# Realisability of Global Models of Interaction

Maurice H. ter Beek[1] (✉), Rolf Hennicker[2], and José Proença[3] (✉)

[1] ISTI–CNR, Pisa, Italy, maurice.terbeek@isti.cnr.it
[2] Ludwig-Maximilians-Universität München, Germany, hennicker@ifi.lmu.de
[3] CISTER & University of Porto, Portugal, jose.proenca@fc.up.pt

**Abstract.** We consider *global models* of communicating agents specified as transition systems labelled by *interactions* in which multiple senders and receivers can participate. A *realisation* of such a model is a set of local transition systems—one per agent—which are executed concurrently using synchronous communication. Our core challenge is how to check whether a global model is realisable and, if it is, how to synthesise a realisation. We identify and compare two variants to realise global interaction models, both relying on bisimulation equivalence. Then we investigate, for both variants, *realisability conditions* to be checked on global models. We propose a synthesis method for the construction of realisations by grouping locally indistinguishable states. The paper is accompanied by a tool that implements realisability checks and synthesises realisations.

## 1 Introduction

We deal with the development of systems of collaborating computing entities which interact by message exchange, like communicating component systems, multi-agent systems (MAS), collective adaptive systems (CAS), groupware systems, multi-party sessions, etc. Such systems are often presented by a set of components whose local behaviour is formally described by labelled transition systems (LTS) or process expressions. Their interaction behaviour is then captured by (synchronous or asynchronous) parallel composition of the local models.

Before designing such local models it is, however, safer to first model the interaction behaviour of the components from a *global* perspective. This led to the investigation of various forms of global models, like global (session) types [7,12, 22,23], global choreographies [34] and global languages [2]; also message sequence charts [19] and UML interaction diagrams [14,29] serve this purpose.

An important question is, of course, whether a global model $\mathcal{M}$ is indeed realisable by a system $\mathcal{S} = (\mathcal{M}_i)_{i\in\mathcal{I}}$ of local component models $\mathcal{M}_i$ (where $\mathcal{I}$ ranges over a set of component names). Possible solutions are investigated for global languages in [2] and, for global session types, in various papers (cf., e.g., [7, 12, 22, 23]) by imposing syntactic restrictions on global types. These approaches use projections to generate local models from global ones.

A different idea is to provide, instead of a global model, a requirements specification *Sp* describing properties of the desired global interaction behaviour by means of some logical formalism like in [8,20,21]. Then local models $\mathcal{M}_i$ are constructed from scratch and their (synchronous) composition $\otimes(\mathcal{M}_i)_{i\in\mathcal{I}}$ must be proven to satisfy the requirements of *Sp*.

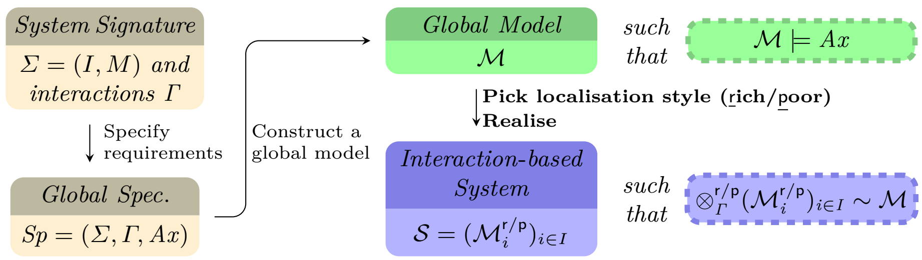



Fig. 1: Workflow for the development of interaction-based systems

**From Requirements to Realisations** We combine the advantages of logical specifications and global models for interaction-based systems by using both in a stepwise manner. Our development method is summarised in Fig. 1.

We start by providing a *(system) signature* $\Sigma = (I, M)$, which determines finite sets $I$ of component names and $M$ of message names. $\Sigma$ induces the set $\Gamma(\Sigma)$ of *(global) $\Sigma$-interactions* of the form $\mathsf{out} \to \mathsf{in} : \mathsf{m}$ where $\mathsf{out}$ and $\mathsf{in}$ are disjoint sets of component names (such that $\mathsf{out} \cup \mathsf{in} \neq \varnothing$) and $\mathsf{m} \in M$. The multi-interaction $\mathsf{out} \to \mathsf{in} : \mathsf{m}$ expresses that all components of $\mathsf{out}$ send message $\mathsf{m}$ to all components of $\mathsf{in}$ such that all send and receive events occur simultaneously. Since usually not all interactions in $\Gamma(\Sigma)$ are desired in an application context (one may wish, e.g., to consider only binary or multicast communication) we consider pairs $(\Sigma, \Gamma)$ where $\Gamma \subseteq \Gamma(\Sigma)$ is a user-defined *interaction set* which restricts the set of all $\Sigma$-interactions to admissible ones. For logical specifications of global interaction behaviour, we propose an action-based logic following a dynamic-logic style which has been successfully applied for specifying ensembles (cf., e.g., [21]). The logic uses the usual diamond and box modalities of dynamic logic ($\langle \alpha \rangle \varphi$ and $[\alpha] \varphi$, resp.) which range over (structured) interactions $\alpha$ built over $\Gamma$ by sequential composition, choice and iteration. A *global interaction-behaviour specification* is then a triple $Sp = (\Sigma, \Gamma, Ax)$, where $Ax$ is a set of formulas, called *axioms*, expressing requirements for the global interaction behaviour (e.g., safety and liveness properties and/or desired and forbidden interaction scenarios).

Given a (global) requirements specification $Sp = (\Sigma, \Gamma, Ax)$, we construct a global model $\mathcal{M}$ for the system's intended interaction behaviour. To formalise such models we use *global LTS* whose transitions are labelled by interactions according to $\Gamma$. If only binary interactions are admitted a global LTS is a choreography automaton as in [1]. Of course, we must check that the constructed global LTS $\mathcal{M}$ satisfies the requirements of the specification $Sp$, i.e. $\mathcal{M} \models Ax$.

The central part of our work concerns the realisation (decomposition) of a global LTS $\mathcal{M}$ in terms of a (possibly distributed) system of interacting components whose individual behaviour is modelled by *local LTS*. First we must determine, for each component name $i \in I$, which local actions component $i$ should provide. To do so, any interaction in which $i$ participates must be mapped to an appropriate local action for component $i$. We study two variants. The first follows approaches to multi-party session types and choreography languages where the names of the communication partners are kept in local actions. For instance, a binary interaction $i \to j : \mathsf{m}$ leads to a local output action $i\,j!\mathsf{m}$ for $i$ and a local input action $i\,j\,?\,\mathsf{m}$ for $j$. In approaches to component-based development, however, transitions describing local behaviour are often labelled just by message names accompanied by information whether it is an output or an input of a component. This makes components better reusable and supports interface-based design [3, 16, 25, 27]. In this case, a binary interaction $i \to j : \mathsf{m}$ leads to a local output action $!\mathsf{m}$ for $i$ and a local input action $?\mathsf{m}$ for $j$. In this paper, we generalise both localisation styles to deal with multi-interactions and call the former "rich local actions" and the latter "poor local actions". From a software designer's point of view the poor localisation style better supports the principle of loose coupling, whereas the rich style better avoids undesired synchronisations.

Once a localisation style $x \in \{\mathsf{r}, \mathsf{p}\}$ is chosen ($\mathsf{r}$ for "rich" and $\mathsf{p}$ for "poor") one can proceed with the actual construction of a realisation of $\mathcal{M}$ in terms of a system presented by a family $\mathcal{S} = (\mathcal{M}_i^x)_{i \in I}$ of local LTS. We say that $\mathcal{M}$ is *realisable* (with localisation style $x$) if such a system exists such that $\mathcal{M}$ is bisimilar (denoted by $\sim$) to the synchronous composition $\otimes_\Gamma^x (\mathcal{M}_i^x)_{i \in I}$ of all $\mathcal{M}_i^x$ taking into account the interactions $\Gamma$ and the localisation style $x$. Hence, our realisation notion is generic w.r.t. $\Gamma$ and parametrised by the chosen localisation style. We show that realisability with poor local actions implies realisability with rich local actions (Theorem 1) but the converse does not hold (Example 3). Since our realisability notion is based on bisimilarity we can deal with non-deterministic behaviour, differently from language-based approaches like [2, 12].

**Race Example** We illustrate our methodology outlined so far by developing a (small) system, called Race, which is meant to model the competition of two runner components R1 and R2 under the control of a third component Ctrl. To start, we provide a signature $\Sigma_{\mathsf{Race}} = (I_{\mathsf{Race}}, M_{\mathsf{Race}})$ with component names $I_{\mathsf{Race}} = \{\mathsf{R1}, \mathsf{R2}, \mathsf{Ctrl}\}$ and message names $M_{\mathsf{Race}} = \{\mathsf{start}, \mathsf{finish}\}$. The idea is that the controller starts the two runners simultaneously, while each runner signals individually to the controller when it has finished its run. Therefore, we use the interaction set on the left of Fig. 2. We do not model the actual running of a runner component, which would be an internal action (cf. [6]).

We require that no runner should finish before starting and that any started runner should be able to finish running. This will be expressed by dynamic logic formulas to be detailed in the requirements specification $Sp_{\mathsf{Race}}$ in Example 1.

Next we construct the global LTS $\mathcal{M}_{\mathsf{Race}}$ shown on the right of Fig. 2, which models the required interaction behaviour of the system so that the requirements of the specification are satisfied. The system starts in the initial (global) state 0, where the controller starts both runners at once. Each runner separately sends a finish signal to the controller (in arbitrary order). After that a new run can start.

$$\Gamma_{\mathsf{Race}} = \left\{ \begin{array}{l} \mathsf{Ctrl} \to \{\mathsf{R1}, \mathsf{R2}\} : \mathsf{start}\,, \\ \mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}\,, \\ \mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish} \end{array} \right\}$$

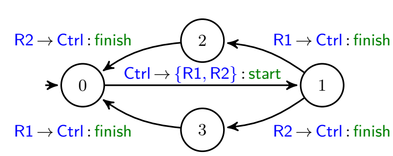
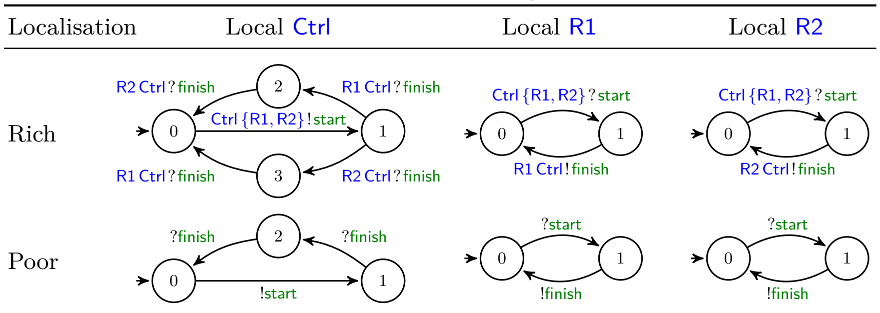



Fig. 2: Interaction set $\Gamma_{\mathsf{Race}}$ (left) and global LTS $\mathcal{M}_{\mathsf{Race}}$ (right); we write Ctrl for $\{\mathsf{Ctrl}\}$ and similarly for R1, R2.

