 &\xrightarrow{\sim} \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2 \\ (\sigma_1, \sigma_2) &\mapsto \sigma_1 \sqcup_{\mathcal{T}} \sigma_2 \end{aligned} \tag{12}$$

This completes the description of the bijection given in (9).

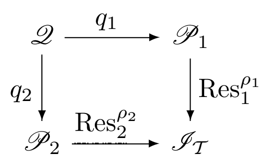
Proof (of Theorem 1). From the definition of $\Sigma_1 \wedge_{\mathcal{T}} \Sigma_2$ and $\Sigma_1 \vee_{\mathcal{T}} \Sigma_2$, it follows that the following diagram in Set:

$$\begin{array}{ccc} \Sigma_1 \wedge_{\mathcal{T}} \Sigma_2 & \xrightarrow{\pi_1} & \Sigma_1 \\ {\scriptstyle \pi_2}\downarrow & & \downarrow{\scriptstyle \mathrm{Res}_1^{\rho_1}} \\ \Sigma_2 & \xrightarrow{\mathrm{Res}_2^{\rho_2}} & \Sigma_1 \vee_{\mathcal{T}} \Sigma_2 \end{array}$$

commutes, which implies, by the definition of morphisms of tagged systems, that the following diagram in TagSys:



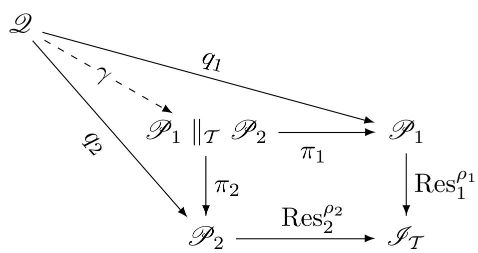
commutes. Consider a tagged system $\mathscr{Q}$ such that the following diagram commutes:



We can define a morphism $\gamma : \mathscr{Q} \to \mathscr{P}_1 \parallel_{\mathcal{T}} \mathscr{P}_2$ by, for $\sigma \in \Sigma_{\mathscr{Q}}$,

$$\gamma(\sigma) = q_1(\sigma) \sqcup_{\mathcal{T}} q_1(\sigma).$$

It follows that there is a commuting diagram:



Moreover, by replacing γ with any other morphism making the diagram commute, say $\tilde{\gamma}$, it follows that for $\sigma \in \Sigma_{\mathscr{Q}}$

$$\tilde{\gamma}(\sigma) = \pi_1(\tilde{\gamma}(\sigma)) \sqcup_{\mathcal{T}} \pi_2(\tilde{\gamma}(\sigma)) = q_1(\sigma) \sqcup_{\mathcal{T}} q_2(\sigma) = \gamma(\sigma).$$

So $\gamma = \tilde{\gamma}$, i.e., γ is unique.

4 Equivalent Deployments of Tagged Systems

Standard composition is just the pullback of a specific diagram in TagSys; this observation naturally allows us to generalize composition. To perform this generalization, we introduce the notion of a general mediator tagged system, $\mathscr{M}$, and define composition to be the pullback of a diagram of the form:

$$\mathscr{P}_1 \xrightarrow{\alpha_1} \mathscr{M} \xleftarrow{\alpha_2} \mathscr{P}_2 \qquad (13)$$

in TagSys. This process will be instrumental later in understanding how to take the composition of more general networks of embedded systems.

We conclude this section by reviewing the definition of semantics preservation and giving necessary and sufficient conditions on when semantics is preserved. We apply these results to the special case of semantics preservation through desyncronization.

Composition through mediation. Given the results of Theorem 1, we can develop a more intuitive notation for composition. Specifically, if $\mathcal{T}$ is the mediator tag structure, then we write $\mathscr{P}_1 \parallel_{\mathcal{T}} \mathscr{P}_2 = \mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2$ (in the case when $\mathcal{T}_1 = \mathcal{T}_2 = \mathcal{T}$ and $\rho_1 = \rho_2 = \mathrm{id}$ in (1), we just write $\mathscr{P}_1 \| \mathscr{P}_2 := \mathscr{P}_1 \|_{\mathscr{I}} \mathscr{P}_2$). The mathematical reason for this is that $\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2$ is (isomorphic to) $\mathscr{P}_1 \times_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2$, i.e., the pullback of the diagram given in (8). The philosophical motivation for this notation is that the composition of $\mathscr{P}_1$ and $\mathscr{P}_2$ can be taken over general mediator tagged systems. In other words, the parallel composition of $\mathscr{P}_1$ and $\mathscr{P}_2$ over a general *mediator tagged system* $\mathscr{M}$, denoted by $\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2$, is defined to be the pullback of the diagram given in (13):

$$\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2 := \mathscr{P}_1 \times_{\mathscr{M}} \mathscr{P}_2 .$$

This implies that if $\Sigma_{\mathscr{M}}$ is the set of behaviors of $\mathscr{M}$, then the set of behaviors for $\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2$ is isomorphic to:

$$\Sigma_1 \times_{\Sigma_{\mathscr{M}}} \Sigma_2 = \{(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2 : \alpha_1(\sigma_1) = \alpha_2(\sigma_2)\}. \tag{14}$$

The explicit construction of $\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2$ as a tagged system is not especially relevant, as we are interested only in its set of behaviors which must be isomorphic to the set of behaviors given in (14). That being said, this construction is a special case of a more general construction (given in Section 6) for which the construction of the tagged system is carried out. We only note that there are bijections

$$(\cdot) \sqcup_{\mathscr{M}} (\cdot) : \Sigma_1 \times_{\Sigma_{\mathscr{M}}} \Sigma_2 \xrightarrow{\sim} \Sigma_{\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2} \tag{15}$$

$$(\pi_1, \pi_2) : \Sigma_{\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2} \xrightarrow{\sim} \Sigma_1 \times_{\Sigma_{\mathscr{M}}} \Sigma_2 \tag{16}$$

which are generalizations of the unification and projection maps given in (12) and (11), respectively.

Specification vs. deployment. Consider two tagged systems $\mathscr{P}_1$ and $\mathscr{P}_2$ with a mediator tag structure $\mathcal{T}$. As in [4] (although with some generalization, since $\mathscr{P}_1$ and $\mathscr{P}_2$ are not assumed to have the same tag structure), we define the following semantics:

Specification Semantics: $\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2$
Deployment Semantics: $\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2$

for some mediator tagged system $\mathscr{M}$. The natural question to ask is when are the specification semantics and the deployment semantics "equivalent."

Formally, and following from [4], we define a mediator $\mathscr{M}$ to be *semantics preserving with respect to* $\mathscr{I}_{\mathcal{T}}$, denoted by

$$\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2 \equiv \mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2 \tag{17}$$

if for all $(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2$,

$$\begin{array}{c} \exists\ \sigma' \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2} \text{ s.t. } \pi_1(\sigma') = \sigma_1 \text{ and } \pi_2(\sigma') = \sigma_2 \\ \Updownarrow \\ \exists\ \sigma \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2} \text{ s.t. } \pi_1(\sigma) = \sigma_1 \text{ and } \pi_2(\sigma) = \sigma_2. \end{array} \tag{18}$$

Utilizing Theorem 1, we have the following necessary and sufficient conditions on semantics preservation.

Theorem 2. *For two tagged systems* $\mathscr{P}_1$ *and* $\mathscr{P}_2$*,*

$$\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2 \equiv \mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2,$$

if and only if for all $(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2$*:*

$$\alpha_1(\sigma_1) = \alpha_2(\sigma_2) \quad \Leftrightarrow \quad \mathrm{Res}_1^{\rho_1}(\sigma_1) = \mathrm{Res}_2^{\rho_2}(\sigma_2)$$

Proof. **(Sufficiency:)** If

$$(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2 \text{ s.t. } \alpha_1(\sigma_1) = \alpha_2(\sigma_2)$$

$$\begin{array}{ll} \overset{\text{by (14)}}{\Rightarrow} & (\sigma_1, \sigma_2) \in \Sigma_1 \times_{\Sigma_{\mathscr{M}}} \Sigma_2 \\ \overset{\text{by (15)}}{\Rightarrow} & \sigma_1 \sqcup_{\mathscr{M}} \sigma_2 \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2} \text{ where} \\ & \qquad \pi_1(\sigma_1 \sqcup_{\mathscr{M}} \sigma_2) = \sigma_1 \text{ and } \pi_2(\sigma_1 \sqcup_{\mathscr{M}} \sigma_2) = \sigma_2 \\ \overset{\text{by (18)}}{\Rightarrow} & \exists\ \sigma \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2} \text{ s.t. } \pi_1(\sigma) = \sigma_1 \text{ and } \pi_2(\sigma) = \sigma_2 \\ \overset{\text{by (16)}}{\Rightarrow} & (\sigma_1, \sigma_2) \in \Sigma_1 \times_{\Sigma_{\mathscr{I}_{\mathcal{T}}}} \Sigma_2 \\ \overset{\text{by (14)}}{\Rightarrow} & \mathrm{Res}_1^{\rho_1}(\sigma_1) = \mathrm{Res}_2^{\rho_2}(\sigma_2). \end{array}$$

The converse direction proceeds in the same manner: if

$$(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2 \text{ s.t. } \mathrm{Res}_1^{\rho_1}(\sigma_1) = \mathrm{Res}_2^{\rho_2}(\sigma_2) \qquad \Rightarrow \qquad \alpha_1(\sigma_1) = \alpha_2(\sigma_2),$$

by (14), (15), (16), and (18).
(Necessity:) We have the following implications:

$$\exists\ \sigma' \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2} \text{ s.t. } \pi_1(\sigma') = \sigma_1 \text{ and } \pi_2(\sigma') = \sigma_2$$

$$\begin{array}{ll} \overset{\text{by (16)}}{\Rightarrow} & (\sigma_1, \sigma_2) \in \Sigma_1 \times_{\Sigma_{\mathscr{M}}} \Sigma_2 \\ \overset{\text{by (14)}}{\Rightarrow} & \alpha_1(\sigma_1) = \alpha_2(\sigma_2) \\ \Rightarrow & \mathrm{Res}_1^{\rho_1}(\sigma_1) = \mathrm{Res}_2^{\rho_2}(\sigma_2) \\ \overset{\text{by (14)}}{\Rightarrow} & (\sigma_1, \sigma_2) \in \Sigma_1 \times_{\Sigma_{\mathscr{I}_{\mathcal{T}}}} \Sigma_2 \\ \overset{\text{by (15)}}{\Rightarrow} & \sigma_1 \sqcup_{\mathscr{I}_{\mathcal{T}}} \sigma_2 \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}}} \mathscr{P}_2} \text{ and} \\ & \qquad \pi_1(\sigma_1 \sqcup_{\mathscr{I}_{\mathcal{T}}} \sigma_2) = \sigma_1 \text{ and } \pi_2(\sigma_1 \sqcup_{\mathscr{I}_{\mathcal{T}}} \sigma_2) = \sigma_2. \end{array}$$

Therefore $\sigma_1 \sqcup_{\mathscr{I}_\mathcal{T}} \sigma_2$ is the element of $\Sigma_{\mathscr{P}_1 \|_{\mathscr{I}_\mathcal{T}} \mathscr{P}_2}$ such that $\pi_1(\sigma_1 \sqcup_{\mathscr{I}_\mathcal{T}} \sigma_2) = \sigma_1$ and $\pi_2(\sigma_1 \sqcup_{\mathscr{I}_\mathcal{T}} \sigma_2) = \sigma_2$, as desired.

The other direction follows in the same way:

$$\exists\ \sigma \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{I}_\mathcal{T}} \mathscr{P}_2} \text{ s.t. } \pi_1(\sigma) = \sigma_1 \text{ and } \pi_2(\sigma) = \sigma_2$$

$$\Rightarrow \qquad \begin{array}{l} \sigma_1 \sqcup_{\mathscr{M}} \sigma_2 \in \Sigma_{\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2} \text{ and} \\ \quad \pi_1(\sigma_1 \sqcup_{\mathscr{M}} \sigma_2) = \sigma_1 \text{ and } \pi_2(\sigma_1 \sqcup_{\mathscr{M}} \sigma_2) = \sigma_2, \end{array}$$

by (14), (15) and (16).

To demonstrate the power of Theorem 2, we prove the following theorem, which is a generalization of one of the two main theorems of [3]. Moveover, we show that the theorem in [3] is a corollary of this theorem; thus, our results are more general. First, we review the general set-up for this theorem.

Desynchronization. Consider the case when $\mathscr{P}_1$ and $\mathscr{P}_2$ have the same tag structure, i.e., $\mathcal{T}_1 = \mathcal{T}_2 = \mathcal{T}$. Consider a mediator tag structure $\mathcal{T}'$ of $\mathcal{T}$, i.e., suppose there exists a diagram in Tag:

$$\mathcal{T} \xrightarrow{\ \rho\ } \mathcal{T}' \xleftarrow{\ \rho\ } \mathcal{T}.$$

In this case, we ask when the mediator tagged system $\mathscr{I}_{\mathcal{T}'}$ is semantics preserving, i.e., when

$$\mathscr{P}_1 \| \mathscr{P}_2 \equiv \mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}'}} \mathscr{P}_2. \tag{19}$$

A very important example of when this framework is useful is in the desynchronization of tagged systems; in this case $\mathcal{T}' = \mathcal{T}_{\text{triv}} = \{*\}$, and $\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}'}} \mathscr{P}_2$ is the *desynchronization* of $\mathscr{P}_1$ and $\mathscr{P}_2$.

Using the notation of this paragraph, we have the following theorem and its corollary.

Theorem 3. *$\mathscr{I}_{\mathcal{T}'}$ is semantics preserving w.r.t. $\mathscr{I}$, $\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}'}} \mathscr{P}_2 \equiv \mathscr{P}_1 \| \mathscr{P}_2$, if and only if for all $(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2$:*

$$\sigma_1^\rho|_{V_1 \cap V_2} = \sigma_2^\rho|_{V_1 \cap V_2} \qquad \Rightarrow \qquad \sigma_1|_{V_1 \cap V_2} = \sigma_2|_{V_1 \cap V_2}.$$

Proof. Note that by the definition of the desynchronization morphism $(\,\cdot\,)^\rho$ (and the fact that it is always surjective), it follows that

$$\sigma_1|_{V_1 \cap V_2} = \sigma_2|_{V_1 \cap V_2} \qquad \Rightarrow \qquad \sigma_1^\rho|_{V_1 \cap V_2} = \sigma_2^\rho|_{V_1 \cap V_2}.$$

Therefore, this result is a corollary of Theorem 2.

Corollary 1. *If $\mathscr{P}_i^\rho$ is in bijection with $\mathscr{P}_i$ for $i = 1, 2$ and $(\mathscr{P}_1 \| \mathscr{P}_2)^\rho = \mathscr{P}_1^\rho \| \mathscr{P}_2^\rho$, then $\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}'}} \mathscr{P}_2 \equiv \mathscr{P}_1 \| \mathscr{P}_2$ ($\mathscr{I}_{\mathcal{T}'}$ is semantics preserving w.r.t. $\mathscr{I}$).*

Proof. We need only show that the suppositions of the theorem imply for all $(\sigma_1, \sigma_2) \in \Sigma_1 \times \Sigma_2$

$$\sigma_1^\rho|_{V_1 \cap V_2} = \sigma_2^\rho|_{V_1 \cap V_2} \qquad \Rightarrow \qquad \sigma_1|_{V_1 \cap V_2} = \sigma_2|_{V_1 \cap V_2}.$$

The result then follows from Theorem 3.