Table 1: Local LTS for each localisation style and for each component




Finally, we want to realise the system by three local LTS such that their composition is bisimilar to the global LTS $\mathcal{M}_{\mathsf{Race}}$. We distinguish the two variants.

**Rich Local Actions** From $\mathit{\Gamma}_{\mathsf{Race}}$ we derive the following sets of rich local actions:

$$\mathit{\Lambda}^{\mathsf{r}}_{\mathsf{Ctrl}} = \big\{\mathsf{Ctrl}\,\{\mathsf{R1},\mathsf{R2}\}\,!\,\mathsf{start}\,,\ \mathsf{R1}\,\mathsf{Ctrl}\,?\,\mathsf{finish}\,,\ \mathsf{R2}\,\mathsf{Ctrl}\,?\,\mathsf{finish}\big\},$$
$$\mathit{\Lambda}^{\mathsf{r}}_{\mathsf{R1}} = \big\{\mathsf{Ctrl}\,\{\mathsf{R1},\mathsf{R2}\}\,?\,\mathsf{start}\,,\ \mathsf{R1}\,\mathsf{Ctrl}\,!\,\mathsf{finish}\big\},\ \text{and}$$
$$\mathit{\Lambda}^{\mathsf{r}}_{\mathsf{R2}} = \big\{\mathsf{Ctrl}\,\{\mathsf{R1},\mathsf{R2}\}\,?\,\mathsf{start}\,,\ \mathsf{R2}\,\mathsf{Ctrl}\,!\,\mathsf{finish}\big\}.$$

For each $i \in \{\mathsf{Ctrl}, \mathsf{R1}, \mathsf{R2}\}$, we use the local LTS $\mathcal{M}^{\mathsf{r}}_i$ in the upper row of Table 1 to build the system $\mathcal{S}^{\mathsf{r}}_{\mathsf{Race}} = \{\mathcal{M}^{\mathsf{r}}_{\mathsf{Ctrl}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{R1}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{R2}}\}$ with rich local actions. One can prove that the "rich" composition (Definition 4) of the three LTS by synchronisation w.r.t. $\mathit{\Gamma}_{\mathsf{Race}}$ is bisimilar (even isomorphic) to $\mathcal{M}_{\mathsf{Race}}$; i.e., we have found a realisation with rich local actions.

**Poor Local Actions** In this case, we derive from $\mathit{\Gamma}_{\mathsf{Race}}$ the following sets of poor local actions, where information on communication partners is omitted:

$$\mathit{\Lambda}^{\mathsf{p}}_{\mathsf{Ctrl}} = \{!\mathsf{start}, ?\mathsf{finish}\} \text{ and } \mathit{\Lambda}^{\mathsf{p}}_{\mathsf{R1}} = \mathit{\Lambda}^{\mathsf{p}}_{\mathsf{R2}} = \{?\mathsf{start}, !\mathsf{finish}\}.$$

For each $i \in \{\mathsf{Ctrl}, \mathsf{R1}, \mathsf{R2}\}$, we use the local LTS $\mathcal{M}^{\mathsf{p}}_i$ in the lower row of Table 1 to build the system $\mathcal{S}^{\mathsf{p}}_{\mathsf{Race}} = \{\mathcal{M}^{\mathsf{p}}_{\mathsf{Ctrl}}, \mathcal{M}^{\mathsf{p}}_{\mathsf{R1}}, \mathcal{M}^{\mathsf{p}}_{\mathsf{R2}}\}$ with poor local actions. Also the "poor" composition (Definition 7) of the three LTS by synchronisation w.r.t. $\mathit{\Gamma}_{\mathsf{Race}}$ is bisimilar (even isomorphic) to $\mathcal{M}_{\mathsf{Race}}$.

**Checking Realisability, Local Quotients, and System Synthesis** So far, we considered the case in which the realisation of a global interaction model $\mathcal{M}$ is "invented". However, there might be no realisation of $\mathcal{M}$ and it would be better to know this as soon as possible to align the global model. Next, we consider the following two important issues and proceed as shown in Fig. 3.
1. How to check whether a given global LTS $\mathcal{M}$ is realisable (rich/poor case)?
2. If it is, how can we build/synthesise a concrete realisation $S$ (rich/poor case)?

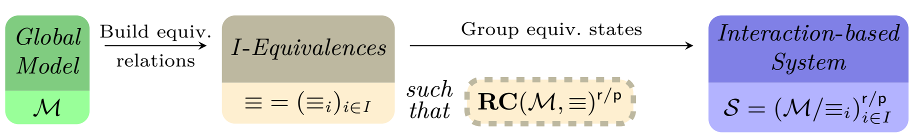



Fig. 3: Approach to check realisability and system synthesis

To tackle the first question we propose, similarly to [13], to find a family $\equiv = (\equiv_i)_{i \in I}$ of equivalence relations on the global state space $Q$ of $\mathcal{M}$ such that, for each component name $i \in I$ and states $q, q' \in Q$, $q \equiv_i q'$ expresses that $q$ and $q'$ are not distinguishable from the viewpoint of $i$. This suggests that $q$ and $q'$, though globally different, can be locally interpreted as the same states. In particular, it is required that any two states $q$ and $q'$ which are related by a global transition $q \xrightarrow{\mathsf{out} \to \mathsf{in} : \mathsf{m}}_{\mathcal{M}} q'$ should be indistinguishable for any $i \in I$ which does not participate in the interaction, i.e. $i \notin \mathsf{out} \cup \mathsf{in}$. On the basis of a given $I$-equivalence $\equiv$, we formulate realisability conditions $\mathsf{RC}(\mathcal{M}, \equiv)^{\mathsf{r}}$ and $\mathsf{RC}(\mathcal{M}, \equiv)^{\mathsf{p}}$ for both localisation styles. We show that in the rich and in the poor case our condition is sufficient for realisability (cf. Theorems 2 and 3).

In both cases, the principle idea how to synthesise a realisation is the same. Given a family $(\equiv_i)_{i \in I}$ of $I$-equivalences for which the realisability condition holds, we construct, for each $i \in I$, a local quotient $(\mathcal{M}/\equiv_i)^{\mathsf{r}/\mathsf{p}}$ by identifying global states (in $\mathcal{M}$) which are $i$-equivalent. Thus we get the desired system (which might still benefit from minimisations w.r.t. bisimilarity).

Note that the $I$-equivalences found for satisfying the realisability condition in the rich case may not be the same as in the poor case and thus also the local quotients may show different behaviour. Moreover, the technique of building local quotients differs from projections used in the field of multi-party session types, since projections are partial operations depending on syntactic conditions (cf., e.g., [7]). A less syntactic and more expressive approach is proposed in [24].

As an example, recall the global LTS $\mathcal{M}_{\mathsf{Race}}$ shown in Fig. 2 (right). The three local LTS with *rich* local actions shown in the upper row of Table 1 are, up to renaming of states, local quotients of $\mathcal{M}_{\mathsf{Race}}$. To construct the local quotient for R1, global states 0 and 2 are identified, as well as states 1 and 3 (and symmetrically for the local quotient for R2). For Ctrl, no proper identification is applied (cf. Example 4 for details). Also the three local LTS with *poor* local actions in the lower row of Table 1 are, up to renaming of states, local quotients of $\mathcal{M}_{\mathsf{Race}}$. In this case, however, to construct the local quotient for Ctrl, two global states of $\mathcal{M}_{\mathsf{Race}}$ are identified, namely states 2 and 3 (cf. Example 6 for details).

**Contributions and Related Work**

1. We propose a rigorous discipline for developing interaction-based systems following a step-wise development method from dynamic-logic requirements specifications over global models of interaction down to systems of (possibly distributed) components. Thus our approach supplements approaches to realisations of global behaviour descriptions (in the form of global languages, e.g. [2], or global session types, e.g. [22]), by an abstract logical layer.
2. Our approach is driven by specified sets of multi-interactions supporting any kind of synchronous communication between multiple senders and multiple receivers. To the best of our knowledge, realisations of global models with arbitrary multi-interactions have not yet been studied in the literature.
3. Our correctness notion for realisation of global models by systems of communicating local components is based on bisimulation, thus letting us deal with non-determinism and going beyond language-based approaches like [2, 12].

Bisimulation also fits well with global requirements specifications since dynamic logic formulas are invariant under bisimulation and therefore hold in any realisation of a global model of a global specification.

4. For constructing realisations we consider two localisation styles (rich and poor local actions) and analyse their relationship. This is a novel result.
5. A global interaction model may, in general, not be realisable. We provide conditions for realisability with respect to both localisation styles. Our conditions are related to the work in [13] which, however, does not deal with multi-interactions and uses a stronger condition ensuring realisation up to isomorphism of LTS; cf. our discussion in Sect. 5.1.
6. For realisable global models, we construct realisations in terms of systems of local quotients. Similar quotient constructions have been used in the proofs of [13], but not for multi-interactions and for different localisation styles. The technique of building local quotients differs from projections used in the field of multi-party session types, since projections are partial operations depending on syntactic conditions (cf., e.g., [7]). In our approach, no restrictions on the form of global models are assumed. However, it must be said that the syntactic restrictions used for global types guarantee some kind of communication properties of a resulting system which we do not consider.
7. We developed a prototypical tool Ceta which checks realisability conditions and, if they are satisfied, generates local quotients and hence realisations.

**Outline** After some formal preliminaries in Sect. 2, we show how to specify requirements for global models of interaction in Sect. 3 and how to realise the latter in Sect. 4. The conditions that guarantee realisability are studied in Sect. 5. In Sect. 6, we present a tool that implements our analyses. It is available at https://lmf.di.uminho.pt/ceta and all examples of the paper are predefined in the tool, like $\mathcal{M}_{\mathsf{Race}}$, including a hyperlink to open the tool with the specific example. Sect. 7 wraps up the paper. A companion report [5] includes all proofs of our results, more details of the tool, and a few additional examples.