To see that the desired implication holds, note that we have the following chain of implications:

$$\sigma_1^\rho|_{V_1 \cap V_2} = \sigma_2^\rho|_{V_1 \cap V_2}$$

$$\begin{array}{lcl} \Rightarrow & (\sigma_1, \sigma_2) \in \Sigma_1 \times_{\Sigma_{\mathscr{I}_{\mathcal{T}'}}} \Sigma_2 \cong \Sigma_{\mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}'}} \mathscr{P}_2} & \\ \Rightarrow & (\sigma_1^\rho, \sigma_2^\rho) \in \Sigma_1^\rho \times_{\Sigma_{\mathscr{I}_{\mathcal{T}'}}} \Sigma_2^\rho \cong \Sigma_{\mathscr{P}_1^\rho \| \mathscr{P}_2^\rho} & \\ \Rightarrow & \sigma_1^\rho \sqcup_{\mathscr{I}_{\mathcal{T}'}} \sigma_2^\rho \in \Sigma^\rho_{\mathscr{P}_1 \| \mathscr{P}_2} & (\text{since } (\mathscr{P}_1 \| \mathscr{P}_2)^\rho = \mathscr{P}_1^\rho \| \mathscr{P}_2^\rho) \\ \Rightarrow & \exists\ \tilde{\sigma} \in \Sigma_{\mathscr{P}_1 \| \mathscr{P}_2} & \text{s.t.} \quad \tilde{\sigma}^\rho = \sigma_1^\rho \sqcup_{\mathscr{I}_{\mathcal{T}'}} \sigma_2^\rho. \end{array}$$

Setting $\tilde{\sigma}_i = \pi_i(\tilde{\sigma})$, the last of these implications implies that $(\tilde{\sigma}_1^\rho, \tilde{\sigma}_2^\rho) = (\sigma_1^\rho, \sigma_2^\rho) \in \Sigma_1^\rho \times_{\Sigma_{\mathscr{I}_{\mathcal{T}'}}} \Sigma_2^\rho$. Now, the fact that $\mathscr{P}_i^\rho$ is in bijection with $\mathscr{P}_i$ for $i = 1, 2$ implies that $\tilde{\sigma}_i = \sigma_i$, or:

$$(\sigma_1, \sigma_2) = (\tilde{\sigma}_1, \tilde{\sigma}_2) \in \Sigma_1 \times_{\Sigma_{\mathscr{I}_{\mathcal{T}}}} \Sigma_2 \qquad \Rightarrow \qquad \sigma_1|_{V_1 \cap V_2} = \sigma_2|_{V_1 \cap V_2}.$$

Example 2. We would like to know semantics is preserved by desynchronization for the tagged systems given in Example 1, i.e., for $\rho : \mathcal{T}_{\text{sync}} \to \mathcal{T}_{\text{triv}}$ the desyncronization morphism, is $\mathscr{P}_1 \| \mathscr{P}_2 \equiv \mathscr{P}_1 \|_{\mathscr{I}_{\mathcal{T}_{\text{triv}}}} \mathscr{P}_2$?

First we apply the necessary and sufficient conditions given in Theorem 3. Since $V_1 \cap V_2 = \{x\}$,

$$\begin{array}{lcl} \sigma_1^\rho(n,x) = \sigma_2^\rho(n,x) & \Rightarrow & \sigma_1(n,x) = \sigma_2(n,x) \\ \sigma_1^\rho(n,x) = \tilde{\sigma}_2^\rho(n,x) & \Rightarrow & \sigma_1(n,x) = \tilde{\sigma}_2(n,x) \end{array}$$

because $\sigma_1(n,x) = \sigma_2(n,x)$ and $\sigma_1(n,x) = \tilde{\sigma}_2(n,x)$. Therefore, semantics is preserved.

Note that Corollary 1 would not tell us whether semantics is preserved, because $\mathscr{P}_2$ is not in bijective correspondence with $\mathscr{P}_2^\rho$, and so the conditions of the corollary do not hold. This demonstrates that Theorem 3 is a stronger result than Corollary 1.

5 Networks of Tagged Systems

In this section, we introduce the notion of a network of tag structures, tagged systems and behaviors. Moreover, we are able to show that these objects correspond to diagrams in Tag, TagSys and Set, respectively. This observation will be fundamental in defining composition for these networks.

Networks of tag structures. We begin by defining a network of tag structures as in [4] (although we state the definition in a slightly different manner). A *network of tag structures* is defined to be a tuple

$$(\Gamma, \mathcal{T}, \mathcal{S}, \rho),$$

where

- Γ is an oriented graph with Q the set of vertices and E the set of edges; for $e = (i, j) \in E$, denote the source of e by $\mathfrak{s}(e) = i$ and the target of e by $\mathfrak{t}(e) = j$.
- $\mathcal{T} = \{\mathcal{T}_q\}_{q \in Q}$ is a set of tag structures.
- $\mathcal{S} = \{\mathcal{S}_e\}_{e \in E}$ is a set of mediator tag structures, mediating between $\mathcal{T}_{\mathfrak{s}(e)}$ and $\mathcal{T}_{\mathfrak{t}(e)}$.
- $\rho = \{(\rho_e, \rho'_e)\}_{e \in E}$ is a set of pairs of morphisms in Tag, such that for every $e \in E$, there is the following diagram in Tag:

$$\mathcal{T}_{\mathfrak{s}(e)} \xrightarrow{\rho_e} \mathcal{S}_e \xleftarrow{\rho'_e} \mathcal{T}_{\mathfrak{t}(e)}.$$

Networks of tagged systems are defined in an analogous manner.

Networks of tagged systems. A *network of tagged systems* is defined to be a tuple

$$(\Gamma, \mathscr{P}, \mathscr{M}, \alpha),$$

where

- Γ is an oriented graph.
- $\mathscr{P} = \{\mathscr{P}_q\}_{q \in Q}$ is a set of tag structures.
- $\mathscr{M} = \{\mathscr{M}_e\}_{e \in E}$ is a set of mediator tagged systems, mediating between $\mathscr{P}_{\mathfrak{s}(e)}$ and $\mathscr{P}_{\mathfrak{t}(e)}$.
- $\alpha = \{(\alpha_e, \alpha'_e)\}_{e \in E}$ is a set of pairs of morphisms in TagSys, such that for every $e \in E$, there is the following diagram in TagSys:

$$\mathscr{P}_{\mathfrak{s}(e)} \xrightarrow{\alpha_e} \mathscr{M}_e \xleftarrow{\alpha'_e} \mathscr{P}_{\mathfrak{t}(e)}.$$

Suppose we have a network of tag structures $(\Gamma, \mathcal{T}, \mathcal{S}, \rho)$ and a collection of tagged systems $\mathscr{P} = \{\mathscr{P}_q\}_{q \in Q}$ such that $\mathscr{P}_q$ has tag structure $\mathcal{T}_q$. Then we can associate to this set of tagged systems a network of tagged systems, $(\Gamma, \mathscr{P}, \mathscr{I}_{\mathcal{S}}, \mathrm{Res}^{\rho})$ with:

$$\mathscr{I}_{\mathcal{S}} = \{\mathscr{I}_{\mathcal{S}_e}\}_{e \in \mathrm{e}}, \qquad \mathrm{Res}^{\rho} = \{(\mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}, \mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)})\}_{e \in E}$$

where $\mathscr{I}_{\mathcal{S}_e}$ is defined as in (6), and $\mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}$ and $\mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)}$ are defined as in (8), i.e., there is a diagram in TagSys of the form:

$$\mathscr{P}_{\mathfrak{s}(e)} \xrightarrow{\mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}} \mathscr{I}_{\mathcal{S}_e} \xleftarrow{\mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)}} \mathscr{P}_{\mathfrak{t}(e)}.$$

for every $e \in E$.

Networks of behaviors. We can define a *network of behaviors* from the network of tagged systems, $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$, as a tuple:

$$(\Gamma, \Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha),$$

where

- Γ is an oriented graph.
- $\Sigma_{\mathscr{P}} = \{\Sigma_{\mathscr{P}_q}\}_{q \in Q}$, where $\Sigma_{\mathscr{P}_q}$ is the set of behaviors for $\mathscr{P}_q$.
- $\Sigma_{\mathscr{M}} = \{\Sigma_{\mathscr{M}_e}\}_{e \in E}$, where $\Sigma_{\mathscr{M}_e}$ is the set of behaviors for $\mathscr{M}_e$.
- $\alpha = \{(\alpha_e, \alpha'_e)\}_{e \in E}$ is a set of pairs of morphisms in Set, such that for every $e \in E$, there is the following diagram in Set:

$$\Sigma_{\mathscr{P}_{\mathfrak{s}(e)}} \xrightarrow{\alpha_e} \Sigma_{\mathscr{M}_e} \xleftarrow{\alpha'_e} \Sigma_{\mathscr{P}_{\mathfrak{t}(e)}}.$$

The association of a network of behaviors from a network of tagged systems can be viewed categorically. For the network of tagged systems $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$, the functor $\mathbf{F} : \mathsf{TagSys} \to \mathsf{Set}$ yields the corresponding network of behaviors $(\Gamma, \Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha)$ because

$$\begin{aligned} \Sigma_{\mathscr{P}} &= \{\Sigma_{\mathscr{P}_q}\}_{q \in Q} = \{\mathbf{F}(\mathscr{P}_q)\}_{q \in Q} \\ \Sigma_{\mathscr{M}} &= \{\Sigma_{\mathscr{M}_e}\}_{e \in E} = \{\mathbf{F}(\mathscr{M}_e)\}_{e \in E} \end{aligned}$$

More generally,

$$\Sigma_{\mathscr{P}_{\mathfrak{s}(e)}} \xrightarrow{\alpha_e} \Sigma_{\mathscr{M}_e} \xleftarrow{\alpha'_e} \Sigma_{\mathscr{P}_{\mathfrak{t}(e)}} = \mathbf{F}\left(\mathscr{P}_{\mathfrak{s}(e)} \xrightarrow{\alpha_e} \mathscr{M}_e \xleftarrow{\alpha'_e} \mathscr{P}_{\mathfrak{t}(e)}\right)$$

for every $e \in E$.

H-cateogries. The goal is to define a network of tagged systems as a diagram in TagSys; to do this, we must first define a specific type of small category termed an *H-category*[4] and denoted by H. This is a small category in which every diagram has the form:[5]

$$\bullet \longrightarrow \bullet \longleftarrow \bullet \longrightarrow \bullet \longleftarrow \bullet \quad \cdots \quad \bullet \longrightarrow \bullet \longleftarrow \bullet$$

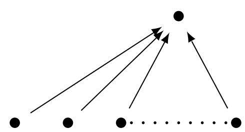
That is, an H-category has as its basic atomic unit a diagram of the form: $\bullet \longrightarrow \bullet \longleftarrow \bullet$, and any other diagram in this category must be obtainable by gluing such atomic units along the source of a morphism (and not the target). More formally, a small category, H, is an H-category if it satisfies the following two axioms:

[4] The category H considered here actually is the opposite category to a H-small category, as defined in [1]. The surprising thing about this is that these categories were considered in the context of hybrid systems; this suggests some interesting relationships between hybrid and embedded systems.

[5] where $\bullet$ denotes an arbitrary object in H and $\longrightarrow$ denotes an arbitrary morphism.

1. Every object in H is either the source of a non-identity morphism in H or the target of a non-identity morphism, but never both; i.e., for every diagram

   in H, all but one morphism must be the identity (the longest chain of composable non-identity morphisms is of length one).
2. If an object in H is the target of a non-identity morphism, then it is the target of exactly two non-identity morphisms, i.e., for every diagram in H of the form



   either all of the morphisms are the identity or two and only two morphisms are not the identity.

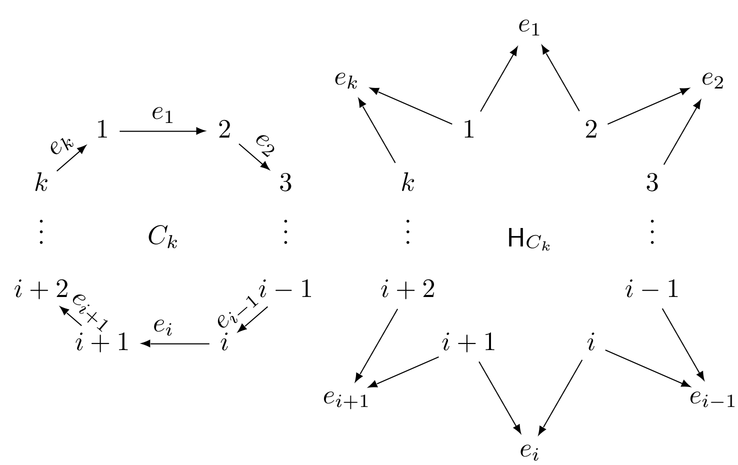
More on H-categories can be found in [2] (where the H-categories defined in [2] are the opposite categories to the ones considered in this paper).

We can associate to an oriented graph Γ an H-category H_Γ by, for every edge in $e \in E$, defining the following diagram

$$\mathfrak{s}(e) \longrightarrow e \longleftarrow \mathfrak{t}(e)$$

in H_Γ. Of course, the identity morphisms must be added to each object in H_Γ in order to complete the definition. More generally, the category of (oriented) H-categories can be formed, Hcat, and it can be demonstrated (cf. [2]) that this category is isomorphic to the category of graphs: $\mathsf{Hcat} \cong \mathsf{Grph}$.

Example 3. The following diagram shows an oriented cycle graph, $\Gamma = C_k$, and the associated H-category H_{C_k}:



Diagrams as functors. A diagram in a category is just a collection of objects together with a collection of morphisms between these objects. Equivalently, a diagram in a category can be viewed as a functor from a small category to this category. This motivates an equivalent, arguably more simple and certainly more abstract, definition of a network of tag structures and a network of tagged systems.

We define a network of tag structures and a network of tagged systems to be functors:

$$\mathbf{T} : \mathsf{H} \to \mathsf{Tag}, \qquad \mathbf{P} : \mathsf{H} \to \mathsf{TagSys},$$

where H is an H-category. For example, if H is given by the diagram: $\bullet \longrightarrow \bullet \longleftarrow \bullet$, then the network of tag structures given in (1) and the network of tagged systems given in (13) are defined, respectively, by the functors:

$$\mathbf{T}(\bullet \longrightarrow \bullet \longleftarrow \bullet) = \left(\mathcal{T}_1 \xrightarrow{\rho_1} \mathcal{T} \xleftarrow{\rho_2} \mathcal{T}_2 \right), \tag{20}$$

$$\mathbf{P}(\bullet \longrightarrow \bullet \longleftarrow \bullet) = \left(\mathscr{P}_1 \xrightarrow{\alpha_1} \mathscr{M} \xleftarrow{\alpha_2} \mathscr{P}_2 \right). \tag{21}$$

More generally, we can associate to a network of tag structures $(\Gamma, \mathcal{T}, \mathcal{S}, \rho)$ and a network of tagged systems $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$ functors:

$$\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)} : \mathsf{H}_\Gamma \to \mathsf{Tag}, \qquad \mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)} : \mathsf{H}_\Gamma \to \mathsf{TagSys},$$

where H_Γ is the H-category associated with Γ, and $\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)}$ and $\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)}$ are defined by:

$$\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)}\,(\mathfrak{s}(e) \longrightarrow e \longleftarrow \mathfrak{t}(e)) := \left(\mathcal{T}_{\mathfrak{s}(e)} \xrightarrow{\rho_e} \mathcal{S}_e \xleftarrow{\rho'_e} \mathcal{T}_{\mathfrak{t}(e)} \right), \tag{22}$$

$$\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)}\,(\mathfrak{s}(e) \longrightarrow e \longleftarrow \mathfrak{t}(e)) := \left(\mathscr{P}_{\mathfrak{s}(e)} \xrightarrow{\alpha_e} \mathscr{M}_e \xleftarrow{\alpha'_e} \mathscr{P}_{\mathfrak{t}(e)} \right), \tag{23}$$

for every $e \in E$.