## 2 Formal Preliminaries

**LTS and Bisimulation** Let $A$ be a finite set of actions. A *labelled transition system* (LTS) over $A$ is a tuple $\mathcal{L} = (Q, q_0, A, T)$ such that $Q$ is a finite set of states, $q_0 \in Q$ is the initial state, and $T \subseteq Q \times A \times Q$ is a transition relation. Note that we consider finite-state LTS, which makes the realisability conditions presented later *decidable*. We write $q \xrightarrow{a}_{\mathcal{L}} q'$ to denote $(q, a, q') \in T$. A state $q \in Q$ is *reachable* if there exists a finite sequence of transitions from initial state $q_0$ to $q$.

Let $\mathcal{L}_i = (Q_i, q_{i,0}, A, T_i)$ be two LTS (for $i = 1, 2$) over the same action set $A$. A *bisimulation relation* between $\mathcal{L}_1$ and $\mathcal{L}_2$ is a relation $B \subseteq Q_1 \times Q_2$ such that for all $(q_1, q_2) \in B$ and for all $a \in A$ the following holds:

1. if $q_1 \xrightarrow{a}_{\mathcal{L}_1} q_1'$ then there exist $q_2' \in Q_2$ and $q_2 \xrightarrow{a}_{\mathcal{L}_2} q_2'$ such that $(q_1', q_2') \in B$;
2. if $q_2 \xrightarrow{a}_{\mathcal{L}_2} q_2'$ then there exist $q_1' \in Q_1$ and $q_1 \xrightarrow{a}_{\mathcal{L}_1} q_1'$ such that $(q_1', q_2') \in B$.

$\mathcal{L}_1$ and $\mathcal{L}_2$ are *bisimilar*, denoted by $\mathcal{L}_1 \sim \mathcal{L}_2$, if there exists a bisimulation relation $B$ between $\mathcal{L}_1$ and $\mathcal{L}_2$ such that $(q_{1,0}, q_{2,0}) \in B$.

**Dynamic Logic** We use (test-free) propositional dynamic logic (PDL) [18] to formulate behavioural properties. Let $A$ be a finite set of *(atomic) actions*. Let the grammar $\alpha \ ::= \ a \mid \alpha;\alpha \mid \alpha{+}\alpha \mid \alpha^*$, with $a \in A$, sequential composition ; , non-deterministic choice $+$, and iteration $*$, define the set $Act(A)$ of *structured actions* over $A$. If $A = \{a_1, \dots, a_n\}$, we write *some* for structured action $a_1 + \cdots + a_n$. We may also refer to all actions of $A$ but one, say $a$, and express this by $-a$.

The set $Frm(A)$ of $A$-*formulas* is defined by the grammar

$$\varphi \ ::= \ \mathit{true} \mid \neg\varphi \mid \varphi \vee \varphi \mid \ \langle\alpha\rangle\,\varphi \qquad \text{(formulas)}$$

where $\alpha \in Act(A)$. Formula $\langle\alpha\rangle\,\varphi$ expresses that at the current state it is possible to execute $\alpha$ such that $\varphi$ holds in the next state.

**Abbreviations** We use the usual abbreviations like *false*, $\varphi \wedge \varphi'$, $\varphi \to \varphi'$, and the modal box operator $[\alpha]\,\varphi$ which stands for $\neg\,\langle\alpha\rangle\,\neg\varphi$ and expresses that whenever in the current state $\alpha$ is executed, then $\varphi$ holds afterwards.

For the interpretation of formulas we use LTS. Let $\mathcal{L} = (Q, q_0, A, T)$ be an LTS over $A$. First we extend the transition relation of $\mathcal{L}$ to structured actions:

$q \xrightarrow{\alpha_1;\alpha_2}_{\mathcal{L}} q'$ if there exists $\hat{q} \in Q$ such that $q \xrightarrow{\alpha_1}_{\mathcal{L}} \hat{q}$ and $\hat{q} \xrightarrow{\alpha_2}_{\mathcal{L}} q'$;
$q \xrightarrow{\alpha_1+\alpha_2}_{\mathcal{L}} q'$ if $q \xrightarrow{\alpha_1}_{\mathcal{L}} q'$ or $q \xrightarrow{\alpha_2}_{\mathcal{L}} q'$; and
$q \xrightarrow{\alpha^*}_{\mathcal{L}} q'$ if $q = q'$ or there exists $\hat{q} \in Q$ such that $q \xrightarrow{\alpha}_{\mathcal{L}} \hat{q}$ and $\hat{q} \xrightarrow{\alpha^*}_{\mathcal{L}} q'$.

The *satisfaction* of a formula $\varphi \in Frm(A)$ by $\mathcal{L}$ at a state $q \in Q$, denoted by $\mathcal{L}, q \models \varphi$, is inductively defined as follows:

$\mathcal{L}, q \models \mathit{true}$;
$\mathcal{L}, q \models \neg\varphi$ if not $\mathcal{L}, q \models \varphi$;
$\mathcal{L}, q \models \varphi_1 \vee \varphi_2$ if $\mathcal{L}, q \models \varphi_1$ or $\mathcal{L}, q \models \varphi_2$; and
$\mathcal{L}, q \models \langle\alpha\rangle\,\varphi$ if there exists $q' \in Q$ such that $q \xrightarrow{\alpha}_{\mathcal{L}} q'$ and $\mathcal{L}, q' \models \varphi$.

$\mathcal{L}$ *satisfies* a formula $\varphi \in Frm(A)$, denoted by $\mathcal{L} \models \varphi$, if $\mathcal{L}, q_0 \models \varphi$. Hence, for the satisfaction of a formula by an LTS the non-reachable states are irrelevant (deviating from the classical semantics of PDL [18]). We can express safety properties, like $[\mathit{some}^*]\,\varphi$, and some kinds of liveness properties like, e.g., $[\mathit{some}^*]\,\langle \mathit{some}^*; a\rangle\,\varphi$.

Satisfaction of formulas in PDL is invariant under bisimulation [9]: Let $\mathcal{L}_1, \mathcal{L}_2$ be two LTS over $A$. If $\mathcal{L} \sim \mathcal{L}'$ then, for any $\varphi \in Frm(A)$, $\mathcal{L} \models \varphi$ iff $\mathcal{L}' \models \varphi$.

## 3 Specifying Requirements for Global Models of Interaction

We focus on the stepwise development of systems whose components interact by synchronous message exchange. We support “multi-interactions”, in which several senders and receivers may participate in a communication. Our starting point are *signatures* $\Sigma = (I, M)$, where $I$ is a finite set of component names (also called participants) and $M$ is a finite set of message names. Any signature $\Sigma$ induces a set $\Gamma(\Sigma)$ of (global) *$\Sigma$-interactions* defined by

$$\Gamma(\Sigma) = \{\mathsf{out} \to \mathsf{in} : \mathsf{m} \mid \mathsf{out}, \mathsf{in} \subseteq I, \mathsf{out} \cup \mathsf{in} \neq \varnothing, \mathsf{m} \in M\}.$$

An interaction $\mathsf{out} \to \mathsf{in} : \mathsf{m}$ expresses that all components whose name occurs in $\mathsf{out}$ send a message named $\mathsf{m}$ to all components whose name occurs in $\mathsf{in}$. Such interactions involving arbitrarily many senders and receivers are also called *multi-interactions*. They will be interpreted by **synchronous** (handshake) communication. As a shorthand notation we write $i$ for $\{i\}$. Special cases are binary interactions between two components $i, j$, denoted by $i \to j : \mathsf{m}$, or multicast communication with one sender $i$ and a group $\mathsf{in}$ of receivers, denoted by $i \to \mathsf{in} : \mathsf{m}$.

Usually not all $\Sigma$-interactions are meaningful for a certain application. Therefore our approach will be driven by user-definable *interaction sets* $\Gamma \subseteq \Gamma(\Sigma)$.

**General Assumption** In the sequel, we assume that $(\Sigma, \Gamma)$ denotes a system signature $\Sigma = (I, M)$ together with an interaction set $\Gamma$. When we talk about a signature we always mean a system signature.

We propose to use interactions as atomic actions in dynamic logic formulas for specifying desired and forbidden interaction properties from a global perspective.

**Definition 1 (global $Sp$).** *A* global interaction behaviour specification *is a triple $Sp = (\Sigma, \Gamma, Ax)$ where $Ax \subseteq Frm(\Gamma)$ is a set of $\Gamma$-formulas, called* axioms.

*Example 1.* A requirements specification for the interaction behaviour of the $\mathsf{Race}$ system is given by $Sp_{\mathsf{Race}} = (\Sigma_{\mathsf{Race}}, \Gamma_{\mathsf{Race}}, Ax_{\mathsf{Race}})$ where $\Sigma_{\mathsf{Race}}$ and $\Gamma_{\mathsf{Race}}$ are defined in Sect. 1 and $Ax_{\mathsf{Race}}$ consists of the following two dynamic logic formulas expressing the two informal requirements described in Sect. 1.

1. *"No runner should finish before it has been started by the controller."*
$$\left[\Big(-(\mathsf{Ctrl} \to \{\mathsf{R1}, \mathsf{R2}\} : \mathsf{start})\Big)^* \ ; \ \begin{pmatrix}\mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}\ + \\ \mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish}\end{pmatrix}\right] \mathit{false}$$
2. *"For any started runner it should be possible to finish its run."*
$$\Big[\mathit{some}^*; \mathsf{Ctrl} \to \{\mathsf{R1}, \mathsf{R2}\} : \mathsf{start}\Big] \begin{pmatrix}\langle \mathit{some}^*; \mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}\rangle\ \mathit{true}\ \wedge \\ \langle \mathit{some}^*; \mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish}\rangle\ \mathit{true}\end{pmatrix}$$ ▷

Given a specification $Sp$, the goal of our next step is to model the global interaction behaviour of the intended system in accordance with $Sp$. For this purpose we use LTS with interactions from $\Gamma$ on the transitions.

**Definition 2 (global LTS).** *A* global LTS *over $(\Sigma, \Gamma)$ is defined as an LTS $\mathcal{M} = (Q, q_0, \Gamma, T)$ over $\Gamma$.*

To check that a global LTS satisfies the axioms of a specification, we may use the mCRL2 toolset [11] and, as explained in [6], the translation of LTS to process expressions as well as the translation of our dynamic logic formulas to the syntax used by mCRL2. For instance, the global LTS $\mathcal{M}_{\mathsf{Race}}$ provided for the race example in Sect. 1 satisfies the axioms of the specification $Sp_{\mathrm{Race}}$ above.

## 4 Realisations of Global Models of Interaction

A crucial step in our development method concerns the realisation of a global interaction model in terms of a system of (possibly distributed) components modelled by *local* LTS (cf. Fig. 1). In this section, we formally define what we mean by a realisation. For modelling local components we must first determine, for each component name $i \in I$, which are the local actions that component $i$ is supposed to support. We study two variants obeying different localisation styles and leading to different instantiations of our realisability notion.