If $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$ is a network of tagged systems, and $(\Gamma, \Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha)$ is the associated network of behaviors, then there is a functor

$$\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)} : \mathsf{H}_\Gamma \to \mathsf{Set},$$

where H_Γ is the H-category associated with Γ, and $\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}$ is defined to be the composite:

$$\mathsf{H}_\Gamma \xrightarrow{\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)}} \mathsf{TagSys} \xrightarrow{\mathbf{F}} \mathsf{Set}\,.$$

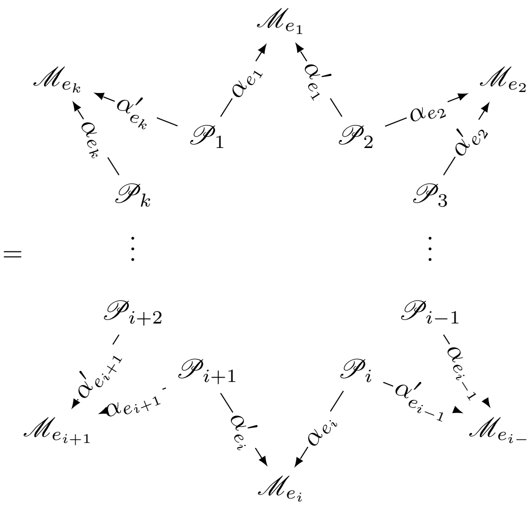
In other words, $\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}$ is defined by:

$$\begin{aligned} \mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}\,(\mathfrak{s}(e) \longrightarrow e \longleftarrow \mathfrak{t}(e)) &:= \left(\Sigma_{\mathscr{P}_{\mathfrak{s}(e)}} \xrightarrow{\alpha_e} \Sigma_{\mathscr{M}_e} \xleftarrow{\alpha'_e} \Sigma_{\mathscr{P}_{\mathfrak{t}(e)}} \right), \\ &= \mathbf{F}\left(\mathscr{P}_{\mathfrak{s}(e)} \xrightarrow{\alpha_e} \mathscr{M}_e \xleftarrow{\alpha'_e} \mathscr{P}_{\mathfrak{t}(e)} \right) \end{aligned}$$

for every $e \in E$.

Example 4. Continuing Example 3, an example of a network of tagged systems associated to the H-category H_{C_k} is given in the following diagram:

$\mathbf{P}(\mathsf{H}_{C_k}) =$



6 Universally Composing Networks of Tagged Systems

In this section, we give a categorical formulation for composition. Since a network is just a diagram, the composition of a network is the limit of this diagram. To illustrate this concept, we first consider a network of tag structures and demonstrate how the taking the composition of this network is consistent with the notion of a common tag structure as first introduced in Section 2. We then discuss how these ideas can be generalized to networks of tagged systems. Finally, we explicitly relate the composite of a network of tagged systems with the composite of a network of behaviors. This relationship will be important when attempting to prove results relating to semantics preservation.

Composing networks of tag structures. Recall that for two tag structures, $\mathcal{T}_1$ and $\mathcal{T}_2$, communicating through a mediator tag structure, $\mathcal{T}$, we obtained a single, common, tag structure that was unique up to isomorphism; the common tag structure, $\mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2$, was the pullback of the diagram given in (1). But the pullback is just a special case of the *limit* (or *inverse limit* or *projective limit*; see [11] for more details) of a functor, i.e.,

$$\mathcal{T}_1 \times_{\mathcal{T}} \mathcal{T}_2 = \varprojlim^{(\bullet \longrightarrow \bullet \longleftarrow \bullet)}(\mathbf{T}) = \varprojlim^{(\bullet \longrightarrow \bullet \longleftarrow \bullet)} \left(\mathcal{T}_1 \xrightarrow{\rho_1} \mathcal{T} \xleftarrow{\rho_2} \mathcal{T}_2 \right),$$

where $\mathbf{T} : (\bullet \longrightarrow \bullet \longleftarrow \bullet) \to \mathsf{Tag}$ is defined as in (20).

Therefore, we can define a tag structure common to an entire network of tag structures by taking the limit of the corresponding diagram in Tag describing this network. If $(\Gamma, \mathcal{T}, \mathcal{S}, \rho)$ is a network of tag structures and

$$\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)} : \mathsf{H}_\Gamma \to \mathsf{Tag}$$

is the corresponding functor and H-category, we define the common tag structure to be $\varprojlim^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)})$, which because of the special structure of an H-category is given by:

$$\varprojlim^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)}) = \left\{ (t_q)_{q\in Q} \in \prod_{q\in Q} \mathcal{T}_q : \rho_e(t_{\mathfrak{s}(e)}) = \rho'_e(t_{\mathfrak{t}(e)}), \quad \forall\ e \in E \right\},$$

which corresponds to the common tag structure defined in [4]. By the properties of the limit, we know that this is in fact the desired common tag structure since for every $e \in E$, we have a diagram of the form

$$\begin{array}{ccc}
\varprojlim^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)}) & \xrightarrow{\pi_{\mathfrak{t}(e)}} & \mathcal{T}_{\mathfrak{t}(e)} \\
{\scriptstyle \pi_{\mathfrak{s}(e)}}\big\downarrow & & \big\downarrow{\scriptstyle \rho'_e} \\
\mathcal{T}_{\mathfrak{s}(e)} & \xrightarrow{\rho_e} & \mathcal{S}_e
\end{array}$$

which is a direct generalization of (3). Moreover, the limit is universal in the same sense as (4).

Composing networks of tagged systems. As with networks of tag structures, we can consider the limit of a network of tagged systems (when viewed as a diagram)—this is the heterogeneous composition of the network. This is justified by the discussion in Section 4, where the composition of a network of two tagged systems, $\mathscr{P}_1$ and $\mathscr{P}_2$, was defined to be the pullback of these systems over the mediator tagged system, $\mathscr{M}$:

$$\mathscr{P}_1 \|_{\mathscr{M}} \mathscr{P}_2 = \varprojlim^{(\bullet\longrightarrow\bullet\longleftarrow\bullet)}(\mathbf{P}) = \varprojlim^{(\bullet\longrightarrow\bullet\longleftarrow\bullet)}\left(\mathscr{P}_1 \xrightarrow{\alpha_1} \mathscr{M} \xleftarrow{\alpha_2} \mathscr{P}_2 \right),$$

where $\mathbf{P} : (\bullet \longrightarrow \bullet \longleftarrow \bullet) \to \mathsf{Tag}$ is defined as in (20).

This indicates a general, and universal, way of taking the composition of a network of tagged systems: through the limit. Consider a network of tagged systems $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$, with the tagged systems $\mathscr{P}_q$ and $\mathscr{M}_e$ given by

$$\begin{aligned}
\mathscr{P}_q &= (V_q, \mathcal{T}_q, \Sigma_q), \qquad q \in Q, \\
\mathscr{M}_e &= (V_e, \mathcal{S}_e, \Sigma_e), \qquad e \in E.
\end{aligned}$$

For the corresponding functor and H-category:

$$\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)} : \mathsf{H}_\Gamma \to \mathsf{TagSys}$$

denote the heterogeneous composition of $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$ by $\|_{\mathscr{M}} \mathscr{P}$ (to be consistent with the notation of [4]) and define it by (unlike [4])

$$\|_{\mathscr{M}} \mathscr{P} := \varprojlim^{\mathsf{H}_\Gamma} (\mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)}) = \left(\bigcup_{q \in Q} V_q, \prod_{q \in Q} \mathcal{T}_q, \Sigma_{\|_{\mathscr{M}} \mathscr{P}} \right), \tag{24}$$

where $\Sigma_{\|_{\mathscr{M}} \mathscr{P}}$ is the set of behaviors

$$\begin{aligned} \sigma : \mathbb{N} \times \bigcup_{q \in Q} V_q &\to \prod_{q \in Q} \mathcal{T}_q \times D \\ \sigma(n, v) &\mapsto ((t_q)_{q \in Q}, d) \end{aligned}$$

such that the following conditions hold: for all $(n, v) \in \mathbb{N} \times \bigcup_{q \in Q} V_q$, there exists unique $(\sigma_q)_{q \in Q} \in \prod_{q \in Q} \Sigma_q$ such that[6] for all $e \in E$

$$\sigma(n, v) = ((t_q)_{q \in Q}, d) \;\Leftrightarrow\; \begin{array}{ll} \text{(i}'\text{)} & \sigma_{\mathfrak{s}(e)}(n, v) = (t_{\mathfrak{s}(e)}, d) \text{ if } v \in V_{\mathfrak{s}(e)} \\ & \text{and} \\ \text{(ii}'\text{)} & \sigma_{\mathfrak{t}(e)}(n, v) = (t_{\mathfrak{t}(e)}, d) \text{ if } v \in V_{\mathfrak{t}(e)} \\ & \text{and} \\ \text{(iii}'\text{)} & \alpha_e(\sigma_{\mathfrak{s}(e)})(n, w) = \alpha'_e(\sigma_{\mathfrak{t}(e)})(n, w) \\ & \forall\ w \in V_e. \end{array} \tag{25}$$