### 4.1 Realisations Using Rich Local Actions

It is common in approaches to global (session) types and choreography languages to preserve the names of communication partners when moving from global interactions to local actions. In [2], e.g., a binary interaction $i \to j : \mathsf{m}$ leads to a local output action $i\,j\,!\,\mathsf{m}$ for $i$ and a local input action $i\,j\,?\,\mathsf{m}$ for j. We generalise this approach to multi-interactions and call the resulting local actions *rich*.

**Definition 3 (rich local actions and local LTS).** *For each $i \in I$, the set of* rich local i-actions *derived from $\Gamma$ is $\Lambda^{\mathsf{r}}_i(\Gamma) = \Lambda^{\mathsf{r}}_{i,out}(\Gamma) \cup \Lambda^{\mathsf{r}}_{i,in}(\Gamma)$ where*

$$\Lambda^{\mathsf{r}}_{i,out}(\Gamma) = \{\mathsf{out}\,\mathsf{in}\,!\,\mathsf{m} \mid \exists\,(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma \text{ such that } i \in \mathsf{out}\} \textit{ and}$$
$$\Lambda^{\mathsf{r}}_{i,in}(\Gamma) \ = \{\mathsf{out}\,\mathsf{in}\,?\,\mathsf{m} \mid \exists\,(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma \text{ such that } i \in \mathsf{in}\}.$$

*A* local LTS *for $i$ with rich local actions is an LTS $\mathcal{M}^{\mathsf{r}}_i = (Q_i, q_{i,0}, \Lambda^{\mathsf{r}}_i(\Gamma), T_i)$.*

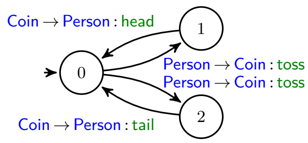
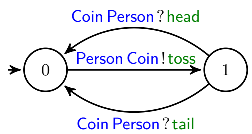
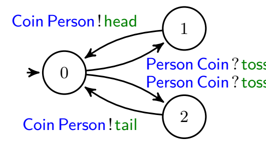
A *system* over $(\Sigma, \Gamma)$ with rich local actions is a family $\mathcal{S}^{\mathsf{r}} = (\mathcal{M}^{\mathsf{r}}_i)_{i \in I}$ of local LTS $\mathcal{M}^{\mathsf{r}}_i$ over $\Lambda^{\mathsf{r}}_i(\Gamma)$ for $i \in I$. The behaviour of such a system is modelled by the synchronous $\Gamma$-composition of its components $\mathcal{M}^{\mathsf{r}}_i$ ($i \in I$) where for all interactions $(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma$ a global transition exists (in a composed state) if for all $i \in \mathsf{out}$ ($i \in \mathsf{in}$, resp.) there is a transition in $\mathcal{M}^{\mathsf{r}}_i$ with the local action $\mathsf{out}\,\mathsf{in}\,!\,\mathsf{m}$ ($\mathsf{out}\,\mathsf{in}\,?\,\mathsf{m}$, resp.) leaving the current local state of $\mathcal{M}^{\mathsf{r}}_i$.

**Definition 4 (synchronous $\Gamma$-composition with rich local actions).** *Let $(\mathcal{M}^{\mathsf{r}}_i)_{i \in I}$ be a family of local LTS $\mathcal{M}^{\mathsf{r}}_i = (Q_i, q_{i,0}, \Lambda^{\mathsf{r}}_i(\Gamma), T_i)$ with rich local actions. The* synchronous $\Gamma$-composition *of $(\mathcal{M}^{\mathsf{r}}_i)_{i \in I}$ with rich local actions is the global LTS, denoted by $\otimes^{\mathsf{r}}_{\Gamma}(\mathcal{M}^{\mathsf{r}}_i)_{i \in I}$, over $(\Sigma, \Gamma)$ with initial state $(q_{i,0})_{i \in I}$ and with (product) states $(q_i)_{i \in I}$ (with $q_i \in Q_i$ for all $i \in I$) and transitions generated from the initial state by the following rule:*

$$\frac{(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma \qquad \forall i \in \mathsf{out} : q_i \xrightarrow{\mathsf{out}\,\mathsf{in}\,!\,\mathsf{m}}_{\mathcal{M}^{\mathsf{r}}_i} q'_i \qquad \forall i \in \mathsf{in} : q_i \xrightarrow{\mathsf{out}\,\mathsf{in}\,?\,\mathsf{m}}_{\mathcal{M}_i} q'_i}{(q_i)_{i \in I} \xrightarrow{\mathsf{out} \to \mathsf{in} : \mathsf{m}}_{\otimes^{\mathsf{r}}_{\Gamma}(\mathcal{M}^{\mathsf{r}}_i)_{i \in I}} (q'_i)_{i \in I} \text{ where } q'_i = q_i \text{ for all } i \in I \setminus (\mathsf{out} \cup \mathsf{in})}$$

**Definition 5 (realisability with rich local actions).** *Let $\mathcal{M}$ be a global LTS over $(\Sigma, \Gamma)$. A system $\mathcal{S} = (\mathcal{M}^{\mathsf{r}}_i)_{i \in I}$ over $(\Sigma, \Gamma)$ with rich local actions is a* (rich) realisation *of $\mathcal{M}$, if $\mathcal{M} \sim \otimes^{\mathsf{r}}_{\Gamma}(\mathcal{M}^{\mathsf{r}}_i)_{i \in I}$ are bisimilar. $\mathcal{M}$ is* realisable *with rich local actions if such a realisation exists.*

Table 2: Non-deterministic toss of a Coin by a Person

| Global $\mathcal{M}_{\mathsf{Toss}}$ | Local $\mathcal{M}^{\mathsf{r}}_{\mathsf{Person}}$ | Local $\mathcal{M}^{\mathsf{r}}_{\mathsf{Coin}}$ |
|---|---|---|
|   |   |   |

Our realisability notion relies on bisimulation. Thus we are able to deal with non-determinism. In particular, according to the invariance of dynamic logic under bisimulation (cf. Sect. 2), we know that global models and their realisations satisfy the same formulas. Hence, once a global model of a global specification is provided, any realisation will be correct with respect to the global specification.

*Example 2.* Consider a non-deterministic example with two participants, a Person and a Coin, and tossing the Coin by the Person is modelled as a non-deterministic action that leads to either head or tail (cf. [28]). Formally, $\Sigma_{\mathsf{Toss}} = (\{\mathsf{Coin}, \mathsf{Person}\}, \{\mathsf{toss}, \mathsf{head}, \mathsf{tail}\})$ and $\Gamma_{\mathsf{Toss}} = \{\mathsf{Person} \to \mathsf{Coin} : \mathsf{toss}, \mathsf{Coin} \to \mathsf{Person} : \mathsf{head}, \mathsf{Coin} \to \mathsf{Person} : \mathsf{tail}\}$. The global LTS $\mathcal{M}_{\mathsf{Toss}}$ and one of its realisations by the two LTS $\mathcal{M}^{\mathsf{r}}_{\mathsf{Person}}$ and $\mathcal{M}^{\mathsf{r}}_{\mathsf{Coin}}$ (with rich local actions) are shown in Table 2. Although $\mathcal{M}_{\mathsf{Toss}} \sim \otimes^{\mathsf{r}}_{\Gamma_{\mathsf{Toss}}}\{\mathcal{M}^{\mathsf{r}}_{\mathsf{Person}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{Coin}}\}$ there would be no bisimulation when considering a deterministic version for both Person and Coin. ▷

### 4.2 Realisations Using Poor Local Actions

We now consider a variant where we omit the communication partners when we move from a global interaction $(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma$ to local actions. In this case only the message name $\mathsf{m}$ is kept together with output information $!\mathsf{m}$ for $i \in \mathsf{out}$ and input information $?\mathsf{m}$ for $i \in \mathsf{in}$. This complies with the idea of component automata used in teams [3,4,6] and many other approaches to component-based design (e.g., I/O automata [27] and interface automata [16]). We call the resulting local actions "poor" since they do not specify communication partners.

**Definition 6 (poor local actions and local LTS).** *For each $i \in I$, the set of* poor local $i$-actions *derived from $\Gamma$ is given by $\Lambda^{\mathsf{p}}_i(\Gamma) = \Lambda^{\mathsf{p}}_{i,out}(\Gamma) \cup \Lambda^{\mathsf{p}}_{i,in}(\Gamma)$ where*

$$\Lambda^{\mathsf{p}}_{i,out}(\Gamma) = \{!\mathsf{m} \mid \exists\, (\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma \text{ such that } i \in \mathsf{out}\} \text{ and}$$
$$\Lambda^{\mathsf{p}}_{i,in}(\Gamma) \;\; = \{?\mathsf{m} \mid \exists\, (\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma \text{ such that } i \in \mathsf{in}\}.$$

*A* local LTS *for $i$ with poor local actions is an LTS $\mathcal{M}^{\mathsf{p}}_i = (Q_i, q_{i,0}, \Lambda^{\mathsf{p}}_i(\Gamma), T_i)$ over $\Lambda^{\mathsf{p}}_i(\Gamma)$.*

The notion of a **system with poor local actions** is defined completely analogously to the rich case in Sect. 4.1. $\Gamma$-composition with poor local actions needs, however, special care since for matching local actions only the message name and input/output information is relevant.

**Definition 7 (synchronous $\Gamma$-composition with poor local actions).** *Let $(\mathcal{M}_i^{\mathsf{p}})_{i\in I}$ be a family of local LTS $\mathcal{M}_i^{\mathsf{p}} = (Q_i, q_{i,0}, \Lambda_i^{\mathsf{p}}(\Gamma), T_i)$ with poor local actions. The* synchronous $\Gamma$-composition *of $(\mathcal{M}_i^{\mathsf{p}})_{i\in I}$ with poor local actions is the global LTS, denoted by $\otimes_\Gamma^{\mathsf{p}}(\mathcal{M}_i^{\mathsf{p}})_{i\in I}$, over $(\Sigma, \Gamma)$ with initial state $(q_{i,0})_{i\in I}$ and with (product) states $(q_i)_{i\in I}$ (such that $q_i \in Q_i$ for all $i \in I$) and transitions generated from the initial state by the following rule:*

$$\frac{(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma \qquad (\forall i \in \mathsf{out} : q_i \xrightarrow{!\mathsf{m}}_{\mathcal{M}_i^{\mathsf{p}}} q_i') \qquad (\forall i \in \mathsf{in} : q_i \xrightarrow{?\mathsf{m}}_{\mathcal{M}_i^{\mathsf{p}}} q_i')}{(q_i)_{i\in I} \xrightarrow{\mathsf{out} \to \mathsf{in} : \mathsf{m}}_{\otimes_\Gamma^{\mathsf{p}}(\mathcal{M}_i^{\mathsf{p}})_{i\in I}} (q_i')_{i\in I} \text{ where } q_i' = q_i \text{ for all } i \in I \setminus (\mathsf{out} \cup \mathsf{in})}$$

The notion of **realisability with poor local actions** is defined completely analogously to the rich case (cf. Definition 5) replacing "rich ($\mathsf{r}$)" by "poor ($\mathsf{p}$)".

An obvious question is whether realisability with respect to the two different localisation styles can be formally compared. This is indeed the case.

**Theorem 1 (poor realisation implies rich realisation).** *Let $\mathcal{M}$ be a global LTS over $(\Sigma, \Gamma)$ which is realisable by a system $\mathcal{S}^{\mathsf{p}} = (\mathcal{M}_i^{\mathsf{p}})_{i\in I}$ with poor local actions. Then there exists a system $\mathcal{S}^{\mathsf{r}} = (\mathcal{M}_i^{\mathsf{r}})_{i\in I}$ with rich local actions which is a realisation of $\mathcal{M}$.*

The converse of Theorem 1 is not true, as demonstrated by the next example.

*Example 3.* We consider a variant of the global LTS $\mathcal{M}_{\mathsf{Race}}$ (Fig. 2) where the transitions $1 \xrightarrow{\mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish}} 3 \xrightarrow{\mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}} 0$ are removed, enforcing R1 to finish before R2. Let us call the resulting LTS $\mathcal{M}'_{\mathsf{Race}}$. Moreover, consider the variant of the local controller $\mathcal{M}^{\mathsf{r}}_{\mathsf{Ctrl}}$ (upper row of Table 1, left) where the local transitions $1 \xrightarrow{\mathsf{R2Ctrl?finish}} 3 \xrightarrow{\mathsf{R1Ctrl?finish}} 0$ are removed and call it $\mathcal{M}'^{\mathsf{r}}_{\mathsf{Ctrl}}$. Now let $\mathcal{S}'^{\mathsf{r}} = \{\mathcal{M}'^{\mathsf{r}}_{\mathsf{Ctrl}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{R1}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{R2}}\}$ be the system with rich local actions (where $\mathcal{M}^{\mathsf{r}}_{\mathsf{R1}}$ and $\mathcal{M}^{\mathsf{r}}_{\mathsf{R2}}$ are shown in the upper row of Table 1, middle and right). It is easy to check that $\mathcal{S}'^{\mathsf{r}}$ is a realisation of $\mathcal{M}'_{\mathsf{Race}}$ with rich local actions, since $\mathcal{M}'_{\mathsf{Race}}$ is even isomorphic to the (rich) $\Gamma_{\mathsf{Race}}$-composition of $\{\mathcal{M}'^{\mathsf{r}}_{\mathsf{Ctrl}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{R1}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{R2}}\}$.

The situation is different if we consider the poor case with controller $\mathcal{M}^{\mathsf{p}}_{\mathsf{Ctrl}}$ (lower row of Table 1, left) which accepts two times in a row a "finish" signal but, due to the poor local actions, cannot fix an acceptance order. The only candidate for a realisation with poor local actions is then the system $\mathcal{S}'^{\mathsf{p}} = \{\mathcal{M}^{\mathsf{p}}_{\mathsf{Ctrl}}, \mathcal{M}^{\mathsf{p}}_{\mathsf{R1}}, \mathcal{M}^{\mathsf{p}}_{\mathsf{R2}}\}$ consisting of the local LTS with poor local actions shown in the lower row of Table 1. Obviously, the $\Gamma_{\mathsf{Race}}$-composition of these local LTS with poor actions does allow a sequence of transitions $1 \xrightarrow{\mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish}} 3 \xrightarrow{\mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}} 0$ and therefore cannot be bisimilar to $\mathcal{M}'_{\mathsf{Race}}$. ▷

## 5 Realisability Conditions

In general a global LTS may not be realisable. Therefore we are interested in (i) conditions that guarantee realisability and (ii) techniques to synthesise realisations from a global LTS $\mathcal{M}$. The notion of $I$-equivalence provides a helpful tool.

The basic idea is to consider the source and target states of a global transition $q \xrightarrow{\mathsf{out}\to\mathsf{in}:\mathsf{m}}_{\mathcal{M}} q'$ to be indistinguishable for a component $i \in I$ if $i$ does not participate in the interaction, i.e. $i \notin \mathsf{out} \cup \mathsf{in}$ (cf. also [13] and the discussion below).

**Definition 8 ($I$-equivalence).** *Let $\mathcal{M} = (Q, q_0, \Gamma, T)$ be a global LTS over $(\Sigma, \Gamma)$. An $I$-*equivalence *over $\mathcal{M}$ is a family $\equiv = (\equiv_i)_{i\in I}$ of equivalence relations $\equiv_i \subseteq Q \times Q$ (reflexive, symmetric, and transitive) such that $q \equiv_i q'$ holds whenever there exists a transition $q \xrightarrow{\mathsf{out}\to\mathsf{in}:\mathsf{m}}_{\mathcal{M}} q'$ with $i \notin \mathsf{out} \cup \mathsf{in}$. The equivalence class of a state $q \in Q$ w.r.t. $\equiv_i$ is the set $[q]_{\equiv_i} = \{q' \in Q \mid q' \equiv_i q\}$.*

### 5.1 Condition for Realisability Using Rich Local Actions

First, we will formulate our realisability condition for the case of rich local actions. We consider a global LTS $\mathcal{M}$ over $(\Sigma, \Gamma)$. Our goal is to find an $I$-equivalence $(\equiv_i)_{i\in I}$ over $\mathcal{M}$ such that for each interaction $(\mathsf{out}\to\mathsf{in}:\mathsf{m}) \in \Gamma$ the following holds. Assume, for simplicity, that $\mathsf{out} \cup \mathsf{in} = \{1, \dots, n\}$. Whenever there is a combination $q_1, \dots, q_n$ of $n$ (not necessarily different) global states together with a global *"glue"* state $g$, i.e. for each $j \in \mathsf{out} \cup \mathsf{in}$, $q_j \equiv_j g$, then we expect: if $\mathsf{out}\to\mathsf{in}:\mathsf{m}$ is enabled in each global state $q_1, \dots, q_n$ then each $j \in \mathsf{out} \cup \mathsf{in}$ should also able to participate in $\mathsf{out}\to\mathsf{in}:\mathsf{m}$ when the global state is $g$, since $j$ cannot distinguish $g$ from $q_j$. Thus the global interaction $\mathsf{out}\to\mathsf{in}:\mathsf{m}$ should be enabled in $g$ and preserve $I$-equivalences.

**Definition 9 (realisability condition (rich case)).** *Let $\mathcal{M}$ be a global LTS over $(\Sigma, \Gamma)$. The* realisability condition *$RC(\mathcal{M})^{\mathsf{r}}$ for $\mathcal{M}$ with respect to rich local actions says that there exists an $I$-equivalence $\equiv = (\equiv_i)_{i\in I}$ over $\mathcal{M}$ such that the following property $RC(\mathcal{M}, \equiv)^{\mathsf{r}}$ holds.*

*For all $\gamma = (\mathsf{out}\to\mathsf{in}:\mathsf{m}) \in \Gamma$ with $\mathsf{out} \cup \mathsf{in} = \{k_1, \dots, k_n\}$ we have:*

$$\forall \begin{pmatrix} q_1 \xrightarrow{\gamma}_{\mathcal{M}} q_1' \quad \cdots \quad q_n \xrightarrow{\gamma}_{\mathcal{M}} q_n' \\ g \in \bigcap_{j=1}^{n} [q_j]_{\equiv_{k_j}} \end{pmatrix} \qquad \exists g' : \begin{pmatrix} g \xrightarrow{\gamma}_{\mathcal{M}} g' \\ g' \in \bigcap_{j=1}^{n} [q_j']_{\equiv_{k_j}} \end{pmatrix}$$

Theorem 2 will provide a constructive argument why condition $RC(\mathcal{M})^{\mathsf{r}}$ ensures realisability with rich local actions. Local quotients are crucial for this.

**Definition 10 (local quotients with rich local actions).** *Let $\mathcal{M} = (Q, q_0, \Gamma, T)$ be a global LTS over $(\Sigma, \Gamma)$ and $\equiv = (\equiv_i)_{i\in I}$ an $I$-equivalence over $\mathcal{M}$. For each $i \in I$ the* local i-quotient *of $\mathcal{M}$ with rich local actions is the LTS $(\mathcal{M}/\equiv_i)^{\mathsf{r}} = (Q/\equiv_i, [q_0]_{\equiv_i}, \Lambda_i^{\mathsf{r}}(\Gamma), (T/\equiv_i)^{\mathsf{r}})$ where*

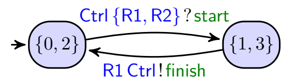
- *$Q/\equiv_i = \{[q]_{\equiv_i} \mid q \in Q\}$,*
- *$(T/\equiv_i)^{\mathsf{r}}$ is the least set of transitions generated by the following rules:*

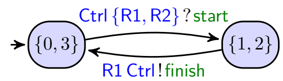
$$\frac{q \xrightarrow{\mathsf{out}\to\mathsf{in}:\mathsf{m}}_{\mathcal{M}} q' \qquad i \in \mathsf{out}}{[q]_{\equiv_i} \xrightarrow{\mathsf{out}\,\mathsf{in}\,!\,\mathsf{m}}_{(\mathcal{M}/\equiv_i)^{\mathsf{r}}} [q']_{\equiv_i}} \qquad\qquad \frac{q \xrightarrow{\mathsf{out}\to\mathsf{in}:\mathsf{m}}_{\mathcal{M}} q' \qquad i \in \mathsf{in}}{[q]_{\equiv_i} \xrightarrow{\mathsf{out}\,\mathsf{in}\,?\,\mathsf{m}}_{(\mathcal{M}/\equiv_i)^{\mathsf{r}}} [q']_{\equiv_i}}$$

Note that $[q]_{\equiv_i} \xrightarrow{\mathsf{out}\,\mathsf{in}\,!\,\mathsf{m}}_{(\mathcal{M}/\equiv_i)^{\mathsf{r}}} [q']_{\equiv_i}$ implies that there exist $\hat{q} \in [q]_{\equiv_i}$, $\hat{q}' \in [q']_{\equiv_i}$ and a transition $\hat{q} \xrightarrow{\mathsf{out}\to\mathsf{in}:\mathsf{m}}_{\mathcal{M}} \hat{q}'$ with $i \in \mathsf{out}$ (and similarly for $\mathsf{out}\,\mathsf{in}\,?\,\mathsf{m}$).