Because $\sigma \in \Sigma_{\|_{\mathscr{M}} \mathscr{P}}$ is uniquely determined by $(\sigma_q)_{q \in Q} \in \prod_{q \in Q} \Sigma_q$ satisfying the right-hand side of (25), we write

$$\sigma = \bigsqcup_{\mathscr{M}} (\sigma_q)_{q \in Q} \in \Sigma_{\|_{\mathscr{M}} \mathscr{P}}$$

for the corresponding element on the left-hand side of (25), and call it the *unification* of $(\sigma_q)_{q \in Q} \in \prod_{q \in Q} \Sigma_q$. Conversely, every element of $\Sigma_{\|_{\mathscr{M}} \mathscr{P}}$ can be written as the unification of an element of $\prod_{q \in Q} \Sigma_q$, so there are projection maps π_q, $q \in Q$, given by

$$\begin{aligned} \pi_q : \Sigma_{\|_{\mathscr{M}} \mathscr{P}} &\to \Sigma_q \\ \sigma = \bigsqcup_{\mathscr{M}} (\sigma_q)_{q \in Q} &\mapsto \sigma_q = \pi_q(\sigma). \end{aligned}$$

We can obtain a better understanding of the behaviors of the tagged system $\|_{\mathscr{M}} \mathscr{P}$ by considering the associated network of behaviors.

Composing networks of behaviors. Let $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$ be a network of tagged systems, $(\Gamma, \Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha)$ the associated network of behaviors, and

$$\mathbf{S}_{(\Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha)} = \mathbf{F} \circ \mathbf{P}_{(\mathscr{P},\mathscr{M},\alpha)} : \mathsf{H}_\Gamma \to \mathsf{Set},$$

[6] Unlike (5), the third condition stated here is no longer redundant.

the associated functor and H-category. Because of the special structure of an H-category, we can explicitly compute the limit of the functor $\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}$; it is given by

$$\underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}) = \left\{ (\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q : \alpha_e(\sigma_{\mathfrak{s}(e)}) = \alpha'_e(\sigma_{\mathfrak{t}(e)}), \quad \forall\ e \in E \right\}. \tag{26}$$

Now, there is a bijection:

$$\Sigma_{\|_{\mathscr{M}}\mathscr{P}} \cong \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}). \tag{27}$$

The map from $\Sigma_{\|_{\mathscr{M}}\mathscr{P}}$ to $\underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)})$ is given by

$$\begin{aligned} (\pi_q)_{q\in Q} : \Sigma_{\|_{\mathscr{M}}\mathscr{P}} &\xrightarrow{\sim} \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}) \\ \sigma &\mapsto (\pi_q(\sigma))_{q\in Q}. \end{aligned} \tag{28}$$

The inverse of this map is given by the unification operator generalized to the network case. That is

$$\begin{aligned} \bigsqcup_{\mathscr{M}}(\,\cdot\,) : \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}) &\xrightarrow{\sim} \Sigma_{\|_{\mathscr{M}}\mathscr{P}} \\ (\sigma_q)_{q\in Q} &\mapsto \sigma = \bigsqcup_{\mathscr{M}}(\sigma_q)_{q\in Q}, \end{aligned} \tag{29}$$

where σ is given as in the left-hand side of equation (25), which is well defined because of the definition of $\underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)})$, i.e., because an element $(\sigma_q)_{q\in Q} \in \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)})$ automatically satisfies the right-hand side of (25) by (26).

Composition over identity mediator tagged systems. An especially interesting case is when the network of tagged systems is obtained from a network of tag structures $(\Gamma, \mathcal{T}, \mathcal{S}, \rho)$, i.e., the network of tagged systems is given by $(\Gamma, \mathscr{P}, \mathscr{I}_{\mathcal{S}}, \mathrm{Res}^\rho)$. Here we explicitly carry out the construction of $\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}$, and demonstrate how this yields the correct definition of $\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}$ so as to be consistent with [4].

If $\mathscr{P}_q = (V_q, \mathcal{T}_q, \Sigma_q)$ for all $q \in Q$, then

$$\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P} = \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{P}_{(\mathscr{P},\mathscr{I}_{\mathcal{S}},\mathrm{Res}^\rho)}) = \left(\bigcup_{q\in Q} V_q, \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)}), \Sigma_{\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}} \right),$$

where $\Sigma_{\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}}$ is defined in the same way as $\Sigma_{\|_{\mathscr{M}}\mathscr{P}}$ with the appropriate modifications, i.e., $\Sigma_{\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}}$ is the set of behaviors

$$\sigma : \mathbb{N} \times \bigcup_{q\in Q} V_q \to \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)}) \times D \subset \prod_{q\in Q} \mathcal{T}_q \times D$$
$$\sigma(n,v) \mapsto ((t_q)_{q\in Q}, d)\,,$$

such that the following conditions hold: for all $(n,v) \in \mathbb{N} \times \bigcup_{q\in Q} V_q$, there exists unique $(\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q$ such that[7] for all $e \in E$

$$\sigma(n,v) = ((t_q)_{q\in Q}, d) \;\Leftrightarrow\; \begin{array}{ll} \text{(i}''\text{)} & \sigma_{\mathfrak{s}(e)}(n,v) = (t_{\mathfrak{s}(e)}, d) \text{ if } v \in V_{\mathfrak{s}(e)} \\ & \text{and} \\ \text{(ii}''\text{)} & \sigma_{\mathfrak{t}(e)}(n,v) = (t_{\mathfrak{t}(e)}, d) \text{ if } v \in V_{\mathfrak{t}(e)} \\ & \text{and} \\ \text{(iii}''\text{)} & \sigma^{\rho_e}_{\mathfrak{s}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}} = \sigma^{\rho'_e}_{\mathfrak{t}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}}. \end{array} \tag{30}$$

Note that the fact that $\sigma \in \Sigma_{\|_{\mathscr{I}_\mathcal{S}}\mathscr{P}}$ takes values in $\underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)})$, rather than $\prod_{q\in Q} \mathcal{T}_q$, is exactly because of condition (iii$''$).

The conditions given in (30) demonstrate that our definition of $\Sigma_{\|_{\mathscr{I}_\mathcal{S}}\mathscr{P}}$ is consistent with the one given in [4] (although our definition of $\Sigma_{\|_{\mathscr{M}}\mathscr{P}}$ is more general that anything defined in that paper). Moreover, (26), (27) and (30) imply that we have the following bijection

$$\begin{aligned} \Sigma_{\|_{\mathscr{I}_\mathcal{S}}\mathscr{P}} &\cong \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_\mathscr{P},\Sigma_{\mathscr{I}_\mathcal{S}},\mathrm{Res}^\rho)}) \\ &= \left\{ (\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q : \sigma^{\rho_e}_{\mathfrak{s}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}} = \sigma^{\rho'_e}_{\mathfrak{t}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}},\ \forall\ e \in E \right\} \end{aligned} \tag{31}$$

as defined in (28) and (29). Here

$$\mathbf{S}_{(\Sigma_\mathscr{P},\Sigma_{\mathscr{I}_\mathcal{S}},\mathrm{Res}^\rho)} = \mathbf{F} \circ \mathbf{P}_{(\mathscr{P},\mathscr{I}_\mathcal{S},\mathrm{Res}^\rho)} : \mathsf{H}_\Gamma \to \mathsf{Set}$$

is the functor corresponding to the network of behaviors $(\Gamma, \Sigma_\mathscr{P}, \Sigma_{\mathscr{I}_\mathcal{S}}, \mathrm{Res}^\rho)$ obtained from the network of tagged systems $(\Gamma, \mathscr{P}, \mathscr{I}_\mathcal{S}, \mathrm{Res}^\rho)$.

7 Semantics Preserving Deployments of Networks

Using the framework established in this paper, we are able to introduce a general notion of semantics preservation. After this concept is introduced, we state the main result of this work: necessary and sufficient conditions for semantics preservation. We conclude this section by applying this result to the specific case of network desyncronization.

Network specification vs. network deployment. Generalizing the notion of specification vs. deployment given in Section 4, we define the following semantics (using the notation of the previous paragraph):

---

[7] Like (5) and unlike (25), the third condition stated here is again redundant because we are taking $\underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{T}_{(\mathcal{T},\mathcal{S},\rho)})$ as our tag structure.