**Theorem 2.** *Let $\mathcal{M}$ be a global LTS over $(\Sigma, \Gamma)$ and let $\equiv\, = (\equiv_i)_{i \in I}$ be an $I$-equivalence over $\mathcal{M}$. If $RC(\mathcal{M}, \equiv)^{\mathsf{r}}$ holds, then $\mathcal{M} \sim \otimes^{\mathsf{r}}_{\Gamma}((\mathcal{M}/{\equiv_i})^{\mathsf{r}})_{i \in I}$.*

*Example 4.* Consider the global LTS $\mathcal{M}_{\mathsf{Race}}$ (Fig. 2). We show that $RC(\mathcal{M}_{\mathsf{Race}})^{\mathsf{r}}$ holds and how to construct, following Theorem 2, a realisation of $\mathcal{M}_{\mathsf{Race}}$. More concretely, we take the family of equivalences $\equiv\, = (\equiv_i)_{i \in \{\mathsf{Ctrl}, \mathsf{R1}, \mathsf{R2}\}}$ that obeys $RC(\mathcal{M}_{\mathsf{Race}}, \equiv)^{\mathsf{r}}$ (see below) and partitions the state space $Q$ as follows: $Q/{\equiv_{\mathsf{Ctrl}}} = \{\{0\}, \{1\}, \{2\}, \{3\}\}$, $Q/{\equiv_{\mathsf{R1}}} = \{\{0, 2\}, \{1, 3\}\}$, and $Q/{\equiv_{\mathsf{R2}}} = \{\{0, 3\}, \{1, 2\}\}$. Using these equivalences, the local quotients for R1 and R2 are as follows:

$(\mathcal{M}/{\equiv_{\mathsf{R1}}})^{\mathsf{r}} =$ 

$(\mathcal{M}/{\equiv_{\mathsf{R2}}})^{\mathsf{r}} =$ 


The local quotient for Ctrl is isomorphic to $\mathcal{M}_{\mathsf{Race}}$ but with local labels. Thus we have obtained a system which is a realisation with rich local actions of $\mathcal{M}_{\mathsf{Race}}$. The local quotients coincide, up to renaming of states, with the local LTS used in the system $\mathcal{S}^{\mathsf{r}}_{\mathsf{Race}}$ considered in Sect. 1.

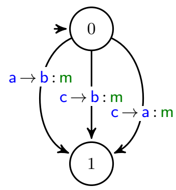
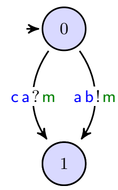
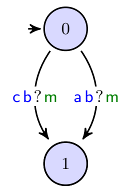
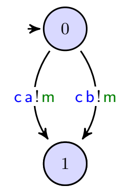
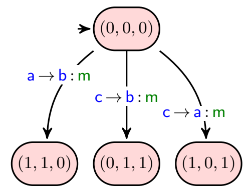
Now, we illustrate how to verify $RC(\mathcal{M}_{\mathsf{Race}}, \equiv)^{\mathsf{r}}$ using, as an example, the interaction $\gamma = \mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}$ which appears twice in $\mathcal{M}_{\mathsf{Race}}$: $t_{12} = (1 \xrightarrow{\gamma} 2)$ and $t_{30} = (3 \xrightarrow{\gamma} 0)$. There are two participants involved: R1 and Ctrl. Hence we need to consider four combinations: $(t_{12}, t_{12})$, $(t_{12}, t_{30})$, $(t_{30}, t_{12})$, and $(t_{30}, t_{30})$. For example, using the combination $(t_{30}, t_{12})$, we compute the glue $[3]_{\equiv_{\mathsf{R1}}} \cap [1]_{\equiv_{\mathsf{Ctrl}}} = \{1\}$. Then, trivially, there exists a transition $1 \xrightarrow{\gamma} 2$, and $2 \in [0]_{\equiv_{\mathsf{R1}}} \cap [2]_{\equiv_{\mathsf{Ctrl}}}$. The same can be shown for all the glues found for the other three combinations. ▷

In general it may happen that a global LTS $\mathcal{M}$ does not satisfy $RC(\mathcal{M})^{\mathsf{r}}$ but nevertheless is realisable. We can prove that $RC(\mathcal{M})^{\mathsf{r}}$ is a necessary condition to obtain a realisation which is related to a global model by a *functional* bisimulation. More interesting, however, would be to weaken $RC(\mathcal{M})^{\mathsf{r}}$ such that it becomes necessary for realisability with respect to arbitrary bisimulations. This is an open and challenging question.

*Example 5.* Consider the signature $\Sigma = (I, M)$ with $I = \{\mathsf{a}, \mathsf{b}, \mathsf{c}\}$, $M = \{\mathsf{m}\}$ and the set $\Gamma = \{\mathsf{a} \to \mathsf{b} : \mathsf{m}, \mathsf{c} \to \mathsf{b} : \mathsf{m}, \mathsf{c} \to \mathsf{a} : \mathsf{m}\}$ of $\Sigma$-interactions. The global LTS $\mathcal{M}$ in Table 3 (left) is realisable by the system $\mathcal{S}^{\mathsf{r}} = \{\mathcal{M}^{\mathsf{r}}_{\mathsf{a}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{b}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{c}}\}$. To see this, we compute the $\Gamma$-composition $\mathcal{M}' = \otimes^{\mathsf{r}}_{\Gamma}\{\mathcal{M}^{\mathsf{r}}_a, \mathcal{M}^{\mathsf{r}}_b, \mathcal{M}^{\mathsf{r}}_c\}$ shown in Table 3 (right). Obviously $\mathcal{M}$ is bisimilar to $\mathcal{M}'$ and hence $\mathcal{M}$ is realisable. However, $\mathcal{M}$ does not satisfy the realisability condition $RC(\mathcal{M})^{\mathsf{r}}$.

We prove this by contradiction. Assume that $\equiv\, = \{\equiv_{\mathsf{a}}, \equiv_{\mathsf{b}}, \equiv_{\mathsf{c}}\}$ is an $I$-equivalence such that $RC(\mathcal{M}, \equiv)^{\mathsf{r}}$ holds. Now consider the interaction $\mathsf{a} \to \mathsf{b} : \mathsf{m}$, the global state 0 of $\mathcal{M}$ and the transition $0 \xrightarrow{\mathsf{a} \to \mathsf{b} : \mathsf{m}}_{\mathcal{M}} 1$. Obviously, $0 \equiv_{\mathsf{a}} 1$ and $0 \equiv_{\mathsf{b}} 1$ must hold since there is the transition $0 \xrightarrow{\mathsf{c} \to \mathsf{b} : \mathsf{m}}_{\mathcal{M}} 1$ where a does not participate and the transition $0 \xrightarrow{\mathsf{c} \to \mathsf{a} : \mathsf{m}}_{\mathcal{M}} 1$ where b does not participate. So we can take 1 as a glue state between the global states $q_1 = 0$ and $q_2 = 0$. Then we consider the transition $0 \xrightarrow{\mathsf{a} \to \mathsf{b} : \mathsf{m}}_{\mathcal{M}} 1$ one time for $q_1$ and one time for $q_2$. Since we have assumed $RC(\mathcal{M}, \equiv)^{\mathsf{r}}$, there must be a transition $1 \xrightarrow{\mathsf{a} \to \mathsf{b} : \mathsf{m}}_{\mathcal{M}'}$

Table 3: Global LTS $\mathcal{M}$ not satisfying $RC(\mathcal{M})^{\mathsf{r}}$ but with realisation $\mathcal{S}^{\mathsf{r}} = \{\mathcal{M}^{\mathsf{r}}_{\mathsf{a}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{b}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{c}}\}$ where $\mathcal{M}' = \otimes^{\mathsf{r}}\{\mathcal{M}^{\mathsf{r}}_{\mathsf{a}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{b}}, \mathcal{M}^{\mathsf{r}}_{\mathsf{c}}\}$

| Global $\mathcal{M}$ | Local $\mathcal{M}^{\mathsf{r}}_{\mathsf{a}}$ | Local $\mathcal{M}^{\mathsf{r}}_{\mathsf{b}}$ | Local $\mathcal{M}^{\mathsf{r}}_{\mathsf{c}}$ | Composed $\mathcal{M}'$ |
|---|---|---|---|---|
|  |  |  |  |  |



leaving the glue state which is, however, not the case. Contradiction! Note that nevertheless the bisimilar global LTS $\mathcal{M}'$ does satisfy $RC(\mathcal{M}')^{\mathsf{r}}$. The example can be checked at $\mathcal{M}$ and $\mathcal{M}'$. ▷

**Discussion** Our realisability condition $RC(\mathcal{M})^{\mathsf{r}}$, based on the notion of an $I$-equivalence $(\equiv_i)_{i\in I}$, is strongly related to a condition for implementability in [13, Theorem 3.1]. In fact, $RC(\mathcal{M})^{\mathsf{r}}$ can be seen as a generalisation of [13] since we consider multi-interactions with distinguished sets of senders and receivers and also specifications for admissible interactions represented by $\Gamma$. Thus we get a generic realisability notion based on $\Gamma$-composition rather than full synchronisation. Moreover, our condition ensures realisability modulo bisimulation instead of isomorphism. Technically, implementability with respect to isomorphism is achieved in [13, Theorem 3.1] by requiring that whenever two global states $q$ and $q'$ are $i$-equivalent, i.e. $(q \equiv_i q')$, for all $i \in I$, then $q = q'$. We do not use this assumption and thus can get realisations modulo bisimilarity which do not realise a global LTS up to isomorphism (cf. [5, Example 8]). Note that [13, Theorem 6.2] also provides a proposal to deal with implementability modulo bisimulation under the assumption of "deterministic product transition systems". In the next section, we study a realisability condition for the case of poor local actions, which deviates significantly from [13].

### 5.2 Condition for Realisability Using Poor Local Actions

We return to the question of how to check realisability, now in the case of poor local actions. The notion of $I$-equivalence is again the key. Note, however, that the computation of an appropriate $I$-equivalence may differ from the rich case.

The realisability condition below is stronger than the one for rich local actions in Definition 9. Intuitively, the reason is that local LTS with poor local actions have, in general, more choices for synchronisation and therefore a global LTS must support these choices in order to be realisable. For each interaction $(\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma$, we require in more cases the enabledness in a glue state $g$. More concretely, $\mathsf{out} \to \mathsf{in} : \mathsf{m}$ must be enabled in $g$ already when in the $j$-equivalent states, say $q_j$, component $j$ is able to output/input message $m$ independently of the communication partners named in $\mathsf{out} \cup \mathsf{in}$, since those would anyway not be known from a poor local action. This is formally reflected by considering the interactions $\gamma_1, \ldots, \gamma_n$ in the next definition.

**Definition 11 (realisability condition (poor case)).** *Let $\mathcal{M}$ be a global LTS over $(\Sigma, \Gamma)$. The* realisability condition $RC(\mathcal{M})^{\mathsf{p}}$ *for $\mathcal{M}$ with respect to poor local actions says that there exists an $I$-equivalence $(\equiv_i)_{i \in I}$ over $\mathcal{M}$ such that the following property $RC(\mathcal{M}, (\equiv_i)_{i \in I})^{\mathsf{p}}$ holds.*

*For all $\gamma = (\mathsf{out} \to \mathsf{in} : \mathsf{m}) \in \Gamma$ with participants $\mathsf{out} \cup \mathsf{in} = \{k_1, \ldots, k_n\}$ we get:*

$$\forall \begin{pmatrix} \gamma_1 = (\mathsf{out}_1 \to \mathsf{in}_1 : \mathsf{m}) \in \Gamma \\ k_1 \in (\mathsf{out}_1 \cap \mathsf{out}) \cup (\mathsf{in}_1 \cap \mathsf{in}) \\ \ldots \\ \gamma_n = (\mathsf{out}_n \to \mathsf{in}_n : \mathsf{m}) \in \Gamma \\ k_n \in (\mathsf{out}_n \cap \mathsf{out}) \cup (\mathsf{in}_n \cap \mathsf{in}) \end{pmatrix} \; \forall \begin{pmatrix} q_1 \xrightarrow{\gamma_1}_{\mathcal{M}} q_1' \\ \ldots \\ q_n \xrightarrow{\gamma_n}_{\mathcal{M}} q_n' \\ g \in \bigcap_{j=1}^{n} [q_j]_{\equiv_{k_j}} \end{pmatrix} \; \exists g' : \begin{pmatrix} g \xrightarrow{\gamma}_{\mathcal{M}} g' \\ g' \in \bigcap_{j=1}^{n} [q_j']_{\equiv_{k_j}} \end{pmatrix}$$

To prove that the condition $RC(\mathcal{M})^{\mathsf{p}}$ indeed guarantees realisability with poor local actions, the idea is again to consider local quotients. Their construction is, however, different from the rich case.

**Definition 12 (local quotients with poor local actions).** *Let $\mathcal{M} = (Q, q_0, \Gamma, T)$ be a global LTS over $(\Sigma, \Gamma)$ and $\equiv = (\equiv_i)_{i \in I}$ an $I$-equivalence over $\mathcal{M}$. For each $i \in I$ the* local i-quotient *of $\mathcal{M}$ with poor local actions is the LTS $(\mathcal{M}/{\equiv_i})^{\mathsf{p}} = (Q/{\equiv_i}, [q_0]_{\equiv_i}, \Lambda_i^{\mathsf{p}}(\Gamma), (T/{\equiv_i})^{\mathsf{p}})$ where*

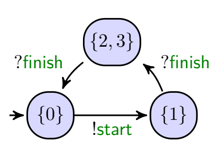
- $Q/{\equiv_i} = \{[q]_{\equiv_i} \mid q \in Q\}$,
- $(T/{\equiv_i})^{\mathsf{p}}$ *is the least set of transitions generated by the following rules:*

$$\frac{q \xrightarrow{\mathsf{out} \to \mathsf{in} : m}_{\mathcal{M}} q' \qquad i \in \mathsf{out}}{[q]_{\equiv_i} \xrightarrow{!\mathsf{m}}_{(\mathcal{M}/\equiv_i)^{\mathsf{p}}} [q']_{\equiv_i}} \qquad\qquad \frac{q \xrightarrow{\mathsf{out} \to \mathsf{in} : m}_{\mathcal{M}} q' \qquad i \in \mathsf{in}}{[q]_{\equiv_i} \xrightarrow{?\mathsf{m}}_{(\mathcal{M}/\equiv_i)^{\mathsf{p}}} [q']_{\equiv_i}}$$

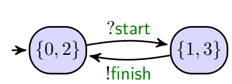
**Theorem 3.** *Let $\mathcal{M}$ be a global LTS over $(\Sigma, \Gamma)$ and let $\equiv = (\equiv_i)_{i \in I}$ be an $I$-equivalence over $\mathcal{M}$. If $RC(\mathcal{M}, \equiv)^{\mathsf{p}}$ holds then $\mathcal{M} \sim \otimes_{\Gamma}^{\mathsf{p}}((\mathcal{M}/{\equiv_i})^{\mathsf{p}})_{i \in I}$.*

*Example 6.* Consider the global LTS $\mathcal{M}_{\mathsf{Race}}$ (Fig. 2). We show $RC(\mathcal{M}_{\mathsf{Race}})^{\mathsf{p}}$ holds and how, following Theorem 3, a realisation of $\mathcal{M}_{\mathsf{Race}}$ with poor local actions can be constructed. The situation differs from the rich case in Example 4, since the equivalence for Ctrl must be chosen differently. We use the family of equivalences $\equiv = (\equiv_i)_{i \in \{\mathsf{Ctrl}, \mathsf{R1}, \mathsf{R2}\}}$ that obeys $RC(\mathcal{M}_{\mathsf{Race}}, \equiv)^{\mathsf{p}}$ (see below) and partitions the state space $Q$ as follows: $Q/{\equiv_{\mathsf{Ctrl}}} = \{\{0\}, \{1\}, \{2, 3\}\}$, $Q/{\equiv_{\mathsf{R1}}} = \{\{0, 2\}, \{1, 3\}\}$, and $Q/{\equiv_{\mathsf{R2}}} = \{\{0, 3\}, \{1, 2\}\}$. Using these equivalences, the local quotients for Ctrl, R1 and R2 are as follows:

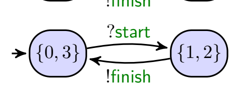
$(\mathcal{M}/{\equiv_{\mathsf{Ctrl}}})^{\mathsf{p}} =$




$(\mathcal{M}/{\equiv_{\mathsf{R1}}})^{\mathsf{p}} =$




$(\mathcal{M}/{\equiv_{\mathsf{R2}}})^{\mathsf{p}} =$



Thus we have obtained a system which is a realisation with poor local actions of $\mathcal{M}_{\mathsf{Race}}$. The local quotients coincide, up to renaming of states, with the local LTS used in the system $\mathcal{S}^{\mathsf{p}}_{\mathsf{Race}}$ considered in Sect. 1.

Now, we illustrate how to verify $RC(\mathcal{M}_{\mathsf{Race}}, \equiv)^{\mathsf{r}}$ using, as an example, the interaction $\mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}$. We have $1 \xrightarrow{\mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}}_{\mathcal{M}_{\mathsf{Race}}} 2$. We also have $1 \xrightarrow{\mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish}}_{\mathcal{M}_{\mathsf{Race}}} 3$ (we must consider the interaction $\mathsf{R2} \to \mathsf{Ctrl} : \mathsf{finish}$ as well since we are in the poor case). Taking 1 as a (trivial) glue state, we thus have, as required, the existence of $1 \xrightarrow{\mathsf{R1} \to \mathsf{Ctrl} : \mathsf{finish}}_{\mathcal{M}_{\mathsf{Race}}} 2$ *but also* it is required that $2 \equiv_{\mathsf{Ctrl}} 3$ must hold which is the case. Note that we wouldn't have succeeded here if we would have taken the identity for $\equiv_{\mathsf{Ctrl}}$ as done for the rich case. ▷

## 6 Tool Support: Ceta

We developed a supporting prototypical tool *Ceta* (Choreographic Extended Team Automata) to analyse global specifications and produce visualisations of state machines. It is open-source, available at https://github.com/arcalab/choreo/tree/ceta, and executable by browsing to https://lmf.di.uminho.pt/ceta.

Ceta starts with a web browser, opening a static webpage that uses our compiled JavaScript built with the Caos framework [31] (cf. screenshot in Fig. 4).

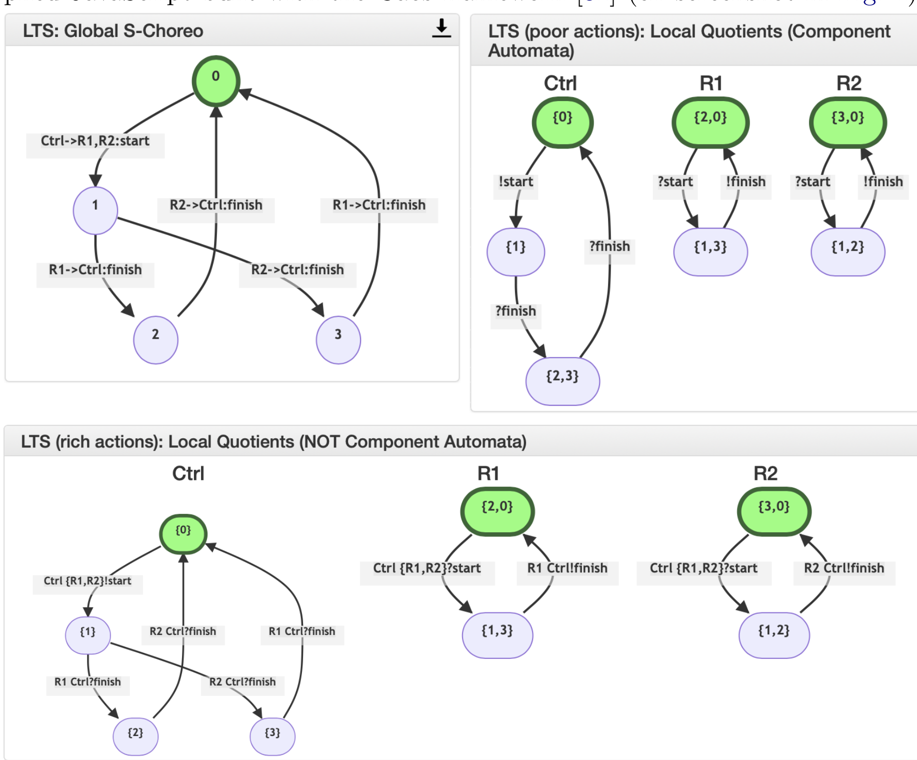



Fig. 4: Screenshots of the Ceta tool (http://lmf.di.uminho.pt/ceta)

The user input is a global protocol described in a choreographic language, resembling regular expressions of interactions. A set of examples with descriptions is also included, covering the examples presented in this paper. The analyses include graphical views of: (i) the global LTS; (ii) local LTS's with rich actions; and (iii) local LTS's with poor actions. Other widgets provide further insights, such as the composition of the local LTS's, the intermediate equivalence classes for both the rich and poor case, the synchronous composition of local LTS's, and bisimulations between the global prototol and composed systems. Readable error messages are provided when the realisability conditions do not hold.