Network Specification Semantics: $\|_{\mathscr{I}_{\mathcal{S}}} \mathscr{P}$
Network Deployment Semantics: $\|_{\mathscr{M}} \mathscr{P}$

The set of mediator tagged systems $\mathscr{M}$ is said to be *semantics preserving with respect to* $\mathscr{I}_{\mathcal{S}}$, denoted by

$$\|_{\mathscr{M}} \mathscr{P} \equiv \|_{\mathscr{I}_{\mathcal{S}}} \mathscr{P}$$

if for all $(\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q$

$$\begin{array}{c} \exists\ \sigma' \in \Sigma_{\|_{\mathscr{M}} \mathscr{P}} \text{ s.t. } \pi_q(\sigma') = \sigma_q \quad \forall\ q \in Q \\ \Updownarrow \\ \exists\ \sigma \in \Sigma_{\|_{\mathscr{I}_{\mathcal{S}}} \mathscr{P}} \text{ s.t. } \pi_q(\sigma) = \sigma_q \quad \forall\ q \in Q. \end{array} \tag{32}$$

We now are able to generalize the results given in Theorem 2 on semantics preservation to the networks of tagged systems case.

Theorem 4. *For the networks,* $(\Gamma, \mathscr{P}, \mathscr{M}, \alpha)$ *and* $(\Gamma, \mathscr{P}, \mathscr{I}_{\mathcal{S}}, \mathrm{Res}^{\rho})$*,*

$$\|_{\mathscr{M}} \mathscr{P} \equiv \|_{\mathscr{I}_{\mathcal{S}}} \mathscr{P}$$

if and only if for all $(\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q$ *and all* $e \in E$*:*

$$\alpha_e(\sigma_{\mathfrak{s}(e)}) = \alpha'_e(\sigma_{\mathfrak{t}(e)}) \quad \Leftrightarrow \quad \mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}(\sigma_{\mathfrak{s}(e)}) = \mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)}(\sigma_{\mathfrak{t}(e)}).$$

Proof. **(Sufficiency:)** If

$$(\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q \text{ s.t. } \alpha_e(\sigma_{\mathfrak{s}(e)}) = \alpha'_e(\sigma_{\mathfrak{t}(e)}) \quad \forall\ e \in E$$

$$\begin{array}{ll} \overset{\text{by (26)}}{\Rightarrow} & (\sigma_q)_{q\in Q} \in \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha)}) \\ \overset{\text{by (29)}}{\Rightarrow} & \bigsqcup_{\mathscr{M}} (\sigma_q)_{q\in Q} \in \Sigma_{\|_{\mathscr{M}} \mathscr{P}} \text{ where} \\ & \qquad \pi_q\left(\bigsqcup_{\mathscr{M}} (\sigma_q)_{q\in Q}\right) = \sigma_q \quad \forall\ q \in Q \\ \overset{\text{by (32)}}{\Rightarrow} & \exists\ \sigma \in \Sigma_{\|_{\mathscr{I}_{\mathcal{S}}} \mathscr{P}} \text{ s.t. } \pi_q(\sigma) = \sigma_q \quad \forall\ q \in Q \\ \overset{\text{by (28)}}{\Rightarrow} & (\sigma_q)_{q\in Q} \in \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}}, \Sigma_{\mathscr{I}_{\mathcal{S}}}, \mathrm{Res}^{\rho})}) \\ \overset{\text{by (31)}}{\Rightarrow} & \mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}(\sigma_{\mathfrak{s}(e)}) = \mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)}(\sigma_{\mathfrak{t}(e)}) \quad \forall\ e \in E. \end{array}$$

The converse direction proceeds in the same manner: if

$$(\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q \text{ s.t. } \mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}(\sigma_{\mathfrak{s}(e)}) = \mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)}(\sigma_{\mathfrak{t}(e)}) \quad \forall\ e \in E$$

$$\begin{array}{ll} \overset{\text{by (29)}}{\Rightarrow} & \bigsqcup_{\mathscr{I}_{\mathcal{S}}} (\sigma_q)_{q\in Q} \in \Sigma_{\|_{\mathscr{I}_{\mathcal{S}}} \mathscr{P}} \text{ where} \\ & \qquad \pi_q\left(\bigsqcup_{\mathscr{I}_{\mathcal{S}}} (\sigma_q)_{q\in Q}\right) = \sigma_q \quad \forall\ q \in Q \\ \overset{\text{by (32)}}{\Rightarrow} & \exists\ \sigma' \in \Sigma_{\|_{\mathscr{M}} \mathscr{P}} \text{ s.t. } \pi_q(\sigma') = \sigma_q \quad \forall\ q \in Q \\ \overset{\text{by (28)}}{\Rightarrow} & (\sigma_q)_{q\in Q} \in \underleftarrow{\mathrm{Lim}}^{\mathsf{H}_\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}}, \Sigma_{\mathscr{M}}, \alpha)}) \\ \overset{\text{by (26)}}{\Rightarrow} & \alpha_e(\sigma_{\mathfrak{s}(e)}) = \alpha'_e(\sigma_{\mathfrak{t}(e)}) \quad \forall\ e \in E. \end{array}$$

(Necessity:) We have the following implications:

$$\exists\ \sigma' \in \Sigma_{\|_{\mathscr{M}}\mathscr{P}} \text{ s.t. } \pi_q(\sigma') = \sigma_q \quad \forall\ q \in Q$$

$$\begin{array}{ll} \overset{\text{by (28)}}{\Rightarrow} & (\sigma_q)_{q\in Q} \in \varprojlim^{\mathsf{H}\Gamma}(\mathbf{S}_{(\Sigma_{\mathscr{P}},\Sigma_{\mathscr{M}},\alpha)}) \\ \overset{\text{by (26)}}{\Rightarrow} & \alpha_e(\sigma_{\mathfrak{s}(e)}) = \alpha'_e(\sigma_{\mathfrak{t}(e)}) \quad \forall\ e \in E \\ \Rightarrow & \mathrm{Res}^{\rho_e}_{\mathfrak{s}(e)}(\sigma_{\mathfrak{s}(e)}) = \mathrm{Res}^{\rho'_e}_{\mathfrak{t}(e)}(\sigma_{\mathfrak{t}(e)}) \quad \forall\ e \in E \\ \overset{\text{by (29)}}{\Rightarrow} & \bigsqcup_{\mathscr{I}_{\mathcal{S}}}(\sigma_q)_{q\in Q} \in \Sigma_{\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}} \text{ and} \\ & \qquad \pi_q\left(\bigsqcup_{\mathscr{I}_{\mathcal{S}}}(\sigma_q)_{q\in Q}\right) = \sigma_q \quad \forall\ q \in Q. \end{array}$$

Therefore $\bigsqcup_{\mathscr{I}_{\mathcal{S}}}(\sigma_q)_{q\in Q}$ is the element of $\Sigma_{\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}}$ such that $\pi_q\left(\bigsqcup_{\mathscr{I}_{\mathcal{S}}}(\sigma_q)_{q\in Q}\right) = \sigma_q$ for all $q \in Q$.

The other direction follows in the same way:

$$\exists\ \sigma \in \Sigma_{\|_{\mathscr{I}_{\mathcal{S}}}\mathscr{P}} \text{ s.t. } \pi_q(\sigma) = \sigma_q \quad \forall\ q \in Q$$

$$\begin{array}{ll} \Rightarrow & \bigsqcup_{\mathscr{M}}(\sigma_q)_{q\in Q} \in \Sigma_{\|_{\mathscr{M}}\mathscr{P}} \text{ and} \\ & \qquad \pi_q\left(\bigsqcup_{\mathscr{M}}(\sigma_q)_{q\in Q}\right) = \sigma_q \quad \forall\ q \in Q, \end{array}$$

by (26), (28) and (29), so $\bigsqcup_{\mathscr{M}}(\sigma_q)_{q\in Q}$ is the element of $\Sigma_{\|_{\mathscr{M}}\mathscr{P}}$ such that $\pi_q\left(\bigsqcup_{\mathscr{M}}(\sigma_q)_{q\in Q}\right) = \sigma_q$ for all $q \in Q$.