## 7 Conclusion

We have proposed a rigorous discipline for developing interaction-based systems. At the heart of our methodology lies the realisation of a global interaction model, i.e. its decomposition into a set of (possibly distributed) components with synchronous communication. We have investigated realisability conditions for two different localisation styles (rich and poor local actions) and techniques to synthesise realisations. Our approach is generic with respect to the choice of admissible interaction sets which may contain arbitrary interactions between multiple senders and receivers but may also be restricted, e.g., to various forms of communication, like multicast or peer-to-peer communication. Due to the generic nature of our notion of an interaction set, our results can be instantiated by different concrete coordination formalisms. For instance, synchronisation type specifications used in the framework of (extended) team automata [4] as well as interactions used in BIP [10] can be represented by interaction sets. Our results should then be directly applicable, to extend the team automata framework as well as BIP by global LTS and to generate distributed component systems for them on the basis of our realisation conditions.

In future research, we plan to (i) integrate the treatment of internal actions using weak bisimulation equivalence for the realisation notions; (ii) consider communication properties (like receptiveness and responsiveness, cf. [4]) when systems are generated from global models; (iii) study open global models (systems) and their composition; and (iv) investigate realisability conditions in the context of asynchronous communication. Moreover, we are still looking for a weaker version of our realisability condition for synchronous systems making it necessary for arbitrary (also non-functional) bisimulations.

Furthermore, we intend to investigate the relation with work in the literature on the decomposition of related formalisms like (Petri net or algebraic) processes into (indecomposable) components [26, 30] used to parallelise (verification of) concurrent systems [15,17] or obtain better (optimised) implementations [32,33].

**Acknowledgments.** *Ter Beek* was supported by MUR PRIN 2020TL3X8X project T-LADIES (Typeful Language Adaptation for Dynamic, Interacting and Evolving Systems) and *Proença* by the CISTER Research Unit (UIDP/UIDB/04234/2020), financed by National Funds through FCT/MCTES (Portuguese Foundation for Science and

Technology); by project IBEX (PTDC/CCI-COM/4280/2021) financed by national funds through FCT; and by project Route 25 (ref. TRB/2022/00061 – C645463824-00000063) funded by the EU/Next Generation, within the Recovery and Resilience Plan.

## References


1. Barbanera, F., Lanese, I., Tuosto, E.: Choreography Automata. In: COORDINATION. LNCS, vol. 12134, pp. 86–106. Springer (2020). https://doi.org/10.1007/978-3-030-50029-0_6
2. Barbanera, F., Lanese, I., Tuosto, E.: Formal Choreographic Languages. In: COORDINATION. LNCS, vol. 13271, pp. 121–139. Springer (2022). https://doi.org/10.1007/978-3-031-08143-9_8
3. ter Beek, M.H., Ellis, C.A., Kleijn, J., Rozenberg, G.: Synchronizations in Team Automata for Groupware Systems. Comput. Sup. Coop. Work **12**(1), 21–69 (2003). https://doi.org/10.1023/A:1022407907596
4. ter Beek, M.H., Hennicker, R., Kleijn, J.: Compositionality of Safe Communication in Systems of Team Automata. In: ICTAC. LNCS, vol. 12545, pp. 200–220. Springer (2020). https://doi.org/10.1007/978-3-030-64276-1_11
5. ter Beek, M.H., Hennicker, R., Proença, J.: Realisability of Global Models of Interaction (Extended Version). Tech. rep., Zenodo (December 2023). https://doi.org/10.5281/zenodo.8377188
6. ter Beek, M.H., Cledou, G., Hennicker, R., Proença, J.: Can We Communicate? Using Dynamic Logic to Verify Team Automata. In: FM. LNCS, vol. 14000, pp. 122–141. Springer (2023). https://doi.org/10.1007/978-3-031-27481-7_9
7. Bejleri, A., Yoshida, N.: Synchronous Multiparty Session Types. ENTCS **241**, 3–33 (2008). https://doi.org/10.1016/j.entcs.2009.06.002
8. Ben-David, S., Chechik, M., Gurfinkel, A., Uchitel, S.: CSSL: a logic for specifying conditional scenarios. In: ESEC/FSE. pp. 37–47. ACM (2011). https://doi.org/10.1145/2025113.2025123
9. van Benthem, J., van Eijck, J., Stebletsova, V.: Modal Logic, Transition Systems and Processes. J. Log. Comput. **4**(5), 811–855 (1994). https://doi.org/10.1093/logcom/4.5.811
10. Bliudze, S., Sifakis, J.: The Algebra of Connectors: Structuring Interaction in BIP. IEEE Trans. Comput. **57**(10), 1315–1330 (2008). https://doi.org/10.1109/TC.2008.26
11. Bunte, O., Groote, J.F., Keiren, J.J.A., Laveaux, M., Neele, T., de Vink, E.P., Wesselink, W., Wijs, A., Willemse, T.A.C.: The mCRL2 Toolset for Analysing Concurrent Systems. In: TACAS. LNCS, vol. 11428, pp. 21–39. Springer (2019). https://doi.org/10.1007/978-3-030-17465-1_2
12. Castagna, G., Dezani-Ciancaglini, M., Padovani, L.: On Global Types and Multi-Party Sessions. Log. Methods Comput. Sci. **8**(1), 24:1–24:45 (2012). https://doi.org/10.2168/LMCS-8(1:24)2012
13. Castellani, I., Mukund, M., Thiagarajan, P.S.: Synthesizing Distributed Transition Systems from Global Specification. In: FSTTCS. LNCS, vol. 1738, pp. 219–231. Springer (1999). https://doi.org/10.1007/3-540-46691-6_17
14. Cengarle, M.V., Knapp, A., Mühlberger, H.: Interactions. In: Lano, K. (ed.) UML 2 Semantics and Applications, chap. 9, pp. 205–248. Wiley (2009). https://doi.org/10.1002/9780470522622
15. Corradini, F., Gorrieri, R., Marchignoli, D.: Towards parallelization of concurrent systems. RAIRO Theor. Informatics Appl. **32**(4-6), 99–125 (1998). https://doi.org/10.1051/ita/1998324-600991

16. de Alfaro, L., Henzinger, T.A.: Interface Automata. In: ESEC/FSE. pp. 109–120. ACM (2001). https://doi.org/10.1145/503209.503226
17. Groote, J.F., Moller, F.: Verification of Parallel Systems via Decomposition. In: CONCUR. LNCS, vol. 630, pp. 62–76. Springer (1992). https://doi.org/10.1007/BFb0084783
18. Harel, D., Kozen, D., Tiuryn, J.: Dynamic Logic. Foundations of Computing, MIT Press (2000). https://doi.org/10.7551/mitpress/2516.001.0001
19. Harel, D., Thiagarajan, P.S.: Message Sequence Charts. In: Lavagno, L., Martin, G., Selic, B. (eds.) UML for Real: Design of Embedded Real-Time Systems, pp. 77–105. Kluwer (2003). https://doi.org/10.1007/0-306-48738-1_4
20. Hennicker, R.: Role-Based Development of Dynamically Evolving Esembles. In: WADT. LNCS, vol. 11563, pp. 3–24. Springer (2018). https://doi.org/10.1007/978-3-030-23220-7_1
21. Hennicker, R., Wirsing, M.: Dynamic Logic for Ensembles. In: ISoLA. LNCS, vol. 11246, pp. 32–47. Springer (2018). https://doi.org/10.1007/978-3-030-03424-5_3
22. Honda, K., Yoshida, N., Carbone, M.: Multiparty asynchronous session types. In: POPL. pp. 273–284. ACM (2008). https://doi.org/10.1145/1328438.1328472
23. Hüttel, H., Lanese, I., Vasconcelos, V.T., Caires, L., Carbone, M., Deniélou, P., Mostrous, D., Padovani, L., Ravara, A., Tuosto, E., Vieira, H.T., Zavattaro, G.: Foundations of Session Types and Behavioural Contracts. ACM Comput. Surv. **49**(1), 3:1–3:36 (2016). https://doi.org/10.1145/2873052
24. Jongmans, S.S., Ferreira, F.: Synthetic Behavioural Typing: Sound, Regular Multiparty Sessions via Implicit Local Types. In: ECOOP. LIPIcs, vol. 263, pp. 9:1–9:29. Dagstuhl (2023). https://doi.org/10.4230/LIPIcs.ECOOP.2023.9
25. Larsen, K.G., Nyman, U., Wąsowski, A.: Modal I/O Automata for Interface and Product Line Theories. In: ESOP. LNCS, vol. 4421, pp. 64–79. Springer (2007). https://doi.org/10.1007/978-3-540-71316-6_6
26. Luttik, B.: Unique parallel decomposition in branching and weak bisimulation semantics. Theor. Comput. Sci. **612**, 29–44 (2016). https://doi.org/10.1016/j.tcs.2015.10.013
27. Lynch, N.A., Tuttle, M.R.: An Introduction to Input/Output Automata. CWI Q. **2**(3), 219–246 (1989), https://ir.cwi.nl/pub/18164
28. Magee, J., Kramer, J.: Concurrency: State Models & Java Programming. Wiley (2006)
29. Micskei, Z., Waeselynck, H.: The many meanings of UML 2 Sequence Diagrams: a survey. Softw. Syst. Model. **10**(4), 489–514 (2011). https://doi.org/10.1007/s10270-010-0157-9
30. Milner, R., Moller, F.: Unique Decomposition of Processes. Theor. Comput. Sci. **107**(2), 357–363 (1993). https://doi.org/10.1016/0304-3975(93)90176-T
31. Proença, J., Edixhoven, L.: Caos: A Reusable Scala Web Animator of Operational Semantics. In: COORDINATION. LNCS, vol. 13908, pp. 163–171. Springer (2023). https://doi.org/10.1007/978-3-031-35361-1_9
32. Teren, V., Cortadella, J., Villa, T.: Decomposition of transition systems into sets of synchronizing state machines. In: DSD. pp. 77–81. IEEE (2021). https://doi.org/10.1109/DSD53832.2021.00021
33. Teren, V., Cortadella, J., Villa, T.: Decomposition of transition systems into sets of synchronizing Free-choice Petri Nets. In: DSD. pp. 165–173. IEEE (2022). https://doi.org/10.1109/DSD57027.2022.00031
34. Tuosto, E., Guanciale, R.: Semantics of global view of choreographies. J. Log. Algebr. Meth. Program. **95**, 17–40 (2018). https://doi.org/10.1016/j.jlamp.2017.11.002