Network desyncronization. Let $\mathcal{T}$ and $\mathcal{T}'$ be two tag structures and $\rho : \mathcal{T} \to \mathcal{T}'$ be a morphism between these tag structures. By slight abuse of notation, let $(\Gamma, \mathcal{T}, \mathcal{T}', \rho)$ denote the network of tag structures such that $\mathcal{T}_q = \mathcal{T}$ for all $q \in Q$ and $\mathcal{T}'_e = \mathcal{T}'$, $\rho_e = \rho'_e = \rho$ for all $e \in E$; denote the corresponding network of tagged systems by $(\Gamma, \mathscr{P}, \mathscr{I}_{\mathcal{T}'}, \mathrm{Res}^{\rho})$. Similarly, let $(\Gamma, \mathcal{T}, \mathcal{T}, \mathrm{id})$ denote the network of tag structures with $\mathcal{T}_{\mathfrak{s}(e)} = \mathcal{T}_{\mathfrak{t}(e)} = \mathcal{T}_e = \mathcal{T}$ for all $e \in E$, and with all morphisms of tag structures being the identity; denote the corresponding network of tagged systems by $(\Gamma, \mathscr{P}, \mathscr{I}, \mathrm{Res}^{\mathrm{id}})$. Therefore, this network consists of a set of tagged systems, all with the same tag structure, communicating through the identity tagged system. A special case in which this framework in interesting is when $\mathcal{T}' = \mathcal{T}_{\mathrm{triv}} = \{*\}$; in this case $(\Gamma, \mathscr{P}, \mathscr{I}_{\mathcal{T}'}, \mathrm{Res}^{\rho})$ is the *desynchronization* of $(\Gamma, \mathscr{P}, \mathscr{I}, \mathrm{Res}^{\mathrm{id}})$.

Utilizing the notation of Section 6, and generalizing the discussion on desynchronization given in this Section 4, we are interested in when

$$\|\mathscr{P} := \varprojlim^{\mathsf{H}\Gamma}(\mathbf{P}_{(\mathcal{T},\mathcal{T},\mathrm{id})}) \equiv \varprojlim^{\mathsf{H}\Gamma}(\mathbf{P}_{(\mathcal{T},\mathcal{T}',\rho)}) =: \|_{\mathscr{I}_{\mathcal{T}'}}\mathscr{P}.$$

In other words, we would like to know when $\mathscr{I}_{\mathcal{T}'}$ is semantics preserving. The following corollary (of Theorem 4) says that this happens exactly when every element of $\mathscr{I}_{\mathcal{T}'}$ is semantics preserving.

Corollary 2. *$\mathscr{I}_{\mathcal{T}'}$ is semantics preserving, $\|\mathscr{P} \equiv \|_{\mathscr{I}_{\mathcal{T}'}}\mathscr{P}$, if and only if for all $(\sigma_q)_{q\in Q} \in \prod_{q\in Q} \Sigma_q$ and all $e \in E$:*

$$\sigma^{\rho}_{\mathfrak{s}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}} = \sigma^{\rho}_{\mathfrak{t}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}} \quad \Rightarrow \quad \sigma_{\mathfrak{s}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}} = \sigma_{\mathfrak{t}(e)}|_{V_{\mathfrak{s}(e)}\cap V_{\mathfrak{t}(e)}}.$$

8 Conclusion

We have presented a categorical approach to the analysis of heterogeneous composition of tagged systems. This approach has allowed us to derive *necessary and sufficient conditions* for semantics preservation in a heterogeneous network of tagged systems. This result can be considered a generalization of the approach of Benveniste et al. who introduced the notion of tagged systems and their heterogeneous composition and derived *sufficient* conditions for semantics preservation.

The main theoretical tool utilized in this paper has been used in hybrid systems [1]. This is not surprising as hybrid systems are a special case of heterogeneous systems. To make this slightly more specific, a *hybrid object over a category* T is defined to be a functor $\mathbf{S} : \mathsf{H} \to \mathsf{T}$, where T is any category. A network of tagged systems is just a hybrid object over the category TagSys, and a hybrid system is a hybrid object over the category Dyn. Therefore, if general results can be obtained for hybrid objects over categories (such as bisimulation relations, cf. [9]) then they can be applied to networks of embedded systems. Conversely, the intuition gained in this paper on composing networks of tagged systems indicates a possible way of composing networks of dynamical systems: hybrid systems.

Acknowledgement. The authors are indebted to Alessandro Pinto for his invaluable feedback. The second author is indebted to Albert Benveniste, Benoit Caillaud, Luca Carloni and Paul Caspi for the many discussions and the work that led to the definition of tagged systems and the issue of semantics preservation of heterogeneous networks of tagged systems.

References

1. A. D. Ames and S. Sastry. A homology theory for hybrid systems: Hybrid homology. In M. Morari and L. Thiele, editors, *HSCC*, volume 3414 of *LNCS*, pages 86–102. Springer Verlag, 2005.
2. A. D. Ames and P. Tabuada. H-categories and graphs. Unpublished Note. `www.eecs.berkeley.edu/~adames`.
3. A. Benveniste, B. Caillaud, L. P. Carloni, P. Caspi, and A. L. Sangiovanni-Vincentelli. Heterogeneous reactive systems modeling: Capturing causality and the correctness of loosely time-triggered architectures (ltta). In G. Buttazzo and S. Edwards, editors, *EMSOFT*, 2004.

4. A. Benveniste, B. Caillaud, L. P. Carloni, P. Caspi, and A. L. Sangiovanni-Vincentelli. Composing heterogeneous reactive systems. Submitted for Publication, 2005.
5. A. Benveniste, L. P. Carloni, P. Caspi, and A. L. Sangiovanni-Vincentelli. Heterogeneous reactive systems modeling and correct-by-construction deployment. In R. Alur and I. Lee, editors, *EMSOFT*, volume 2855 of *LNCS*, pages 35–50. Springer Verlag, 2003.
6. A. Benveniste, B. Cauillaud, and P. Le Guernic. From synchrony to asynchrony. In J.C.M. Baeten and S. Mauw, editors, *CONCUR*, volume 1664 of *LNCS*, pages 162–177. Springer Verlag, 1999.
7. P. Caspi. Embedded control: from asynchrony to synchrony and back. In T. A. Henzinger and C. M. Kirsch, editors, *Porc. of the Workshop on embedded software systems*, volume 2211 of *LNCS*, pages 80–96. Springer, 2001.
8. L. de Alfaro and T. A. Henzinger. Interface theories for component-based design. In T. A. Henzinger and C. M. Kirsch, editors, *Porc. of the Workshop on embedded software systems*, volume 2211 of *LNCS*, pages 148–165. Springer, 2001.
9. E. Haghverdi, P. Tabuada, and G. J. Pappas. Bisimulation relations for dynamical and control systems. *Electronic Notes in Theoretical Computer Science*, 69:1–17, 2003.
10. H. Kopetz. *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Kluwer Academic Publishers, 1997.
11. S. Mac Lane. *Categories for the Working Mathematician*, volume 5 of *Graduate Texts in Mathematics*. Springer, second edition, 1998.
12. E. A. Lee and A. Sangiovanni-Vincentelli. A denotational framework for comparing models of computation. Technical report, University of California, Berkeley, 1997.
13. E. A. Lee and A. Sangiovanni-Vincentelli. A framework for comparing models of computation. In *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 1998.
14. A. Pinto, A. Sangiovanni-Vincentelli, R. Passerone, and L. Carloni. Interchange formats for hybrid systems: Review and proposal. In M. Morari and L. Thiele, editors, *HSCC*, volume 3414 of *LNCS*. Springer Verlag, 2005.
15. J. W. Polderman and J. C. Willems. *Introduction to Mathematical Systems Theory: A Behavioral Approach*. Springer, 1998.
16. A. J. van der Schaft and A. A. Julius. Achievable behavior by composition. In *Procedings of the 41st IEEE Conference on Decision and Control*, Las Vagas, Nevada, 2002.
17. A. J. van der Schaft and J. M. Schumacher. Compositionality issues in discrete, continuous and hybrid systems. *Int. J. Robust Nonlinear Control*, 11:417–434, 2001.
18. G. Winskel and M. Nielsen. Models for concurrency. In *Handbook of Logic and the Foundations of Computer Science*, volume 4, pages 1–148. Oxford University Press, 1995.